Zur Erlangung des akademischen Grades eines
Doktors der Ingenieurwissenschaften (Dr.-Ing.)
von der Fakultät für Wirtschaftswissenschaften
des Karlsruher Instituts für Technologie (KIT)
genehmigte Dissertation von
Dipl. Medienwissenschaftler Fabian Flöck

MAKING SOCIAL DYNAMICS AND CONTENT EVOLUTION TRANSPARENT IN COLLABORATIVELY WRITTEN TEXT

FABIAN MORITZ FLÖCK

Tag der mündlichen Prüfung: 28. Januar 2016
Referent: Prof. Dr. Rudi Studer
Korreferent: Prof. Dr. Markus Strohmaier

Karlsruhe/Köln, 2016

ABSTRACT

This dissertation presents models and algorithmic procedures for accurately and efficiently extracting data from revisioned content in Collaborative Writing Systems about (i) the provenance and history of specific sequences of text, as well as (ii) interactions between editors via the content changes they perform, especially disagreement. Based on these techniques, services and tools are presented that leverage the extracted data for further use. We also discuss systematic collaboration processes that can be researched with these new data and tools.

On the use case of the English Wikipedia, and based on empirical observations and research literature, the thesis first outlines evidence for certain "social mechanisms" that can systematically influence collaboration patterns of editors in articles and can lead to harmful consequences for content quality, e.g., xenophobia and eristic arguments. We specify which data would be necessary to study these dynamics in depth, in particular data for precisely tracking the history and evolution of certain content elements and the interactions of editors with the text and with each other. As most Collaborative Writing Systems keep track of the text revisions of documents, but not of meta-data about specific text sequences, interactions and alterations, we present a technique that enables to exactly determine the provenance of – and changes to – single content elements. To this end a hierarchical, k-partite graph is proposed to model the nested elements of a revisioned document, i.e., paragraphs, sentences and single tokens; an algorithm is then implemented based on this model. By means of the English Wikipedia we show by conducting user experiments that this method distinctly outperforms state-of-the-art approaches in the accuracy of determining the origin of content changes in the revision history. Our technique further achieves a reduction of one order of magnitude in runtime over current approaches, which makes it an appropriate tool for the analysis of large amounts of data. We briefly present an API service for extracting the enriched data from Wikipedia that our tools offer on-demand. Next, it is demonstrated how the extracted data can be formally encoded as interactions between editors in a directed, weighted and signed network graph. For this purpose we define antagonistic and supporting interactions, e.g., mutual deletion or the reintroduction of removed content. We show through a user survey that, with our model of disagreement, the detection of interaction behavior between editors can be notably refined and extended compared to the most commonly used approach for detecting reverts. Lastly, two novel, interactive and Web-based visu-

alization tools are presented: (i) whoCOLOR, a browser-plugin that provides an overlay for Wikipedia articles, showing original authors of content elements as well as their specific editing history and conflicts surrounding them and (ii) whoVIS, a network graph of disagreement between editors in an article that is explorable over time and allows inspection of concrete disagreement edges. These tools can be employed to explore the extracted data in a way that makes it possible for end-users to achieve better insights into the complex editing dynamics of articles.

ACKNOWLEDGMENTS

My gratitude goes first of all to my family for supporting me, especially my mother, who always assured me of my abilities when I was doubting them.
Further, most of the credit for this thesis being initiated at all goes to my first advisor, Elena Simperl. She believed that I, with a Social Science background, actually had a place in a Computer Science group, put a lot of trust in me and taught me a great deal about what it means to write a successful scientific paper or proposal, and to navigate the scientific landscape in general.
Similarly, I owe much to Rudi Studer for actually taking me on as a PhD student without knowing what to expect; and for being the kindest, most patient superior that anyone can hope for in any kind of career.
I also want to thank Denny Vrandečić for introducing me to the "wonders" of Wikipedia, being always enthusiastic, and giving me innumerable hints and pointers.
Further, Felix Stadthaus and Andriy Rodchenko have helped considerably to push the presented research forward with the work they have done under my supervision.
My gratitude extends to Markus Strohmaier, for being my second examiner and providing me with very constructive criticism.
Thanks go also to the rest of my colleagues at the Knowledge Management research group at AIFB for the support, the feedback on my work and not least the great Kicker sessions and Friday beers.
Lastly, I'm most indebted to Maribel, my motivator to keep going when it gets roughest and from whom I probably learned the most, in work and life. Without her, many things would not have turned out as well as they did.

CONTENTS

1 INTRODUCTION

When readers gauge the trustworthiness and factual accuracy of a newspaper report, an article on a news website, or even a blog, they employ a range of heuristics: they consider for instance writing style and grammar, logical consistency of the text, and, not least, supporting references and quotes [72].

Yet one of the major factors used for judging these dimensions are the characteristics of the *sources*, namely the media outlet and the author(s) – in contrast to references or cited sources. These primary originators of content bear various indicators for a reader to estimate the credibility of the offered information. Perceived expertise, general trustworthiness, credentials, similarity to own beliefs and goodwill of a source have all been shown, in a range of communication studies, to significantly influence a readers evaluation of the offered content, on and off the Web [72, 165]. Similar factors regarding publication venues and authors have likewise been demonstrated to affect audiences' assessment of quality in scientific writing [97]; a process which you, dear reader, might experience at this very moment.

In comparison, when we read products of *digital collaborative writing*, such as a Wikipedia article or, in the work place, an organization-internal Wiki page or a digital Google spreadsheet[1] composed by various co-workers, we are prone to apply similar heuristics for judging the credibility of the presented information [100, 102]. However, the main differences to the previous news media examples are that the authors of such a document are commonly multiple individuals, and that the identity of the authors of specific content that we read is mostly unknown in the first place, simply because individual author attribution is in the main not available or would be very arduous to retrieve (i.e., by manually going trough edit log files) in those systems [6].[2] We hence have to place our "source trust" in the brand, platform or community body that publishes the information (in the case of Wikipedia) or the individuals that are primarily responsible for the document as a whole (in the office document case) – if we think the heuristic to be applicable *at all* under these conditions.

One step to enable this quality-estimation mechanic in such collaborative, revisioned text systems like Wikipedia would be the ability to **accurately identify individual authors or editors of specific pieces of content** and even get additional information on their personal expertise or other characteristics. While an author's character-

1 Part of the Google Docs service: http://docs.google.com

2 Some collaborative writing platforms like Etherpad include explicit provenance-annotation of text pieces by default, but the majority doesn't.

istics might not be the main barometer to decide on the credibility of a source, they certainly are important complements to the inherent attributes of the content itself (like easily verifiable facts or retrievable references), and especially when the latter ones are not available, reputation and expertise of an author can act as valuable assurances of credibility.

On the other hand, simply knowing *who* wrote a document or specific parts of it can, in many cases, prove to be insufficient or even misleading on its own. More telling for assessing the trustworthiness or general quality of a text might it be to understand *how* the text was composed, i.e., the (collaborative) processes behind the production of the presented content. In journalism, to draw from our comparative example, processes and team routines behind news production have been shown to have a marked impact on quality [65] – while collaborative content production is becoming more common in the domain [116]. Intuitively, this makes sense to us: we would want to know if the writer of the news report on a major chemical plant leak was coerced by his editor to leave out specific information pointing to responsible actors because it was not "airtight" enough. It would change our interpretation of the breaking story on mass surveillance if there was a major disagreement in the editorial staff meeting about two different angles to the narrative and only one survived to see the printing press. And we would certainly be interested if the main writer of an article on a collaborative news blog received several revisions to misstated facts from her co-editors, but did not accept the corrections to her piece.

What is true for journalistic content production certainly holds likewise for the relatively new forms of writing and editing digital documents together, and even goes beyond, as we will explore deeper in Chapter 3: the dynamics of how editors of single Wiki pages or similar collaborative documents interact with the content – and with each other over that content – is decisive for the end-result of the common document, which is eventually presented to the reader. The collective interactions, especially how disagreement is resolved, are crucial levers for the eventually produced information. Certain behavioral patterns – such as claiming ownership of a collaborative document or non-cooperative behavior – between co-authors and editors on article level can systematically prevent the creation of high-quality content and might thus, as a complement to content-based analysis, serve as indicators for possible flaws in quality; or even enable an interpretation of the text that is inspired by its social writing history. This information can provide key insights into why certain parts of the document say what they say, especially in those cases or portions of the text where one cannot easily consult external sources for cross-checking. A method for revisioned, Collaborative Writing Systems (CWS) to **extract the fine-grained interac-**

tions of editors with each other and with the content over time and represent them in a way that best models the underlying reality of the socio-technical system, based on the text revisions produced by the editors, would therefore help to understand the collaboration mechanisms behind the content construction. Apart from a possible **quality assessment**, such data could enable a much broader study to **understand the social mechanisms that drive and form digital collaborative writing** in general. It could also enable a kind of **social transparency, accountability and self-monitoring** of the editors writing documents together – either on individual or group level, in order to learn from and hence avoid counter-productive interaction mechanisms. The foundation for gaining these insights is the **efficient and accurate mining** of such data and the quantitative and qualitative analysis that can be performed on top of it.

One reason why accountability and social transparency are hard to provide is that understanding the collaborative writing history of a document as a casual user, or even editor, in a straightforward, intuitive way is still a hard task. Although some of the systems (esp. Wikis) fully document all edits in the revision history and provide some aid in navigating them, there is no uncomplicated way to browse, inspect and analyze the creation process in all its intricacy. This information would be key to enable deeper insights into the writing process; but it is effectively hidden from the user due to the innate complexity. In this light, it would also be helpful to provide readers, editors and researchers of such systems with **intuitive visual interfaces as a low-threshold way of exploring authorship and collaboration dynamics** of revisioned documents.

1.1 RESEARCH QUESTIONS

In view of these deliberations, this dissertation aims at answering the following research questions:

- **RQ1**: *In collaboratively writing and editing specific digital documents together, what systematically appearing social mechanisms can be identified that have the potential to influence the quality of the eventual document produced and what methods do we need to model and detect them?*

  As an answer to this question, we extract such social mechanisms from research work on Wikipedia in Chapter 3; and we conclude that reliable and scaleable approaches for change tracking as well as mining provenance[3] and user interactions via text revisions are the main, crucial parts that are needed, but not yet available. Hence we formulate **RQ2**.

3 "Provenance" here refers to the source revision where a piece of content was first introduced to the article.

- **RQ2**: *How do models and algorithmic methods have to be designed to help us extract content provenance and interactions of editors with the content and with each other in a way that is (i) efficient and (ii) produces accurate representations of the socio-technical dynamics in digital collaborative writing platforms?*

  After devising and evaluating those models and algorithms in Chapter 4 for provenance and change tracking and in Chapter 5 for editor interactions, we are further interested in how to employ the mined data in making editing processes and content provenance more transparent for end-users, leading to **RQ3**.

- **RQ3**: *Which novel end-user tools, especially visualizations, can be built on top of the extracted data that provide casual readers as well as editors and scientist with a low-threshold, intuitive way to explore and understand the collaboration and content provenance dynamics in Collaborative Writing Systems?*

Two manifest forth and fifth research questions will not be answered in this thesis due to feasibility constraints, although they are certainly very related: **RQ4:** *Can we detect certain systematic behavioral dynamics (cf. RQ1) through revision history data with statistical models?* And **RQ5:** *Can we empirically link these patterns to specific quality changes in the content?* Although we cannot answer these question here to the extent they deserve, the methods and tools contributed in this thesis constitute the novel and necessary stepping stones to enable their exploration in the first place. We will discuss the further research agenda in Chapter 7.

1.2 FOCUS ON WIKIPEDIA AS THE USE CASE

We research the presented questions on the example of the English Wikipedia. The focus on one of numerous existing Collaborative Writing Systems (CWS) has several reasons:

- Of all revisioned CWS whose edit history (and other data) is completely and publicly available, Wikipedia is arguably the largest, in terms of data, documents, editors, readers and views – and has, if readership is taken as the measure, also the biggest societal impact, being the World's 6th most visited website overall.[4]

- A large amount of research on collective content production – and certainly the largest part of the investigation on digital collaborative writing – has been done on the English Wikipedia, hence providing the necessary theoretical and methodological

4 According to Alexa: http://www.alexa.com/siteinfo/en.wikipedia.org as of March 19, 2015

background for our research. Without this research foundation it would be hard to impossible to understand which particular behavioral patterns humans systematically exhibit when collaborating in a large CWS and to infer which mining methods should be developed to augment the revision history data.

- Tens of thousands of open online as well as intra-organizational Wikis worldwide are deployed, based on the same principle of collective contribution as the world's largest encyclopedia, thus being prone to showing similar collaborative writing mechanisms and likely to be suited to transfer the methods and insights provided in this thesis. To a large part, they even use the same software, MediaWiki.[5]

Although the answers to our research questions can vary to some degree in different systems, our approach is to first find answers in the confined environment of one platform to enable the possible transfer of these insights and methods to similar systems in future work. Hence, while certain phenomena we describe in Chapter 3 might in some cases be not directly transferable to other CWS, we believe that (i) it is reasonable that they exist in other systems as well in their general nature, as they are not overly specific for the English Wikipedia. Moreover, (ii) our main contribution, the models and algorithmic methods for mining and representing these social dynamics – derived in Chapters 4 and 5 – are surely applicable to other systems in general, given slight adaptations. We will discuss the general transferability and limitations of the answers to the research questions further in Section 2.3.

1.3 CONTRIBUTIONS AND STRUCTURE OF THIS THESIS

In order to answer the research questions, this dissertation contributes the following:

1. **A literature review and systematization** of (i) the existence of certain recurring social mechanisms in the collaborative writing and editing of Wikipedia articles and (ii) their possible effects on the collectively produced output in Chapter 3. This includes determining for each social mechanism which data is needed to identify and analyze it properly, an insight which motivated the development and design requirements of the algorithms that are proposed thereafter.

2. **A formal model to represent provenance and (dis)agreement in revisioned content, and the development of algorithmic methods** that help to efficiently and accurately mine the data

5 https://www.mediawiki.org/

needed to identify and analyze the previously described social mechanisms. This includes mainly:

a) **An efficient and reliable algorithm to determine provenance and change revisions of single text tokens** in the revision history of a document. Provenance – and therefore also authorship – attribution so far was either inaccurate (or mostly: not tested) and/or too inefficient to compute. The solution presented in this thesis is the first to be evaluated at over 90% correct attributions on average; it also increases computational speed by at least one order of magnitude while scaling very well to larger amounts of data, which is crucial for the application to big data like the complete set of all revision histories of all articles in the Wikipedia project, especially if the results should be continuously updated.

b) **Proposing an algorithmic method to increase accuracy when mining disagreement between users** from the article revision history. We show that our method can find 12% more "full reverts" than previous work and additionally enables the detection of "partial reverts". The knowledge gained is built upon in the interaction mining presented thereafter.

c) **An extension of the provenance and change tracking algorithm, which mines detailed "agreement" and "disagreement" relations between users** on top of the accurate modification attributions of individual text tokens. This is relevant as building editor-to-editor networks from content changes so far was either inefficient (as it relied on costly text difference methods) or much too coarse-grained (as it relied on simple identity revert detection).

d) **Experiments and surveys involving end users** to determine how certain interactions of users with the content and – via the content – with each other should be translated into explicit representations of social interactions via algorithmic methods. These insights informed the design of our models and algorithms.

3. **Working prototypes of Web-based visualizations for end users** that are built on top of the extracted data and which make editor-editor interactions, authorship and disagreement transparent. Also an Application Programming Interface (API) as a service to query for the provenance of words in an arbitrary English Wikipedia article on demand and the production of datasets for provenance attribution and editor interactions.

The main contributions of this thesis are undoubtedly related to research question **RQ2** and the models and algorithms for mining provenance and editor interactions. They therefore also receive the most coverage in the remainder. Before presenting the contributions in the above-described order, we will first give a short general introduction to Collaborative Writing Systems and the theoretical background in Chapter 2. The work will conclude with a summary and discussion.

1.4 IMPACT

The impact attained trough our contributions has several facets.

- *What:* **For researchers**, our work enables the extraction of **provenance and changes on world-level as well as fine-grained editor interaction data in the most accurate and efficient way yet**, even for large datasets (e.g., a whole Wikipedia language edition, including all article revision histories). This newly available data permits for a deeper investigation of the collaborative interaction patterns of editors than ever before, especially to investigate the causal relation between the appearance of systematic patterns of editing between individuals and article quality, but also the general study of interaction of editors with the content and each other at a level of detail that was not available previously for these large data sets. It can, to give just one example, help to better understand conflicts (e.g., edit wars in Wikipedia) that have so far only been researched via complete reverts of documents to older revisions and allows to explore the whole spectrum from small corrections over substantial content disagreements to outright disputes.
  By outlining several social mechanisms expected to be present in the Wikipedia editor community, based on behavior observed in related empirical work, we moreover make concrete suggestions on which crucial social dynamics to focus on in future research.

  For end-users (readers or editors): The data made available through our mining methods can be used in a variety of tools to enable **accountability and social transparency** of the often complex collaborative editing process. Foremost, these will likely be visualizations, such as the ones we developed and present in this thesis, but could also be other tools, like, e.g., automatically generated warnings or conflict indicators. In any way, by exposing content and editor interactions in detail and for large amounts of data, these tools have the potential to enable complexity reduction on-demand, not only for involved editors but even for casual readers or journalists. And they can even help

researchers to get a first intuitive, explorative look at the intricacies of the collaboration process. More "hands-on" applications are also probable, such as the attribution of certain parts of the content to its main authors, which is in some cases needed under the CC-BY-SA license[6] in Wikipedia, for example, and has not been easily extractable so far.

- *Where:* As mentioned previously, the algorithmic and visualization methods developed in this thesis can be **in principle be applied to any revisioned CWS**; yet the focus of interest of end-users and researchers will most likely be on the Wikimedia Foundation's projects.[7] Foremost Wikipedia.org and its different language editions will be application scenarios of our techniques, but likely also other Wikimedia projects like Wiktionary, Wikimedia Commons or even Wikidata, which is possible as these MediaWiki-based systems all use similar text-based markup to represent the data they display in the frontend. Larger Wiki communities (and research on them) like Wikia.com[8] as well as organization-internal Wikis are also likely to benefit from the application of our methods and tools. Repositories for collective code-writing, such as GitHub, provide further promising application scenarios; although code-writing follows different patterns for collaboration (as discussed in Section 2.3.2) than composing natural language documents, the principal technologies can still be applied to gain more insight into the interplay of users and content. These are the main application scenarios where we see the output of this thesis to have potential impact on the way the processes behind writing, editing and collaboration in CWS are made transparent and better understood. Yet, numerous revisioned CWS exist that could apply the insights and tools gained through this thesis, as we will discuss further in Section 2.1.1.

1.5 RELATION TO PREVIOUS WORK

Most of the content used in this thesis has been published in peer-reviewed conferences and workshops. Chapter 3 is to a notable extent based on the work "Towards a diversity-minded Wikipedia" published at the ACM WebScience Conference 2011 (full paper) [50] and

6 https://en.wikipedia.org/w/index.php?title=Wikipedia:Reusing_Wikipedia_content&oldid=669612767, asking for "a list of all authors" in specific cases, which is often approximated by listing the editors with the most changes, although these might not be the originators of most of the content in the re-used version; CC-BY-SA: http://creativecommons.org/licenses/by-sa/3.0/

7 https://wikimediafoundation.org/wiki/Home

8 A platform hosting Wikis as a service on the subdomains of http://wikia.com, currently (Dec. 11, 2015) ranked 103rd most visited website in the World by Alexa: http://www.alexa.com/siteinfo/wikia.com

to a smaller degree as well on "What Web Collaboration Research Can Learn from Social Sciences Regarding Impairments of Collective Intelligence and Influence of Social Platforms", a workshop paper at the "Harnessing the Power of Social Theory for Web Science" workshop at the ACM WebScience Conference 2013 [47]. Chapter 4 is based on "WikiWho: Precise and Efficient Attribution of Authorship of Revisioned Content", presented at the World Wide Web conference 2014 (full paper) [48] as well as to a much smaller part on its precursor "Whose article is it anyway? – Detecting authorship distribution in Wikipedia articles over time with WIKIGINI" presented at the Wikipedia Academy 2012 (full paper) [46]. The first part of Chapter 5, Section 5.1 is a close adaptation of "Revisiting Reverts: Accurate Revert Detection in Wikipedia" [51], which was a full paper at the ACM Conference on Hypertext and Social Media 2012. The subsequent Section 5.2 was mostly written for this thesis, although the model and algorithm already existed and were used and partly described in "whoVIS: Visualizing Editor Interactions and Dynamics in Collaborative Writing Over Time", which was a demonstration paper at the World Wide Web conference 2015 [49]. Lastly, Chapter 6 is based on several publications. The "whoVIS" paper [49] is the basis for Sections 6.2 and 6.3, describing the editor network visualization tool of the same name. The description of the "whoCOLOR" tool is based in equal parts on the bachelor thesis "User interfaces for tracing social editing dynamics in Wikipedia" by Felix Stadthaus I supervised [136] as well as "Towards Better Visual Tools for Exploring Wikipedia Article Development – The Use Case of 'Gamergate Controversy'", a paper at the "Wikipedia, a Social Pedia" workshop, collocated with the International AAAI Conference on Web and Social Media 2015 [52]. The latter paper also makes up the largest portion of Sections 6.4 and 6.5.

2 BACKGROUND AND MOTIVATION

In this chapter, we will elaborate on the theoretical background and thoroughly motivate our work.

2.1 ON DIGITAL COLLABORATIVE CONTENT PRODUCTION

Twenty years after its inception, the Web is *the* platform for the publication, use and exchange of information, on a planetary scale on virtually every topic; and it represents an amazing conglomerate of individual contributions and artifacts of collaborative production. The success of the Web can be attributed to several factors, most notably to its principled scalable design, but also to a number of subsequent developments such as platforms for user-generated content, smart mobile devices, and cloud computing. These trends are said to be responsible for a dramatical lowering of the barriers of entry when it comes to producing and consuming information online, leading to an unprecedented growth and mass collaboration. They have been empowering millions of users all over the globe to publish terabytes of multimedia content on sharing and networking platforms, communicate and exchange their points of view, and ask and answer questions, among many other activities – publicly expressing and sharing their ideas, knowledge and resources for the collective good [148]. One of the biggest promises of this human-machine apparatus we call the World Wide Web is the ability of social collectives to produce meaningful content through the online software systems they populate. They, for instance, gather and structure knowledge in encyclopedias (Wikipedia, Wikia.com) or question & answer sites (Stackexchange[1], Quora[2]), build software in Free/Libre Open Source Software (FLOSS) projects (Linux[3], Mozilla[4]), filter and rank current topics (Twitter[5], reddit[6]), and add meaningful metadata descriptions to content (del.icio.us[7], digg[8]), to name a few.

While some of these collective products are more or less straightforward aggregations of individual actions (the sum of upvotes on a reddit post, the stars on a product rating, the retweets under a Twitter

1 http://stackexchange.com/
2 https://www.quora.com/
3 http://www.linuxfoundation.org/about/about-linux
4 https://www.mozilla.org/
5 https://www.twitter.com/
6 https://www.reddit.com/
7 https://delicious.com/
8 https://web.archive.org/web/20120303014802/http://digg.com/

hashtag), others require a crowd of users to coordinate their efforts when coming together online to produce specific digital artifacts in a collaborative manner, such as the code for a piece of software or a coherent Wiki article about a certain topic. These can sometimes be exceptionally complex, and mostly require an ex-ante goal setting of at least a broad scope, to coordinate the work between all participants [99].

Because of the vast numbers of contributing users, the artifacts generated by collaborative platforms are plenty; while additionally, due to the "wisdom of the crowds" or the "many eyes principle" [143] that is assumed to guide and control meaningful output of these systems, they are often also trusted to be of high quality and importance, as can be well seen by the high consumption of knowledge extracted from Wikipedia, the wide usage of open source software like Mozilla's Web browser and email client or the Linux operating system [135].

2.1.1 *Collaborative Writing Systems*

One of the most popular forms of the above-mentioned, goal-oriented and collaborative content creation online is users writing and editing some kind of text-based document(s) together. Such document generation can be seen most prominently in public Wikis (e.g., thousands of encyclopedic projects at Wikia.com, or Wikipedia, which we will employ as our primary use case in this work) as well as in organization-internal Wikis, used in non-governmental organizations, clubs or companies to capture their common knowledge. Business solutions like Microsoft's Sharepoint or Office 365 are frequently implemented in intra-organizational settings and typically also allow for composing and coediting non-Wiki, but rather page-wise electronic documents together in a "what you see is what you get" (WYSIWYG) manner, imitating well-known single-user desktop office applications in their look and functionality.[9] Similarly, but rather used for semi-public or private closed-group writing projects, (office) document sharing like the free Google Docs or the freemium Zoho Docs allow to create textual files for a specific purpose.[10] An alternative to those "big brand" tools is to employ one of numerous spin-off products based on the synchronous writing platform Etherpad,[11] to name only a few of the solutions that are available nowadays. Dedicated (scientific) writing tools like Sharelatex[12] use markup languages (similar to Wikis), but produce "traditional", page-wise documents bounded by

9 See https://goo.gl/ezI5a0 and https://goo.gl/k2YPdU
10 http://docs.google.com, https://www.zoho.com/docs/
11 http://etherpad.org/
12 https://www.sharelatex.com/

a physical size. These examples give a glimpse into the abundance of solutions currently available and used in a broad array of settings.

What these tools offer is often referred to as *computer-supported collaborative writing* and is understood as a subfield of computer-supported collaborative work (CSCW) [11, 71, 132]. Although no canonical definition exists for *collaborative writing* (CW), a survey of existing work on the topic by Lowry et al. concluded that broadly, it can be understood as "an iterative and social process that involves a team focused on a common objective that negotiates, coordinates, and communicates during the creation of a common document"[13] [99], which encompasses "traditional" offline CW forms as much as those that are enabled through online platforms and tools. Here, we adopt this definition, as well as Lowry et al.'s denotation of *collaborative writing software*: "Software that allows collaborative writing groups to produce a shared document and assists collaborative writing groups perform the major collaborative writing activities". They further point out that "based on the desired writing task, CW includes the possibility of many different writing strategies, activities, document control approaches, team roles, and work modes", which we will in aggregation call *CW settings* here. The concrete writing task, e.g., an encyclopedic article vs. a work of fiction such as a novel vs. a piece of code, can be dubbed the *CW task*. Finally, each writing task can be carried out by a certain group of individuals, which we will denote *CW community*. Hence, we define a *Collaborative Writing System*[14] as a specific instantiation of a combination of (i) particular CW software, (ii) a pre-set CW task under (iii) specific CW settings, (iv) performed by a certain CW community. An example: A Wiki might be set up using the popular MediaWiki[15] software, for the task of recording work processes, by the members of the electrical engineering department of company X, given the setting of clear-cut roles (such as several writers under a manager that approves all changes) and a work mode that has team members write on distinct paragraphs before merging them into a coherent document. In contrast, we can have a distinct CWS, consisting of a set of documents in a public Etherpad instance, dedicated to recording financial accounting best practices, but open to the general public, with minimal rules and not having imposed on it any writing procedures.

Even FLOSS and many other software projects are commonly built through text-based code document sharing and collaboration, often

13 And continued: "The potential scope of CW goes beyond the more basic act of joint composition to include the likelihood of pre- and post-task activities, team formation, and planning."

14 While sometimes also called "Collaborative Editing Systems" [99] , we will solely use the term "Collaborative Writing Systems" in this thesis, which we take to encompass all acts related to writing a document together, including editing the article in ways that are not strictly considered "writing" content, such as deleting, spelling corrections and reverting to previous versions.

15 http://www.mediawiki.org

via repositories using a subversion or Git system, such as Github.[16] However, writing software code together deviates somewhat from writing a natural language document together. It does not fit neatly with Lowry et al.'s definition, as argued by Hill [71], as the contributors do not necessarily all follow a clearly defined, overall objective and create one common document, and secondly often work on code branches in parallel, later merging their contributions together; arguably, they are thus not "collaborating" in the sense of Lowry et al.'s CW definition or engaging in what he calls "reactive writing". We will therefore treat software-code writing as a special case of CW here and and discuss the relation further in Section 2.3.2.[17]

2.1.1.1 *Relevant Features of Collaborative Writing Software*

The software of such digital collaborative writing systems typically exhibits some characteristics we want to point out in the light of our research objectives and subsequent approach:

- A detailed revision (or: version) history of every state of the document resulting from each edit action taken by the individual contributors is recorded and made available; thus offering the possibility to recreate, for each document, the whole history of editor interactions with the content that led to its emergence and current state; as well as the ability to trace the recent ongoing editing processes and revert to a previous version.
- Notwithstanding that some separate, secondary channel for communication and coordination is usually present (commentaries, discussion/talk page or chat, associated with the document), the document itself is the central action and interaction space, with the secondary channels not necessarily used and not unconditionally needed for the document writing task, although they can be of importance to undergird the process.
- Although algorithms (and usually also interfaces) are featured for detecting textual differences between different revisions of the document and the editor that caused them, there are no explicitly encoded editor-to-editor interactions or relations recorded related directly to the main process of editing the document. For instance, while it is extractable which editors worked on the document at the same time and even the same part of the document, an explicit recording of who agreed or disagreed or

16 http://www.github.com

17 Note that Lowry et al.'s definition per se also excludes (i) crowdsourcing (micro)tasks or similar parallel contribution set-ups, where typically, the individuals don't know the document to be created as a whole and cannot react to it or other users' edits, i.e., the process is non-iterative and non-reactive, and (ii) tagging or labeling tasks, where no coherent documents but unordered collections of keywords are created and, arguably, no common specific objective exists.

"collaborated" with each other on which part of the document is lacking.[18]

- In most (e.g., MediaWiki, Google docs[19]), but not all cases (e.g., Etherpad), the single elements of the text are *not* explicitly annotated with provenance information at the time of writing, i.e., lacking information about the source revision of that specific element, which is linked to the author and the time it was created. Therefore, when retrieving a particular revision, this information about the provenance of single text pieces contained in that revision is not readily available and has to be ex-post extracted from existing revision histories.

- In the majority of systems, readers see preceding editors' actions at least in form of an updated document (plus optionally recorded activity logs), and are able to freely react to them by adding content or altering the effect of changes made previously. Users can thus truly *collaborate* on the content in the sense of being aware of, recursively reacting to and building on each others' contributions; hence, the software is supporting "reactive writing" and exertion of "shared control", as described by Lowry et al. in respect to CW strategies and CW control mechanisms [99].[20,21]

While the CWS that fit this description encompass many different systems, in this work, we will constrain ourselves to investigate the use case of Wikipedia and the MediaWiki Software it runs on, further elucidated in Section 2.3.

2.1.1.2 *Spread and Impact of Collaborative Writing Systems*

Multiple CWS have received considerable attention in academic research.[22] At the same time, the content produced on these platforms is ever increasing; and while also often used in intra-organizational or private settings, much of it is available in the public domain and

18 If present, explicit relations of editors are based on extra-document links that users exhibit either implicitly or explicitly on the collaboration platform, e.g., friendship ties of Google Plus accounts or talk-page entries of other users in Wikipedia. But they are not a direct product or representation of the editing process.

19 At time of writing, with not all internal features of Google Docs made public.

20 Like in the example of the intra-company engineering Wiki given above, the actual implementation of a non-centralized control and other CW settings is of course also dependent on deliberately taken (management) choices especially in such cases where the CW community is organized in some form outside of the CWS. This is however generally not the case in open online CWS.

21 Etherpad, Google Docs and related solutions even allow for real-time synchronous editing of a document. These interactions still can be seen as "micro-turns" in a technical sense.

22 7510 studies alone mention "Wikipedia" in their title, 6910 only "Wiki", and 1480 "Collaborative Writing" since 2001 in a Google Scholar search. ("allintitle:" search on https://scholar.google.com)

viewed (or otherwise used) by millions of users. As an example, all Wikipedia language editions together today feature over 35 million publicly available articles, receiving around 12 million edits each-month,[23] and the English version alone attracts around 9.5 billion page views each month,[24] being one of the World's top 10 websites by visits.[25] Its sister-project Wiktionary hosts over 23 million articles (233 million page views for the English edition)[26] and other Wikimedia projects such as Wikivoyage, Wikinews and Wikiversity reach similar impact. The for-profit platform Wikia.com hosts over 330,000 individual Wikis that attract over 140 million visitors per month.[27] Wikiindex.org further lists over 21,000 publicly accessible, active Wikis as of June 2015, with the number of non-indexed Wikis conceivably being even larger.[28] Also, many enterprises today are likely to employ one or several Wikis for internal knowledge management or other purposes. While the exact number has to be guesswork, a 2009 survey by consulting firm McKinsey revealed that over 60% of companies used Wikis, a number unlikely to have notably decreased since then.[29]

Public Etherpad (Lite) servers are hosted by a range of websites including the Etherpad foundation itself, each of which allows the free creation of any number of single Etherpad instances.[30] Numbers of actual usage are not available, but it can be speculated that numerous organizations employ Etherpad instances internally for collaboration. As it is open source, many software solutions include Etherpad into their architecture[31] or even build new systems on top of it.

Regarding cloud office solutions like Microsoft Sharepoint / Office365 and its main competitor Google Drive / Google Apps for Work (all including CW capabilities), a 2013 Gartner study found that "there were 50 million business people provisioned in whole or part with cloud office systems capabilities at the start of 2013"[32] and predicts that by 2022, 695 million users will employ such online productivity suites [147]. These quantities do not even include the informal, free usage of, e.g., Google Docs or Zoho Docs by possibly millions of private users such as students.

These exemplary numbers of a subsample of CW solutions suffice to showcase the enormous usage and impact of CWS. They also underpin how much log data on writing and collaboration behavior

23 As of June 2015: https://stats.wikimedia.org/EN/TablesWikipediaZZ.htm
24 As of June 2015: http://stats.wikimedia.org/EN/ReportCardTopWikis.htm
25 As of June 2015: http://www.alexa.com/siteinfo/wikipedia.org
26 As of June 2015: https://stats.wikimedia.org/wiktionary/EN/Sitemap.htm
27 As of June 2015: http://www.wikia.com/Wikia
28 http://wikiindex.org/Category:All
29 http://www.mckinsey.com/insights/business_technology/how_companies_are_benefiting_from_web_20_mckinsey_global_survey_results
30 https://github.com/ether/etherpad-lite/wiki/Sites-that-run-Etherpad-Lite
31 https://github.com/ether/etherpad-lite/wiki/Third%2Dparty%2Dweb%2Dservices%2Dthat%2Dhave%2Dsupport%2Dfor%2DEtherpad%2DLite
32 https://www.gartner.com/doc/2492216/new-developments-cloud-office-market, reported in [147]

exists, recorded along with the produced textual content, that has the potential to enable unique research insights on a large scale.

2.2 THE RESEARCH AFFORDANCES OF COLLABORATIVE WRITING SYSTEMS

In the wake of the exceptional success story of digital collaboration platforms for content production – and CWS in particular – emerge great challenges associated with making sense out of the sheer amount of information continuously being generated through the interaction of human agents with each other and with the software systems they inhabit [142]. This holds particularly for understanding which socio-technical dynamics are responsible for which emerging structures, processes and artifacts of online collaboration. Therefore, when many users gather online to purposefully create and refine specific digital documents together – e.g., public Wikis and FLOSS projects as opposed to platforms where, arguably, the main goal is to socialize (Facebook)[33] or trade goods (eBay)[34] – we see two broader research affordances standing out, regarding the social content-creation aspect of CWS: The large amount of detailed data on how human agents interact in a CWS with each other and the documents they are creating can help us **(i) understand the dynamics of the collective process by learning about the presence of typical social patterns and mechanisms in human online collaboration behavior**. And more specifically, it can enable **(ii) understanding which typical social behavioral patterns lead to what (positive or negative) outcome in terms of a desirable end-product of the collective effort** (in our case: textual documents). Id est, the first view may lead to a finer understanding of human collaboration and work behavior online (and maybe even in general); while the latter point can serve to improve the concrete socio-technical systems in question in terms of usability, efficiency or overall quality of the produced artifacts if systematic harmful (or beneficial) behavioral patterns can be identified and prevented (or supported). We will discuss these two perspectives below.

2.2.1 *Research Potential 1: Finding Systematically Occurring Social Mechanisms in Collaborative Writing*

We will base our research approach on the concept of "Social Mechanisms" to explain why middle-range theorizing about systematic cause-effect relationships between individuals is of interest in CWS.

33 http://facebook.com
34 http://ebay.com

2.2.1.1 *A Brief Excursus: Social Mechanisms*

Peter Hedström and Richard Swedberg, inspired by the middle-range sociological approaches first promoted by Merton and Lazarsfeld [109, 92], made the case for the concept of "Social Mechanisms" underlying many higher-level social theories or patterns: "there are general types of [social] mechanisms, found in a range of different social settings, that operate according to the same logical principles" [69]. As an illustration, they list the "belief-formation" mechanism which they see at the foundation of several (macro-level) social phenomena: (i) the self-fulfilling prophecy [108] (e.g., a collective bank run out of fear of money shortage), (ii) network diffusion [30] (e.g, the adoption of a new drug in a network of physicians) and (iii) threshold-based behavior [62] (e.g., passer-bys propensities to visit a restaurant based on already present guests). In all cases, Hedström and Swedberg argue, the strength of the belief of an individual that the action considered (withdrawing savings, adopting a drug, visiting a restaurant) at time t is a function of the observed number of other individuals having carried out the action at time $t-1$. Hence predecessors always signal to successors the ostensibly correct conduct, which they have a probability to adopt as a blueprint for action that increases with the amount of others observed (especially with no pre-formed beliefs and little personal information). This mechanism, also called "social proof" by Cialdini [29] (and related to the concept of "information cascades" [14]), underlies these and many other theories about behavior on the macro level. Territorial defense (the individual owns resources and/or territory and reacts with marking and active aggression against intruders) [128] or the bystander effect (given the impression that enough potential helpers are present, no individual takes steps to help in an emergency situation, as she feels her marginal contribution is unnecessary) [26] can be given as further examples of social mechanims.

Gambetta [56] illustrates the analytic thinking pattern behind this approach with a study where the superficial "causality" between the increasing age of a student's father (independent variable) and the decreasing likelihood to stay in school (dependent variable) was clearly visible in all statistical models, even if controlled for many other explanatory factors. For most statistical ends and purposes, this level of "cause and effect" would suffice. Yet, just looking at variable interdependence does not explicate the *mechanisms* that explain *why* this phenomenon happens and what exactly it is composed of at an inter-individual level. Gambetta investigates and deduces that his original suspected mechanism (paraphrased) "Increasing age of the main provider diminishes projected income for the family (cause), therefore a child is asked to support the family by working and dropping out of school (effect)" was a major – but only one of several – mechanism at play, and mechanisms were differing notably for genders

and social classes, in aggregate culminating in the observed variable correlation.

These samples from related literature serve to exemplify that for a given social phenomenon at a larger scale, a social mechanism at the inter-individual level may (and probably will) interplay with other social mechanisms and might, depending on the individuals and situation, produce completely different results in aggregate [56]. As a bottom line, this approach aims to model social reality on a lower, more generalizable level than grand social theories, without delving into the intra-personal, psychological aspects, but staying at a inter-individual level, and do so with a clear *how* and *why* explanation of a cause-effect relationships.

2.2.1.2 *Social Mechanisms in Collaborative Writing Systems*

Given that social mechanisms describe fairly basic social interaction (or often: reaction) patterns, commonly comprised of heuristics to chose an optimal course of action in a given situation, we argue that such mechanisms exist of course as well in the space of digital collaborative writing and that we can employ this analytical approach to obtain a better understanding of the inner workings of CWS. Even more so, with the constrained action space of adding and deleting text (plus some secondary discussion and message space) and the clearly outlined frame of goals (i.e., the subgoals of creating and refining documents), the complexity in terms of number and nature of potential mechanisms should be reduced compared to more open human interaction scenarios. Yet, we expect to find certain social mechanisms also known from offline, more general contexts, such as territoriality or social proof mechanisms.

By employing this approach, we do not focus on the whole system, respectively macro level, of a CWS (e.g., user activity development on the whole Wikipedia), but on the middle-range arena of individual actions: the single documents to be created and refined and the actors and actions that shape them. While being influenced by the macro level and retroacting on it, basic article-level social mechanisms have the likely tendency to systematically appear (e.g. defending self-built content, extensively evaluating content by unknown actors). And since they should be generalizable to a considerable extent – as they always emerge from (i) human nature, (ii) in a CW software environment, (iii) with the common goal of writing documents together – such mechanisms might well not only be transferable between document building processes in one system, but also between distinct CWS.

To this end, in Chapter 3, we will present some of the social mechanisms we identified and extracted from research literature on Wikipedia and discuss what data would be necessary to model them statistically.

2.2.2 *Research Potential 2: Social Mechanisms and Quality*

Given the enormous user base of CWS and the trust users often put in the retrieved information [100], understanding the dynamics affecting the quality of the individual emerging products of collaborative text-creation projects is an issue of *societal relevance* in regard to publicly accessible platform content. This is especially true in the case of Wikipedia, that has become one of the primary go-to sources for knowledge for large parts of western civilization (e.g., college students [67, 80], medical practitioners [12], as well as the general population[35]), but also for many other public Wikis. The relevance is underlined by findings that public knowledge repositories like Wikipedia have been shown to exhibit great lacks of quality in some parts (notwithstanding their excellent quality in others) [38, 58, 86, 96]. Aside from public interest, organizations such as companies of course have their own concern for intra-organizational quality control and optimization for their internal collaboration systems, a fact that requires little underscoring and has been documented extensively (e.g., [35, 98, 170]). It is thus desirable to gauge the quality of the collectively produced content as accurately as possible to alert an unsuspecting reader or editor of possible flaws or to even deploy countermeasures that amend the erroneous material.

One major question in respect to the quality of collaboratively produced documents (especially by large userbases in open online systems) regards the "collective intelligence" whose assumed presence is the basis for much of the trust placed in the output of CWS [77, 85, 100, 143], particularly if no external quality management is present (like in the vast majority of open, online CWS). Despite the often high confidence in the content exhibited by its eventual consumers, research has shown that this "wisdom of the crowds", generally and in digital collaboration, is a fragile thing that hinges greatly on the behavioral dynamics the human contributors exhibit, shaped not only by the composition of the user-base itself, but also by the environment the software system provides [53, 77, 85]. Certain dynamics can lead to unwelcome results of the collective process, just like it is the case for offline scenarios, covered by decades of research on the emergence of harmful interaction patterns in certain populations, resulting in a vast body of scientific work aiming to understand, predict, and even prevent the occurrence of such phenomena. These include mass hysteria and panics, stock market bubbles and disease spread; the mechanisms at work have been tried to explain with the help of

35 http://www.pewinternet.org/2011/01/13/wikipedia-past-and-present/, http://www.ard-zdf-onlinestudie.de/index.php?id=502

organizational theory, social imitation theories and psychological approaches, to name only a few.[36]

We suspect that social mechanisms in writing digital documents together can have homologous negative effects when they "turn bad". To wit, social proof and "rational imitation" [68] are initially useful heuristics to gauge market opportunities, but if each individual acts by the same social mechanism, stock market bubbles might be a probable outcome. And just as likely could ignoring one's own personal information or cognitive resources – but relying on the group decision – when writing or revising content lead to negative outcomes when this mechanism triggers for all involved users. When building a document, a strong sense of commitment and attachment to the content by one or a few users might result in a beneficial guardianship, while with another document, triggered by certain social cues, it might turn into an embargo of "outsider" contribution, shutting out valuable information – yet, the underlying social mechanism would be the same. In Chapter 3 we will thus not only expand on this and other examples of social mechanisms we identified in Wikipedia but also on how they can have negative effects on document quality. There, we will also discuss what "quality" and its impairment mean in each case, as the term is not unambiguous.

2.3 BASICS OF – AND FOCUS ON – WIKIPEDIA

In the remainder of this work, we will focus on the English Wikipedia as our use case. In this section, we (i) recapitulate the basics of Wikipedia as a CWS as well as (ii) explain the decision for focussing on it more in detail and elucidate how representative (the English) Wikipedia can be for other CWS.

2.3.1 *Basics of Wikipedia*

Wikipedia is a top-ten Web site providing a community-built encyclopedia for free.[37] We have already discussed some of the enormous impact of Wikipedia in Section 2.1.1.2, as well as its wide coverage by research. The English edition, apart from receiving over 9 billion pageviews, also attracts over 3 million edits per month and hosts over 4.8 millions articles. It is the largest among the over 280 language editions, in articles, users and almost every other measure. This makes the English Wikipedia the single largest online CWS in existence for which nearly all data is openly available, including all edits ever made by users, all revisions and articles created and much more data on

36 A plethora of research work exists on these and many related topics. In the scope of this work we cannot introduce them in their entirety. See [107, 106, 159, 137] (panics), [118, 1, 133, 9] (stock market) and [44, 111, 112, 28] (disease spread) as examples.

37 Measured by visitors: http://www.alexa.com/siteinfo/wikipedia.org

activities carried out on the site. Its status as a prime scientific investigation subject should thus be apparent.

The completely open contribution possibilities for the general public (everyone can edit, even without an account) makes it predestined for studying social dynamics in a CWS. The English Wikipedia allows completely free editing (edits by any user take immediate effect) in contrast to some other language editions like the German one, where edits have to be cleared by senior editors.[38] Also, the users themselves decide what to edit when and how. There is no structured revision process for edits implemented, although notifications via the so-called "watchlist"[39] and other means are available to users for following the changes to specific articles. Often, however, edits are also made because readers simply stumble upon text passages they find change-worthy and thus correct them – sometimes to the disagreement of others. Users can easily "undo" changes by reverting back to an older version of the article.

Further, the features for CW software we outlined in Section 2.1.1.1 all apply to the MediaWiki Software that the Wikipedia projects run on, including that no word-level provenance and changes or editor-to-editor interaction data is explicitly tracked. Additionally, Wikipedia has the following characteristics:

- In regard to the CW task, the content to be generated for each article is a non-fiction, informative documentation of factual knowledge from secondary sources on a specific topic, written in natural (i.e., human-readable) language, in an encyclopedic style. Concurrently, non-fictional writing is commonly defined as "any informative work [...] whose creator, in good faith, assumes responsibility for the truth or accuracy of the events, people, and/or information presented".[40]

- As for the CW settings, the writing project is not overseen systematically by a central authority but is generally conducted and controlled by the editors that carry out the editing and self-organize.[41]

- Although technically everybody can be an editor, research has shown that the CW community of Wikipedia covers only a

38 Note that while this holds for most of the articles on the English Wikipedia, there are some articles that are under special protection, meaning that they can only be edited by registered users or even users with special rights. These are however a minority of the overall article body. Article creation in the English Wikipedia is also restricted to registered users.

39 https://en.wikipedia.org/wiki/Special:Watchlist

40 According to https://en.wikipedia.org/wiki/Non-fiction#cite_ref-litfiction_1-0, paraphrased from [45].

41 Although some internal systematic organization emerged over the years (admins, policies, committees, etc.). Yet, no external organizational structure (like a company owning the Wiki) is interfering with the community's efforts. The Wikimedia Foundation has a strict policy of not getting involved in any article writing issues.

narrow section of the offline-population's socio-demographic scope. Section 3.1.5 elaborates further on the most notable imbalances.

Further, we will introduce some basic terms denoting functions and entities in Wikipedia that will be mentioned in the remainder:

- Article: The main documents (which we will deal with mostly in this work) in Wikipedia are articles. Each article has an article talk page that can be optionally used for discussion article content and changes. There are also user pages, Wikipedia "meta" pages for rules and other types of pages we will not address here specifically.
- Edit: An edit denotes the action of changing the wikitext of a specific article and then saving the changes. An edit always creates a new revision of the article. An edit can, in the basic conception, include only four actions: add text, delete text, replace text and/or move text somewhere else in the article.
- Revision: A revision is a version of an article. Each edit creates a new revision, that is, edit e creates revision i, edit $e+1$ creates revision $i+1$, and so on. Revisions are therefore sequential and an edit creating a revision i is always performed on revision $i-1$.
- Revision history: The revision history is a list of sequential revisions, ordered by timestamp of creation, annotated with the editor and other metadata. A visual representation of the revision history is also linked atop each article in Wikipedia's interface.
- Revert: In the official Wikipedia guidelines the definition of a 'revert' reads as follows: *"Reverting means undoing the effects of one or more edits, which normally results in the page being restored to a version that existed sometime previously. More broadly, reverting may also refer to any action that reverses the actions of other editors, in whole or in part."*[42] We will build on this definition in Section 5.1.
- Wikipedia diff (text difference): A Wikipedia diff denotes the result of using MediaWiki's internal text differencing algorithm for comparing two revisions. It is also available to users in the graphical interface, showing removed and added text pieces from one revision to the other according to the internal algorithm.
- Editors: Users performing edits. These can be registered users – in this case, their user name is recorded in the edit logs – or

42 http://en.wikipedia.org/wiki/Wikipedia:Reverting, also compare http://en.wikipedia.org/wiki/Help:Reverting (accessed 13.09.14, italics added).

unregistered users, these are only identified by their IP address at the moment of editing.

- Authors: Editors adding text, not only moving or deleting text.
- Wiki markup-text (or Wikitext, or Wikicode): The textual representation of an article that is parsed by MediaWiki to produce the front-end HTML shown to the readers. It consists of the textual content in the front-end plus Wiki markup characters that use a specific syntax to denote particular transformations into HTML. E.g., double square brackets around a word will result in a link to a Wikipedia article page of the same name in the HTML version.[43]

2.3.2 *Wikipedia as Our Use Case for Collaborative Writing Systems*

We see Wikipedia as the best use case for CWS on which to demonstrate the applicability and usefulness of the methods and tools presented in this thesis. The main reasons for our focus on Wikipedia have been laid out in Section 1.2. First, we have already outlined why Wikipedia is a worthwhile research subject by itself, and that insights gained about the social side of Wikipedia have their own merit, as the plethora of research on the platform proves. Second, we think that many of the insights gained in this thesis are transferable: Wikis as knowledge management tools arguably make up one of the largest shares of CWS in use today (cf. Section 2.1.1.2), and given that the underlying technology and technical procedures for creating content and interaction are alike in most Wikis and Wikipedia, we are confident that many of the lessons from this research can be transferred. We believe however, that at least for systems (including Wikis) used for fictional writing, the Wikipedia credo of "finding and describing the truth" does no apply to Wikis and other CWS and likewise, the processes of disagreement and content negotiation do not work in the same way and neither will many social mechanisms. Also, particularly intra-organizational Wikis and CWS, while exhibiting similar CW tasks and software, can follow very different CW settings and have very distinct communities. While some social mechanisms we will explain in the following chapter might occur in the same way in, for example, a company-internal Wiki (e.g., a territorial behavior regarding specific content), many social effects might be transformed or mitigated by the much stricter formal and organizational structure of a business as well as the type of users that operate in it.

Similarly, as hinted at in Section 2.1.1, collective code writing differs from collaborative "human-parsable" text writing in several aspects. Most prominently, software projects can exhibit a rather hierarchical planning mode as well as a modular and parallel contribution

43 Cf. https://en.wikipedia.org/wiki/MediaWiki#Markup

process especially when carried out inside organizations with a predefined chain of command; but also in FLOSS projects such as Linux that exhibit clear planning structures [71, 90]. Modular parts of the code are often assigned to specific programmers/editors, making the coding process less of a constant "reactive" endeavor, but a clear division of labor [34]. Even for unorganized open projects, e.g., private open source code on GitHub, work is commonly carried out in parallel and later merged, not conforming with collaborative writing by Lowry et al., as we mentioned already in Section 2.1.1. Also, the negotiations between users about what constitutes "correct" or "incorrect" content are certainly carried out on a different semantic basis then it is the case in encyclopedic articles or other natural language documents: software code is often much easier to prove to be sub-optimal or wrong than natural language facts, and social processes with many participants to determine factual correctness or optimal writing styles as complex as in Wikipedia are unlikely to be needed.

For digital page-wise office document writing in CWS, such as Google Docs and professional solutions like Microsoft Sharepoint, we also see differences for some CWS in CW software, tasks, communities and settings that might impair the transferability of the results of this thesis.

Yet, for most of our contributions presented hereafter, we are confident that applicability is possible to many other CWS and scenarios and that they will provide new ways to shed light on the collaboration behind the scenes of CWS. This applies specifically to most of the technological advancements we describe in Chapters 4 to 6, i.e., our approaches to mining authorship and interaction as well as visualizations. We will come back to the transferability discussion in the conclusion.

3

SOCIAL MECHANISMS IN COLLABORATIVE WRITING ON WIKIPEDIA

In Section 2.2 we explained why it might be of interest to find social mechanisms in the collaborative writing process of documents to (i) gain general insights into the collaboration behavior of humans in CWS and (ii) identify such mechanisms that can negatively influence the quality of the document on *article level*. In this chapter, we provide a collection of indicators for the existence of such mechanisms and their influence on the resulting content, which we extracted from research literature on the English Wikipedia. There has been a fair share of research regarding social interaction, editing behavior and collaborative content production in Wikipedia that we can draw from.

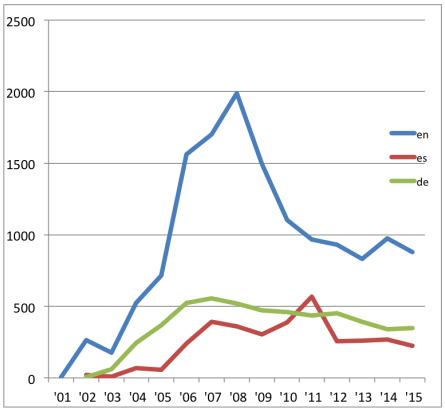
We concentrate here on the *main* recurring editing and collaboration mechanisms we found in the research literature that have the potential to have notable influence on the collective content production and vetting process. These mechanisms have often emerged as heuristics to deal with the challenges of maintaining Wikipedia and collaborate efficiently. This means that they originally serve as beneficial strategies in response to the socio-technical environment. Yet, they might "turn bad" and impair article quality in certain contexts and should therefore at least be made effectively transparent for inspection by readers, editors and researchers. Such transparency motivates the need for better data mining and representation, including visualization methods for exploring the social processes underlying Wikipedia. We will therefore, for each mechanism, outline what data would be needed to model the social and content interactions for a quantitative analysis.

Before listing the single mechanisms, starting from Section 3.2, we will give a brief introduction into dynamics that have been observed in the English Wikipedia as a whole, to provide some background information.

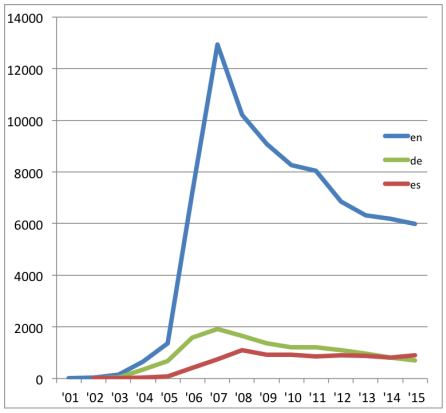
3.1 PREFACE: GENERAL DEVELOPMENT

Some "circumstantial evidence" exists of patterns in article and usage data on the system-wide level that could be interpreted as *symptoms* caused by the social mechanisms occurring between editors on the meso-level that we will describe in the following.[1] They give, in any

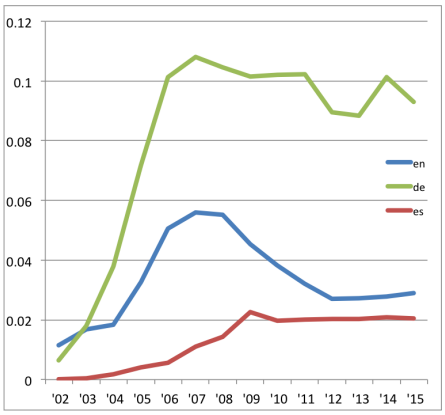
1 I.e., these observations have been made for the platform as a whole but can vary in their extent between articles or categories.



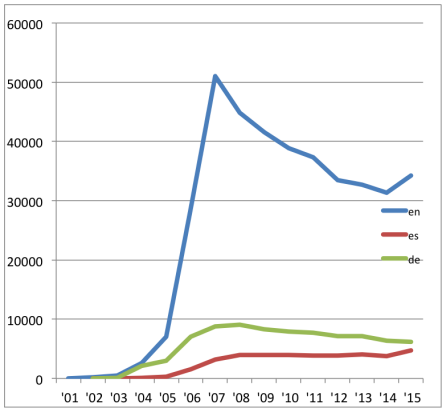
(a) New articles per day



(b) Successfully retained newcomers (Wikipedians who edited at least 10 times since they registered)



(c) Reverts per edits (percentage)



(d) Active wikipedians: Wikipedians who contributed 5 times or more in a given month

Figure 1: **Development of key metrics over time** of the English Wikipedia (blue), with the Spanish (red) and German (green) language versions given for comparison (not referenced in text). X-Axes steps denote years.[2]

case, a frame for interpretation of the mechanisms discussed further below.

3.1.1 *Less Articles and Edits*

As can be seen in Figure 1a, the number of new articles created has been decreasing dramatically or at least stagnating during the last years, a phenomenon most notably visible for the English and German versions of the encyclopedia, which are considered to be "mature" in terms of overall article count and development of the project.[2]

2 Figures are based on http://en.wikipedia.org/wiki/Wikipedia:Modelling_Wikipedia_extended_growth and http://stats.wikimedia.org.

Although especially the English Wikipedia has shown exponential growth in the past [24, 83, 154, 163], since 2008 this trend has been declining, not only in articles created, but also in the number of edits and editors per month, which stagnated and started shrinking in a similar way.[3]

Suh et al. [140] discuss this growth decline in the English Wikipedia and explain it by comparing Wikipedia to a kind of ecosystem which has reached a state of matureness where many articles are close to complete on a factual level. Accordingly they hypothesize that the decline can be attributed to "(a) the slowing growth of the editor population due to limited opportunities in making novel contributions;" – this would mean that readers encounter (obvious or easy) chances for contributing less frequently – "and (b) increased patterns of conflict and dominance due to the consequences of the increasingly limited opportunities".

3.1.2 *More Edits Reverted, Failing Newcomer Retention*

As support for their increased-conflicts hypothesis Suh et al. [140] point out that the number of reverted edits significantly increased in the English Wikipedia from 2005 to 2008, with occasional and new editors experiencing greater resistance compared to high-frequency ones. Halfaker et al. [64] corroborate this trend, and point out that the contributions of *desirable* newcomers (non-vandals, contributing useful content) have been increasingly more reverted from 2006 to 2011. Reverts relative to edits have indeed been increasing over the years in several Wikipedia editions (cf. Figure 1c; the English Wikipedia showed a decline in relative reverts after 2008, but as a general trend, the revert rates are increasing or stagnating.)

Halfaker et al. [64] have also shown that being often reverted is a contributing factor to the decrease in users trying to attempt changes at all, and makes desirable newcomers leave for good (Cf. Figure 1b). They empirically demonstrate that during Wikipedia's exponential growth phase up to 2007, "the rate of rejection for edits made by desirable newcomers rose and the survival rate of desirable newcomers fell" [64]. This means that not only more malicious users and vandals have been targeted by reverts, but good-faith, desirable editors as well. Lastly, newcomers often just do not manage to "fit in" – even without being reverted – as they either do not find work to do, or are simply not adequately introduced into the (often complicated) Wikipedia culture by senior editors [64].

3 Not shown, cf. https://stats.wikimedia.org/EN/SummaryEN.htm

3.1.3 *Increase of Rule Pages and Governance Work*

Research has shown a shift of work (expressed by the number of edits) to coordination, policy setting, and governance in the English Wikipedia [83], resulting in a surge in the amount of explicit rule and procedure pages – arguably converting Wikipedia into somewhat of a "bureaucracy" [22, 55] – but at least giving it more of an organizational and hierarchical character than a completely open collective treating every user equally. This growth continues steadily, for official guidelines and policies and even more so for hard-to-enforce "essays" that are a rather informal way of explicating norms [64]. Rule creation and enforcement are quite decentralized [13], but with the slowed-down growth after 2007 took a turn to more centralization by allocating interpretive primacy and implicit change permissions to the more senior editors [64].

3.1.4 *Concentration*

Research indicates that the widely observed phenomenon of a small minority of overall users [105, 162] providing most of the contributions [154, 156] and content [82] to an online collaboration system holds true for Wikipedia as well. For instance, 0.5% of all editors provide 74.3% of all article edits in the English Wikipedia.[4] Priedhorsky et al. [122] did early work related to concentration of authorship in articles,[5] indicating that 86% of word-views are attributed to words from the 10% most frequently editing users.[6]

3.1.5 *Population Bias*

Research findings suggest that the editors actually contributing Wikipedia's content are not socio-demographically or mindset-wise representative of society in general, of the average Internet user, or even the average reader of Wikipedia. For instance, the UNU-Merit Wikipedia Survey [59], an online survey conducted at the end of 2008 revealed that, among other results, of all 176,192 Wikipedia users interviewed, 65.9% were mere readers, 23.3% were occasional contributors and only 7.4% were regularly editing the encyclopedia. Less than 13% of contributors were female, some countries such as Germany were vastly over-represented in terms of number of editors in relation to their population size and 75% of the respondents were under 30 years old. 46% of those who contribute had undergraduate

4 https://stats.wikimedia.org/EN/TablesWikipediaEN.htm#editdistribution

5 Aaron Swartz did a somewhat related analysis on a smaller sample of articles, based on Python's basic longest-sequence matching [146].

6 "Persistent word views" was defined as "the number of times any given word introduced by an edit is viewed". The method was however not tested for accuracy or efficiency, cf. Section 3.3.3.

or higher tertiary education. A strong self-selection process in certain subpopulations is hence present in regard to joining Wikipedia as an editor; and it is very likely also present when becoming a top-contributor, exacerbating potential bias-effects. It appears that while the openness of an online collaboration system such as Wikipedia is a pre-requisite for the diversity of its contributing users, self-selection can certainly lead to a lack of plurality inside the system [93].

3.2 LEARNED TERRITORIAL DEFENSE BEHAVIOR, THREAT HEURISTICS, AND XENOPHOBIA

The social mechanism we describe in the following is based on two main concepts: learned territorial defense behavior and xenophobia.

Thom-Santelli [149] gives an overview of basic territorial behavior: "From the perspective of socio-biology, anthropology and geography, territoriality is strategic and is defined as an individual or group's attempts to influence or control animals (including people), phenomena and relationships within a territory (Ardrey, 1968; Sack, 1986). Territoriality serves a spatial- organizational purpose (Dyson-Hudson Smith, 1978) [...]. In this context, actors assert territoriality as a means of control and the maintenance of power by limiting access to a territory, particularly when it contains valuable resources (Taylor, 1988)". While in a traditional socio-biological sense, the defense of territory is aimed at securing resources for the defenders (inhabitants), in digital collaborative production systems, it is foremost aimed at the digital artifacts, which may have different meaning or value to the defenders [151, 150] but are generally not sparse. Further, in the specific case we describe here, we argue (a) that the territorial behavior is merely adapted as a *defense against harm* to the digital artifacts and (b) that it is *learned* by members of the defending group by witnessing harmful attacks (or trough second-hand reports by other users).

Xenophobia is, in its most basic and literal meaning, the "fear and hatred of strangers or foreigners or of anything that is strange or foreign" (Merriam-Webster dictionary [167]). Xenophobia usually has its origin in an individual decision process that is personally believed to be rational (e.g., a specific group of people poses a danger to personal security), but in its unreflected generalization becomes irrational [127]. As such, defensive rules of conduct against outsiders might, once introduced, not be reassessed for their actual value for the defending community. Xenophobia is naturally linked to territorial and defensive behavior.

3.2.1 *Suspected Dynamics on Wikipedia*

The gain in reverts we could observe in several language editions (compare Section 3.1.2), especially between 2003 and 2007, can to

some extent be the result of an "article defense behavior" editors exhibit, particularly for those articles on their "watchlist",[7] which involves a general wariness, and often rejection, of *seemingly untrustworthy contributors*. This repudiation generally stems from past experience: Wikipedia articles are very often the target of vandalism, including all kinds of malevolent edits.[8] It has been estimated that about 7% of all edits to Wikipedia are vandalism [41]. Secondly, spammers[9] frequently attempt to promote certain products or services through setting hyperlinks or favorably altering content [161]. The amount is so high that as early as 2006, as Goldmann reports [60], editors were asked by Wikipedia's legal council to "shoot on sight" when seeing spammers, as "outsiders were increasingly using Wikipedia for promotional ends" [7]. Lastly, "trolls" are mischievous users that either want to test Wikipedia's vigilance in terms of incorrect content or just find entertainment in placing misleading or incorrect information in articles, observing how long the (often well-disguised) misinformation survives [131]. Such, in summary, "disruptive edits" are very frequently carried out from unregistered IP addresses or new accounts, at times so-called "Sock Puppets", created for just that purpose.[10]

A 7% rate of vandalistic edits – plus other, less obvious disruptive changes such as trolling, spam or paid editing – results in a vast amount of total edits to be filtered out at around 3 million edits per month, with ever-fewer Wikipedians to do the work (cf. Figure 1d).[11,12] As a result of this experience, users are likely to acquire straightforward heuristics to swiftly pre-classify and accordingly interpret edits if certain characteristics of the editor are fulfilled. This includes most prominently checking if the account is new (or simply unknown), rarely used or if the editor is unregistered, and maybe even browsing the account's edit history [60, 20]. Content-related heuristics might comprise certain general text patterns, such as linking to apparent commercial sites or using particular (inappropriate) vocabulary. Certain vandal-typical behaviors can be used as markers

7 Many regular editors patrol articles by adding them to their watchlist, a feed of all changes to articles the editor has subscribed to. See https://en.wikipedia.org/w/index.php?title=Help:Watchlist&oldid=676506955

8 "Vandalism is any addition, removal, or change of content, in a deliberate attempt to damage Wikipedia.", according to https://en.wikipedia.org/w/index.php?title=Wikipedia:Vandalism&oldid=692313087

9 "There are three main types of spam on Wikipedia. These are: advertisements masquerading as articles; external link spamming; and adding references with the aim of promoting the author or the work being referenced", according to https://en.wikipedia.org/w/index.php?title=Wikipedia:Spam&oldid=688415022

10 Sock Puppet Accounts (SPA) are accounts registered aside from the primary account of a user, often to carry out incognito edits reinforcing her own position in a discussion or edit conflict; seemingly from a third party. Cf. https://en.wikipedia.org/w/index.php?title=Wikipedia:Sock_puppetry&oldid=682708900

11 As of March 2015, http://stats.wikimedia.org/EN/TablesDatabaseEdits.htm

12 Bots are nowadays employed to help shoulder the workload: http://stats.wikimedia.org/EN/PlotsPngEditHistoryTop.htm

for malevolent intents, such as not discussing changes before editing or not (appropriately) commenting edits. Using such rules of thumb, which can be quite simple at times, the workload of filtering damaging edits of any kind can be more efficiently tackled by not spending excessive time on analyzing the content of each edit in detail, while still successfully intercepting most damage intents.

The implementation of some of these rules as explicit technological barriers support the notion that such heuristics are in fact widespread in the community: Blocking of certain IP ranges[13] and restricting the creation or editing of articles to only auto-confirmed users[14] are the most prominent examples. Yet, although in recent years these technical restrictions as well as anti-vandalism bots and semi-automatic tools have been increasingly and successfully deployed on Wikipedia to complement manual labor [57], most spam and troll edits as well as many vandalism cases still have to be inspected by hand and the general heuristics-based social mechanism described previously is likely to remain in place; the manual "cleanup" workload is still high and the main filters remain "social, not technological" [60].

It can be theorized that with increasing exposure to such edits and experience in fighting them, "patrolling" editors get more suspicious of edits from unknown accounts, reinforcing their defensive behavior against such contributors, and generalizing the use of their heuristics especially under high workload.

3.2.2 *Potential Harmful Consequences*

Like every heuristic, identifying malevolent or merely low-quality edits by user account information or rather superficial content and conduct features entails the risk of false-positives, i.e., reverting good or at least *good-faith* work of editors, as some (especially new) editors might be acting in ways contradicting the behavioral guidelines of Wikipedia.[15] At the same time, editors might adapt their perception of what a "disruptive" edit is and deliberatively reject also obviously well-meaning, but underperforming newcomers in an attempt to exclude editors that "do not know what they are doing" [150].

The defensive revert reaction becomes harmful when it is applied to a more general set of users and more precipitously so, an effect aggravated as more perceived attacks are being repelled. Goldman describes such reinforcing defensiveness among editors as culminating in *xenophobia* [60]. Goldman writes: "Due to the constant threat of spam and vandalism, some Wikipedia editors become socialized to

13 https://en.wikipedia.org/wiki/Wikipedia:Database_reports/Range_blocks

14 Registered accounts at least 4 days old and with at least 10 edits: https://en.wikipedia.org/w/index.php?title=Wikipedia:User_access_levels&oldid=693983853#Autoconfirmed_and_confirmed_users

15 See specifically: https://en.wikipedia.org/wiki/Wikipedia:Assume_good_faith#Good_faith_and_newcomers

assume that site edits are made by bad folks for improper purposes, thus developing a 'revert first' mentality", and further notes that "the adverse presumptions especially apply to unregistered or unsophisticated users who do not comply with Wikipedia's cultural rituals [...]. By failing to conform to the rituals, these contributors implicitly signal that they are Wikipedia outsiders, which increases the odds that Wikipedia insiders will target their contributions as a threat." [60, p.168]. Especially not being logged in makes one the target of increased scrutiny, even for minor infractions of Wikipedia's code of conduct – so far that unregistered users are seen as "second-class citizens" according to Ayers [7]. The raising of barriers for appropriate conduct in form of innumerable explicit policies and guidelines apart from implicit social norms (cf. Section 3.1.3) further assists in identifying potential "drive-by" attackers. Yet, these rules are so convoluted that also well-meaning newcomers frequently fail to adhere to them due to the effort and time needed to fully comprehend and internalize them, and can hence be used to rationalize reverts of those newcomers.

Confronted with a deluge of potentially damaging edits (especially in the growth phase between 2003 and 2007, but still today), and lacking clearly distinguishable signals from new or anonymous, good-willed editors, patrolling Wikipedians hence face an information asymmetry [134]: they cannot efficiently and quickly enough distinguish well-done or at least well-intended edits from malicious changes, and, as a result, employ a "shot gun" method of keeping out attackers, in the process reverting many good-faith editors. Moreover, this mechanism is exacerbated by the use of semi-automated revert tools like Huggle,[16] which allow the custodians to quickly skim over and revert a huge amount of edits and which leads to a rather impersonal interaction with the reverted editors, as Halfaker et al. point out [64]. Accordingly, Thom-Santelli et al. [150] highlight the positive effects of a territorial watch for deterring vandalism, but at the same time "[...] also observe that these defensive behaviors may run the risk of deterring new community member participation", a view shared by other researchers [163, 64] and community members.

In summary, while most likely effective against much of the malicious edit intents, the potential negative consequences of this territorial defense behavior and xenophobia are apparent: Reverted useful changes and content from good-will editors, be it occasional or newcomers, will not be included in articles. And, after experiencing being reverted, less "established" editors might give up editing an article or Wikipedia altogether, a dynamic which could be one of the reasons for the decrease in successful newcomer retention that we saw in Section 3.1.2; a damaging mechanism for the article if the edit was actually useful but also for the survival of the overall community even if

16 https://en.wikipedia.org/wiki/Wikipedia:Huggle

their contribution was suboptimal in the first attempt [64]. This effect is exacerbated by the daunting amount of rules to master by newcomers for signaling their "worthiness" of having their contributions accepted.

3.2.3 *Open Questions and Metrics Needed to Answer Them*

The main theorized mechanism is that problems caused by "foreigners" will lead to a generalization of reactive, xenophobic resistance behavior against this group, using overly unspecific features for identification.

One question is therefore if the development of xenophobic behavior is systematically triggered for editors, overall or in specific articles, by a particular amount of malevolent edits they are involved in defending against – or at least witness as active editors in articles. Another question would be to what extend a socialization of wariness against outsiders, as pointed out by Goldmann, takes place through interpersonal influence (e.g. through accounts given on talk pages). It is also plausible that the effect differs highly between individuals and in certain article environments. In fact finding a systematic mechanism in place would warrant an estimation of single editors' potential to show xenophobic traits by looking at their editing histories. On single-article level, the result could be a labeling of editors and their edits into severity of estimated xenophobic behavior, enabling a monitoring of the editors by themselves and others. Even editors that join an article later and have not been directly involved in defensive reverts themselves might be affected, if an article that is permanently under attack signals caution of suspicious editors for example through respective talk page entries or warnings in the form of templates.

To conduct such an analysis one needs to know (i) how often, and how, malevolent users carry out disruptive edits in an article, through either an automatic vandalism classification tool that is *not* only trained on the revert actions by editors themselves – to not encode possible xenophobia into the algorithm – or an analogous hand-curated gold-standard data set of adequate size. Further, we need to learn (ii) when and how "guarding" editors undo those edits: How often, over what time span? Are they fully or partially reverting contributions? Are editors taking turns in reverting vandalism, in a form of a coalition against attackers? How much vandalism has the editor witnessed and/or fought in a target or any article, hence how much defensive behavior might have already been learned? What are the reasons given for reverting? Lastly, as a potential dependent factor: (iii) do over-generalized features of the malicious edits (unregistered user, no comments, no discussion entries, etc.) become the de facto

trigger for reverts, effectively extending them to non-vandalistic edits as well?

Apart from a *reliable and unbiased vandalism-indicator*, such analysis calls for an *accurate algorithmic method for detecting reverts* or, more generally, all disagreements in edits. In other words, it should enable to distinguish the degree of "defense" against a perceived malicious edit on the continuum between a full revert of an edit on one end and mere corrections on the other. It would moreover be helpful to explore the interactions and relationships between the defending editors to account for collaboration. Do, e.g., multiple "defenders" revert the same editors together? Do they reinstate each others content if deleted? Further, a *reliable method for text-mining* of edit comments – to extract reasons for reverts – and of talk pages – to detect expressions of xenophobia – would help to gain more knowledge about the general hostility towards outsiders among the article editors as well as point out editors that show particularly strong territorial defense behavior. Such data could be complemented by conducting *editor surveys* regarding revert motivations, the exact features of an edit used when determining to revert it, and acceptance of unknown users among editors.

These type of data in conjunction would enable an investigation into the assumption that indeed editors get conditioned to perceive certain broad features of edits and editors as reliable thread indicators the more attacks they are exposed to and afterwards spend less cognitive resources on corroborating suspicions through thorough content review but instead rely on these simple indicators for their revert decision.

3.3 OWNERSHIP BEHAVIOR

The psychological concept of personal "ownership" has been defined (cf. Van Dyne and Pierce [152]) as "the state in which an individual feels that an object (i.e., material or immaterial) is experienced possessively (i.e., it is 'MINE' or it is 'OURS')." This concept can result in a variation of the defensive territoriality we described in Section 3.2, but is not primarily an act of protection triggered by intrusion to any kind of clearly delimited area (here: the article) but first and foremost resources that belong to oneself. In the case of an article, the object of possession can be the article as a whole (e.g., if the editor created it) but more often will be the specific content segements that an editor has added herself. While similar to the territorial defense discussed above, it is not necessarily learned or dependent on negative prior experiences like threats, but based on the general human tendency to feel ownership for things in one's possession, and even more so for things that one has created [152].

3.3.1 *Suspected Dynamics on Wikipedia*

Although explicitly discouraged by Wikipedia,[17] strong feelings of ownership for an article including protective behavior are not uncommon, as Thom-Santelli et al. [150] show. This might have to do with the hours of work many authors put into an article, the self-perceived level of expertise they possess on a given topic, and other reasons that lead to a personal attachment to an article.

Halfaker et al. [63] summarize this and other work to conclude that researchers have "found that there are editors who assert ownership over articles and use their previous work on an article to exert control over which contributions will be accepted". Halfaker et al. also empirically show that the number of editors whose originally authored words are deleted during an edit is a very strong and stable predictor of the probability that the deleting edit will be reverted itself. They find this 'stepping on toes' effect to be in place independently of any other feature of the editor and hence infer that "[...] Wikipedia's review system suffers from a crucial bias: editors appear to inappropriately defend their own contributions". Editors ergo in this mechanism do not apply any threat heuristics nor do they necessarily (although this often coincides) aim to protect the article as a whole. The mechanism in place is rather that change to one's own content is much more likely to be reverted by oneself than changes to other content, no matter the nature of the change or the changing editor.

3.3.2 *Potential Harmful Consequences*

Content guarded in such a way naturally runs a higher risk of primarily reflecting the "owning" author's point of view. Although many editors do a good job of incorporating all relevant viewpoints on a topic even if they are not their own (many featured articles are for example predominantly written by just one or two editors), the inherent bias of the Wikipedia population (cf. Section 3.1.5) and a possible inclination of certain subgroups to more tenaciously defend their content (e.g., hypothetically, males under 20, religious users) makes it likely that if articles are written by just a few editors, these users might share several characteristics, be it demographics, political views or simply behavioral traits. They are hence prone to deliberately or unknowingly exclude relevant points of view of other population groups. And if editors are unreceptive to corrections of text just because they deem it their property and not because of the content itself, this undermines the very core of Wikipedia's principle of peer-review.

Lastly, an ownership mentality of course amplifies the problem of new users feeling unwelcome and not needed, as pointed out in Section 3.1.2.

17 http://en.wikipedia.org/wiki/Wikipedia:Ownership_of_articles

3.3.3 *Open Questions and Metrics Needed to Answer Them*

Halfaker et al. [63] studied reactions of editors to changes of their owned words by building on the "content persistence" metric adopted from Priedhorsky et al. [122], but did so only for a subsample of articles. The method was not shown to be efficient to run on whole language editions of Wikipedia, nor was there any accuracy testing reported on it.[18] What is hence needed is an *efficient authorship attribution* (and change/persistence tracking) technique that is *tested for its accuracy*. Also we would need *accurate revert detection* to track all changes to an editor's content on single word level and her reaction to it. Extracting edits per user (for determining edit concentration) can be easily done from Wiki log data, yet exact and efficient authorship extraction and change tracking is a non-trivial task in comparison [36]. Based on this data, it would be possible to investigate several aspects of ownership behavior:

- To what extent are "owners" of content overly protective of their content segments in comparison to other editors' text in general on Wikipedia?
- Where (in which articles, which thematic domains) is such behavior more pronounced than on average? Are some content types more protected than others? Does strong ownership behavior appear independently from present threats like vandalism? The latter would help to delineate ownership behavior from defensive territoriality for the whole article as outlined in Section 3.2.
- Who is exhibiting strong ownership behavior? Do such users share common characteristics like demographics (as far as extractable), editing patterns, talk page activity, added content types, etc.?
- Is this self-protection stronger with more already owned content in an article, indicating a reinforcement of ownership mentality with increasing amounts of owned words ? How does ownership behavior generally develop over time?
- Is ownership behavior usually successful in terms of "amassing" owned words? Does it lead to higher concentration of content on editors that show such behavior? It is possible that aggressive defending of own content draws repercussions from the

18 As far as discernible from [122] the used method is based on conventional text-differentiation algorithms, which are neither necessarily accurate nor efficient for determining provenance, authorship or for persistence tracking, as we will see in Chapter 4. An evaluation in this regard was also not provided. The method further does only take into account identity reverts, which we have shown to omit a notable amount of undo/delete actions [51].

community and simply fails eventually, and that many editors with high amounts of owned content do not show such behavior.

- Does ownership behavior appear as frequently in featured or good articles as in average, unmarked articles or even in such that are tagged with templates indicating point of view flaws? One could evaluate specific pieces of content written by distinct subgroups of self-defending editors in regard to viewpoint bias. This could also be accomplished by querying independent human raters for assessment of content heavily defended by its owners as well as the unsuccessful changes attempted by other editors.

Effective and efficient authorship mining and change tracking would enable investigation into these questions; not only on an aggregate statistical level but also through appropriate, understandable visualization interfaces for end-users, to inspect dynamics in articles they are interested in.

3.4 SOCIAL PROOF AND A "NO-CHANGE CULTURE"

Social proof [29] describes a socio-psychological phenomenon in which, in a setting of uncertainty, where personal knowledge about the optimal behavior is low, individuals emulate the actions of others to arrive at the best choice for action. The effect is stronger (i) the more uncertain a person is about the right behavior and the ambiguity of the situation, (ii) the more trust is placed in the informedness of the individuals showing the behavior and (iii) the number of people showing the behavior. It can often result in *herd behavior* [10] and behavior driven by *information cascades* [14]. While assuming the conduct of an observed crowd might in some scenarios serve the goal of integrating or identifying oneself with a social group, it is in other cases driven by purely *rational imitation* to arrive at better individual choices [69, 10].

An auxiliary sociological and psychological concept applicable in our setting is *conservatism*, a "belief in the value of established and traditional practices in politics and society" by members of a social group or more generally "the tendency to prefer an existing or traditional situation to change" [31].[19] As such it often informs the actions of individuals holding these beliefs and tendencies, striving to consolidate the status quo [88]. The social mechanism behind it can be interpreted as avoidance of risks associated with change.

Where a conservative mindset and behavior dominates, naturally, innovation might get stifled; an effect exacerbated when additionally

19 Please note that we aim to decouple "conservative" here from any specific political view but only use it to describe a type of social behavior that aims at preserving given states over new ones.

this dominant mindset is visible and a social-proof-driven dynamic of "imitation of inaction" takes place. A related mechanism has been called the *spiral of silence* in other contexts [113]. It describes a phenomenon were individuals are not voicing their opinions in public and neglecting any personal knowledge contradicting the *perceived* majority opinion to not dissent from the crowd. In extreme cases, the effect can be that a large minority or even a majority exists that is silently holding different viewpoints than those publicly expressed, but not being aware of the relative popularity of their views and therefore not uttering them.

3.4.1 *Suspected Dynamics on Wikipedia*

In many cases, a consensus between Wikipedians has been reached for an article or subsection of an article, meaning that the content has been (re-)viewed by many users and eventually reached a stable state, where editors have brought in their ideas and which has not been changed for a longer time.[20] Establishing an explicit consensus on content, especially after disagreements, is often the result of a lengthy negotiation process. But it might also happen "silently", by merely not changing content for extended periods of time (also, e.g., only for a specific subsection, while other changes are made in the article). Altering such content afterwards from its agreed-on form is at least subject to increased examination by some editors, often such with a longer tenure in the article. Hence, as a first mechanism, these editors might request specific reasoning as to why content, which apparently "worked" for a longer time for most visitors – or was negotiated between several editors – should now be changed; especially in change attempts whose subject are no explicit novel developments or previously missing facts for which sound sources and evidence exist. In the worst case, the change in question can directly result in a revert by such "watchmen" if it adds no "hard" facts, or otherwise benefits the article in an obvious way.[21] In justifying reverts, they might also point to social proof consisting of former edits, talk page entries or templates in favor of the old state of content.

As a second effect, editors that are aware of the article developments – through following the edits in the revision history, reading the talk page or seeing templates – might perceive an established consensus and take this social proof as a signal to not attempt any further changes in the first place. This holds especially if they are not confident in their own latent plans for change – as in: “If those words where read by so many people and were not changed, they

20 Cf. https://en.wikipedia.org/wiki/Wikipedia:Consensus

21 Cf. the common practice of reverting edits with "Violates consensus" edit comments, see https://en.wikipedia.org/wiki/Wikipedia:Consensus and associated talk page.

certainly have to be right". Even if the alteration is implemented, a revert by another editor pointing to the ostensible consensus might again trigger uncertainty and serve as social proof of the correctness of the established information and lead to a reevaluation of the own edit. This rational imitation means that if a large group of people has seemingly assessed a content to be right by not altering it (and this consensus is visible to a user who might consider alterations), it becomes a less likely target of subsequent changes and hence is even more perceived as to be correct by social consensus, and so on.

In many cases, this mechanism is a viable and completely rational social heuristic when trying to reach and maintain high-quality, stable articles. While random vandalism attacks and other clearly damaging edits are easy to spot, less obvious suboptimal changes are more complicated to revert with objective reasoning; here, a pointer to previous discussions and consensus finding processes can act as quick legitimation for "quality-keeping" reverts and provide stability without the need for redundant renegotiation. And educating herself about former discussions and change attempts can persuade a well-meaning, but ill-informed contributor to refrain from introducing yet another non-fitting version of a specific text piece.

From related research, we know of indicators for the fact that much content has become consolidated insofar that it is difficult for editors to change it. Halfaker et al. [63] show that if an editor removes words in her edit, the probability of her edit to be reverted increases significantly with the number of article revisions the removed words had "survived" before (normalized for the number of removed words). Adler et al. [2, 3] base their WikiTrust metrics on an author's reputation derived from the persistence of her revisions and demonstrate that reputation is a good predictor for her later edits to be less likely removed. In both cases, the interpretation of these findings by the authors is that word and edit persistence can be used as a measurement of quality according to Wikipedia rules, for instance, a predictor for "featured articles" [2, 3]. Still, in a nutshell, the results of the studies cited in fact make the case for basically one thing: that words which have been in an article for a longer period of time are harder to remove. Of course, the intuitive reasoning is that if a piece of content is indeed factually or otherwise "correct" according to the individual judgement of most readers and editors, it will less likely be removed and more likely be reinstated once deleted; this can be seen as collective decision making at work. Yet, as a secondary possible explanation, the perceived consensus between editors might lead to less change attempts or more reverts, based on social proof of consensus or simply the age of calcified content – instead of the content itself.

3.4.2 *Potential Harmful Consequences*

When social proof is acquired by a user for informing the decision if a change should be made (for instance, by reading the discussion pages or studying the revision history and edit comments), this can lead to misguided conformity behavior. In some cases, there might be indeed simply not enough personal knowledge about the subject present to make an educated edit; and social proof might rightfully convince the user that previous negotiations between other editors about the content arrived at better solutions than the one she is about to (maybe hastily) apply. In other instances, however, the individual might, in face of the apparent social proof of correctness and completeness of the content (i) not even try to tap into her own cognitive resources and knowledge, or to do independent research, to reduce cognitive effort [171], as the article seems to be "taken care off". This can be likened to the so-called "bystander effect" that has been extensively discussed in sociological literature [110, 61].[22] In another possible scenario she could (ii) already have new information and hold a diverging opinion on how the text should be written, but shy away from implementing it. As Schulz-Hardt et al. [129] put it: "[...] groupthink theory [74] and research on groupthink [43] stress that formal and informal conformity pressures and the desire to preserve harmony within a group can override the motivation to critically appraise the relevant facts, thus (often) leading to poor decisions". The mechanism in the second scenario is the same that underlies the "spiral of silence" dynamic described previously. It can result in potentially beneficial changes not being applied and untapped knowledge being excluded. With such mechanisms being the norm, the result can be a "no-change" culture in an article (or for a subsection), where adjustments are not common anymore and whose contents may be seen by editors as a near-ideal status quo that can only be changed if special justification is given. Such an environment can hence lead to less change attempts being made in the first place, as users do not learn from observation that "being bold" when editing pays off. This notion is relevant as *boldness*, conceived as challenging old content structures by correcting them or bringing in additional new content, is an important pillar for quality, and as such it is actively encouraged by Wikipedia's guidelines.[23]

Social proof (esp. apparent consensus) in combination with a conservative inclination can also be used as a misguided rationale for rejecting changes already made. Even if an article was considered "complete" at one point in time, new events or changing societal views

22 A popular example is the case of a person in need of medical assistance in a public space. If a large-enough crowd is present, an observer will automatically assume someone in the crowd will call an ambulance due to the sheer amount of onlookers and probability. Where this reasoning occurs (at least for some time span) it transforms the observer into a bystander instead of a potential aide.

23 http://en.wikipedia.org/wiki/Wikipedia:Be_bold

regarding the topic might require updates. While new factual data based on substantial sources will have less trouble to be accepted in Wikipedia (so far as it is notable and relevant), new viewpoints on certain topics that emerge in society may not be easily "provable" in terms of their relevancy and correctness. Examples could be the changing views of society on topics like discrimination, politics in general, or morals and ethics, which often lack "hard facts" to support them; or simply rewrites of any existing text to reflect a different approach to the topic without adding new factual proof or references. Particularly with a more diverse set of editors joining Wikipedia or taking interest in new topics (e.g. from different cultures, born in a different decade), novel points of view are likely to be claiming their right of inclusion; and in many cases, rightfully so. Yet, rejecting such contributions is likely to succeed if the only proof of what should or should not be included lies in the (perceived) opinion of the majority and the "traditional" form of the article or the policies that govern allowed editing behavior; said changes might thus face serious difficulties entering an article in a very conservative, consolidated article environment.

Furthermore, incorporating these viewpoints into an article might prove a harder task later in the article life than if the same inclusion had been attempted while the page was still evolving, e.g., in its first 200-300 revisions, and even more so for articles that were started in the infant years of Wikipedia. Viewpoints added in these early days may have a much higher probability of being eventually represented in the article, because social proof behavior can result in information cascades [14] where earlier content has a higher chance to survive than such that was later introduced.

3.4.3 *Open Questions and Metrics Needed to Answer Them*

To answer if such mechanisms in fact exist in Wikipedia, we have to investigate several aspects:

- What is the relation of age of content and its likelihood (i) to receive and (ii) to survive change attempts on average for whole Wikipedia language editions? While it is reasonable to believe that changes will see a marked drop in the chance to be reverted once they survived a first phase of intense scrutiny of around 5 to 10 revisions, it is unclear which development generally occurs afterwards. Does the effect, e.g., consistently grow stronger with increasing age of the content or the article and/or will it decrease again at one point as content becomes outdated?
- How does simple age (in revisions or time) compare with other factors in predicting change attempts and/or survival of content for the whole Wikipedia or specific categories and thematic

domains? Can age alone be a predictor, even if one controls for most other relevant factors, especially the quality of the present text and the content to be inserted in its stead?

- Is the effect of content age on survival reinforced by the presence of markers for consensus or explicit "no change" signals – such as talk page entries, templates, notes in the Wikitext (e.g., "do not change as per consensus"), or comments associated with the edits in the article history?
- Do users in fact don't attempt changes because they are influenced by explicit markers such as templates or behavior of other editors, convincing them no more alterations should be made?
- Do "conservative" users have to be present that specifically protect older content from change (by signaling or reverting) for this effect to take place? Does such a type of users exist?

To enable such investigation, certain metrics are necessary which we do not have access to yet for all revisioned documents in reliable quality:

- Age of content: How many revisions (or views or time) has a particular word been present in an article? Such knowledge allows for gauging how often content was "inspected" (i.e., seen) and not changed. It would also be useful to know if a word has been in the article constantly or if it had been removed for some time and then reintroduced (cf. next item).
- Change attempts: Related to word age is a detailed measure of how often a piece of content has been "challenged" (i.e., deleted or changed and then reinstated) in total and per a specific amount of revisions or time and how often it was restored after such an attempted change or removal. This would also allow an aggregate measure of "boldness" over time for the whole article to assess the overall editing dynamics in an article and compare it to other articles.
- Signaling and its effect on boldness: A reliable indicator of potential "no-change" signals like templates, edit comments and talk page entries is required to be matched against a possible decrease in (successful) change attempts, of articles or smaller text sequences. However, surveys of users (both addressees and originators) about their perception of such signals would be advisable to complement this approach.
- Edit quality indicators: Either (i) dependable automated classifiers for edit quality are needed that are trained *without* using content age as a feature or (ii) effective and efficient crowdsourcing methods to judge quality of edits.

For content age and change attempts, the metrics correspond to the efficient and accurate provenance and change tracking needed, as discussed in the previous sections.

3.5 CONFLICT

Conflict can be defined as "an expressed struggle between at least two interdependent parties who perceive incompatible goals, scarce rewards, and [potential] interference from the other party in achieving their goals" [164]. Conflict is often an integral part of negotiation and the exchange of arguments is essential to merge opposing viewpoints on a topic. Putnam [123] writes congruently: "The recognition that conflict is productive is not new. Theorists of the 1950s and 1960s address the functional and the productive side of conflict [33, 39]. Conflict [...] enhances adaptation, growth, and stability of organizations; it guards against groupthink; and it facilitates effective decision making through challenging complacency and illusions of invincibility."

3.5.1 *Suspected Dynamics on Wikipedia*

Many different kinds of disagreements over content exist in Wikipedia and their transition into "conflicts" is mostly a question of definition. So much research on "disputes" and "conflict" in Wikipedia has been done that the existence and central role of conflicts regarding content negotiation in Wikipedia does not have to be emphasized (to name a few works: [85, 83, 84, 81, 87, 5, 141, 168, 169]).[24] As a basic denominator, research agrees, in line with traditional sociopsychological literature, that conflicts are essential to negotiating which content should stay in articles. E.g., Kittur et al. [83] point out that specific kinds of conflict can be beneficial in peer collaboration systems, while Vuong et al. [155] make a congruent point, but stress that this holds only for non-personal, purely content-related disputes.

3.5.2 *Potential Harmful Consequences*

Although the beneficial functions of disagreement and outright conflict cannot be denied, scholars concede that an over-abundance of conflict can lead to negative results, as is for example frequently the case when "edit wars" are mentioned [141]. While in some scientific work, the negative effect of conflict in Wikipedia is researched, a fine-grained discussion of the exact mechanisms of *why* conflict is happening, and *how exactly* it is diametrical to quality, is often lacking.

24 Cf. https://scholar.google.com/scholar?q=allintitle:conflict+wikipedia

A specific pattern that recurrently appears during editing conflicts is the continuous deletion and reintroduction of the same content between two or more editors, without exchanging substantial new arguments for their respective changes. This behavior, which Wikipedia actively tries to suppress (among other policies) via the "3-revert-rule"[25], is a reliable indicator for an "unhealthy" conflict, as no new knowledge is generated and no progress is made in negotiating the content. One mechanism that is supposedly behind this dynamic is the common human pursuit of winning an argument once it has been started; not necessarily due to the absolute personal conviction regarding the correctness of the content but simply for the sake of prevailing in the contest with an opponent or because of an agenda in pushing a certain point of view.[26] This type of dispute is known as *eristic argumentation* in the philosophical and psychological literature [114]. Wikipedia's closest equivalent concept is "battleground behavior",[27] which is as strongly discouraged, as is attempting to "win" an argument.[28] If such arguments happen, however, they tend to be self-sustained, as no involved party is aiming for a middle-ground solution to the conflict and logical or evidence-based resolutions might not be effective. Such arguments can also be linked to personal animosities and emotional motivations [25]. While this can happen for a whole article, it is much more likely to concern specific parts of articles. Further, eristic arguments develop between individuals as well as between groups (or "opinion camps"). One probable outcome of such a conflict – if not both sides are reconciled and forced to a consensus through a mediating third actor such as another editor or an administrator – is that one side will renounce and surrender the field to the more tenacious party if a consensus seems unattainable. In these instances, the inferior side of an editing conflict might not have any of their views represented in the eventual article, and the content would hence be biased towards the "winning" faction.[29]

Another negative effect of conflicts can be the voluntary exclusion of more moderate editors from the discourse when argumentation and editing get more intensive or even aggressive. Avoidance is a common social mechanism to deal with conflict [94, 126]. When a

25 https://en.wikipedia.org/wiki/Wikipedia:Edit_warring#The_three-revert_rule Excerpt: "An editor must not perform more than three reverts on a single page – whether involving the same or different material – within a 24-hour period".

26 https://en.wikipedia.org/wiki/Wikipedia:NPOV_dispute#POV_pushing

27 https://en.wikipedia.org/wiki/Wikipedia:What_Wikipedia_is_not#Wikipedia_is_not_a_battleground

28 https://en.wikipedia.org/wiki/Wikipedia:Wikipedia_is_not_about_winning

29 This is showcased by the example of Conservapedia, a clone of Wikipedia which represents views on many topics that are typically referred to as politically "conservative". It is a direct result of conflicts in a few articles, where no consensus could be found between two opposing camps, resulting in the conservative party not re-entering the discussion arena but founding their own encyclopedia: http://www.conservapedia.com

dispute is already ongoing, third-party editors might chose not to participate in editing at all in the article to not get "sucked into" the conflict, and for instance be subjected to personal attacks. We will call this retreat mechanism *sidelining* of neutral editors. The effect might not only be that the viewpoints of these editors on the topic in question are lost, but also their intervention as mediators between warring factions. Out of similar reasoning, Wilkinson and Huberman [163] recommend increasing the number of users with diverse viewpoints to a high-conflict article instead of having "the same few people arguing back and forth" as a means to avoid deadlock situations and achieve progress.

3.5.3 *Open Questions and Metrics Needed to Answer Them*

Several measures could serve for detecting *eristic behavior* in conflict:

- Non-evolving disagreements: A method is needed to track disagreements that go back and forth on the same or adjacent content between the same (group of) users for an extended amount of revisions or without shifting to new topics, and not developing the content further. While approaches have been proposed to find reverts of revisions, these are only able to detect specific kinds of disagreements, where whole revisions are undone (which do not cover all disagreements, as we will discuss in Chapter 5). A more fine-grained detection of disagreements including repeated add and undo actions on word-level would therefore be essential to cover all potential conflict. E.g., it would be helpful to detect one editor adding, and another editor deleting, the same words multiply times, while other, unrelated actions are carried out at the same time in their edits. Similarly important, this world-level conflict tracking method would allow using part-of-speech analysis on top of its results, to allow detecting the exact content of disagreements and discerning if they center on specific kind of text (e.g., certain adjectives), and if antagonists in a conflict advance the content only in so far as they use very similar text elements (e.g., synonyms) but still cling to their viewpoints.

- Disappeared content after conflict: For learning about the consequences of eristic argumentation, it would be essential to detect if one opponent in a conflict loses their content after the dispute dies down and the other side keeps their content in the article, thereby effectively "winning" the argument. This also makes a sensible measure necessary of when a conflict starts and when it ends, on word level, as well as the exact tracking of which user authored or has defended which words and how long these content pieces indeed remain in the article.

- Opinion camps: When multiple users engage in such back-and-forth conflicts we would like to know if they can be partitioned into "opinion camps" working against each other on the same content. To this end, on top of the aforementioned measures, one would need precise data about disagreement interactions between the editors on word level and an approach to cluster them into plausible groups.
- Sidelining: To identify a possible sidelining mechanism, it is necessary to reliably determine the intensity of conflict or classify it as aggressive or "edit war", to then relate these measures to the potential decrease of activity of editors in the article which are not involved in those conflicts. This necessitates accurate detection of disagreements and inter-editor relations (cf. the notion of reverts discussed above). Additionally, a method to classify aggressive or conflict-cultivating edit comments and talk page entries would be valuable to enhance the analysis based on the main article content.

3.6 CONCLUSIONS

A quote from a user on the arbitration case of the highly controversial article "Gamergate controversy" supplies an illustrative example of how several of the aforementioned mechanisms can come into play to create a non-productive article environment:[30] "I'm a semi-involved editor in this case, in that I've attempted to make one minor edit to the article and was turned off from any further contribution. [...] I'm sure the article was filled with people trying to push an agenda without actual regard for neutrality. [...] this type of mentality is damaging since it quickly pushes away new editors, so ultimately the page becomes effectively owned by the same group of pro-[Gamergate] and anti-[Gamergate] Wikipedians with most other people watching from the sidelines, at best".[31] As we will also discuss in our visualization use case in Section 6.4, in this specific case, a handful of highly active users in this article were (i) engaging in a several eristic arguments about different sections, (ii) formed two distinguishable opinion camps, (iii) were very aggressive in their disagreements (and comments), (iv) eventually owned most of the article content and (v) eventually sidelined many more neutral users.

30 The article was the arena of one of the most notorious edit wars of Wikipedia's recent history. Refer to Section 6.4 for a detailed description of the article and its topic. Find the full arbitration case here: https://en.wikipedia.org/wiki/Wikipedia:Arbitration/Requests/Case/GamerGate

31 https://en.wikipedia.org/w/index.php?title=Wikipedia:Arbitration/Requests/Case/GamerGate&oldid=635696903#Statement_by_.28somewhat_involved.29_Remorseless_Angel

In this chapter, we presented some of the patterns that can point to such social mechanisms which are possibly counterproductive to the collaborative effort. We also explained what kind of data would be necessary to inspect if those mechanisms are in effect and in what way they indeed turn harmful. We gave an overview of those of Wikipedia's social mechanisms that we deemed most influential and that can occur in each article. Note again that the general social heuristics behind these mechanisms are mostly beneficial, and that unfavorable corollaries are just one possible outcome that has to be carefully checked for.

As a summary, we identified the following main methods and metrics to be elementary to detect the presented social mechanisms and their impact:

- accurate and efficient provenance (authorship) detection,
- tracking of word age, removals and reintroductions,
- word-level disagreement (conflict) detection and classification into stages of disagreement,
- construction of temporal inter-editor relationship networks per article based on edits,
- reliable, unbiased and efficient vandalism detection,
- collection of features of editors beyond editing behavior (e.g., demographics, political views),
- reliable and unbiased edit and content quality assessment (automated or through crowdsourcing),
- a classification of article and talk page templates as well as other possible markers in the article into "no-change" signals, ownership claims or other relevant categories,
- text-mining methods for identifying aggressive, possessive, "xenophobic" or otherwise interesting speech patterns in edit comments and talk pages.

Research has already made important steps towards viable solutions for some of these missing methods. Regarding vandalism detection, academia [119, 27, 160, 41] as well as the Wikipedia community[32] have come up with working prototypes. For automatically judging the quality of articles and edits, promising machine learning approaches have been published [157, 158, 145, 144, 66]. There have moreover been advancements in the area of text-mining as well as regarding inference of demographics for online users, which could

32 https://en.wikipedia.org/wiki/Category:Wikipedia_counter-vandalism_tools

serve as a base to provide us with methods and data as described above.

In the remainder of this work, we will however focus on what we consider a central missing piece for studying social editing mechanisms on Wikipedia. Namely, we will propose an efficient and accurate provenance detection method, which will also enable us to track all changes to single words (including reintroductions, repeated deletes, age). From this data, we then derive (dis)agreement interactions between editors, which can be used to detect conflicts on word level. While conflict detection has been studied mostly for article level [141, 83, 168, 169], some approaches have been proposed to track conflicts, more fine-grained, on below-article level [23, 17]. This thesis will provide a more accurate and easily extractable data basis, particularly the word-level changes and interactions between editors that are needed to compute such fine-grained conflict metrics. Lastly, we will demonstrate how to construct inter-editor interaction networks out of this information.

4

PROVENANCE AND CHANGE DETECTION

In this chapter we will present a model, and an algorithm on top of this model, to attribute the revision and author of origin to content tokens (mostly whole words delimited by whitespaces) in revisioned text. In addition, this will allow us to trace every change made to a specific token over the whole lifetime of the article and the number of revisions it was present. The concrete method is inspired by and tested on Wikipedia editing data.

The need for such an approach was already motivated in the preceding chapters. Additionally, the usefulness of a trustable and efficient word-level change tracking has been previously discussed in research [36, 46], and is as well corroborated by the fact that Wikipedians might be motivated by the recognition by their peers that comes with authoring content [54]. Further, more practical purposes also exist [36]: To reuse a Wikipedia article under the CC-BY-SA license, for instance, might require to list the main authors of the article, which are not easily retrievable as there exists no straightforward way in the MediaWiki software to show authors of single pieces of text for a particular revision.[1] Lastly, the Wikipedia community has come up with a number of intended solutions related to the original revision attribution problem on word level, which highlights the utility of such a solution for Wikipedians.[2]

The attribution problem at this fine-grained level and in highly dynamic environments like Wikipedia is not trivial, as has been outlined in previous work [36]. We will discuss the challenges when introducing our content model in Section 4.2.1. Frequent reintroductions, content moves and other actions can be hard to monitor. In software code revisioning, similar issues can emerge and refined attribution techniques can have similar merits, as small changes of a few characters might have great effects, just as (re)introducing larger code chunks.

Against this background, the main contributions of this chapter are: the model for revisioned content we propose (Section 4.2), the algorithm we build upon that model (Section 4.3), the generation of a gold standard for precision testing (Section 4.4.1.1) and the experimental evaluation of precision, runtime and materialization size in comparison to the state-of-the-art (remainder of Section 4.4).

1 https://en.wikipedia.org/wiki/Wikipedia:Reusing_Wikipedia_content The Wikipedia reuse policy is asking for "a list of all authors" in specific cases, which is currently often simply approximated by listing the editors with the most changes, although these might not have added the most – or any – of the reused content; cf. also CC-BY-SA: http://creativecommons.org/licenses/by-sa/3.0/legalcode

2 https://en.wikipedia.org/wiki/Wikipedia:Tools

Although we use the example of the English Wikipedia as inspiration and testing ground, the proposed model and algorithm can be understood as components of a more generally applicable method for revisioned content. We are convinced that many of the assumptions made for the use case of Wikipedia also hold true not only for other Wikis but also for other revisioned content systems.

4.1 RELATED WORK

In the context of Software Engineering, content attribution has been studied in terms of code ownership. In programming, lines of code are still used for measuring technical quality of source code [8], as well as a basic unit to identify contributors. Therefore, solutions to trace code ownership are designed to operate on a coarse-grained level [117, 120]. Decentralized Source Code Management systems such as Apache Subversion [117] or Git[3] provide a feature to keep track of changes line-by-line. This functionality is denominated `blame` or `praise` depending on the system. When a contributor performs a change on a line of code, the system attributes the whole line to that user. In this way, `blame` allows to identify who last modified each line in a given revision of a file, but the information about the origin of the content is unaccounted for. The `blame` approach is a suitable solution to detect defects in collaborative software development [124] as well as to select expert developers for implementing required changes in programs [95], yet does not provide an appropriate mechanism to trace the *first* revision the content appeared in at a more fine-grained level such as single words or special characters, to which we refer as "tokens" in the remainder.

To detect provenance information in Wikipedia article text, several analysis approaches have been employed. *HistoryFlow* by Viegas et al. [153] assigns sentences of a text to the editor who created or changed them. It does not, however, acknowledge deleted content that was later reconstructed as being written by the original editor. More importantly, by operating on a sentence level, small changes like spelling mistake corrections lead to wrongly recognizing the correcting editor as the author of the whole sentence.

Wikitrust generates a visual mark-up of trusted and untrusted passages in any Wikipedia article [4, 2, 3].[4] To track provenance and authorship, longest matches for all word sequences of the current revision are searched for in the preceding revision and in previously existing (but now deleted) word-chunks. In this way, *Wikitrust* can as well detect reintroduced words and assign the original author – an important feature, as "reverts" to formerly existing revisions are commonplace in Wikipedia. The underlying algorithm is, however, a

3 http://git-scm.com/
4 http://www.wikitrust.net/

variation of a greedy algorithm [2], known to look for local optima, which in the case of determining word authorship can lead to grave misinterpretations when word sequences are moved rather than inserted or deleted only [21].

In earlier work [46], we introduced an authorship attribution approach for Wikipedia based on a tree model of paragraphs and sentences. The precision of the results according to the evaluation lies at 59.2% compared to 48.4% for *Wikitrust*, values that are rather unsatisfactory for productive application requiring users to place confidence in the computed attributions. The work presented here builds on the foundations laid by [46], but formalizes a model based on a k-partite graph to represent paragraphs, sentences and tokens much more efficiently compared to the tree model of [46]. We also gain over 30% in precision in respect to [46] by refining the conceptualization of authorship and tokenization of text.

The most relevant related work, by de Alfaro and Shavlovsky [36], proposes an algorithm for determining provenance of tokens in revisioned content based on the concept of processing text as sequences of neighboring tokens and finding "relevant" matches in the revision history given a parameterized rarity function, for the work in question defined as the length of the sequence.[5] This means that a token is uniquely identified solely by its local neighbors. To store the provenance attributions, the annotated revision history is remembered by means of a trie structure. A trie is a tree where the provenance information is stored in the leaves, while the intermediate nodes are empty. The arcs of this trie are labeled with tokens. All the arcs leaving a given node correspond to neighbors of the token(s) represented in the label of the arc incoming to that node. The algorithm was tested in terms of runtime and materialization size (storage in secondary memory) of the results. It takes into account reintroduction of text and can keep track of provenance on a word or smaller token level and therefore conforms exactly to the goals set out for our work. However, it was never evaluated regarding the accuracy of its attributions and we suspect that the technique of selecting the most likely provenance revision by using only the local neighboring tokens as described in [36] is prone to mismatches as it does not take into account all relevant change information; if the used matching sequence is too long, it could for example ignore the true origin of a token, if it is too short it might wrongfully select a recent introduction of the same tokens as the true origin.

5 I.e., in their paper, the authors define the appearance of tokens in specific 4-grams as "rare enough" (paraphrased) to trace their origin. Another rarity function could, e.g., be based on the general probability of a word or n-gram to appear in a natural language text.

4.2 MODELING REVISIONED CONTENT

The following subsections outline our model for representing revisioned content.

4.2.1 *A Model Based on Observations of Real-World Revisioned Writing*

When observing collaborative writing in systems that rely on long-term, incremental refinement by a large group of users, namely Wikipedia and its sister-projects, certain patterns become salient. In the following we list our conclusions from these observations and from studying related literature (e.g., [76, 83, 153]). These assertions build the conceptual foundation for the content model developed in Section 4.2.2.

The first assessment is that a considerable part of editing activity after the initial basic writing phase consists of moving, deleting and reintroducing content, but not adding much new subject matter per edit. A notable number of edits consists of reintroductions, which are often reverts due to vandalism; another reason for reverts is, e.g., a conflict between disagreeing factions. Moves of text sequences are also a regular sight, where a sentence or paragraph gets shifted to another position in the article without alterations. Another sizable amount of edits is predominantly changing only very small parts of the article per edit, incrementally revising the text. This pattern is occasionally interrupted by a burst of new content; for instance, in case of a larger addition or a fundamental rewrite. Still, very often the changes implemented per edit do not span more than one section or paragraph – frequently, they do not even transgress the boundary of a sentence. These assertions point out that methodically keeping track of reused and relocated content plays an important role when intending to efficiently monitor token provenance and change histories over large data in such a system.

Regarding the conceptual definition of "original provenance" (respectively, "authorship"), the larger context of a token plays a crucial role. The paragraph or the section it is embedded in can be as important for the interpretation of its meaning as its immediate token neighbors in a sequence. The same exact string of tokens, even up to the length of a sentence (e.g., a figure of speech), might mean something completely different in one section of a text (e.g., an introduction) than in another segment, where it was potentially introduced for a different purpose. It can therefore not necessarily be seen as an exact copy of the same sequence in another position, entailing the attribution of the same revision of origin. An example: One editor writes the sequence "A theory is" in front of a statement A at the top of an article in revision i. Later, a different editor adds the same three words ahead of a completely different statement B in revision $i + x$.

Both authors use the same terms and add the assertion that the subsequent statements are mere theories instead of proven facts. Yet, the declaration that "statement *B* is a theory" can only be attributed to the later editor as the first author used the same chain of words in a different context and with a completely different goal (namely to call statement *A* a theory). This also applies, e.g., if two authors use an identical literature reference in order to prove different facts. Basically, just by comparing local neighbors of words, as it is done by de Alfaro and Shavlovsky [36], which in practice use four-word sequences, the larger context of the tokens is not taken into account. Extending these neighbor-tracking sequences to sentence or paragraph length, on the other hand, is not constructive, as this would contradict the initial idea of exact word-level provenance and author attribution.

Tracking provenance hierarchically, by assigning tokens to a larger enclosing unit (sentences), and linking these to another superordinate element like a paragraph provides a more exact identification of tokens than mere local contextualization. Another key advantage compared to the method proposed by de Alfaro and Shavlovsky [36] is that not all tokens in the text have to be analyzed if the enclosing unit has already been identified as unchanged or as reintroduced from an earlier revision. This enables a more rapid processing of the data, if changes, as mentioned above, often only affect fractions of the whole article.

4.2.2 *A Graph-based Model for Revisioned Text Content*

We propose a model to represent revisioned content as a k-partite graph, where the content is partitioned into units of discourse in writing, i.e., paragraphs, sentences and tokens (which can consist of words or single characters as we will explain in Section 4.3.3).

In order to illustrate the representation of revisioned content with the proposed model, consider a Wiki page with three revisions r_0, r_1 and r_2 as depicted in Figure 2. The first revision, r_0, contains a single paragraph, p_0, which is composed of only one sentence, s_0, with two tokens (t_0 and t_1). The labels over the arcs represent the relative position of the nodes. For instance, the tokens t_0 and t_1 are located in positions 0 and 1 of the sentence s_0, respectively. The second revision, r_1, creates two new paragraphs, p_1 and p_2. The paragraph p_1 is written by reusing the sentence s_0 from revision r_0 followed by a new sentence, s_1. The third revision, r_2, reuses paragraph p_2 from the previous revision and creates two new paragraphs, p_3 and p_4. In addition, p3 contains a new sentence which reuses the token t_2 originally inserted in the previous revision.

Definition 1 *(A Graph-based Model for Revisioned Text Content). Given a revisioned content document, it can be represented as a* k*-partite graph, with* $k = 4$, $G = (V, E, \phi)$ *defined as follows:*

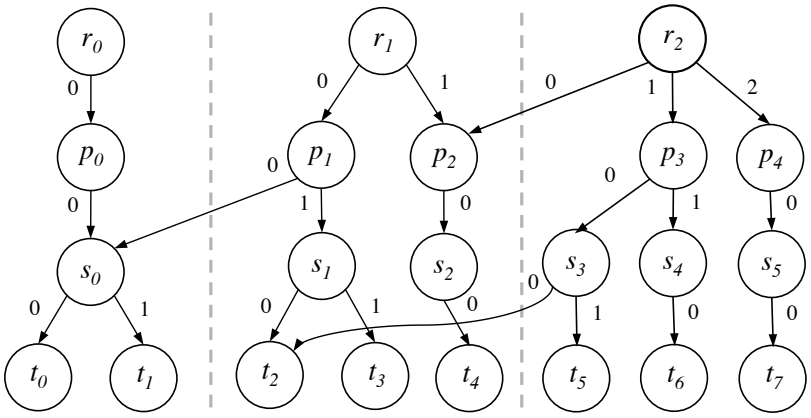


Figure 2: **Example of the revisioned content graph.** Revisions are represented by nodes r, paragraphs by p, sentences by s, and tokens by t. Arcs between nodes correspond to the *containment* relation. Labels on arcs represent the relative position of a content element in the revision.

- *The set of vertices* V *in* G *is composed of four different subsets* R, P, S, T, *i.e.,* $V = R \cup P \cup S \cup T$. *The subset* R *represents the revisions of a given document,* P *the paragraphs of the document,* S *the sentences that compose the paragraphs, and* T *the tokens (words, special characters, etc.) in the sentences. The subsets* R, P, S, T *are pairwise disjoint.*

- *The set of arcs* E *in* G *is partitioned into* $k-1$ *cuts as follows:* $E = \langle R, P\rangle \cup \langle P, S\rangle \cup \langle S, T\rangle$. *The arcs in* G *represent the relationship of containment, e.g., if* $p \in P$, $s \in S$ *and* $(p, s) \in E$ *then the paragraph* p *contains the sentence* s.

- *A labeling mapping* $\phi : E \to \mathbb{N}_0$ *over the arcs in* G *represents the relative position of a token, sentence or paragraph in a sentence, paragraph or revision, respectively. Each arc is labeled only once, therefore these labels are not updated.*

- *Additionally, it is necessary to keep record of the sequence in which the revisions were generated. Since adding arcs between revision nodes violates the partite graph definition, we represent this information by annotating the revision nodes with a labeling function* $\mathtt{label} : R \to \mathbb{N}_0$ *such that if revision* r_i *is a predecessor of revision* r_j, *then the following condition is met:* $\mathtt{label}(r_i) < \mathtt{label}(r_j)$.

4.2.3 *Restrictions Over the Model*

In the following we present the restrictions to guarantee consistency within the model. We refer to paragraphs, sentences or tokens as "content elements".

The first property refers to the number of content elements within a revision. Particularly, this property allows the definition of an empty revision (with no content elements).

Property 1 *Given* R *as the set of revision vertices (according to Definition 1), the number of arcs that leave a revision vertex (denoted* $deg^+(\cdot)$*) must be greater than or equal to zero.*

$$\forall v \in R(deg^+(v) \geqslant 0) \tag{1}$$

The second property restricts the existence of empty paragraphs or sentences, i.e., each paragraph or sentence must contain at least one content element.

Property 2 *Given* P *as the set of paragraph vertices and* S *as the set of sentence vertices (according to Definition 1), the number of arcs that leave a paragraph or sentence vertex (denoted* $deg^+(\cdot)$*) must be greater than zero.*

$$\forall v, v \in P \ \vee \ v \in S(deg^+(v) > 0) \tag{2}$$

The following property establishes that paragraphs, sentences or tokens must be associated with at least one revision, paragraph or sentence, respectively.

Property 3 *Given* P *as the set of paragraph vertices,* S *as the set of sentence vertices and* T *as the set of tokens (according to Definition 1), the number of incoming arcs of a paragraph, sentence or token vertex (denoted* $deg^-(\cdot)$*) must be greater than zero.*

$$\forall v, v \in P \ \vee \ v \in S \ \vee \ v \in T(deg^-(v) > 0) \tag{3}$$

The last property refers to the labelling of the arcs that leave a given vertex. This property states that each content element (paragraph, sentence or token) can only occupy a single relative position.

Property 4 *Given* R, P, *and* S *as the set of revision, paragraph, and sentence vertices, respectively (according to Definition 1), the label of an arc that leaves a vertex in* R, P *or* S *must be between zero and the number of arcs that leaves that vertex. The set of arcs that leave a vertex is denoted as* $d^+(\cdot)$*. Moreover, the labels of the arcs that leave a given vertex must be unique.*

$$\begin{aligned}\forall v, v \in R \ \vee \ v \in P \ \vee \ v \in S(\exists e \in d^+(v)(0 \leqslant \phi(e) < deg^+(v)) \\ \wedge \ \forall e_1, e_2 \in d^+(v)(e_1 \neq e_2 \rightarrow \phi(e_1) \neq \phi(e_2))\end{aligned} \tag{4}$$

4.2.4 *Operations Over the Model*

We define four different operations over the model that correspond to the actions that can be performed by editing a document. In the following, $\mathtt{path}(a, b)$ is defined as the set of arcs from vertex a to reach vertex b. The first operation defines the creation of a new (initially empty) revision, which consists of adding a vertex to the set of revision vertices.

Definition 2 *(Creation of a New Revision). Let* r_{i-1} *be the last revision in the graph* $G = (V = \{R \cup P \cup S \cup V\}, E, \phi)$*. The operation* $\mathtt{createRevision}(r_i)$ *represents the creation of a new revision (denominated current revision) and is defined as follows:*

$$R := \{r_i\} \cup R \tag{5}$$

After the newly created revision r_i *is added to the set of revisions, the corresponding label of* r_i *is assigned as follows:*

$$\mathtt{label}(r_i) := |R| - 1 \tag{6}$$

The following operation allows creating a new paragraph, sentence or token in a certain position of a given revision, paragraph or sentence, respectively. This operation consists of adding a vertex to the corresponding vertex partition, and an edge in the corresponding arc cut annotated with the position of the element.

Definition 3 *(Creation of Content). Let* x *and* y *be content elements such that* y *is a new element to be added in* x *at position* α*,* (x, y) *denoting an arc between* x *and* y*. The operation* $\mathtt{createContent}(x, y, \alpha)$ *in* $G = (V = \{R \cup P \cup S \cup V\}, E, \phi)$*, with* $\alpha = \phi((x, y))$*, is defined as follows:*

$$\begin{cases} P := \{y\} \cup P \ \wedge\ \langle R, P\rangle := \{((x, y), \alpha)\} \cup \langle R, P\rangle & \textit{if } x \in R \\ S := \{y\} \cup S \ \wedge\ \langle P, S\rangle := \{((x, y), \alpha)\} \cup \langle P, S\rangle & \textit{if } x \in P \\ T := \{y\} \cup T \ \wedge\ \langle S, T\rangle := \{((x, y), \alpha)\} \cup \langle S, T\rangle & \textit{if } x \in S \end{cases} \tag{7}$$

Where $\langle R, P\rangle \subseteq E$*,* $\langle P, S\rangle \subseteq E$*, and* $\langle S, T\rangle \subseteq E$ *are denominated arc cuts. In addition, the creation of an element* y *in a given revision* r_i *meets the following condition:*

$$\exists \rho(\rho \in \mathtt{path}(r_i, y)) \tag{8}$$

The third operation defines the action of copying, or reintroducing, content from an old revision. This operation consists of creating an edge from the content element of the current revision to the copied element, and labeling the edge with the relative position of the element in the current revision.

Definition 4 *(Copying Content from an Old Revision). Let* r_i $(i > 0)$ *be the current revision in the graph* $G = (V = \{R \cup P \cup S \cup V\}, E, \phi)$ *and* r_{i-k} $(0 < k \leqslant i)$ *an old revision. Let* x *and* y *be content elements such that* y *is an element copied from revision* r_{i-k} *in the element* x *of revision* r_i *at position* α*. The operation* $\texttt{copyContent}(x, y, \alpha)$ *in* G*, with* $\alpha = \phi((x, y))$*, is defined as follows:*

$$\begin{cases} \langle R, P\rangle := \{((x, y), \alpha)\} \cup \langle R, P\rangle & \text{if } x \in R, y \in P \\ \langle P, S\rangle := \{((x, y), \alpha)\} \cup \langle P, S\rangle & \text{if } x \in P, y \in S \\ \langle S, T\rangle := \{((x, y), \alpha)\} \cup \langle S, T\rangle & \text{if } x \in S, y \in T \end{cases} \tag{9}$$

Where $\langle R, P\rangle \subseteq E$, $\langle P, S\rangle \subseteq E$, *and* $\langle S, T\rangle \subseteq E$ *are denominated arc cuts. In addition, copying an element* y *from revision* r_{i-k} *to revision* r_i *meets the following condition:*

$$\exists \rho(\rho \in \mathtt{path}(r_{i-k}, y)) \wedge \exists \rho'(\rho' \in \mathtt{path}(r_i, y)) \tag{10}$$

The last operation is the deletion of content, which models the case when content from the previous revision is removed. This operation requires no alteration on the structures of the model, since elements are never removed from revisioned content.

Definition 5 *(Deletion of Content). Let* r_i $(i > 0)$ *be the current revision in the graph* $G = (V = \{R \cup P \cup S \cup V\}, E, \phi)$ *and* y *the element from the previous revision to be removed. The deletion of* y *in* r_i *of* G *is denoted as* $\texttt{deleteContent}(r_i, y)$ *and meets the following condition:*

$$\exists \rho(\rho \in \mathtt{path}(r_{i-1}, y)) \wedge \nexists \rho'(\rho' \in \mathtt{path}(r_i, y)) \tag{11}$$

4.3 COMPUTING PROVENANCE IN REVISIONED CONTENT

This section describes the implementation of an provenance attribution algorithm based on the presented model.

4.3.1 *The Provenance Attribution Problem*

The provenance attribution problem consists of identifying for each token the revision in which the token originated. This problem has been previously introduced [36], where each token is annotated with an origin label denoted as Θ. In the following we devise a theoretical solution to the provenance attribution problem for a given token, built on top of the proposed graph-based model.

Theorem 1 *(A Solution to the Provenance Attribution Problem). Let* $G = (V, E, \phi)$ *be the graph of a given revisioned content, modeled according to Definition 1. The provenance of a token* t *can be determined by identifying all the revisions where the token occurs and selecting the revision that was generated first in sequential order, i.e., the revision with the minimum label.*

$$\forall t \in T(\Theta(t) := \min\{\mathtt{label}(r_i) | \exists \rho(\rho \in \mathtt{path}(r_i, t)) \wedge r_i \in R\})$$

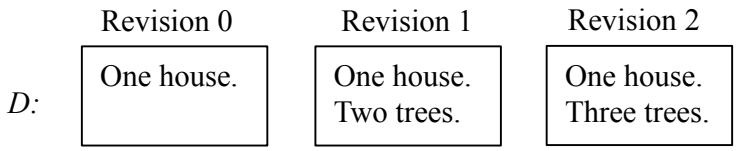


Figure 3: **Example of a document** D **with revisioned content.** D contains three revisions, each one with a single paragraph.

Proof 1 *We want to demonstrate that if* t ($t \in T$) *originated in revision* r_i ($r_i \in R$)*, then* $\Theta(t) = \texttt{label}(r_i)$*. By contradiction, lets assume that* t *originated in* r_i*, but* $\Theta(t) \neq \texttt{label}(r_i)$*. Therefore, we have two cases:* $\Theta(t) < \texttt{label}(r_i)$ *or* $\Theta(t) > \texttt{label}(r_i)$*. Furthermore, there exists a revision* r_j ($r_j \in R$) *such that* $\Theta(t) = \texttt{label}(r_j)$*. By hypothesis,* t *did not originate in* r_j *but in* r_i*, and by Definition 3 there exists a path from* r_i *to* t*. Therefore,* t *must have been copied from* r_i *to* r_j*. According to Definition 4, an element can only be copied from an old revision, thus the case* $\Theta(t) < \texttt{label}(r_i)$ *is discarded. In the other case, we can affirm that* r_i *is a predecessor of* r_j*, therefore* $\min(\texttt{label}(r_i), \texttt{label}(r_j)) = \texttt{label}(r_i)$*. By the definition of* $\Theta(t)$*, the only possibility for not selecting* r_i *as the origin of* t *is that there does not exist a path from* r_i *to* t *(contradiction to Definition 3).*

□

4.3.2 *Implementation of the Proposed Solution*

We have demonstrated that our proposed model provides a straightforward solution to the provenance attribution problem in revisioned content. In the following we devise an algorithm to build this model, while generating origin labels of tokens simultaneously. The source code and further information are available online.[6]

Algorithm 1 outlines our proposed solution, WikiWho, an algorithm that constructs a graph according to Definition 1 to represent a document with revisioned content. WikiWho follows a breadth-first strategy (BFS) to build the graph structures for each revision and assigns the corresponding origin labels to each token. To illustrate the execution of Algorithm 1, consider the revisioned content presented in Figure 3. In this example, the document D is composed of Revision 0, Revision 1 and Revision 2. The algorithm starts processing Revision 0, and creates the corresponding revision node r_0 (Algorithm 1, line 4). Then, the content is split into paragraphs; in our example there is only one paragraph (p_0), which is split into a single sentence (s_0). Once the algorithm has tokenized all sentence nodes, it proceeds to calculate the `diff` operation (line 13) between the current text and token nodes from the previous revision.[7] For the first revision, this

6 http://f-squared.org/wikiwho/, https://github.com/maribelacosta/wikiwho

7 Via the *difflib* Python library: http://docs.python.org/2/library/difflib.html

Algorithm 1: WikiWho Algorithm

Input: A document D with revisioned content $r_0, r_1, ...r_{n-1}$.
Output: A graph $G = (V, E, \alpha)$ representing the authorship graph for D.

```
1  Create an empty graph G = (V, E, ϕ)
2  Create an empty queue Q
3  for i in 0, 1...n − 1 do
4      G.createRevision(r_i)
5      label(r_i) ← i
6      y′ ← tokenize(r_i)
7      Enqueue (r_i, y) onto Q for all y in y′
8      x_prev ← NULL
9      diffed ← FALSE
10     while Q is not empty do
11         (x, y) ← Q.dequeue()
12         if x is a sentence ∧ !diffed then
13             Calculate diff of unmarked tokens of r_{i−1} against
               unmarked tokens of r_i (i > 0)
14             diffed ← TRUE
15         if x = x_prev then
16             α ← α + 1
17         else
18             α ← 0
19             x_prev ← x
20         if y ∈ V ∧ y is not marked then
21             G.copyElement(x, y, α)
22             Mark all the nodes reachable from y, including y.
23         else
24             G.createElement(x, y, α)
25             if y is a token then
26                 Θ(y) ← label(r_i)
27             else
28                 z′ ← tokenize(y)
29                 Enqueue (y, z) onto Q for all z in z′
30     Unmark all the marked nodes
31 return G
```

operation corresponds to `diff('', 'One house .')`, i.e., empty content `diffed` vs. the tokens of r_0. The `diff` output states that all the tokens in revision r_0 are new and the algorithm creates the corresponding token nodes (line 26), and annotates them with $\Theta = 0$. The current state of the graph is presented as the leftmost of the three sections of Figure 4.

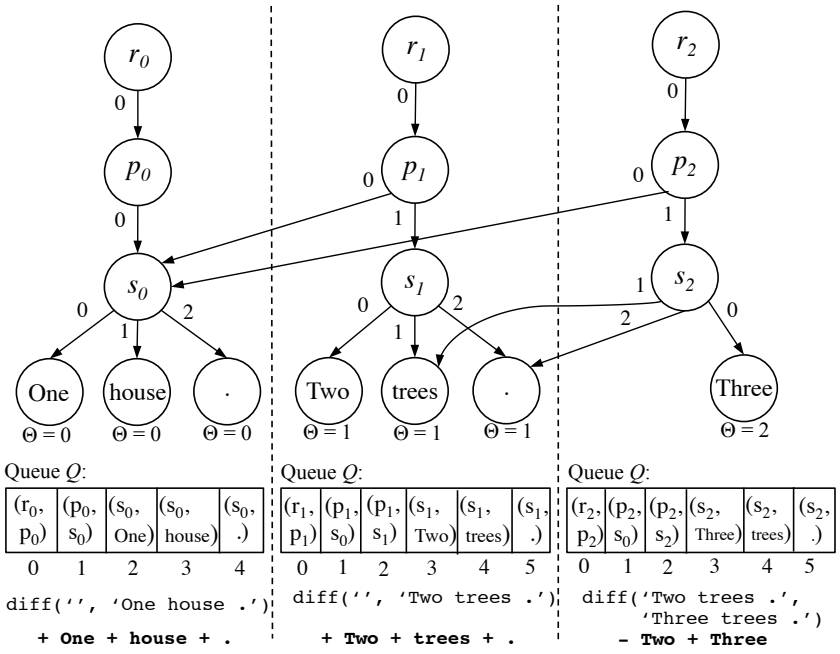


Figure 4: **Execution of WikiWho for the example from Figure 3.** Sections delimited by dashed lines represent the state of the graph after each revision. At the bottom, the progress of the queue Q and the output of the `diff` for each revision iteration are depicted.

After processing Revision 0 in Figure 3, in the next iteration the algorithm creates the revision node r_1. In this revision the paragraph has changed w.r.t. to the previous revision, therefore the node p_1 is created. One of the sentences of p_1 corresponds to s_0 – created in revision r_0 – and the algorithm marks all the nodes reachable from s_0, including s_0. This is a case of reuse of content and it is detected by the Algorithm 1 on line 22, since the analyzed vertex is already in the graph and has not been used previously in the current revision since it is not marked. The other sentence in p_1 is new, therefore the node s_1 is created. At this step, the `diff` is calculated over the sentence `'Two trees .'` and the set of unmarked token nodes from r_1 (which is $\emptyset$). The three tokens are identified as new and annotated with $\Theta = 1$. The state of the graph at the end of this iteration is illustrated by the combination of the left and the middle section of Figure 4.

In the last iteration of the example, the node r_2 is created. This revision contains a new paragraph p_2, composed of s_0 and a new sentence s_2. After processing the sentences, the algorithm calculates `diff('Two trees .', 'Three trees .')`. Note that the sentence `'One house .'` is not considered by the `diff`, since these nodes were marked when s_0 was processed. According to the `diff`, only the token `'Three'` is new and is annotated with $\Theta = 2$. Figure 4 depicts the current state of the graph.

4.3.2.1 *Time Complexity of the Proposed Solution*

In each iteration, the WikiWho algorithm performs two types of operations: the graph-based operations, and the diffing of text (tokens) that did not match with the content of previous revisions. Assuming that the graph is represented as an adjacency list [32] – in which every vertex stores a list of its adjacent vertices – all the operations over the graph presented in Section 4.2.4 have time complexity $\mathcal{O}(1)$. However, in order to decide whether content is new or copied from a previous revision, Algorithm 1 (line 20) verifies whether a given node already belongs to the graph ($y \in V$). Regarding the diffing of the text, in each iteration Algorithm 4.3 performs the `diff` (line 13) of unmatched (unmarked) tokens from current and previous revision. In the following, we present an analysis of time complexity of the proposed solution for the worst and average cases and we denote:

- $|T|$ is the total number of tokens in the revision history of a given article,
- n is the total number of revisions of a given document,
- the time complexity of the `diff` algorithm is originally defined as $O(NM)$ [21] where:
  - N is the sum of the length of the two texts that are diffed,
  - M is the size of the minimum edit script from one text to the other.

Worst Case. The worst case occurs when each token is written in a different paragraph *and* all the content has changed from one revision to another, therefore everything has to be diffed. In this case, the algorithm always visits all the nodes in each iteration; the time complexity of these operations is then $O(n|T|)$. In addition, in each iteration the algorithm performs the `diff` of two consecutive revisions, i.e., the algorithm performs $n-1$ `diff` comparisons. For simplification, consider that the average size of a revision is $\frac{|T|}{n}$ and the size of the minimum edit script for each `diff` operation is M. Taking into consideration the time complexity of the `diff` algorithm [21], the time complexity of these operations in the proposed solution is $\mathcal{O}(n-1 \cdot (\frac{|T|}{n} + \frac{|T|}{n}) \cdot M) \sim O(|T|M)$

$$\underbrace{O(n|T|)}_{\text{Graph operations}} + \underbrace{O(|T|M)}_{\text{Diff operations}}$$

which is equivalent to:

$$O(|T|(n+M))$$

Average Case. We have observed that frequently, only small portions of content are changed to actual new content from one revision

to another.[8] Such cases include, for instance, spelling corrections or replacements of a few words in one sentence. Rewriting larger sequences of text without simply moving them or reintroducing old content are far more uncommon. Hence, in the average case, since most of the content (usually paragraphs) remains the same, the algorithm visits portions of the graph in each iteration; the time complexity of this is $O(\omega n|T|)$ with $0 < \omega < 1$ since the algorithm always visits certain nodes in the graph, but not all of them. Analogously, since in each iteration the algorithm performs the `diff` of portions of tokens, the time complexity of these operations is $O(\epsilon|T|M)$ with $0 < \epsilon < 1$. Notice that $\epsilon > 0$ models that new content is added to revisions *and* tokens from previous revisions are removed. The case $\epsilon = 1$ models when for all revisions the algorithm performed the `diff` of all the token; this would only happen if the content is entirely new in each revision.[9] The time complexity of the proposed solution in the average case is:

$$\underbrace{O(\omega n|T|)}_{\text{Graph operations}} + \underbrace{O(\epsilon|T|M)}_{\text{Diff operations}}$$

which is equivalent to:

$$O(|T|(\omega n + \epsilon M))$$

with $0 < \omega < 1$ and $0 < \epsilon < 1$.

4.3.2.2 *Representation of Content Nodes*

As explained earlier, revision nodes are uniquely annotated with a `label` (line 5 of Algorithm 1), usually representing the sequential order in which the revisions were generated. In a Wiki environment, the revision identifier provided may serve as a label in WikiWho. Paragraph and sentence nodes are identified by a hash value. The hash value is the result of applying the MD5 algorithm [125] over the content of the paragraph or sentences, respectively. Token nodes contain the actual text of the revisions, i.e., the single tokens that compose the revisioned content. WikiWho annotates the token nodes with their corresponding origin label (Θ), as shown on line 26 of Algorithm 1. This avoids the calculation of all the paths from revision nodes to a token to retrieve its provenance information.

4.3.2.3 *Implementation of Operations*

One crucial operation of WikiWho is determining whether a certain node y belongs already to the graph (line 22 of Algorithm 1). Depend-

8 During our experiments described in Section 4.4.

9 We have observed that these cases may occur from one revision to another when under vandalism attacks. Section 4.3.4 presents simple heuristics implemented in our work to avoid diffing vandalistic content.

ing on the type of the node, this step can be implemented differently. When y is a paragraph or a sentence, the algorithm only checks the corresponding node partition, i.e., if $y \in P$ or $y \in S$, respectively. When y is a token, the decision whether the token is new or not relies on the output of the `diff` operation (line 13).

On lines 6 and 30, Algorithm 1 performs the tokenization of the text unit that is currently being processed. Tokenization refers to the process of splitting the text into more fine-grained units. We define a token as the smallest unit, be it indivisible. Details regarding the definition of grammatical units are discussed in Section 4.3.3.

Once the graph is built, accessing the provenance information for each computed revision is straightforward. The origin labels of the content in r_i can be retreived by traversing the graph G with any search approach, e.g., depth-first search (DFS), considering r_i as the root node and visiting the nodes in the order induced by ϕ.

4.3.3 *Design Issue: Tokenization*

To achieve provenance attributions that correspond to the "original" revision of a given token, it is important to chose the tokenization very carefully in respect to the specific context of the system and its usage. An overly fine-grained tokenization – e.g., on character level – might prove counter-productive if it is not necessary to determine the origin of single characters and make the interpretation of the results too complex for end-users. On the other hand, using only demarcations such as periods to identify sentences as the smallest unit can prove too coarse. Whole tokens can in that case spuriously be reattributed to new authors even when only minor adjustments are applied. As the optimal tokenization can vary immensely between different contexts, we concentrated here on the Wikipedia environment. We believe, however, that the presented design choices are applicable as well for other CWS that follow roughly the same patterns of editing and collaborative writing, particularly other Wikis.

When processing the complete source text of articles, as we do with WikiWho, it is not only important to take into account the intricacies of natural language (as it appears on the article front-end) and a corresponding optimal tokenization, but also the function-oriented Wikitext of the Mediawiki software. Small changes in the markup can entail important changes in the front-end of an article, be it for instance the inclusion of a template via only a few pasted characters or the setting of links. Consider the following example, where a contributor writes the word Germany and another contributor in a subsequent revision adds markup content to represent the same word as an internal link ([[Germany]]) to the respective article. Using only white spaces as separators would lead to counting the latter string as a new token; using a similarity-metric like a Levenshtein distance might lead to an

attribution of the whole string to the former author. What actually happens here is that the word "Germany" was written by one author and the link (specified with "[[" and "]]") was set by another. Both are essentially important and distinct actions in Wikipedia. Many more of these examples could be given, pertaining to templates, language links, references and numerous others.

Besides white spaces we thus chose the most commonly used functional characters of the Wiki markup as delimiters, such as "|", "[", "]", "=", "<", ">", to name some. We further split sentences (as defined in our model) at common sentence delimiters such as ".", "?", etc., and paragraphs at double line breaks, which are used in Wikipedia to begin a new paragraph in the text. All delimiters were also treated as tokens, as they fulfill important functions in the text. We determined all of these demarcations after extensive testing with real article data until we reached a splitting we deemed optimal to achieve the best balance of precision and efficiency.[10]

4.3.4 *Optimization for Wiki Environments: Vandalism Detection*

The most expensive operation of the proposed algorithm is the `diff`, as shown in the time complexity analysis presented in Section 4.3.2.1. We are interested in detecting those vandalism attacks that change large amounts of content from one revision to another, significantly affecting the performance of the `diff` operation. There are different types of vandalism in Wikipedia, such as removing large parts of a page, or modifying a page in a way that adds a lot of vandalistic content.[11] In order to avoid the computation of the `diff` in the previous cases, we implemented two simple vandalism detection techniques that do not impose a large computational overhead and filter out only the most obstructive cases that would increase runtime notably.

PERCENTAGE OF SIZE CHANGE FROM ONE REVISION TO ANOTHER

This mechanism is triggered when a large amount of content gets removed at once, by comparing the current content size versus the size of the previous revision. An example of this type of vandalism is *page blanking*, which signifies deleting all the content of a Wiki page.[12] Since the size of the early revisions of a Wiki page can fluctuate notably, this technique is fired only when the article has reached a certain size in terms of revision count. To not filter out revisions where

10 The list appears in Text.py as part of the WikiWho algorithm implementation available at https://github.com/maribelacosta/wikiwho.

11 http://en.wikipedia.org/wiki/Wikipedia:Vandalism_types

12 http://en.wikipedia.org/wiki/Wikipedia:Page_blanking

much content is moved to a different article in good faith, we analyze the edit comment log provided in the article history dumps.[13]

TOKEN DENSITY This proposed technique aims at detecting vandalism that consists of adding large amounts of disruptive content, often composed of the same text, repeated numerous times. For these cases, we propose a measurement called *token density*, defined as follows. Consider the bag of tokens of a revision r as the result of splitting the revision's content with a tokenization mechanism. This bag can be formally represented as a multiset T_r, which consists of a set T'_r – constructed from removing duplicates in T_r – and a function $m : T'_r \to \mathbb{N}_0$ that denotes the number of times an element of T'_r occurs in T_r. We calculate the token density as follows:

$$\text{tokenDensity}(T_r) = \begin{cases} \frac{\sum_{t \in T'_r} m(t)}{|T'_r|} & \text{if } T_r \neq \emptyset \\ 0 & \text{if } T_r = \emptyset \end{cases}$$

A high token density suggests that the content is composed of a collection of repeated words. From this computation we discard tokens corresponding to Wiki markup elements, since they appear several times in a revision and might be misinterpreted as vandalism. These vandalism filters must be configured with very relaxed thresholds such that no false positives are generated, which was successfully achieved with the values set in the experiments of this work (cf. Section 4.4.2.2). Note that the objective of implementing these techniques is solely to avoid the computation of the `diff` operation on large amounts of irrelevant content. We do *not* aim at applying these techniques as general solutions for the problem of vandalism detection in Wikis.

4.4 EXPERIMENTAL STUDY

We empirically analyzed the performance of the proposed algorithm WikiWho and compared it to the algorithm "A3" by de Alfaro and Shavlovsky [36], which can be considered the benchmark for the given task at the time of writing.[14] In our experiments we report on the execution time of the evaluated algorithms as well as their precision in finding the correct revision of origin for a token. Regarding precision, this is the first evaluation for both algorithms. The datasets, gold standard and further details of the experimental results are available online.[15] For all articles analyzed in the following evaluations,

13 To avoid false positives, the size threshold in the implementation used here was set to a relatively high count of over 1000 revisions (average revision size of articles: <200, cf. https://en.wikipedia.org/wiki/Wikipedia:Pruning_article_revisions), the comment text patterns are .*?moved.*? and .*?redirect.*?.

14 Retrieved from: https://sites.google.com/a/ucsc.edu/luca/the-wikipedia-authorship-project

15 http://f-squared.org/wikiwho

the full history in XML format was retrieved from the English Wikipedia via the MediaWiki `Special:Export` mechanism.[16]

4.4.1 *Evaluation of Precision*

In the following we explain the three-step procedure to construct and validate the gold standard. We measured the quality of the evaluated algorithms by comparing their provenance attributions with the results of the gold standard.

4.4.1.1 *Creating a Gold Standard for Provenance in Revisioned Content*

To create the gold standard we selected 40 English Wikipedia articles. Ten articles each were randomly picked from the following four revision-size ranges:[17] articles with i) over 10,000 revisions, ii) 5,000-10,000 revisions, iii) 500-5,000 revisions and iv) 100-500 revisions. The reason for this stratified sampling process was to include a sufficient number of articles that present a challenge to the algorithms when picking the correct revisions of origin, as a higher number of revisions naturally increases the difficulty of the task, as more candidate solutions exist.[18] The latest revision at the point of retrieval of the articles was the "starting revision" for whose tokens the provenance was determined.The text plus markup of each of the 40 articles was split into tokens as described in Section 4.3.3. Out of this tokenized content, for each article, six instances were randomly selected, resulting in a total of 240 tokens. For each of these, the final gold standard contains the revision in which they first appeared (revision of origin) and the starting revision. To assign the correct revision of origin to all of these tokens, we followed three consecutive steps.

STEP 1: Three researchers of the AIFB institute manually searched the "Revision History"[19] of the respective 40 articles for the origin of each of the 240 tokens in the gold standard independently from each other. No common interpretation of what constitutes a "correct origin" was agreed on beforehand but was entirely up to the individuals. If the researchers initially disagreed on the correct origin of a token, this disagreement could in most cases be resolved, as it in almost all cases stemmed from one researcher overlooking an earlier addition of the token. Only in three cases was this not achieved, so

16 http://en.wikipedia.org/wiki/Special:Export

17 The "random article" feature of Wikipedia was used. Redirect or disambiguation pages where skipped.

18 Articles under 100 revisions are not challenging for the task. We did sample test-runs with non-crowdsourced test answers and both algorithms scored very close to a precision of 1.0.

19 Example for the article "Korea"': https://en.wikipedia.org/w/index.php?title=Korea&action=history

that they were excluded from the gold standard and replaced with new randomly selected tokens.[20]

STEP 2: Next, the gold standard was validated by users of the crowdsourcing platform Amazon Mechanical Turk (hereafter called "turkers").[21] We selected two random tokens for each article in the gold standard. We then created a Human Intelligence Task (HIT) on Mechanical Turk for each of these 80 tokens to be validated by 10 distinct turkers each. We paid 15 US$ cents and selected turkers with a past acceptance rate of over 90% and at least 1,000 completed HITs.[22] A HIT was composed of the following elements (cf. Figure 5):

a) A link to a copy of the starting revision of the Wikipedia article with the highlighted token (Fig. 5c). If the token only appeared in the markup, we represented an excerpt of the markup as a picture next to the front-end text where it appears in the article HTML, explaining to look for it in the markup.

b) A link to the Wikipedia "Difference View" of the revision of origin proposed by the gold standard (Fig. 5d). It shows which changes the edit introduced that lead to that revision.[23]

c) Detailed instructions explaining how to use the above mentioned pages and a description of what solution was sought.

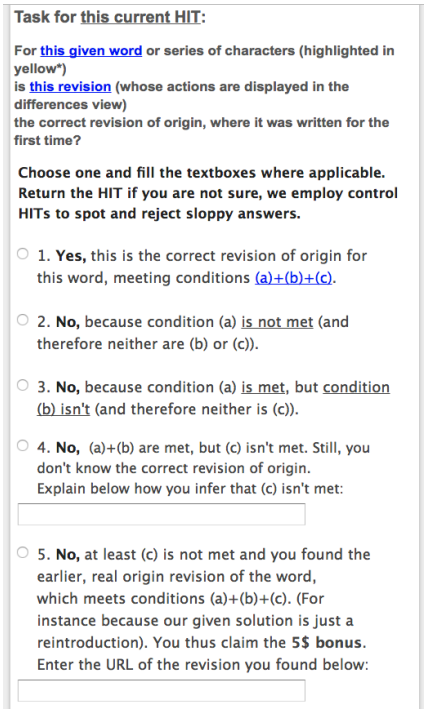
Three different conditions had to be fulfilled by the proposed revision: First, a string equivalent to the token should indeed have been added in that revision (and not only be moved inside the article text). Second, the token added should be the "same" token as highlighted in our gold standard solution. We explicitly left it open to the turkers to interpret what "same" meant to them and gave only one simple, unambiguous example, explaining that not any string matching the gold standard token was looked for but the specific token in the context that it is presented in (e.g., if the token was a specific "and", we would not be looking for any "and"). The third condition was that the token was actually added in the given revision for the *first* time and not just reintroduced, e.g., in the course of a vandalism revert. Turkers could chose between one answer option indicating "correct revision", three choices pointing out the violation of any of the three conditions and a fifth option with a text box if they had found a revi-

20 The situation for the rating of these three tokens was either that (i) it was not possible to objectively decide between an early addition plus deletion of the token and a notably later readdition as the source or (ii) the context (neighbors) of the token changed heavily, but gradually, so that it was not possible to unanimously answer the question "Is this still the same token?". The exclusion of tokens that are even hard for human coders to agree on of course raises the suspicion of such cases also being hard to determine by any of the tested algorithms and could therefore have lead to a slightly better outcome for the tested cases in the evaluations presented hereafter.

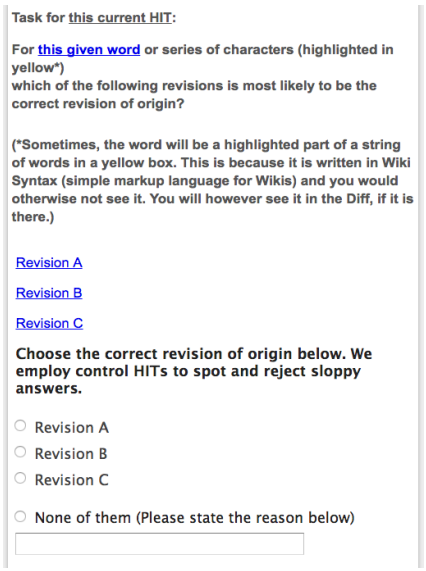
21 http://www.mturk.com

22 The pay rate was the result of a number of tries with rates at 10 and 13 cents that did not attract enough turkers.

23 Example diff: https://en.wikipedia.org/w/index.php?title=Korea&diff=574837201

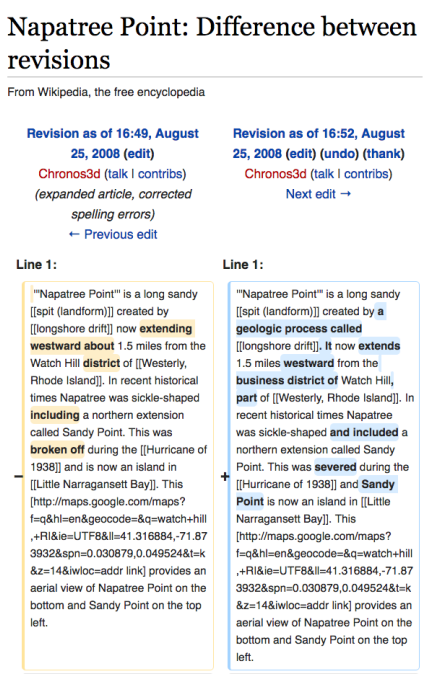


(a) Step 2: Question and answer options



(b) Step 3: Question and answer options

(c) An instance of the highlighted word to be searched



(d) An instance of a presented text difference view that could be a candidate for the first origin of a token

Figure 5: **Screenshots of the Mechanical Turk tasks** for steps 2 and 3 of the evaluation of accuracy. The complete and more detailed task descriptions including instructions can be found in Appendix A.1.

Table 1: Results of step 2

Agreement per 10 asked turkers	Number of tokens with respective score
10/10	9
9/10	56
8/10	12
7/10	3
< 7/10	0

sion that was more likely to be the origin of the token (Fig. 5a). For option 5, we offered a bonus payment of 5 US$ to propose a better solution than the one presented and gave a detailed manual on how to search the revision history page of a Wikipedia article by hand as well as a list of tools by the Wikimedia community that can be helpful with the task.

Results of Step 2: The 800 answers we received as the result of this experiment included 24 answers suggesting a better solution, but none of them fulfilled all three conditions. We therefore reposted these HITs once we assessed them. As these turkers had spent 17 minutes on average for the task and obviously had tried to find a better solution, they were paid bonuses ex-post. Overall, turkers spent from 40 seconds to 13 minutes solving the task, with an average of 4 minutes and 49 seconds. Turkers thus spent considerable time assessing the correctness of the presented solutions.[24]

In Table 1, we report the results of aggregating the answers of the "incorrect" options (option 5 was handled as mentioned above). On average, the solutions received 89% agreement. 65 tokens received nine to ten out of ten agreement votes. 12 solutions received 8/10 and three received 7/10 "correct" votes. In the latter cases the disagreeing turkers pointed in 7 of 9 answers at the lack of a matching string being added in the suggested revision, although this was in fact the case.[25] Overall, we consider the result of this experiment to compellingly support the proposed gold standard solutions.

Step 3: As a further test we ran the WikiWho algorithm as well as the A3 algorithm (in two different variants), as explained in the following Section 4.4.1.2. For 67 of the 240 tokens in the gold standard at least one of the algorithms produced a result deviating from the gold standard. For all of these 67 tokens we set up a Mechanical

24 This excludes 12 turkers whose HITs were rejected and reposted for obviously incorrect answers, such as choosing option 5 and not reporting a better solution.

25 We believe this could have been because of particular nature of the respective Wikipedia Difference Views, where the token was hard to track.

Turk experiment with the same settings as explained in Step 2. In this HIT, however, we presented the turkers with three differing possible revisions of origin and asked them which one was most likely correct or if none of them was (option 4, cf. Fig. 5b). One of the three solutions was always the gold standard answer and one or two were solutions by one of the algorithms, depending on how many algorithm results disagreed. If only two differing solutions were available, the third one was filled with an incorrect control answer. Answer positions were randomly changed in each HIT.

Results of Step 3: 670 single answers were retrieved for the 67 tokens. The general agreement score for the gold standard solution was 81%, with 7/10 or more votes validating the gold standard as correct for 63 tokens. Given the nature of the task and the different possible interpretations, we consider the gold standard to have gained a solid affirmation for these tokens. In four cases, however, the turkers disagreed decisively with the gold standard. In two of these instances, there was complete disagreement over the right solution, while in two other examples four users each endorsed the differing WikiWho and the differing A3 solution, respectively. We therefore removed these tokens from the following evaluation in 4.4.1.2 since a certain solution for these cases is lacking.[26] The remaining 63 tokens achieved an agreement of 83%.

As a conclusion to these three steps of quality assurance we can assume that the gold standard is sufficiently robust to test algorithm precision against it. We are however publishing the gold standard and encourage the community to assess and expand it further.[27]

4.4.1.2 *Measuring the Precision of the Algorithms*

After validating the gold standard, WikiWho and A3 algorithms were tested for their ability to correctly detect the revisions of origin for each token. The evaluation metric was precision defined as: $p = \frac{TP}{TP+FP}$ where a true positive (TP) means that the provenance label computed by the algorithm is matching the gold standard described in 4.4.1.1 and otherwise is a false positive (FP).

Three articles in the gold standard from the revisions-size bracket over 10,000 had to be excluded due to technical reasons and are hence exempt from all following experiments to guarantee the same data basis.[28] The remaining 37 articles encompass 218 tokens.

The A3 algorithm we retrieved includes a filter that seems to be intended to remove the Wiki markup that does not appear on the HTML front-end of an article.[29] More important is however that all

26 We marked these cases in the published gold standard accordingly.

27 See http://f-squared.org/wikiwho/#paper

28 The A3 algorithm did not process these articles despite several intents to resolve the issue. The files were unaltered XML-dumps from the Wikipedia servers. The articles are "Vladimir Putin", "Apollo11" and "Armenian Genocide".

29 The filter is not described in [36].

Table 2: Precision comparison of WikiWho and A3

x ∈	ALL	[10k,∞)	[5k,10k)	[500,5k)	[100,500)
Full sample					
p WikiWho	0.95	0.97	0.93	0.95	0.95
p A3 MF-OFF	0.77	0.77	0.64	0.76	0.87
Gain in *p* by WikiWho	0.18*	0.20*	0.29*	0.19*	0.08
Available results *n*	218	58	42	58	60
Sample restricted to output of A3 MF-ON (*n* -80)					
p WikiWho (restricted)	0.96	0.97	0.89	1.00	0.95
p A3 MF-ON	0.81	0.69	0.70	0.88	0.95
Gain in *p* by WikiWho	0.15*	0.28*	0.19	0.12*	0.00
Available results *n*	138	39	27	34	38

Notes: *n* = number of tokens, k = one thousand, *p* = precision, x = number of revisions per article, * = difference significant at 0.05 (paired t-test)

citations and references get discarded, although they appear in the front-end and can in some cases make up large parts of the article, not to mention their functional importance for the credibility of Wikipedia articles. Hence we ran one variant of the A3 algorithm with this markup filter disabled (henceforth **"A3 MF-OFF"**)[30], also because our aim was to compare WikiWho to another algorithm that is able to process the entire source text. The unaltered version of the A3 algorithm will be referred to as **"A3 MF-ON"**. A3 MF-ON yielded results for 138 of the 218 tokens as the remaining part was filtered out. We therefore compared its output to the result for the same 138 tokens by WikiWho, as can be seen in the lower part of Table 2. A3 MF-OFF produced output for the whole set and we compared it to the full results of WikiWho, listed in the upper part of Table 2.

WikiWho scores at 18% and 15% higher precision overall, respectively, for the full and the restricted token sample. As becomes evident from the results, the gain in precision by WikiWho turned out especially high for the two biggest revision-size brackets, while it

30 Apart from this change the settings used in [36] were replicated.

is lower for the 5,000 to 50,000 bracket and much lower and non-existent, respectively, for the smallest-size bracket. On one hand, this seems to indicate that for articles with up to 500 revisions, the difference between the two approaches is negligible and both have a very high precision. Given the long tail of small articles in Wikipedia, this is a very encouraging result. On the other hand, with increasing editing activity and therefore growing number of revisions of an article, it seems to become harder for the A3 algorithm to correctly determine the provenance of certain tokens, while WikiWho can sustain a high level of precision, even for articles with over 10,000 revisions. Given the steady growth of Wikipedia and the size of other revisioned content these approaches might be adaptable to, such as GitHub or large office document sharing platforms, this is an important aspect of scalability. Moreover, particularly when processing the much "dirtier" Wiki markup, WikiWho seems to have a notable advantage when it comes to precisely determining provenance.

4.4.2 *Evaluation of Execution Time*

We measured the algorithm time for computing provenance labeling of revisioned content from Wikipedia pages.

4.4.2.1 *Experimental Set-up*

We used two datasets created by retrieving the full revision history content for each article from the English Wikipedia in XML format.[16] *Dataset 1* was generated by randomly selecting Wiki pages in the article namespace that were no redirects or disambiguation pages. This dataset is comprised of 45,917 revisions in 210 articles, i.e., an average number of 219 revisions per article; the average revision size is 2,968 KB. *Dataset 2* contains the Wiki pages used in the quality evaluation presented in Section 4.4.1.1. Its articles are larger, with an average number of revisions of 5,952 and an average revision size of 461,522 KB per article. This allowed for some "heavy load" testing. This last dataset is composed of 36 articles with a total of 214,255 revisions.[31]

We defined execution time as the time elapsed between the point when the algorithm reads the first revision and the point when the provenance labeling of the last revision of a given article is computed. Both algorithms are implemented in Python and the time was measured with the `time.time()` command from the Python library. The experiments were all executed on a dedicated OS X machine with a 2.5 GHz Intel Core i5 processor and 4GB RAM.

31 We excluded again the three articles mentioned in Section 4.4.1.2 as well as "Jesus", as it would run over 5 hours for some settings.

4.4.2.2 *Algorithm Settings*

The A3 algorithm was set according to the configuration presented by de Alfaro and Shavlovsky [36].[32] The tokenization implemented by A3 uses only whitespaces as delimiters. In addition, A3 employs two types of filters. First, a content aging filter that limits the number of revisions to be analyzed by excluding the content from old revisions according to the values of the thresholds Δ_N and Δ_T; in our experiments, we used the original configuration of the algorithm ($\Delta_N = 100$, $\Delta_T = 90$).[33] Second is the Wiki markup filter, which we discussed in Subsection 4.3.3. The Wiki markup affects the amount of content to be processed in each iteration and we thus studied the performance of the algorithm with this filter on (**A3 MF-ON**) and disabled (**A3 MF-OFF**).

Regarding the WikiWho vandalism detection mechanisms (cf. Section 4.3.4), we empirically set up their thresholds by performing tests on a random article sample. In the experiments, the value for the change percentage filter was equal to -0.40, and the token density was set to 10. This resulted in 0.5% of revisions being filtered. As discussed in Section 4.3.3, the definition of tokenization units is an important factor that affects the quality as well as the performance of the algorithm. We studied the performance of WikiWho in two variations of tokenization plus one additional setting:

- **WikiWho complex tokenization (CT)**: We implemented the tokenization described in Section 4.3.3, considering the Wiki markup. This is the original algorithm we propose.
- **WikiWho simple tokenization (ST)**: Tokens are obtained by splitting the raw content using only whitespaces as delimiters. This setting was used to assess which additional load the complex tokenization adds by generating a much higher number of tokens to track.
- **WikiWho ST and content aging filter on (ST/AF-ON)**: We implemented the content aging filter described for A3, with $\Delta_N = 100$. This setting and the A3 MF-OFF configuration allow to compare the algorithms under similar conditions.

4.4.2.3 *Results*

We executed each setting 5 times and report on the average time per article, per revision and per Kilobyte. The runtime results for all settings are listed in Table 3. Figure 6 plots the time results in relation

32 I.e., with sequence length as the rarity function and a threshold equal to 4, cf. [36].

33 Δ_N limits the processed content to a maximum of N most recent revisions, while Δ_T further filters out revisions older than T days.

Table 3: Execution time of algorithm settings

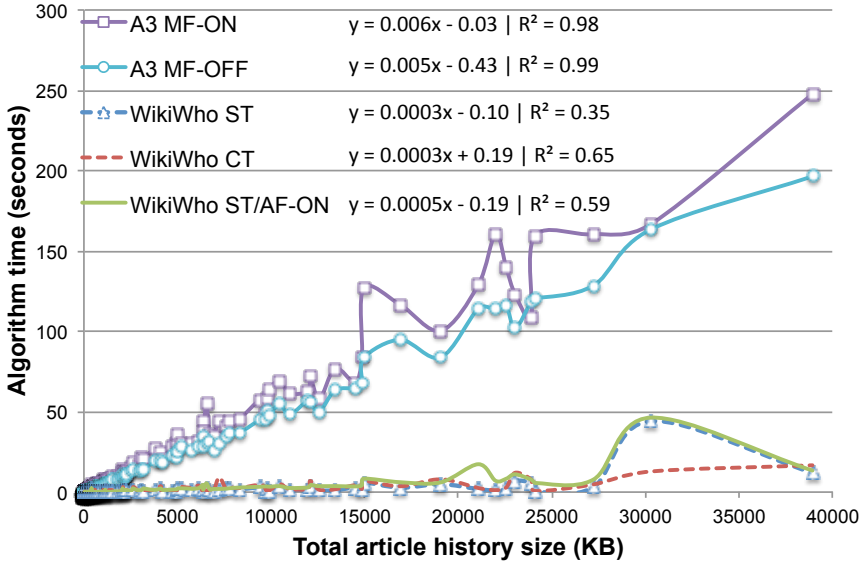
Algorithm setting	Avg. time per article (secs.)	Ratio of runtime (base: ST)	Avg. time per revision (secs.)	Avg. time per KB (secs.)
		Dataset 1		
ST	0.84	1 : 1	0.0038	$2.84x10^{-4}$
CT	1.04	1 : 1.24	0.0047	$3.49x10^{-4}$
ST/AF-ON	1.32	1 : 1.57	0.0061	$4.46x10^{-4}$
A3 MF-OFF	14.30	1 : 17.02	0.0654	$4.82x10^{-3}$
A3 MF-ON	17.69	1 : 21.05	0.0809	$5.96x10^{-3}$
		Dataset 2		
ST	184.97	1 : 1	0.0322	$4.01x10^{-4}$
CT	284.44	1 : 1.54	0.0495	$6.16x10^{-4}$
ST/AF-ON	290.97	1 : 1.57	0.0506	$6.30x10^{-4}$
A3 MF-OFF	2834.37	1 : 15.32	0.4931	$6.14x10^{-3}$
A3 MF-ON	2559.38	1 : 13.84	0.4452	$5.55x10^{-3}$

to increasing total article history size – meaning average revision size times number of revisions[34] – for both datasets.

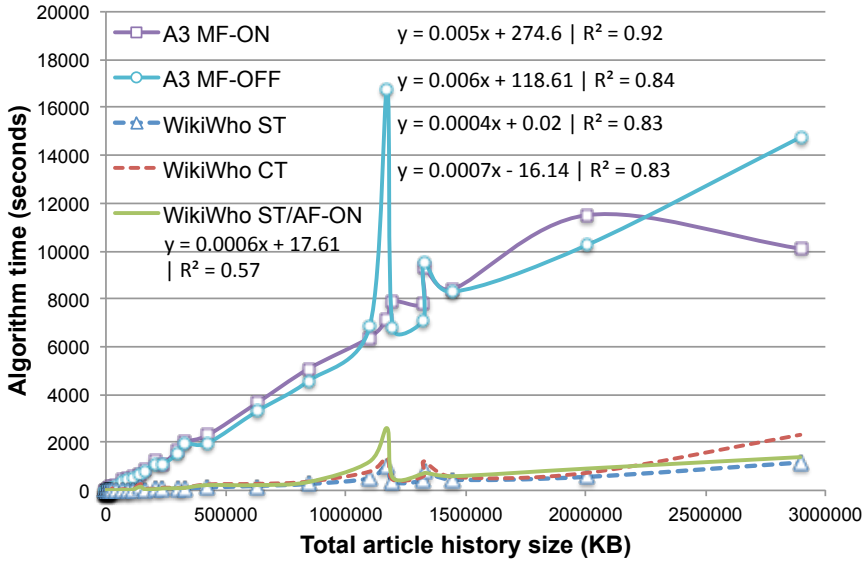
The figures include the functions for fitted linear regression lines, showing that for the A3 settings the runtime increases with growing article size by a much larger factor than it is the case for the WikiWho variants. We can observe that the behavior of all the settings in both algorithms is consistent in general and increases in a linear or almost linear fashion with an increasing content size. Fluctuations between data points suggest that the execution time is also influenced by other properties of articles, e.g., the amount of content modified from one revision to another.

The runtime decrease by WikiWho in contrast to A3 is in the range of one order of magnitude, differing over the settings. The two most comparable settings ST/AF-ON and A3 MF-OFF differ by a factor of 10.83 and 9.760, respectively for the two datasets, while the originally proposed setting CT completes the task in an even shorter time. It appears that the time filter is in fact no accelerator for the WikiWho algorithm, supposedly because it creates more overhead than is saved by not processing older revisions. The A3 algorithm shows the same behavior in *Dataset 1*. Still, for *Dataset 2* the markup filter seems to take effect, presumably because in longer revisions the amount of filtered content is larger.

34 We show total article size on the x-axis, as average revision size and number of revisions both influence the runtime. Text includes Wiki markup.



(a) Performance in *Dataset 1* (Articles randomly selected)



(b) Performance in *Dataset 2* (Articles used in quality evaluation)

Figure 6: **Algorithm execution time evaluation** for different settings of WikiWho and A3 in *Dataset 1* and *Dataset 2* – the fitted linear functions are denoted by y, respectively for the data series on the left (fit lines omitted and data points partly omitted for readability).

Overall, WikiWho is able to execute the given task of computing provenance in a very efficient manner and outperforms the A3 algorithm significantly in runtime in all variants. This is possible due to the construction of paragraph and sentence nodes comparable to creating indexes over the text. This allows to efficiently detect large chunks of text that remained unchanged between revisions, vastly reducing the number of necessary comparisons at the token level.

4.4.3 *Evaluation of Materialization Size*

Since revisioned content is in constant production – particularly in the English Wikipedia, where over 3 Million revisions are created monthly[35] – it might be useful to materialize partial computation in order to allow incremental data processing, i.e., the algorithm can be stopped at a certain point in time and then resume its execution. Therefore, we implemented a JSON serialization mechanism to optionally materialize partial computation. We measured the overhead caused by the serialization in terms of space.

4.4.3.1 *Experimental Set-up*

We used the articles contained in the two datasets presented in Section 4.4.2, and serialized the computation of the provenance labels of the whole page history for each article. We compared the serialization mechanism of WikiWho and A3 under similar conditions with the settings ST/AF-ON and MF-OFF, respectively. Both algorithms WikiWho and A3 utilize the `cjson` Python library[36] to implement the (de-)serialization mechanisms. Since we are comparing the agorithms with content aging filter set to $\Delta_N = 100$, we report on the size of the JSON serialization in relation to the size of the last $N = 100$ revisions of each article.

4.4.3.2 *Results*

Figure 7 plots the results of the materialization for WikiWho and A3 for dataset 1 (dataset 2 showed consistent results in its 36 cases). The behavior of the two algorithms is in general consistent. When the size of the revisioned content increases, the relative size of the serialization decreases exponentially. This suggests that both algorithms efficiently represent redundant content. Figure 7 further depicts the percentage difference of the WikiWho minus the A3 materialization with increasing content size. It shows a volatile behavior with a linear trend. On average, the size of the serialization is 66% for WikiWho and 56% for the A3 algorithm with respect to the total size of the last 100 revisions.

Weighing the cost of storing the results for small articles versus the average time to calculate provenance labels with WikiWho, materializing these results does not bring additional benefits; on the contrary, it incurs on extra space and time. Therefore, the proposed serialization mechanisms should be executed only when the time to compute the provenance labels over the whole article history exceeds a "reason-

35 According to the Wikimedia Statistics of June 2013: http://stats.wikimedia.org/EN/SummaryEN.htm

36 https://pypi.python.org/pypi/python-cjson

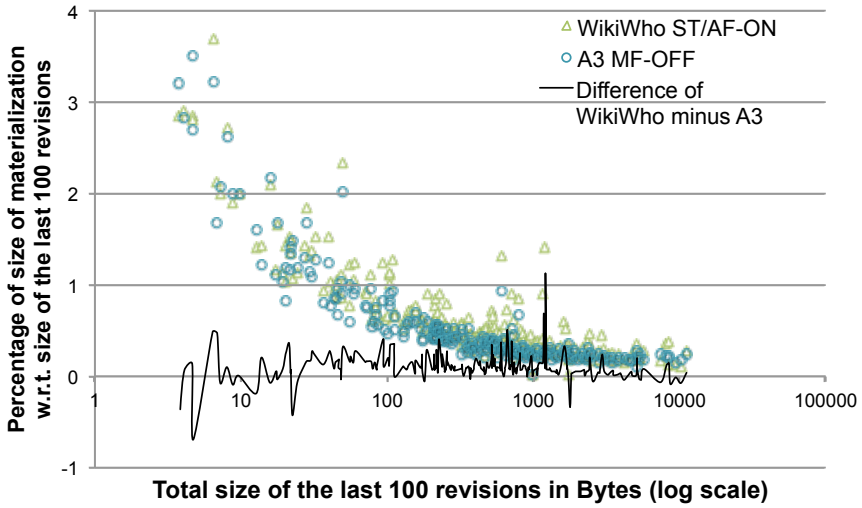


Figure 7: **Size performance in Dataset 1** (Wiki pages randomly selected) – article "Rankin County" at y-axis values 10.69 (WikiWho) and 11.13 (A3) not shown for readability.

able" response time, e.g., wait time for end users. Using the worst case linear estimation of the originally proposed setting CT for *Dataset 1* (cf. Figure 6), a hypothetical maximum runtime of 5 seconds would allow to process all articles with up to 16,033 KB complete revision history text size without the need for materialization. As far as we can estimate by a random sampling from the Wikipedia database, at least half of all articles in the English Wikipedia currently stay under this size limit.[37] The needed storage space can of course be further reduced by relaxing the runtime constraint. For articles over this limit, intervals of revisions can be determined when a materialization becomes necessary, although this is beyond the scope of this work.

4.5 CONCLUSIONS AND FURTHER DEVELOPMENT

In this chapter we have proposed WikiWho, a solution for the attribution of provenance in revisioned content. We built a graph-based model to represent revisioned content, and provided a formal solution to the provenance problem. In order to measure the quality of WikiWho, we created a gold standard of over 240 tokens from Wikipedia articles, and corroborated it via crowdsourcing. It is, to our expertise, the first gold standard of this kind to measure the precision of provenance attributions on token-level. We compared WikiWho against the state-of-the-art, exceeding it by over 10% on average in precision, and outperforming the baseline execution time by one order of magnitude. Our experimental study confirmed that WikiWho is an effective and efficient solution.

37 https://wiki.toolserver.org/view/Database_access

Inter-article content tracking: Although in this work we restricted the use of WikiWho to single articles, it is also possible to operate it over several articles in a Wiki, tracking the movement of text between different pages. Alas, this would be much more resource-intensive, as for each article revision potential revisions from all other articles have to be assessed. The feasibility of this extension is to be evaluated.

Further use cases: We used the English Wikipedia as inspiration and testing ground, yet the proposed solution can be understood as a more generally applicable method for revisioned content. We are convinced that many of the assumptions made for our use case also hold true for other Wikis and also for other revisioned content systems. Regarding different language editions of Wikipedia, we are confident that for all languages that use a similar splitting of language into word units, the employed tokenization would be applicable with comparable results. We expect this to be the case for most Romance and Germanic languages. Still, at least for other languages beyond those, this assumption will have to be evaluated and the tokenization has possibly to be adapted.

Different Wiki systems running on MediaWiki or a similar software can easily be analyzed with WikiWho in the same fashion, given they supply the structured revision-log as provided by Wikipedia. Other natural-language-based CWS should be possible to analyze likewise, given similar version histories. For collaborative code writing, it has to be explored (i) which hierarchical units of a document should optimally be used to generate the content graph (e.g., code functions instead of paragraphs, sentences, tokens), and (ii) how the tokenization should be performed. However, we see a broad array of use cases for WikiWho with minor adaptations.

Potential for improvements: Further techniques to optimize the materialization of intermediate computation might be desirable. Since each article may show different editing patterns (in terms of size and number of revisions), it is beneficial to adapt the frequency of the serialization routine for each article. In terms of the hierarchical splitting into paragraphs, sentences and words, other splitting settings could be tested to assess changes in performance and accuracy. Lastly, a different approach for finding changes on word level than the longest-common-subsequence matcher implemented by Python might speed up the process and produce even more accurate results.

WikiWho API: We set up an adapted implementation of the WikiWho algorithm that can be queried over the HTTP protocol to retrieve the provenance information of all tokens in any revision of an article in the English Wikipedia. It downloads and processes the content of all revisions up until the requested revision and returns the provenance information (ordered token list, with origin revision id, optional author name) as a JSON document. The service incre-

mentally adds new downloaded and processed revision content to its database, avoiding redundant downloads.[38]

38 An example call: http://wikiwho.net/wikiwho/wikiwho_api_api.py?revid=670069989&name=William_Hamilton_Maxwell&format=json&params=author. Explanation of parameters: "revid" takes as arguments a single revision id of the requested article or a range of revision ids defined by start and end id, separated by "|", "name" is the name of the English Wikipedia article, "format" is for now only JSON, the optional "params" accepts so far only "author", which additionally to the revision ids of origin also provides the original author of each token.

5

EDITOR-EDITOR INTERACTION MINING

After we have developed the WikiWho approach to trace the creation of a token and all following changes to it back to a specific revision and author, we are now able to set editors in relation to each other by the exact changes they carried out on each other's words. It is however not yet clear how to best encode these actions into an *explicit* interaction relationship between two users, due to the large variety of interpretations of the different actions that can be performed on another user's content.

There are a several user actions that potentially inform editor relations or interactions and can be derived from the editing logs available. One example used at times in the research literature [103, 15, 75] is "co-editing", which refers to editing of an article by multiple editors in a predefined timeframe, while not necessarily touching each others content. Some research approaches interpret such co-editing of users as them "working together". Alas, such an interpretation is seldom grounded in evidence or even on theoretical footing and we argue that in the majority of cases, simply co-editing an article without touching each others content allows no sound inferences whatsoever about the interactions between two editors in a given article.[1]

We therefore resort strictly to those cases where editors interact directly with other editors' content. From this selection of edits, especially disagreements between editors are of interest, as they are arguably by far the most common direct article-based interactions of this sort [42, 63, 82, 83, 122, 140, 89]. "Disagreement" here broadly signifies that certain text pieces originally written or reintroduced by one user get altered or deleted by another user and vice versa (we will refine this notion below). Almost unanimously, disagreement in Wikipedia-related research is modeled as so-called reverts, which we initially defined in Section 2.3.1. To make a first step towards learning what type of actions carried out on another editor's content can mean what type or level of disagreement, in Section 5.1 we concentrate on such reverts. This helps us to distinguish the main disagreement edits, "full reverts", from all other deleting and undo actions and to model them accordingly. In particular, we show that the state-of-the-art approach for detecting and modeling reverts has a number of important drawbacks - it only detects a limited number of reverts, while simultaneously misclassifying too many edits as reverts, and not distinguishing between full and partial reverts. These insights al-

1 They might, however, shed light on several editors' relation regarding working in Wikipedia in general. E.g., working on many of the same articles together, always at the same time. Still, this is not in the scope of our research questions here.

low us to perform a first disambiguation between different kinds of disagreements.[2]

In Section 5.2 we then formalize a model for editor disagreements (and agreements), based on the insights gained through our reverts study and other related work on the topic. This enables us to build a meaningful social graph of editors that can be used in further analyses and visualizations or build upon for more advanced disambiguations of interactions.

5.1 PRE-STUDY: DETECTION OF DISAGREEMENTS THROUGH REVERTS

One common tool when analyzing Wikipedia editor behavior is referred to as "revert detection", which is typically defined as the task of finding edits that undo the actions of one or more previous edits, and is canonically understood to represent disagreement between reverting and reverted editors [42, 63, 82, 83, 122, 140, 89] (a full discussion about the notion of a revert is given in Section 5.1.1). In practice, and taking into account the very nature of the editing process in Wikipedia, revert detection forms a foundational step for many (more elaborated) research ideas, and its purposeful handling leads to a better understanding of Wiki-like systems and collaboration in them. It is also virtually the sole method used to model and detect disagreement between editors based on article edits.

Despite its importance as a pre-processing step, most research work capitalizing on reverts-related information relies on a very basic detection approach. In a nutshell, the method discovers identical revisions of an article based on the MD5 hash values [125] of the corresponding full revisions, and considers all edits lying between two consecutive revision duplicates as being reverted.[3] Although its coarseness is discussed in some work (see Section 5.1.2), the extend of its oversimplification of revert behavior has not yet been acknowledged in its full range of implications. We argue that this approach is neither sound nor complete; it detects only a limited share of the actual reverts – as they are defined and understood in Wikipedia editing practice, cf. Section 2.3.1 – and falsely classifies several co-occurring editing activities as reverts, thus leading to lower-quality analysis data which is very likely to hamper the accurate interpretation in Wikipedia research regarding disagreements (and therefore, most interactions) between editors.

Below, we introduce an algorithm improving the detection of reverts in terms of accuracy and coverage. This is achieved by comparing edits based on the actions they perform, which are measured by

2 Future research will have to explore this categorization of interactions further, as we will discuss as well in Section 5.3.1

3 Hash values are here unique short sequences encoding a text of arbitrary length.

means of the word tokens they add or delete.[4] To evaluate its performance against the more naïve approach currently in use, we conducted a user study and other tests, which provide clear evidence of significant gains in accuracy and coverage.

5.1.1 *Reverts as Basis for User Disagreement Modeling*

To recount the official Wikipedia definition we cited in Section 2.3.1: *"Reverting means undoing the effects of one or more edits, which normally results in the page being restored to a version that existed sometime previously. More broadly, reverting may also refer to any action that reverses the actions of other editors, in whole or in part"*. Text passages from other Wikipedia pages concur with this definition, such as *"A revert means undoing the actions of another editor"* on the Wikipedia edit warring page.[5] Furthermore, we define some basic terms used in the remainder as follows:

- A "reverting edit" is an edit that carries out a revert, i. e., reverts one or more other edits. The editor of this edit is a "reverting" editor.
- A "reverted edit" is an edit whose changes to an article are undone partly or completely by a reverting edit ex-post. The article version that is changed by the reverting edit is the "reverted revision". The editor of this edit is a "reverted" editor.

Within Wikipedia, reverting can be carried out in different ways:[6]

1. Manual reverting, which means deleting text from or adding text to an article by hand.
2. Activating the "undo" button next to an edit in the article history dialog. It enables to undo the actions performed by only that specific edit (which is not necessarily the latest edit).
3. Using the "rollback" feature, which immediately reverts all top consequent edits made by the last editor, going back to a previous version of the corresponding article. It is available to administrators and editors who have been explicitly granted the right to use this function.

Let us illustrate how inferences about editor interrelations could be drawn based on rather shallow knowledge of the content of an edit through a simple example: If edit 1 establishes an article consisting only of the word "apple", edit 2 adds "pie" after "apple" and edit 3

4 The algorithm is available under an open license at http://people.aifb.kit.edu/ffl/reverts/

5 http://en.wikipedia.org/wiki/Wikipedia:Edit_warring (accessed 13.09.11, italics added).

6 http://en.wikipedia.org/wiki/Help:Reverting, (accessed 13.09.11)

deletes only the word “pie”, we can conclude intuitively that edit 3 deleted the content introduced by edit 2. This action could have been carried out by hand, by the "undo" feature for edit 2 or by doing a "rollback" to edit 1, the result is the same in this case. If, moreover, these were edits by distinct editors, we can further assume that the editor of edit 3 wanted to undo the action of the editor of edit 2; and we can do so without considering the meaning conveyed by the text strings that are added and deleted in the process.

As briefly explained through the example above, reverts are relatively easy to extract and interpret compared to other types of editing activities that comprise editor interaction, without automatically necessitating knowledge of the *meaning* of the changed content (as is the case for, e.g., editors writing about a topic with the same point of view, but never touching each others content), while providing an essential insight into the behavior of editors. A number of works on Wikipedia have been using reverts as a metric in their studies, be it on general trends involving reverts [140], correlation of the chances of getting reverted with specific editor or edit characteristics [63], using reverts as an indicator of damage repair and vandalism fighting [84, 122], or considering them in some other ways when analyzing editing behavior [82]. In the case of social network modeling, it is especially important to not only detect who was reverted or who was reverting, but also *who was reverted by whom*, i. e., to model the antagonistic dynamics in an article on a detailed and accurate level to unveil the fine-grained disagreements and interrelations between users.

Data quality is always a relevant issue when it comes to interpreting behavioral data, but it is essential in making sense of the social dynamics a the level of individual articles, which can sometimes mean interpreting conflicts among only a handful of users who influence the direction of the entire article. Achieving this aim poses therefore higher demands on the quality of the method as is currently delivered by available techniques, since a large number of false-positives and false-negatives in the results might lead to grave misinterpretations of editor relations. We give examples for this in Section 5.1.2. It is for these reasons necessary to be able to rely on the accuracy and completeness of the identification of edits as "reverts", i.e., clear disagreements. This, in turn, requires a precise and purposeful notion of what a "revert" is, and a revision of existing methods operationalizing revert detection to accommodate this theoretical understanding.

5.1.2 *State-of-the-art in Revert Detection*

Wikipedia-related research using reverts as a metric [42, 63, 82, 83, 122, 140, 89] almost invariably deems only so called "identity reverts"

Table 4: Example of the result of the simple identity revert detection method

Edit	Revision content	Words deleted or added in the edit	MD5 hash	Detected reverts
1	Zero	+'Zero'	Hash1	
2	Zero Apple Banana	+'Apple' +'Banana'	Hash2	Reverted by edit 5
3	Zero Apple Banana Coconut Date	+'Coconut' +'Date'	Hash3	Reverted by edit 5
4	Zero Coconut Date	-'Apple' - 'Banana'	Hash4	Reverted by edit 5
5	Zero	-'Coconut' -'Date'	Hash1	Reverting e. 2, 3, 4

as an appropriate means to investigate revert behavior.[7] This approach relies on finding two revisions containing exactly the same content via MD5 hashes [125].[8] Subsequently, all edits lying between two identical revisions are considered as *reverted*, with the second identical revision as the *reverting* edit, and the first one as the one *reverted to*. As defined by Halfaker et al. in [63]: "A revert is a special kind of edit that restores the content of an article to a previous revision by removing the effects of intervening edits". Table 4 shows an example of how reverts are detected in this manner.

Beside such work using identical revisions to detect reverts, there is research discussing types of reverts – and actions to be considered reverts – more in-depth, as well as the implications and complexity of reverts in Wikipedia (e.g. [121]). But these publications do not introduce a model or algorithm evaluated to detect reverts in a more accurate way. The works by Adler, Chatterjee and de Alfaro [2, 3] implement elaborated approaches for keeping track of addition and removal of words in an article by different editors. But it is neither aimed at, nor evaluated in respect to precisely detecting revert relations between editors.

7 There is some other work that identifies reverts solely based on regular expressions (e. g., using keywords such as "rv", "revert") in edit comments, calling it a "reasonable proxy" for revert detection [84] (also [81]). But this is to be neglected here as the work cited as source for this statement clearly states that "MD5 (identity) reverts actually capture more revisions than user-labeled (comment) reverts (3.7M vs. 2.4M), suggesting that a substantial number of reverts are not labeled as such" [83].

8 The MD5 hash sum is commonly used to check if two file or text contents are identical.

Where used in the research literature mentioned above, the simple identity revert detection method (henceforth: SIRD) is never explicitly ex ante motivated by a theoretical concept of revert behavior or by any definition established by the Wikipedia editor community. Rather, the motivation for using SIRD as stated (representatively) by Halfaker et al. is that it "is computationally simple and determining exactly which editors' revisions were lost due to the revert is straightforward" [63]. The underlying (implicit) notion of what a revert is can be seen as an over-simplification of how Wikipedia defines this concept: "[...] which **normally** results in the page being restored to a version that existed sometimes previously." (bold added, cf. Section 2.3.1). It does not require the reverting edit to actually *undo the actions* of an edit identified as reverted, although that characteristic is unequivocally stated in Wikipedia's definition (compare the indicated revert of edit 4 by edit 5 in Table 4, which does not undo 4's actions). As such, the revert detection method is also not able to make distinctions concerning the relationship between reverting and reverted edit: It is not possible to indicate if the reverting edit fully, partly or not at all undid the actions of the reverted edit (again, compare the example in Table 4). It also does not require the intention of the reverting edit to revert any other edit.

SIRD supposedly detects most of the existing reverts: Kittur et al. suggest that by combining a method based on edit comments, (looking for the keywords "revert" and "rv") and SIRD, 95% of the *reverting edits* identified as a result could be found using only identity reverts [83].[9] In a number of subsequent publications [63, 122, 140], this finding was used to conclude that the mere 5% additional reverting edits found by looking at comments do not justify the effort of using this additional source of information on top of the SIRD method. But there was no dedicated investigation so far if other detection methods might find even more reverts,[10] as many users do not attach comments to their edits [83, 122, 140] and MD5 hashes cannot be used to find partial reverts that do not produce identical revisions [63].

In terms of a "real-world-check" of the conceptualization underlying the SIRD method, there has not been any testing of the false-positive rate of the delivered results, in the sense of evaluating it against the Wikipedia definition of a revert or what is perceived as

9 In the paper, the authors refer to such edit pairs as the found "reverts" while actually they report the number of found reverting edits that either have an identical previous version or a comment mentioning the two keywords [83]. Where identified by comments, it cannot be in all cases concluded *what* revisions were actually reverted when there is no identical version (i. e., in the case of a partial revert) and no indicator in the comment (e. g., in the case of a comment consisting only of one of the keywords).

10 Ekstrand et al. [42] compared cosine similarity and adoption coefficient approaches with the SIRD for finding revision "history trees". They come to the conclusion that the SIRD algorithm is a better solution for representing revision relationships than the other two approaches.

a revert by Wikipedia users. This means there was no evaluation, for instance, if the actions of identified *reverted* edits are really undone in subsequent *reverting* edits. This is a crucial issue especially in the light of the very simplistic, technology-driven definition of a revert the SIRD method implicitly builds upon. Although we know of at least one analysis toolkit (pyMWDat)[11] that extends to some degree the above described basic definition of reverting and reverted edits, we are so far not aware of an elaborate and working algorithm modeling revert (or disagreement) behavior, which is designed to capture reverts based on a more realistic definition of how reverting and reverted edits are related.

The inability of the SIRD method to detect reverts that do not create a duplicate revision is acknowledged by Priedhorsky et al., who state that beside the identity revert, there exists an "*effective revert*, where the effects of prior edits are removed (perhaps only partially)" [122]. Such cases cannot be fully detected using only MD5 hashes [63]. In Table 4, this is exemplified by the actions of revision 4, which deletes all words introduced by revision 2, while still generating a completely new revision content. Intuitively, one could say that revision 4 is *effectively* reverting revision 2, as it undoes all its actions; this interpretation conforms with the Wikipedia revert definition. The SIRD method, however, will not detect the revert relationship in this way, but instead, as shown in the example, detects revision 5 as reverting revision 2, solely because it (incidentally) lies between two identical revisions. In a scenario where revision 5 would be non-identical to revision 1, the method would not even detect any revert of revision 2. This illustrates the logical inconsistencies of the conceptual model on which SIRD implicitly operates. Note that this is only one of a number of many example scenarios in an edit history we found, where the SIRD method leads to a questionable result. Additional examples are given in Section 5.1.3.

The coarseness of SIRD is further discussed by Priedhorsky et al. [122], who note that understanding and taking into account the intention of a revert is challenging (and thus not feasible), while the method already covers one of the most common types of reverts (producing identical revisions) at an, arguably, sufficient level of quality. This latter has, however, never been proven.

As a conclusion, when setting the Wikipedia revert definition as a benchmark for the understanding of revert and disagreement behavior in current Wikipedia editing practice, the coverage (finding all actual reverts) and accuracy (finding only true-positives) of the SIRD method are suboptimal. Edits are always and only detected as

11 Available at http://code.google.com/p/pymwdat/ (accessed on 06.11.11) – As noted in the documentation of the tool, pyMWDat works similar to the SIRD method, but differentiates between the revisions marked as "reverted"; in other words, the first revision after the "reverted-to" revision is marked as "possible vandalism", while the remaining reverted edits are classified as a separate group of "good-will edits".

reverted if they lie between two identical revisions for reasons which are not further taken into account. This has a number of important consequences:

- edit pairs are detected as "reverting" and "reverted" that cannot be seen as reverts when compared to known edit behavior in Wikipedia and the general understanding of editing practice of the contributors (see Wikipedia's definitions);
- there might be many reverts still to be found by untested methods; and
- for those reverts that are found by the SIRD approach, it cannot be distinguished to which extent a revert took place: a full revert (all actions undone), or only 20%, 70%, etc. of actions of a previous edit undone.

We will discuss the distinction between a "full" and a "partial" revert in the following section. In any way, it seems obvious that a new, more fine-grained method for detecting reverts and disagreement might be beneficial.

5.1.3 *An Improved Revert Detection Method*

We have exemplified why the currently used approach of encoding disagreement through reverts might be suboptimal. We therefore propose a revert detection method that can identify all undo-actions at word level and does not rely on identical revisions to be produced. This, in turn enables to additionally identify cases where an edit was not fully, but only partially undone. Yet, partial reverts are more complicated to use in modeling social interactions than full reverts: for the latter, it is known that *all* actions of a previous edit have been undone, constituting clear *disagreement*; while for the former, the range of possible interpretations of the revert action is comparatively much wider. To give an example, the removal of 20 words from a recent 600-word-entry could mean only a small correction, while deleting the single word "not" in a certain position could change the meaning of the whole entry. The deletion of all 600 words on the other hand can be safely interpreted as that text being of contention. When using the results of a revert analysis in scientific work, it should therefore at least be possible to distinguish partial from full reverts and leave it up to the investigator to select down to which degree of undoing the detected partial reverts should be treated as disagreement. This applies in particular to those scenarios where a thorough analysis based on a comparatively smaller data corpus is of interest, such as the editing behavior in one specific Wikipedia article. These deliberations motivated the development and testing of the method presented in the next section.

5.1.3.1 *Revert Definition*

The first step towards devising a more accurate revert detection method is to establish a clear conceptual foundation of what a revert is, followed by an algorithm that detects all and only those edits that fit the corresponding definition.

The Wikipedia revert definition is used as a reference point, as it states what actions constitute a revert as a behavior of an editor, and as it is grounded in the common editing practice of the Wikipedia community. For assessing the results of our method versus SIRD, we give priority to:

1. detecting edits that are no false-positives, i. e., only reverts fitting the used definition; and

2. distinguish full reverts from all other, partial undo actions, as only for those full reverts we can safely assume former edit actions were completely undone and thus unambiguously indicate a "reverted-reverting" disagreement relationship. We thereby focus stronger on finding and evaluating full reverts, as partial reverts are too ambiguous in their meaning to provide useful explicit disagreement relations without a further, much more elaborate classification approach.[12]

According to these pre-requisites and taking into account what kind of data can be reliably used to identify reverts, the following definitions are set up.

Definition 6 *(Full Revert). An edit* A *is fully reverted if all of the actions of that edit are completely undone in subsequent edits. If all of these undo actions are carried out in one single edit* B*, edit* B *has then fully reverted edit* A*.*

Definition 7 *(Partial Revert). An edit* A *is partly reverted if at least one but not all of the actions of that edit are undone in subsequent edits. An edit* B *carrying out an undo action targeting another edit* C *is considered a partial revert of* C *if it doesn't undo* all *actions that* C *has carried out.*

An "action" is for our purposes defined as the deletion or addition of a single token of text (mostly words delimited by whitespaces, cf. Section 4.3.3).[13] Note that these definition does not rule out that edit

12 E.g., a user might add a twenty-word sentence and just one word might get removed (reverted) by another editor subsequently. While the removal of a complete edit is relatively easy to classify as disagreement, by algorithmic heuristics as well as human raters, the former case requires more sophisticated approaches to assess its meaning.

13 Note that in the used implementation, also spelling corrections of a word are considered as a deletion and addition of a new word. This is however a question of how one defines "new" and "old" tokens. One might consider using a certain Levenshtein distance or similar measures to treat such cases as corrections. Synonym detection approaches could also be viable.

Table 5: Example of the result of the improved revert detection method

Revision #	Revision content (text tokens)	Words deleted or added (actions taken)	Detected reverts DIFF FR = Full revert PR = Partial revert	Detected reverts SIRD (cf. Table 4)
1	Zero	+'Zero'		
2	Zero Apple Banana	+'Apple' +'Banana'	FR by 4	FR by 5
3	Zero Apple Banana Coconut Date	+'Coconut' +'Date'	FR by 5	FR by 5
4	Zero Coconut Date	-'Apple' -'Banana'	FR of 2	FR by 5
5 (≡1)	Zero	-'Coconut' -'Date'	FR of 3	FR of 2, 3, 4
6	Zero Fig	+'Fig'	FR by 8	
7	Zero Fig Grape	+'Grape'	FR by 8	
8	Zero Huckleberry	-'Fig' -'Grape' +'Huckleberry'	FR of 6, 7	FR by 11
9	Zero Huckleberry Grape	+'Grape'	PR of 8	FR by 11
10	Zero Huckleberry Fig Grape	+'Fig'	PR of 8	FR by 11
11 (≡7)	Zero Fig Grape	-'Huckleberry'	PR of 8	FR of 8, 9, 10

B performs other actions on top of undoing A's actions or the actions by a number of distinct edits (for an example see Table 5, where edit 8 is reverting edits 6 and 7, while on top adding new content).

In Table 5, comparing the results of the SIRD method to our new approach reveals important differences. For revisions $1-5$, we see that adhering to our definition, revision 5 is only reverting revision 3, while revision 4 is reverting revision 2. This indicates higher accuracy (complying with Wikipedia's definition) in comparison to SIRD. We additionally detect the revert by revision 8 of revisions 6 and 7, where no duplicate revisions can be found. This means our method is potentially able to find more reverts than SIRD.

If all of an edit A's actions have been undone in a collective effort by many partial reverts, A is counted as fully reverted, but no single "full reverting" edit can be identified. An example for this case is given with edits 9 to 11 in Table 5, which are reverting edit 8 only in aggregate. For our evaluation later on we took only such edits into account that were a full revert by *one* specific edit (i.e., not edits $9-11$ in our example). This is due to the fact that, in the multiple-reverters-case, it would not be possible to assign a single reverting edit B, and thus not unambiguously determine the reverting and the reverted edit in every case. And as each of the reverting edits might have undone small parts of the fully reverted edit, but did each individually not intend to revert it completely (hence maybe did not "disagree" with it, but just slightly "corrected" it), we cannot clearly classify these cases, just as for all partial reverts.[14]

Apart from those given in Table 5, there are other examples of reverts where our method can extract more meaningful revert information compared to the baseline approach. One frequently occurring scenario is the repair of a well-intended, but erroneous revision by several subsequent revisions. If the last one in this row of repair edits restores the last error-free revision, all other repair edits will be marked as reverted with the SIRD method, although each of the edits did only implement a partial revert. With our method and the definition introduced earlier in this section, we do not incorrectly assign reverts in this setting. An example: let the actions by edit 8 in Table 5 be considered a contribution containing factual mistakes. We then assume that edits 9, 10 and 11 are trying to repair damage caused by edit 8. While edits 9 and 10 overlooked some inaccuracies, edit 11 eventually restores the last error-free revision 7. In this case, the SIRD algorithm would have assigned edit 11 as the reverting edit of edits 8, 9 and 10, although, according to Wikipedia's definition, at least the actions of edits 9 and 10 were not undone.

5.1.3.2 *Implementing the Revert Detection*

Like for WikiWho, to operationalize the actions of the editors we use added and deleted word tokens, i. e., character chains separated by white spaces, and we operate on the Wikitext, not on the front-end article content. Before taking a look at the specific word changes of an edit, we eliminate unchanged paragraphs in the manner explained in Chapter 4, to reduce the amount of text for word-level text difference comparisons (DIFF).

To compare the remaining (edited) paragraphs, DIFFs are calculated. For every revision r_n in the article history, we check via DIFFs if its previous i edits r_{n-1} to r_{n-i} (with $i > 1$) performed the exact opposite of a subset of actions of r_n, starting with r_{n-1} and going

14 Partial reverts that do not entail a fully reverted target revision are not shown in the example, but of course occur frequently as well.

sequentially until r_{n-i}. For a deleted text token the opposite action is a re-addition (also called reintroduction), for an added text token it is a deletion of the same token. A formal definition of these actions is provided in Section 5.2.

The maximum scan-range i is in our example set to 20 previous revisions (we discuss the size of i in Section 5.1.4). If an opposite matching subset is found in r_{n-1} we do not look for the same subset in the following r_{n-2} to r_{n-20}, as the action of r_{n-1} can only be undone once. If r_n, on top of reverting other edits, added or deleted additional content that does not have a matching opposite subset in the previous 20 edits, no revert action is registered.

When performing the computation discussed above, we filter out so-called "blankers" from the list of possible reverting edits. Blankers are edits deleting the whole content of an article, which should not be treated as common reverting edits, as their behavior cannot be interpreted as an intentional act aimed at undoing the specific edits whose added content they delete. Rather, the vandalistic intention is aimed at harming the article as a whole.

We implemented the SIRD approach and our new algorithm in Python. The input for both algorithms is an XML file of all revisions for a given article, consisting of the revision ID and the text of the revision plus some metadata. Both scripts also implement the detection of blankers, as noted earlier. For generating the text DIFFs, the difflib library of Python is used on token level.The resulting algorithm produces revert indicators according to our definition and exactly as laid out in Table 5.

5.1.4 *Evaluation*

We evaluated both the accuracy and the coverage of the reverts found by the DIFF method against the baseline SIRD approach. Runtime is not reported as only the quality of the extracted disagreement interactions was of concern here and all calculations can be carried out via the WikiWho algorithm.

5.1.4.1 *Accuracy Evaluation*

For comparing the accuracy of the revert detection we set up a revert assessment survey with Wikipedia editors.[15] Only Wikipedia editors were chosen as we are interested in results that conform with what the Wikipedia community perceives as a revert (and outsiders generally have no concept of a "revert"). The survey was conducted for 11 days in October 2011. Participants were recruited through several

15 All participants were editors who performed reverts on a regular basis themselves. We asked for the length of their tenure as editor and their experience with reverts, which both had no significant impact on the answers. We also have no reason to believe the self-selection of participants introduced a bias in the answers.

internal Wikipedia outlets such as the Community Portal and the Village Pump.[16] We set up two samples of 20 assessment steps to be evaluated by the participants.[17]

The first sample (referred to as sample A) of 20 assessment steps consisted of 9 edit pairs detected as a full revert only by the DIFF method, 9 detected only by the SIRD method and, as a control group, 2 pairs detected by both methods in the same way – each randomly selected. 29 users completed this first sample and all assessment steps.[18] The second sample (sample B) consisted of 8 edit pairs detected as a revert solely by the DIFF method, 8 detected only by the SIRD method and 4 pairs detected by both methods. 16 participants, distinct from the assessors of the first sample, completed all steps in this sample.[19]

Following the rationale laid out already in Section 5.1.3.1 we used only such edit pairs that were identified as full reverts by our algorithm. This was done also because SIRD implicitly sees all revisions between two duplicate revisions as fully reverted, and hence the cases it identifies can only be compared to what we define as a full revert.

The samples were designed to include more SIRD- and DIFF-only edit pairs than results identified by both approaches, because the aim of the evaluation was to compare them against each other in those cases that differed. The samples of 9, 9, 2 (A) and 8, 8, 4 (B) edit pairs, respectively, were randomly drawn from the pool of all unique revert-pairs detected by the two methods in five randomly selected Wikipedia articles. To generate the edit pairs, the number of i previous revisions to be scanned for reverted edits (as explained in Section 5.1.3.2) was set to 20.[20]

An assessment step consisted of two text DIFFs, exactly as known from the Wikipedia revision history feature.[21] Figure 8 depicts a sample assessment step. The first DIFF, shown on the top of the page, represented edit 1 (the DIFF showed what was changed by edit 1). The second DIFF depicted edit 2, which was a subsequent edit from

16 Community Portal: http://en.wikipedia.org/wiki/Wikipedia:Community_Portal Village Pump: http://en.wikipedia.org/wiki/Wikipedia:Village_pump (both accessed 25.10.11).

17 We preferred to restrict the number of assessment steps to 20, as first tests showed that participants did otherwise abort the survey prematurely due to its perceived over-length.

18 Nine users were excluded in total from the two samples, as they aborted the survey after only one or two questions. This was done to prevent a potential bias in more and possibly different answers for earlier questions.

19 Unfortunately, no more than 16 participants volunteered to complete the assessment steps for sample *B*, which was set up after 29 users completed sample *A*. The number of edit pairs detected by both methods was raised slightly in sample *B* to get assessments for a bigger sub-sample of these edits.

20 The rationale behind this design decision was that an intentional revert targeting specific content is likely to happen within a limited window of edits after the original edit took place, as changes stay in focus of the community (and the change logs) for a limited amount of edits. This was confirmed through additional evidence collected by manual assessment we conducted. In this assessment, we observed the best accuracy at $i = 20$.

the same article identified as the reverting edit of edit 1 by one of the revert detection methods. The use of colors and the "+/-" signs were adapted without changes from Wikipedia, as explained on the respective "Help:Diff" page.[21]

Participants were asked if, according to the Wikipedia definition of a revert, edit 2 had reverted edit 1. The answer options included "Full revert", "Partial revert", "No revert" and "I don't have a clue". The participants were provided with the Wikipedia revert definition in each assessment step (omitted in Figure 8) and were particularly asked to apply this definition in their assessment, and not their own definition, if different. At the end of the survey we asked the respondents if the Wikipedia definition indeed conformed with their own definition of a revert, on a Likert-scale from 1 (no agreement) to 5 (full agreement). 17 editors answered this question, with 15 voicing full agreement and two agreeing only partly (scores 2 and 3).

For each of the two samples, containing 20 edit pairs each, the overall agreement of the participants that the corresponding pair was either a full revert, a partial revert or no revert, was computed. The assessed edit pairs received 29 (16) votes, distributed over the three revert types (or "I don't have a clue").[22] Consequently, the revert types could achieve a score from 0% to 100% (29/16) of the participants agreeing in each assessment step. Figure 9 and Table 6 show the average agreement of participants over all edit pairs for each of the two methods and each of the three types of revert.[23] When asked if the displayed edit-pair was a full revert, 77.2% (78.8%) of the participants expressed their agreement for the pairs found only by the DIFF method, while only 25.5% (23.8%) did so for the edit pairs detected only by the SIRD method. This difference between the agreement score means was significant at $p < 0.01$.[24] When asked if an edit pair was a partial revert, the agreement was at a mean of 23.1% (24.4%) for the SIRD method and at 4.1% (6.9%) for the DIFF method. With $p < 0.1$, this difference was, however, not significant for sample A, but significant at $p < 0.01$ for sample B. When asked if an edit-pair was no revert at all, i.e., a false positive, a mean of 49.3% (50.0%) participants agreed for the edit pairs detected only by the SIRD, while only a mean of 17.2% (14.4%) said so for the pairs detected solely by our DIFF method. This difference was significant, at $p < 0.05$ ($p < 0.05$). The means of agreement for the control group of the edit pairs found by both methods were generally aligned with

21 https://en.wikipedia.org/w/index.php?title=Help:Diff&oldid=452412954

22 Numbers of sample B are put in brackets, henceforth.

23 The raw data of the survey answers can be found at http://people.aifb.kit.edu/ffl/reverts/. The box plot for sample B was omitted as it showed very similar agreement scores to sample A, thus yielding no additional information other than confirming our previous findings.

24 The mean agreement for the pairs found by both methods was 86.2% (92.5%). It was significantly different from the SIRD-method-only sample at $p < 0.01$, but not significantly different from the new-method-only sample.

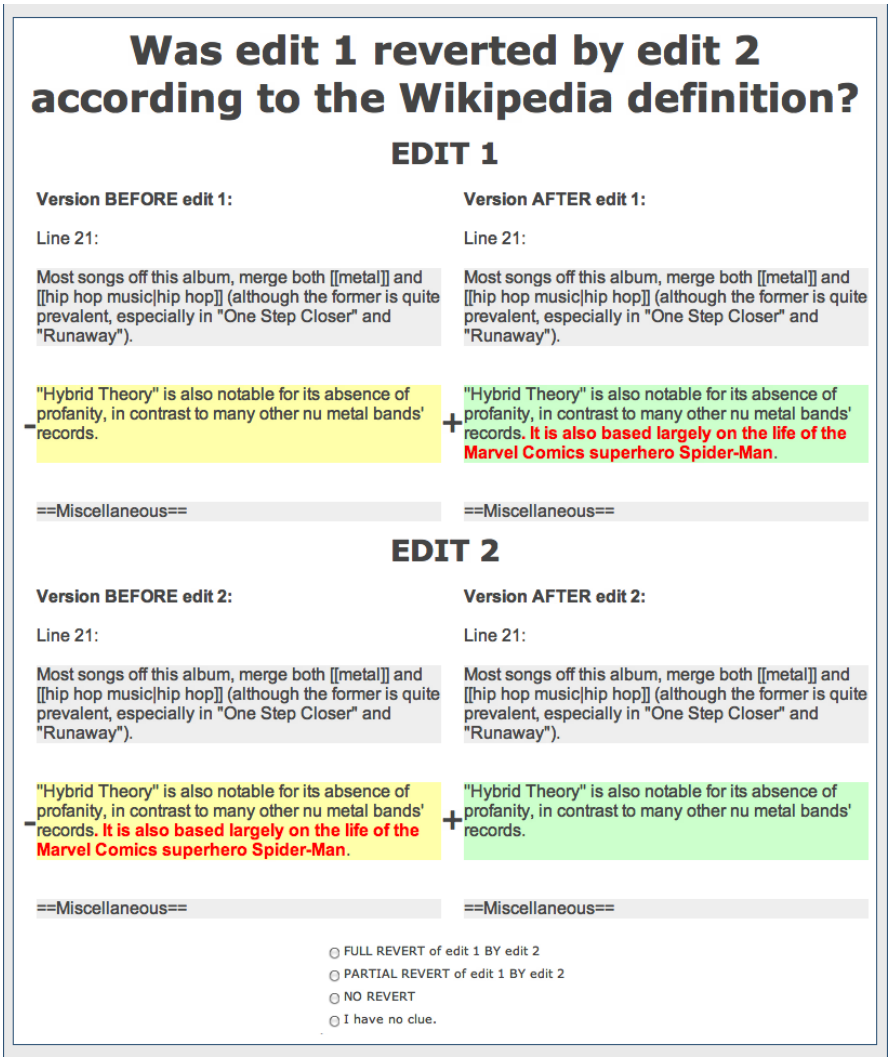


Figure 8: **Screenshot of an assessment step in the survey**. Two text difference views for two different edits are shown with the old (left) and the new (right) version of the affected portion of the article. On the very bottom are the four response options.

those of the new-method-only edit pairs in both samples, as they revealed no significant differences (therefore excluded from the summary of results in Table 6). They were not aligned with the means of the SIRD-only sample, as can be observed in Table 6. For sample A, only the difference in the agreement scores for full reverts differed significantly ($p < 0.01$), but for sample B, all differences were significant. The means of agreement for the answer "no clue" are not listed here as they were very low ($< 3.7\%$) for both methods and samples and all revert types, and there were no significant differences.

5.1.4.2 *Coverage of Revert Detection*

To evaluate the performance of our method against SIRD with respect to the number of reverts detected, we analyzed a sample of 5,000 ran-

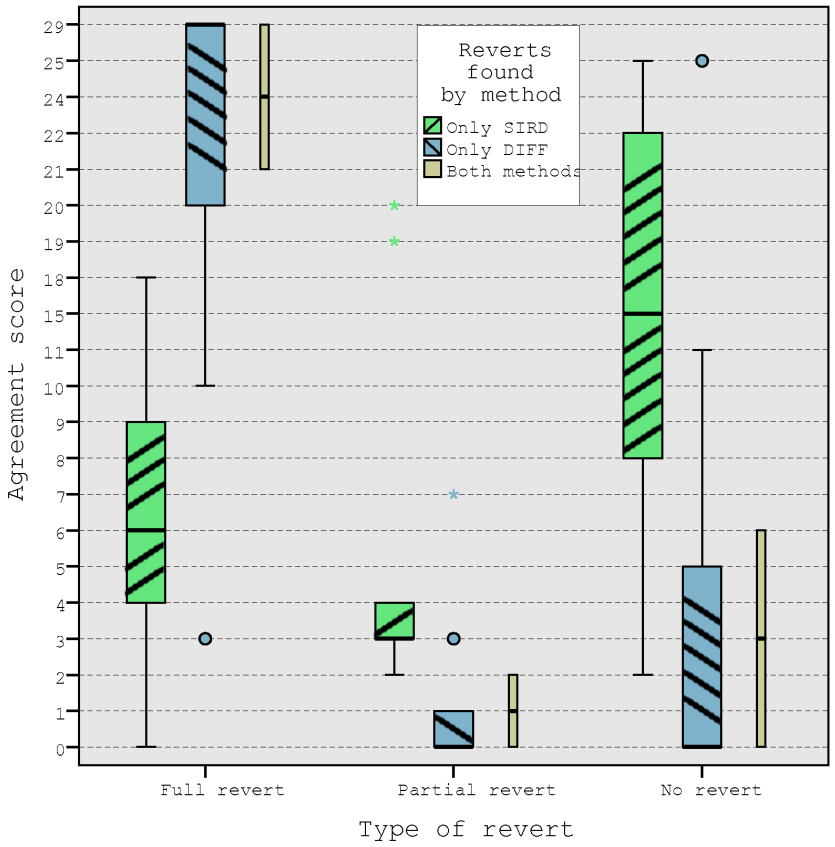


Figure 9: **Boxplot comparison of the means of the absolute votes of agreement** between all three methods, grouped by indicated type of revert for sample A (25th and 75th percentiles as box, 1.5x interquartile range (IQR) as whiskers, outliers > 1.5*IQR, extremes > 3*IQR).

domly selected articles, which were no redirect pages. They contained a total of 392,724 edits. The results were generated via two scripts implementing the two methods under evaluation. First, we report on the results produced with a maximum scan-range of $i = 20$ previous revisions for identical hashes and, respectively, corresponding DIFFs, as used for the accuracy evaluation in Section 5.1.4.1.

In total, 75,278 unique edit pairs were detected as reverts in the articles by the two methods. Of those reverts, 39,816 were found by both methods in the exact same manner, 14,495 were found only by the SIRD method, and 20,976 were found only by the new method. Table 7 gives an overview of the results. Thus, for 27% of the reverts found by the SIRD in total, the new method found different reverts.[25] On top, the new method found about 12% more reverts than the SIRD method.[26]

When carried out with a maximum scan-range of $i = 50$ and $i = 100$, the number of reverts found by both methods as well as the difference

25 $14,495/(14,495 + 39,816) = 0.27$
26 $(20,967 + 39,816)/(14,495 + 39,816) = 1.12$

Table 6: Means of agreement scores for different methods and revert types, for both survey samples

Sample A	n	Full revert	Partial revert	No revert
DIFF only	9	77.2%	4.1%	17.2%
Difference		$p < 0.01$		$p < 0.05$
SIRD only	9	25.5%	23.1%	49.3%
Difference		$p < 0.01$		
Both methods	2	86.2%	3.4%	10.3%
Sample B	n	Full revert	Partial revert	No revert
DIFF only	8	78.8%	6.9%	14.4%
Difference		$p < 0.01$	$p < 0.05$	$p < 0.05$
SIRD only	8	23.8%	24.4%	50.0%
Difference		$p < 0.01$	$p < 0.05$	$p < 0.01$
Both methods	4	92.5%	5%	1.9%

Differences tested via Student's T-test. P-values only shown where significant with $p < 0.05$ and always referring to the differences between scores below and above. n stands for the number of assessment steps (=edit pairs).

between the two methods grew considerably. The new method found 19% and 24% more reverts, respectively, as can be seen in Table 7.

5.1.4.3 *Discussion of the Evaluation Results*

Regarding accuracy, participants of the revert assessment agreed that in the mean, for the new method, the found edit pairs are significantly (i) less often false positives, i. e., no actual reverts and (ii) more often full reverts than those pairs found only by SIRD. When we consider the basic definition of a revert SIRD operates with, and the scenarios in which this oversimplification can lead to suboptimal revert classification, as demonstrated in Section 5.1.3, the results make a strong case for the more accurate revert detection offered by our method. It also seems that our suspicions regarding the type of misclassifications made by SIRD are confirmed: Mostly, when misclassifying an edit pair, the pair is actually no revert at all. But, to a lesser extent (and more so than our new method) SIRD actually identifies partial reverts as full reverts.

The explanatory power of the results is to some extent impaired by the relatively small sample of assessment steps and survey participants. Nevertheless, the high significance of the key findings, and the fact that two distinct groups of Wikipedians assessed two distinct

Table 7: Number of detected reverts (≡ edit pairs) in article sample by methods, for different levels of *i*

i		Pairs detected by		Gain by
		SIRD	DIFF	DIFF
20	n	54,311	60,783	6,472
	%	-	-	12
50	n	55,647	66,115	10,468
	%	-	-	19
100	n	56,101	69,549	13,448
	%	-	-	24

(a) Reverts detected by each of the methods (in absolute numbers, sets intersect) and gain in absolute amount by DIFF (also in percent)

i		Sum of *unique* detected pairs	Detected by both	Detected only by SIRD	Detected only by DIFF
20	n	75,278	39,816	14,495	20,967
	%	100	52.9	19.3	27.8
50	n	81,714	40,048	15,599	26,067
	%	100	49.0	19.1	31.9
100	n	85,604	40,076	16,025	29,503
	%	100	46.8	18.7	34.5

(b) Results of the two methods with duplicates removed (intersection represented in the "both" column), showing each method's uniquely detected revert pairs. One can see that the DIFF method finds more unique pairs, an effect that increases with larger *i*.

sets of edit pairs in an almost identical fashion speaks for the generalizability of the observations. The survey results for the new DIFF method appear to be in accordance with the part of the detected reverts which both methods are able to find. It becomes clear from the answers of the editors that the share of reverts that is solely identified by the SIRD method is in the mean significantly more often wrong and finds considerably fewer full reverts compared to the total number of reverts detected.

Looking at this number, the new method is able to detect from 12% (at $i = 20$) up to 24% (at $i = 100$) more reverts than SIRD. When operating on larger editing windows the accuracy might decrease because the new method will more likely match negative subsets of word tokens that were not meant to be reverted by the potential reverting edit. For $i = 20$, we can thus postulate the following based on the analysis of the evaluation: Given the result of a revert analysis with the SIRD method on a set of edits, our method is able to detect different

revert-pairs that are significantly less likely to be false-positives and more likely to be full reverts for 27% of the revert-pairs detected by the baseline approach. In addition, the new method finds 12% more revert-pairs than the SIRD method. Note again that we excluded partial reverts from the detection for this evaluation. The resulting number of detected reverts would increase considerably when including these.

Another aspect that speaks in favor of the new method is the following: SIRD gives preference to detecting reverts that produce an identical revision. This is more likely to happen when the rollback function (see Section 5.1.1) is used, as rollbacks invariably return an article to a pre-existing revision. It is less likely to happen for undo-based and manual reverting. As the rollback function is available only to administrators and editors with special rights, the majority of editors has to make use of the remaining two procedures in order to revert. It is therefore plausible that reverts detected via identical revisions were conducted by a disproportionately high number of users with special rights and administrators. In turn, the reverts left undiscovered by SIRD are more prone to being carried out by "common" editors. In this manner, SIRD introduces a bias towards a special user group. So, even given a theoretical high accuracy in detecting reverts that can be identified with identical hashes, this bias would exists when relying solely on this method.

Looking at these results, it must be concluded that Wikipedia research work that bases its inferences on data derived via the SIRD method runs the danger of being misguided. Not fully acknowledging these impairments or dismissing them as ignorable noise in the data is tenuous, at least if the aim is to model the complex dynamics in specific articles on a detailed and accurate level.

5.1.5 *Summary of the Pre-Study*

We provided new substantial evidence that simple revert detection via hash values is not sufficient to accurately capture all relevant revert – and therefore disagreement – interactions between Wikipedia editors and that these shortcomings seem to be more grave than generally suspected in the research work that applies this method.[27] We presented a new method for the detection of reverts in Wikipedia which compares edits based on the actions they undertake at the level of word tokens added or deleted. Our method relies on a revert notion which is congruent with the official Wikipedia guidelines, and

27 It of course depends heavily on the specific research work if and what actual effect these shortcomings have on the eventual outcome of that research, as identity reverts indeed make up the largest parts of all reverts, as the large overlap of found reverts by both methods suggests.

with the general understanding of the Wikipedia community with respect to reverting behavior.

As revealed by a user study, our method, without implementing a very complex algorithm, is more accurate in identifying full reverts as understood by Wikipedia editors. More importantly, our method detects significantly fewer false positives than the SIRD method; this is due to the simplified revert model the SIRD method operates on, which does not perform optimally in practice when extracting revert data for realistically modeling editor-editor behavior in Wikipedia. A limiting factor for these encouraging results is the fact that the assessed samples of edit pairs and editors were by no means large. However, the answers of two distinct groups of Wikipedians on two distinct samples of edit pairs showed almost identical assessments. Given these observations, combined with the key findings being highly significant and an algorithm built on a solid theoretical foundation, rooted in the Wikipedia community's revert definition, we are confident that our results can be further generalized. Concerning the number of identified reverts, we found that an average 27% of the revert pairs detected by SIRD are not accurate in the above regard and that the DIFF method we developed can not only detect the same amount of reverts with better accuracy, but on top finds 12% more revert pairs than the baseline approach.

When calculating interactions between editors at the article level and dealing with other tasks that require a in-depth look at reverts, using SIRD introduces the risk of misinterpreting and wrongly modeling revert actions. In particular, the editorial social system of an article, to be studied via social network analysis or visualizations, requires an accurate and complete capturing and depiction of what is happening among key editors and editor camps. We are confident that research relying on SIRD for that purpose due to the lack of alternatives [83] or proposing similar modeling [50] will profit from the accuracy and coverage gain of our method.

5.2 FORMALIZING THE EDITOR INTERACTION NETWORK

Provided with the insights about disagreement interactions of editors and a computational method to accurately attribute single changes of words to the corresponding edits, we can now formalize the network of interactions between revisions and – in extension – editors. To this end, we will first formalize a *revision-revision network*, as revisions (respectively: the edits that led to them) are the basic units for the changes carried out. In the second step, as one revision is ascribed to exactly one editor, the revisions are aggregated per editor to generate the *editor-editor network*. This two-step separation of the description helps to better follow the procedure to arrive at the eventual social interaction graph.

5.2.1 *Related Work*

Several approaches exist to construct networks between editors from Wikipedia article history data with edges (neutral, negative or positive) signifying edit interactions (e.g., [130, 73, 91, 83, 138, 79]). Most, however, employ simple co-editing of articles, some sort of sequential editing of the same articles by two editors, identity reverts, or other measures that do not take into consideration the exact word content changed by each editor. In fact, and to the best of our knowledge, just a few works have concerned themselves with formalizing a network among editors built from the explicit *word-level interactions editors carry out on each others content* in Wikipedia articles.

One very promising approach in this regard is proposed by Brandes et al. [19]. It aims to improve on the straightforward identity-revert-based method used, e.g., by Suh et al. [138], noting that this technique does "not consider who deletes how much of whose edits or who restores whose edits deleted by whom. However, [...] it is exactly this information that enables us to characterize individual authors and groups of authors" (much along the lines of our argumentation in Section 5.1). To allow a more fine grained network construction and depiction, Brandes et al. hence improve this method in [19] to infer an edge (v, u), weighted with the exact words written by editor u that were subsequently (dis)agreed on by editor v. They propose "disagreement" edge types for deleting and reintroducing content plus an "agree" type for restoring. Similarly, Maniu et al. [103] infer a signed network of positive (agree) and negative (disagree) relations between editors by – among other relationships – extracting changed words via text deltas. However, to verify if an edit de facto constitutes a revert to a former revision, the actual text editing actions are not taken into account but only the fairly sparse and unreliable edit comments. While these two approaches are agnostic to the semantics of the deleted and added tokens (i.e., text strings), Bogdanov et al. [16] propose to extract (dis)agreement between editors with the help of topic models. By employing Latent Dirichlet Allocation (LDA), they assign a predefined set of latent topics to an article. The article is then split into paragraphs and by measuring (via cosine similarity) how much an edit of a user increases or decreases the presence of a latent topic compared to another user's edit, a (dis)agreement relation is computed. While this approach tries to tackle some inherent problems in ignoring the meaning of deleted and added tokens, its utility and accuracy was only preliminarily tested and defining the right number of latent topics automatically for a large number of articles is still a hard task.

In the remainder, we will built upon the ideas for modeling the interaction network brought forth by the approach of Brandes et al. [19], as it is the most methodically sound in our view and has

been adopted most in subsequent work. The interactions captured in the model by Brandes et al. are deletion, undo of deletion, and reintroduction of content. We will however expand on this approach and add additional interaction relations we deem necessary as well as the notion of full and partial reverts. Further, we formally define a revision-revision network as a first step in building the editor-editor graph.

5.2.2 *Revision-Revision Network*

First, we define the revision-revision network. Like [19], we distinguish between positive and negative directed interactions, which represent agreement and disagreement among editors, respectively. In line with [19] we are only defining actions of *edits manipulating pre-existing content*; hence, simply adding text without affecting another revisions' text tokens or the act of simply consecutively editing (without knowledge about what was changed) are not taken into account, unlike in other approaches, e.g., [130, 73, 79]. Interpreting such interactions as any kind of (dis)agreement is simply too ambiguous.

The model we present in this section can be understood as generally applicable to all CWS and is based only on revisions. It is built upon the formal model of content provenance and tracking we introduced in Chapter 4. It is therefore also the most comprehensive formal model of word-level edit interactions to date.

Note that each edit (leading to a revision) is carried out by exactly one editor, so that in a later step we can aggregate the nodes representing revisions for each editor. Yet, for the sake of simplicity, we will forgo the notion of editors for now. This is equal to assuming the hypothetical case that each editor generates exactly one revision.

Definition 8 *(Revision-Revision Network.) Given the graph of a revisioned content document* $G = (V = \{R \cup P \cup S \cup T\}, E, \phi)$ (cf. Definition 1)*, the revision-revision network of* G *is a directed graph* $\overline{G} = (\overline{V}, \overline{E}, (\overline{l}, \overline{w}))$ *defined as follows:*

- *The set of vertices* $\overline{V}$ *is composed of revision nodes from* G *, i.e.,* $\overline{V} = R$*.*
- *The set of arcs* $\overline{E} \subseteq \overline{V} \times \overline{V}$ *represents interactions between revisions. A revision interacts with another revision when portions of content are modified, i.e., interactions can be deletions, undo of deletions, reintroductions, redeletions, or undo of reintroductions of tokens.*
- *A labeling function* $\overline{l} : \overline{E} \rightarrow$ *{'deletion', 'undo-deletion', 'reintroduction', 'redeletion', 'undo-reintroduction'} that denotes the type of interaction between revisions.*
- *A weight function* $\overline{w} : \overline{E} \rightarrow \mathbb{N}$ *over the arcs in* $\overline{G}$ *that represents the number of tokens that were involved in the interaction.*

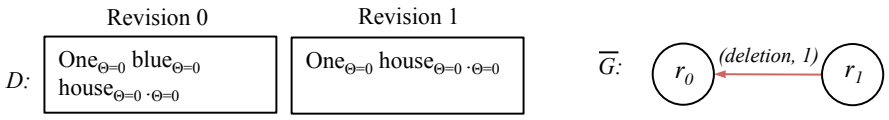


Figure 10: **Example of a deleting content interaction.** Revision 1 (represented by node r_1) deletes one token ('blue') from Revision 0 (represented by node r_0). This interaction is represented with an arc from r_1 to r_0 with label *'deletion'* and weight 1 (Note that the period "." is part of the text, not the notation).

Given is $G = (V = \{R \cup P \cup S \cup T\}, E, \phi)$ a graph of a revisioned document, and a revision-revision network $\overline{G} = (\overline{V}, \overline{E}, (\overline{l}, \overline{w}))$. Assume that $\mathtt{path}(a, b)$ corresponds to the set of paths from vertex a to vertex b in G, as defined in Chapter 4.

We devise five different types of interactions between revisions. The first interaction defines the deletion of content which consists of removing content that was originally created in a previous revision. For example, consider the revisioned document D from Figure 10. Revision 1 removes the token 'blue' which originated in Revision 0 ($\Theta = 0$), therefore, an arc from Revision 1 to Revision 0 exists in $\overline{G}$. The arc is annotated with the label *'deletion'* and its weight equals 1, since one token was deleted.

Definition 9 *(Interaction: Deleting Content). Let r_i, r_j be two different revisions ($r_i, r_j \in R$, with $j > i$). The interaction of deleting content in r_j from r_i occurs when r_j removes a set of tokens D_{ij} that were originally created in r_i. Formally, D_{ij} can be defined as follows:*

$$
\begin{aligned}
D_{ij} := \{t \mid t \in T \wedge & && \textit{t is a token in the document} \\
\Theta(t) = \mathtt{label}(r_i) \wedge & && \textit{the origin of } t \textit{ is } r_i \\
\mathtt{deleteContent}(r_j, t)\} & && \textit{token } t \textit{ is deleted in } r_j \textit{ (Definition 5)}
\end{aligned}
$$

An arc $\overline{e}$ from r_j to r_i ($\overline{e} = (r_j, r_i)$) with label 'deletion' ($\overline{l}(e)$ ='deletion') exists in $\overline{G}$ if and only if the set of removed tokens D_{ij} is non-empty. The weight of the arc $\overline{e}$ ($\overline{w}(\overline{e})$) is computed as $|D_{ij}|$. Deleting content is considered a negative interaction between r_j and r_i.

The second interaction defines when the deletion of tokens is undone. Undoing a deletion consists of restoring tokens that have been removed in a previous revision. For instance, consider the example from Figure 11 where Revision 2 has reintroduced the token 'blue' previously deleted by Revision 1. In this case, an arc from Revision 2 to Revision 1 exists in $\overline{G}$, and it is annotated with the label *'undo-deletion'* and weight equal to 1 (Revision 2 also newly adds a second sentence in our example, but this does not influence or induce any other action).

Definition 10 *(Interaction: Undoing a Deletion). Let r_i, r_j be two different revisions ($r_i, r_j \in R$ with $j > i$). The interaction of undoing a deletion in r_j*

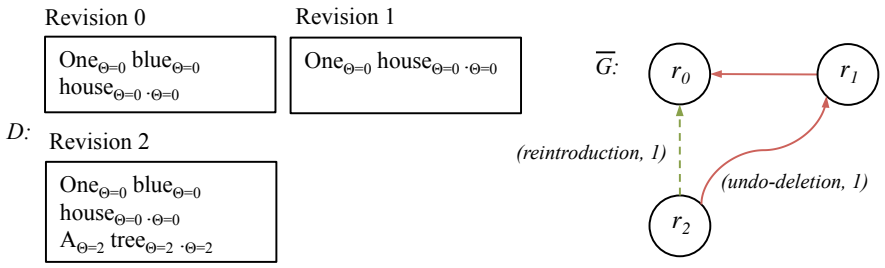


Figure 11: **Example of interactions: undoing a deletion and reintroduction of content.** Solid (red) arcs represent negative interaction between revisions; dashed (green) arcs represent positive interactions. Revision 2 (represented by node r_2) undoes the deletion performed by Revision 1 (node r_1). This interaction is represented in $\overline{G}$ with an arc from r_2 to r_1 with label *"undo-deletion"* and weight 1. In addition, when undoing the deletion of Revision 1, Revision 2 reintroduced the token from Revision 0. Therefore, the corresponding arc with label *'reintroduction'* is created in $\overline{G}$ from r_2 to r_0.

that was performed in r_i *consists of restoring a set of tokens* UD_{ij} *in* r_j *that were removed by* r_i*. Formally,* UD_{ij} *is defined as follows:*

$$
\begin{aligned}
UD_{ij} := \{t \mid\ & t \in T \wedge && t \textit{ is a token in the document}\\
& label(r_i) < label(r_j) \wedge && r_i \textit{ was created before } r_j\\
& deleteContent(r_i, t) \wedge && \textit{token } t \textit{ is deleted in revision } r_i\\
& \exists \rho(\rho \in path(r_j, t))\} && \textit{token } t \textit{ appears in revision } r_j\\
& \nexists \rho'(\rho' \in path(r_{j-1}, t))\} && t \textit{ does not appear in revision } r_{j-1}
\end{aligned}
$$

An arc $\overline{e}$ *from* r_j *to* r_i *(*$\overline{e} = (r_j, r_i)$*) with label 'undo-deletion' (*$\overline{l}(e)$ *='undo-deletion') exists in* $\overline{G}$ *if and only if the set of tokens* UD_{ij} *is non-empty. The weight of the arc* $\overline{e}$ *(*$\overline{w}(\overline{e})$*) is computed as* $|UD_{ij}|$*. Undoing a deletion is considered a negative interaction between* r_j *and* r_i*.*

The third type of interaction defined in the network is a direct consequence of undoing deletions. The action of undoing the deletion of tokens is equivalent to reintroducing (or restoring) tokens from a previous revision. Following our running example, consider the revisioned document D presented in Figure 11. When Revision 2 undid the deletion performed in Revision 1, it reintroduced the token 'blue' originally written by Revision 0. Therefore, $\overline{G}$ contains an arc from Revision 2 to Revision 0 labelled as *'reintroduction'* with weight 1.

Definition 11 *(Interaction: Reintroducing Content). Let* r_i, r_j *be two different revisions (*$r_i, r_j \in R$ *with* $j > i$*). The interaction of reintroducing content in* r_j *from* r_i *occurs when* r_j *restores a set of tokens* RI_{ij} *that were*

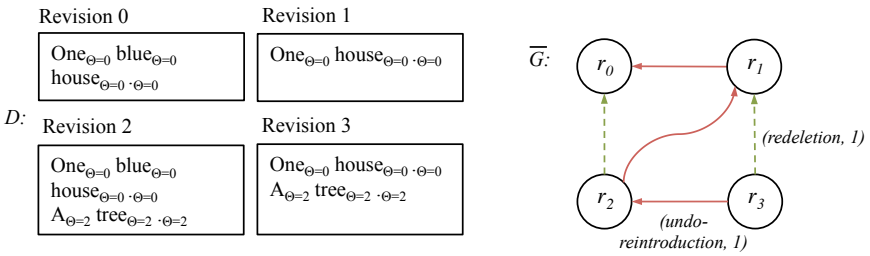


Figure 12: **Example of interactions: redeletion of content and undoing a reintroduction.** Solid (red) arcs represent negative interaction between revisions; dashed (green) arcs represent positive interactions. Revision 3 (represented by node r_3) redeletes the token that was already removed by Revision 1 (node r_1), but reintroduced by r_2. This interaction is represented with an arc in $\overline{G}$ from r_3 to r_1 with label *"redeletion"* and weight 1. In addition, Revision 3 undid the reintroduction of the token 'Blue' in Revision 2. Therefore, the corresponding arc with label *'undo-reintroduction'* is created in $\overline{G}$ from r_3 to r_2.

originally created in r_i *and were removed in another revision* r_k *(*$r_k \in R$ *with* $j > k > i$*). Formally, the set* RI_{ij} *is defined as follows:*

$$
\begin{aligned}
RI_{ij} := \{t \mid t \in T \wedge \quad & && t \textit{ is a token in the document}\\
\Theta(t) = label(r_i) \wedge \quad & && \textit{the origin of } t \textit{ is } r_i\\
label(r_k) < label(r_j) \wedge \quad & && r_k \textit{ was created before } r_j\\
deleteContent(r_k, t)) \wedge \quad & && t \textit{ is deleted in } r_k\\
\exists \rho(\rho \in path(r_j, t)) \wedge \quad & && t \textit{ appears in revision } r_j\\
\nexists \rho'(\rho' \in path(r_{j-1}, t))\} \quad & && t \textit{ does not appear in revision } r_{j-1}
\end{aligned}
$$

An arc $\overline{e}$ *from* r_j *to* r_i *(*$\overline{e} = (r_j, r_i)$*) with label 'reintroduction' (*$\overline{l}(e)$ *='reintroduction') exists in* $\overline{G}$ *if and only if the set of reintroduced tokens* RI_{ij} *is non-empty. The weight of the arc* $\overline{e}$ *(*$\overline{w}(\overline{e})$*) is computed as* $|RI_{ij}|$*. Reintroducing content is considered a positive interaction between* r_j *and* r_i*.*

Property 5 *Given are three different revisions* r_i, r_k, $r_j \in R$*. Assume that* r_k *deletes tokens that originated in revision* r_i*, and that* r_j *restores those tokens. An arc* $\overline{e}_1 = (r_j, r_i)$ *exists in* $\overline{G}$ *with* $\overline{l}(\overline{e}_1)$ = *'reintroduction' if and only if there is an arc* $\overline{e}_2 = (r_j, r_k)$ *in* $\overline{G}$ *such that* $\overline{l}(\overline{e}_2)$ = *'undo-deletion'.*

The fourth interaction represents the redeletion of content. This interaction is performed when content that was previously removed (and subsequently reintroduced) is deleted once again. Consider the example from Figure 12 in which Revision 3 redeletes the token 'blue' that was removed by Revision 1. Analogous to previous examples, $\overline{G}$ contains an arc from the node of Revision 3 to Revision 1, with label *'redeletion'* and weight 1.

Definition 12 *(Interaction: Redeleting Content). Let* r_i, r_j *be two different revisions (*$r_i, r_j \in R$ *with* $j > i$*). The interaction of redeleting content in* r_j

consists of removing a set of tokens RD_{ij} *that were previously deleted in* r_i. *Formally, the set* RD_{ij} *is defined as follows:*

$$\begin{aligned} RD_{ij} := \{t \mid t \in T \wedge & && t \textit{ is a token in the document} \\ & label(r_i) < label(r_j) \wedge && r_i \textit{ was created before } r_j \\ & deleteContent(r_i, t) \wedge && t \textit{ was deleted in revision } r_i \\ & deleteContent(r_j, t)\} && t \textit{ is also deleted in } r_j \end{aligned}$$

An arc $\overline{e}$ *from* r_j *to* r_i *(*$\overline{e} = (r_j, r_i)$*) with label 'redeletion' (*$\overline{l}(e)$ *='redeletion') exists in* $\overline{G}$ *if and only if the set of redeleted tokens* RD_{ij} *is non-empty. The weight of the arc* $\overline{e}$ *(*$\overline{w}(\overline{e})$*) is computed as* $|RD_{ij}|$. *Redeleting content is considered a positive interaction between* r_j *and* r_i.

The last operation corresponds to undoing a reintroduction. Following the example presented in Figure 12, Revision 3 has undone the reintroduction of the token 'blue' performed by Revision 2. In this case, $\overline{G}$ contains an arc between Revision 3 and Revision 2, with label *'undo-reintroduction'* and weight 1.

Definition 13 *(Interaction: Undoing a Reintroduction.) Let* r_i, r_j *be two different revisions (*$r_i, r_j \in R$ *with* $j > i$*). The interaction of undoing a reintroduction in* r_j *consists of removing tokens that were previously reintroduced in* r_i.

$$\begin{aligned} UR_{ij} := \{t \mid t \in T \wedge & && t \textit{ is a token} \\ & \exists r_k (r_k \in R \wedge && r_k \textit{ is a revision} \\ & label(r_k) < label(r_i) \wedge && r_k \textit{ was created before } r_i \\ & deleteContent(r_k, t)) \wedge && r_k \textit{ deletes } t \\ & \nexists \rho' (\rho' \in path(r_{i-1}, t)) \wedge && t \textit{ does not appear in revision } r_{i-1} \\ & \exists \rho (\rho \in path(r_i, t)) \wedge && t \textit{ appears in } r_i \\ & deleteContent(r_j, t)\} && t \textit{ is deleted in } r_j \end{aligned}$$

An arc $\overline{e}$ *from* r_j *to* r_i *(*$\overline{e} = (r_j, r_i)$*) with label 'undo-reintroduction' (*$\overline{l}(e) =$ *'undo-reintroduction') exists in* $\overline{G}$ *if and only if the set of tokens* UR_{ij} *is non-empty. The weight of the arc* $\overline{e}$ *(*$\overline{w}(\overline{e})$*) is computed as* $|UR_{ij}|$. *Undoing the reintroduction of content is considered a negative interaction between* r_j *and* r_i.

Property 6 *Given are three different revisions* r_k, r_i, $r_j \in R$. *Assume that* r_i *undoes the deletion of tokens performed by revision* r_k, *and that* r_j *redeletes those tokens. An arc* $\overline{e}_1 = (r_j, r_k)$ *exists in* $\overline{G}$ *with* $\overline{l}(\overline{e}_1)$ = *'redeletion' if and only if there is an arc* $\overline{e}_2 = (r_j, r_i)$ *in* $\overline{G}$, *such that* $\overline{l}(\overline{e}_2)$ = *'undo-reintroduction'.*

5.2.3 *Proposed Algorithm*

In order to detect the previously defined interactions, we have extended the WikiWho algorithm presented in Section 4.3. The extensions (see blue-colored lines in Algorithm 2) consist of verifying in

certain parts of the algorithm whether the conditions to construct the set of involved tokens in the different interactions hold true. The outcome of Algorithm 2 is a graph $\overline{G}$ that contains the interactions among the revisions in D. To illustrate the execution of the extensions of the algorithm, consider the revisioned document D from Figure 12. In this example, the document D is composed of four revisions. Similarly to WikiWho, the algorithm starts processing Revision 0 ($i = 0$) and computes the authorship for each token that is contained in this revision. Note that since this is the first revision, there are no interactions in the multigraph $\overline{G}$.

In the next iteration, Algorithm 2 processes Revision 1 ($i = 1$). In this revision, the tokens 'One', 'house', and '.' are copied from Revision 0 (lines 20-22). Given that these three tokens have not been deleted in the past, UD_{ki} and RI_{ki} are both equal to $\emptyset$ for all $k < 1$ (lines 23-26). After the algorithm has processed the tokens within the revision, it checks wether tokens were removed (lines 34-39). In this case, the token 'blue' has been deleted for the first time by Revision 1. Therefore, $D_{1,0}$ is not empty and the algorithm adds the interaction 'deletion' from r_1 to r_0 (line 36). The procedure *createInteraction* (see Algorithm 3) updates the corresponding structures of the multigraph $\overline{G}$.

After processing Revision 1, Algorithm 2 issues the next revision ($i = 2$). When processing the content of Revision 2 the algorithm detects (lines 20-22) that the tokens 'One' and 'house' have been copied from the previous revision, and the token 'blue' from Revision 0. Moreover, the token 'blue' was deleted in Revision 1, therefore the following interactions are created in $\overline{G}$ (lines 23-26): *'undo-deletion'* from r_2 to r_1, and *reintroduction* from r_2 to r_0. According to WikiWho, the tokens in the sentence 'A tree .' are new and are annotated with the authorship $\Theta = 2$ (lines 29-30). Once all the tokens in Revision 2 have been processed, the algorithm checks if tokens from previous revisions have been deleted. Since this is not the case, no further interactions are created.

Lastly, Revision 3 is processed by Algorithm 2. In this case, all the tokens have been copied from previous revisions (lines 20-22) and no reintroductions (or undo-deletions) have been performed (lines 23-26). The algorithm then proceeds to check whether tokens have been deleted in the current revision (lines 34-39). As depicted in Figure 12, the token 'blue' has been removed once again, therefore the sets $RD_{3,1}$ and $UR_{3,2}$ are non-empty. The algorithm then creates the following interactions in $\overline{G}$: 'redeletion' from r_3 to r_1 (line 37), and 'undo-reintroduction' from r_3 to r_2 (line 39). The algorithm finalizes when all revisions have been processed and returns the graph $\overline{G}$ that contains the identified interactions.

Algorithm 2: Algorithm to Build a Revision-Revision Network

Input: A document D with revisioned content $r_0, r_1, ...r_{n-1}$.
Output: $\overline{G} = (\overline{V}, \overline{E}, (\overline{l}, \overline{w}))$, the revision-revision network from D.

```
1  Create an empty graph G = (V, E, ℕ_0, ϕ)
2  Create an empty multigraph Ḡ = (V̄, Ē, (l̄, w̄))
3  Create an empty queue Q
4  for i in 0, 1...n − 1 do
5      G.createRevision(r_i)
6      label(r_i) ← i
7      y′ ← tokenize(r_i)
8      Enqueue (r_i, y) onto Q for all y in y′
9      x_prev ← NULL
10     diffed ← FALSE
11     while Q is not empty do
12         (x, y) ← Q.dequeue()
13         if x is a sentence ∧ !diffed then
14             diff unmarked tokens of r_{i−1} against unmarked tokens of
               r_i (i > 0)
15             diffed ← TRUE
16         if x = x_prev then
17             α ← α + 1
18         else
19             α ← 0 ; x_prev ← x
20         if y ∈ V ∧ y is not marked then
21             G.copyContent(x, y, α)
22             Mark all the nodes reachable from y, including y.
23             for revision r_k(k < i) such that UD_ki ≠ ∅ do
24                 createInteraction(r_i, r_k, 'undo-deletion', |UD_ki|, Ḡ)
25             for revision r_k(k < i) such that RI_ki ≠ ∅ do
26                 createInteraction(r_i, r_k, 'reintroduction', |RI_ki|, Ḡ)
27         else
28             G.createContent(x, y, α)
29             if y is a token then
30                 Θ(y) ← label(r_i)
31             else
32                 z′ ← tokenize(y)
33                 Enqueue (y, z) onto Q for all z in z′
34         for revision r_k(k < i) such that D_ki ≠ ∅ do
35             createInteraction(r_i, r_k, 'deletion', |D_ki|, Ḡ)
36         for revision r_k(k < i) such that RD_ki ≠ ∅ do
37             createInteraction(r_i, r_k, 'redeletion', |RD_ki|, Ḡ)
38         for revision r_k(k < i) such that UR_ki ≠ ∅ do
39             createInteraction(r_i, r_k, 'undo-reintroduction', |UR_ki|, Ḡ)
40     Unmark all the marked nodes
41 return Ḡ
```

Algorithm 3: Procedure to Update the Structures of a Revision-Revision Network

```
def createInteraction(source, target, label, weight, G = (V, E, (l, w))):
    V ← V ∪ {source, target}
    e ← (source, target)
    l(e) ← label
    w(e) ← weight
    E ← E ∪ {e}
```

5.2.4 *Addendum: Full and Partial Reverts*

Following the running example and our interaction definitions, it is apparent that certain edit actions reverse the actions of other edits. For instance, in Figure 12, Revision 2 reverses the deletion of the token 'blue' performed in Revision 1, which itself reversed the addition of 'blue' done in Revision 0. In the following, we provide a definition of inverse interactions that may occur in a revision-revision network. Inverse interactions provide the grounds to formally define partial and full reverts, which were introduced in Section 5.1.3.1.

Definition 14 *(Inverse Interactions.) Given a revision-revision network, an inverse interaction consists of reversing the actions of a graph operation or another interaction. The inverse interactions that generate negative relationships between revisions are as follows:*

- *The inverse of the graph operation 'creation of content' (Definition 3) is the interaction 'deletion' (Definition 5).*
- *The inverse of the interaction 'deletion' (Definition 5) is the interaction 'undo-deletion' (Definition 10).*
- *The inverse of the interaction 'reintroduction' (Definition 11) is the interaction 'undo-reintroduction' (Definition 13).*

First, we formally define a full revert which takes place when all edit actions performed in a revision r_x are reversed by another revision r_y *and* the inverse interactions of r_y are applied to all the tokens modified in r_x.

Definition 15 *(Full Revert). Let r_x and r_y be two different revisions ($r_x, r_y \in R$ with $y > x$). Revision r_y fully reverts r_x if the following conditions are met simultaneously:*

- *For each interaction (or graph operation 'Creation of content') f performed by revision r_x exists an interaction g performed by r_y such that g is the inverse of f.*

- *Given* r_x *and* r_y*, where* f *is an interaction (or graph operation 'Creation of content') performed by* r_x*, and* g *is an inverse interaction in respect to* f *performed by* r_y*, assume that* T_f *(*$T_f \neq \emptyset$*) and* T_g *(*$T_g \neq \emptyset$*) are the sets of tokens involved in* f *and* g*, respectively. Revision* r_y *is a full revert of* r_x *if* $T_f \cap T_g = T_f$.

Next, we introduce the definition of partial revert, which occurs between revisions when *at least one but not all* actions are reversed by another revision, *or* when at least one inverse interaction is applied only to a subset of tokens.

Definition 16 *(Partial Revert). Let* r_x *and* r_y *be two different revisions (*$r_x, r_y \in R$ *with* $y > x$*). Revision* r_y *partially reverts* r_x *if one of the following conditions is met:*

- *There exists at least one interaction (or graph operation 'Creation of content')* f *performed by revision* r_x *such that there is no interaction* g *performed by* r_y *where* g *is the inverse of* f and *at least one interaction* g' *is performed by* r_y *that is an inverse of* f' *performed by* r_x.
- *Given* f *and* g*, where* f *is an interaction (or graph operation 'Creation of content') performed by* r_x*, and* g *is the inverse interaction of* f*, performed by* r_y*, assume that* T_f *(*$T_f \neq \emptyset$*) and* T_g *(*$T_g \neq \emptyset$*) are the sets of tokens involved in* f *and* g*, respectively. Revision* r_y *is a partial revert of* r_x *if* $T_f \cap T_g \subset T_f$.

5.2.5 *Editor-Editor Network*

Lastly, to represent how the edit actions of an editor relate her to other editors, we extend the model of revision-revision networks previously introduced in Section 5.2.2. Below, we formalize how the nodes and edges of the graph in a revision-revision network are mapped to the corresponding structures of an editor-editor network.

Definition 17 *(Mapping a Revision-Revision Network to an Editor-Editor Network). Given the graph of a revision-revision network* $\overline{G} = (\overline{V}, \overline{E}, (\overline{l}, \overline{w}))$*, an editor-editor network* $\overline{G'} = (\overline{V'}, \overline{E'}, (\overline{l'}, \overline{w'}))$ *is a directed multigraph built as follows:*

- *For each revision* r_i *in* $\overline{V}$, $\mathtt{editor}(r_i)$ *belongs to* $\overline{V'}$ *where* $\mathtt{editor}(r_i)$ *denotes the creator of revision* r_i.
- *Each arc* $e = (r_i, r_j)$ *in* $\overline{E}$ *is mapped to* $e' = (\mathtt{editor}(r_i), \mathtt{editor}(r_j))$ *belonging to* $\overline{E'}$*, with* $\overline{l'}(e') = \overline{l}(e)$ *and* $\overline{w'}(e') = \overline{w}(e)$.

For most ends and purposes, such as visualizations and statistical analyses, the editor-editor network is a more practical representation of the interactions mined from revisions. In a further transformation

step, the edges between editors can for example be simply aggregated per type of interaction or per polarity (negative/positive) to form a more simple-to-analyze social graph. This will result in a graph similar to that proposed by Brandes et al. [19]. Yet, the information attached to the revision-revision network can also be preserved in the editor-editor network, for example to discount older interactions between editors, if needed in a specific analysis use case.

5.3 CONCLUSIONS

In this chapter, we first investigated the nature of disagreement relationships between editors by challenging the common technique applied by research to mine "reverts". By adhering more closely to Wikipedia's own definition and incorporating human assessment, we were able to more accurately decide what disagreements to represent in an explicit model and how. Apart from better detecting full reverts and distinguishing full and partial reverts, this study also provided learnings about the nature and types of disagreement in a large CWS. These insights were built upon in modeling the revision-revision network, and on top, the editor-editor network, and helped us to seamlessly extend the actual WikiWho algorithm to capture such interactions. Thereby, our social network mining method from word-level interactions is the first one to be based on a text-difference and provenance tracking method that was actually tested for its accuracy. This is not the case for any other algorithm used so far in the literature to mine editor-editor networks.

5.3.1 *Future Research*

While different types of disagreement and agreement are captured in our presented approach, the weight of a (dis)agreement is based solely on the number of changed tokens. A more sophisticated method could as well include the semantics of the changed tokens or of the whole enclosing linguistic units (sentence, paragraphs, sections) and weigh certain interactions higher or categorize them along dimensions, especially for disagreement. Such could be spelling and grammar corrections, updates because of current events, small corrections while leaving most of an author's content intact, major corrections and complete disagreement, to name some possible instances. These categories might be mappable to a continuum of disagreement to be translated into specific numerical weights on edges and could thereby help to give increased prominence to more substantial disagreements between editors. We have already experimented with part-of-speech tagging to find simple update-related deletions and undos as well as edit-distance measures to classify minor changes.

Regarding agreement interactions, it might be possible to develop a reliable model for inferring agreement by co-editing specific parts of articles together, although it will have to be based on strong reasoning about editor motivation and attention as well as evaluations. The method for interaction extraction presented here, although reliable in mining the raw text changes and unambiguous in translating them into a network graph must also be evaluated in future work in terms of the meaningfulness of the provided data, particularly taking into account the different meaning interactions can take as discussed above. However, it provides a first solid foundation for modeling editor interactions that can be built upon.

6 VISUALIZATION TOOLS

Wikipedia as a socio-technical system has received its fair share of attention over the last decade and very much has been learned by scholars and community members about its inner workings. Yet, understanding the collaborative writing history of a *specific article* as a casual user, editor or even researcher in an easy, intuitive way (i.e., without relying on elaborate statistical analysis) is still a hard task. There is a lack of transparency regarding the editing process on Wikipedia: it is fully documented in the revision history, but not in a way that is straightforward to browse, inspect and analyze by humans in all its intricacy. For instance, one cannot easily discover which words were contributed by what author or what specific dynamics governed the rise of disagreement between editors on particular content in the article. This information would be key to enable *accountability and social transparency*, as has been argued by Suh et al. [139], but is hidden from the user due to its innate complexity.

Some visualization tools, e.g., "Wikitrust" [41], "History Flow" [153], "Wikidashboard" [139] and community solutions have been proposed;[1] but most have since been discontinued as a service and further provide only solutions to specific subproblems of the complex phenomenon that is understanding the collaborative writing of a Wiki article. Therefore, tools that allow users to visually explore the dynamic relationships between editors which emerge from the main activity in the system – the collaborative process of adding, deleting and restoring specific content – are not available or not equipped for the purpose of accurately reflecting all relevant interactions of editors with each other and the content.

We hence argue in this chapter that better software tools that allow end-users to visually explore the dynamic, collaborative process of adding, deleting and restoring specific content have to be made available for the purpose of accurately reflecting all relevant interactions of editors with each other and the emergence of content; and that those should be *integrated* to allow a seamless exploration of all relevant editing activity. We also make the case why such tools would be key to (i) enable more transparency and thus reduce the complexity that is inherent to the writing processes of (especially controversial) articles on Wikipedia; and (ii) why such transparency empowers readers, editors and researchers to better comprehend the context of an article's emergence, and to interpret its content accordingly (e.g., by acknowledging opinion camps and biased behaviors, ownership or

1 For community tools see http://en.wikipedia.org/wiki/Wikipedia:Tools

edit warring behavior of single authors, etc. and the effect they have on the eventual content presented). Providing suited visual tools to explore the article history in terms of content and editor interactions is therefore an essential step towards assuring *fully informed readers and editors*.

In this Chapter we (i) present our proposed tools whoCOLOR and whoVIS, and show how these tools can help tackling the issue of insufficient transparency of the editing process and content emergence; further we (ii) outline how an integration and further development of such and similar tools can provide a user with insights each of these implementations could not supply individually; and (iii), as a by-product, we conduct a qualitative analysis of the evolution and disputed content of the "Tropical Storm Alberto (2006)" and "Gamergate controversy" articles, which we employ as exemplary use cases.

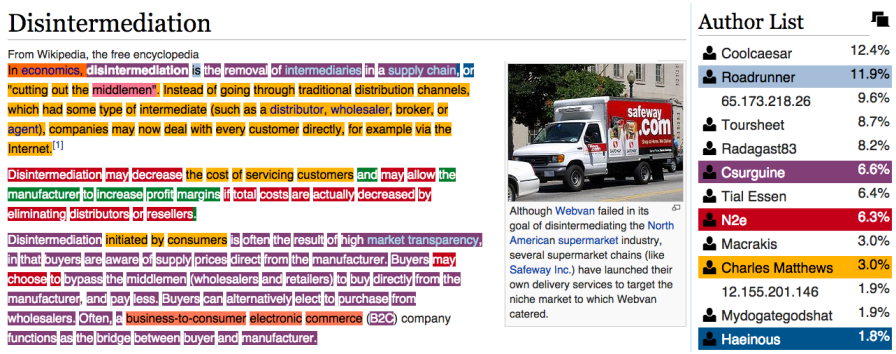
The first tool we present is whoCOLOR, a visual overlay on Wikipedia articles enabling users to inspect provenance, historic development and disagreements regarding any content element of an article, computed on the latest revision data of an article. The second tool, whoVIS, is aimed at visualizing the disagreement network between editors and enabling its exploration over time, together with auxiliary information.

6.1 WHOCOLOR

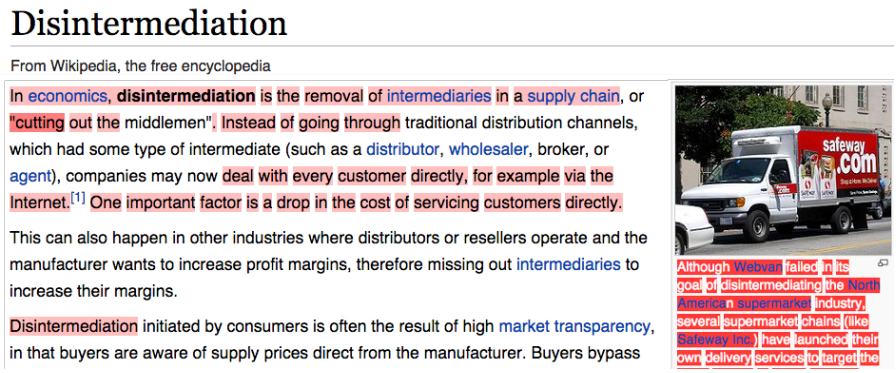
whoCOLOR consists of a userscript meant to be loaded in a client's Web browser with the Tampermonkey/Greasemonkey[2] extensions and a server-side service to provide the needed data for the userscript. The userscript in the client's browser is activated when an article page of the English Wikipedia is loaded. Once loaded, the tool offers an overlay for the article content in three views: (i) the Provenance View, (ii) the Conflict View and (iii) the Word History View, as exemplified in Figure 13.

In the Provenance View, by hovering over text passages, the user is notified of (i) the author of the selected words through a highlight-effect in the author-list that is located on the right-hand side of the article content, with authors ranked by the percentage of text they have originally written; and the user (ii) also sees all other text highlighted written by the same author. By clicking on words or authors, the user can then make the highlighting of the author and all her written words permanent in a unique color. The approach is heavily inspired by similar work done for the community solution "WikiPraise" by Wikipedia User Netaction, that was based on the now-defunct Wik-

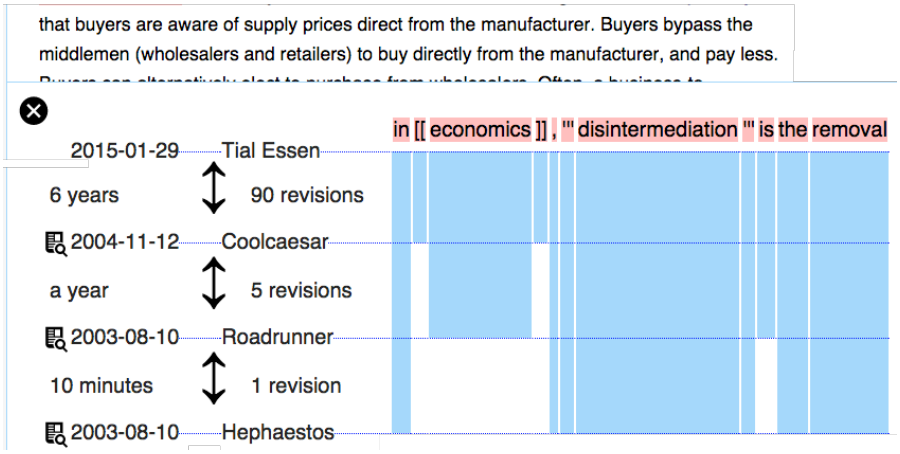
2 Both extensions enable executing JavaScripts that post-process webpages loaded in the browser. https://tampermonkey.net for Google Chrome, https://www.greasespot.net for Mozilla Firefox. The userscript can be found and tried out under http://f-squared.org/whovisual/, although as of writing only the Provenance View is enabled for arbitrary articles.



(a) Provenance View, depicting the words originally written by certain editors as colored markup, with author listed on the right, ordered by the percentage of content they have authored in the whole article.



(b) Conflict View, highlighting more controversial parts of the article in a darker shade of red.



(c) Word History View for a selection of tokens. Blue horizontal lines denote edits that added or deleted one of the inspected tokens, including timestamp and editor. If the area under an edit is white and above blue, the token in that column was added in the revision, and vice versa for deletion. Added tokens in a revision are included in the article for a specific amount of "*(X)* revisions" and "*(X) (time units)*" (left of blue vertical bars).

Figure 13: **The three different views of whoCOLOR** for the article "Disintermediation".

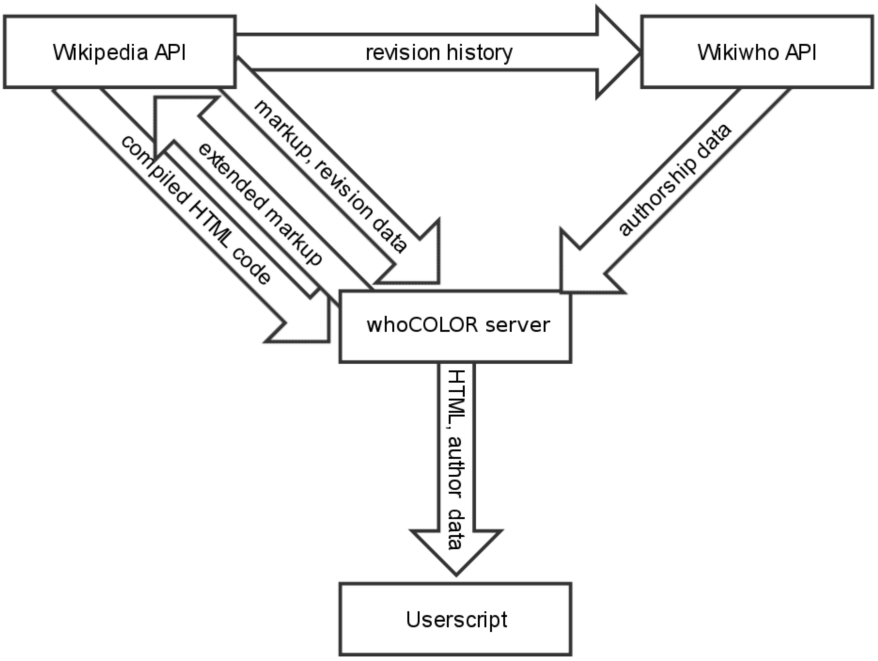


Figure 14: **Architecture of the whoCOLOR infrastructure.** Revision data is requested from the Wikipedia API (revision meta-data, markup text), then enriched with provenance (+ other change data) from the WikiWho API, then sent again to the Wikipedia API, which responds with the final, highlighted HTML that is passed to the userscript in the browser, exchanging the original article HTML with the enriched version. Figure taken unaltered from [136].

itrust API.[3] The Provenance View (like the other views) also works for historic article revisions.

whoCOLOR also features a "Conflict View". It colors those words in the article in a stronger tone of red the more deletes and reintroductions (hence: disputes) they were subjected to in the past (see Figure 13b). The current version implements a conflict measure that sums up the disappearances and reappearances of a word. Future work will likely (i) exclude vandalism and (ii) use a more elaborate metric of conflict. One possible measure for the latter could be the approach proposed by Bykau et al. [23]. By aggregating historic disagreements about specific content parts into one single measure and markup, this view enables a critical inspection of content and is relatively straightforward to interpret.

A third feature is the "Word History", which is available in both previously described views. If the user marks up a sequence of words with the mouse, a dialog appears on the bottom of the page. Once activated, the interface (see Figure 13c) can be used to inspect the periods of time when the selected words were present (blue background) in the article or when they were not (white background) on a timeline.

3 See https://de.wikipedia.org/wiki/Benutzer:NetAction/WikiTrust.

It also shows who (when) removed the content, for how long it was absent, and (if applicable) which user reintroduced it (when). It can hence be particularly helpful in understanding who the antagonists were in possible disputes indicated by the "Conflict View"; or it can simply be used to query for the age of a specific word and jump to its source revision via the links provided next to each revision line.

In order to retrieve the needed data for the different views, the userscript invokes a server-side service of whoCOLOR regarding the specific revision of an article that is viewed by the user. As shown in Figure 14, the whoCOLOR server first queries the Wikipedia API[4] for revision meta-data and content (as Wiki markup text). It then retrieves the needed information about the provenance of each individual word in the text, as well all individual changes to it in the past from the WikiWho API (cf. Section 4.5). With this data, the whoCOLOR server generates a modified version of the Wiki text of the article including annotations, and sends it again to the Wikipedia API for compilation of the interface HTML to be shown to the user. Once this annotated HTML is received, it is passed back to the userscript, which then exchanges the original HTML of the article in the client browser.

6.1.1 *Evaluation*

The "Provenance View", the "Conflict View" and the "Word History" were tested on three dimensions with a small group of eight users.[5] All were "casual" to "frequent" readers (with infrequent editing), none described themselves as "Wikipedian", i.e., active editor. While a more elaborate testing process is needed, this assessment gives a first insight regarding promises and challenges of the proposed solution. Participants were provided with an instruction how to use the views and were further posed a task for which features of a specific view had to be used. The quality of answers was ensured by checking task results, for which gold standard solutions exist.

The participants were then asked how useful the view could be in the users' typical Wikipedia usage, how easy the execution of the task was and how well they understood the relevant view after using it for a while.

6.1.1.1 *Results*

When asked for the *general usefulness of a view for everyday* reading of a Wikipedia article on a scale from 1 to 5, the "Conflict View" was judged to be most useful with an average rating of 3.75, while the "Word History View" scored 3.0 and the "Provenance View" 2.25. One

4 See https://en.wikipedia.org/w/api.php?action=help&modules=main

5 All university mostly students, recruited over personal acquaintance. For further information, also regarding other aspects of the evaluation, please refer to [136].

explanation for the low score of the "Provenance View" could be that users which do not belong to the core editor group of Wikipedia might gain little from user names and profiles of other editors. The "Conflict View" however is interpretable without knowing the concrete actors; this could also be the root of its very high average score of 4.25 regarding the *ease of use* on a 1 to 5 scale. "The Provenance View" scored 3.75 in this regard, with the "Word History View" trailing at 3.13, a fact that was confirmed by free feedback of users: the tracing of the word history seemed not intuitive enough in the interface. Lastly, the participants were requested to provide a rating of 1 to 5 for the *understanding of the different views* they had acquired. Here, the "Conflict View" surprisingly scored slightly lower (4.0) than the "Provenance View" (4.25), while the "Word History View" again reached the lowest average (3.63). This seems to indicate that although the "Provenance View" is easy to understand it is not as practical as the "Conflict View" in general for average readers.

In conclusion, this very preliminary evaluation only gives a first glimpse of the applicability of the proposed tool and more sophisticated and better grounded studies have to follow. Also, we expect the assessment for frequent editors to be considerably distinct, as they face different challenges as readers and are more adept with the use of rather technical tools. As a valuable learning, however, we can conclude that at least the "Conflict View" seems to have potential for usage even for casual users, as was confirmed as well in free comments made by the participants; feedback we have received by early adopters in the Wikipedia community indicates the same. The latter group found, apart from the "Conflict View", especially the "Provenance View" very practical for everyday tasks. With further improvement of the interfaces there seems to be prospects for application for both user types.

6.2 WHOVIS

whoVIS is a Web-based tool to visualize the editor-editor disagreement network in a Wikipedia article over time, derived from the collaborative editing actions on word level in an article. It implements an algorithm to transform and depict the multigraph of an editor-editor network as described in Chapter 5 and makes it navigable over time, enriched with meta-information on editors and edges. This visualization takes form as a network graph with editors as nodes (the node size being the words editors have authored in the selected revision) and disagreements among them as edges, as shown in Figure 15. Users can navigate the editor network over time via a slider or skip buttons (see top of Figure 15) and re-arrange nodes for a better overview. The tool also allows to inspect individual edges for the disagreed-on content by clicking on them. This latter function,

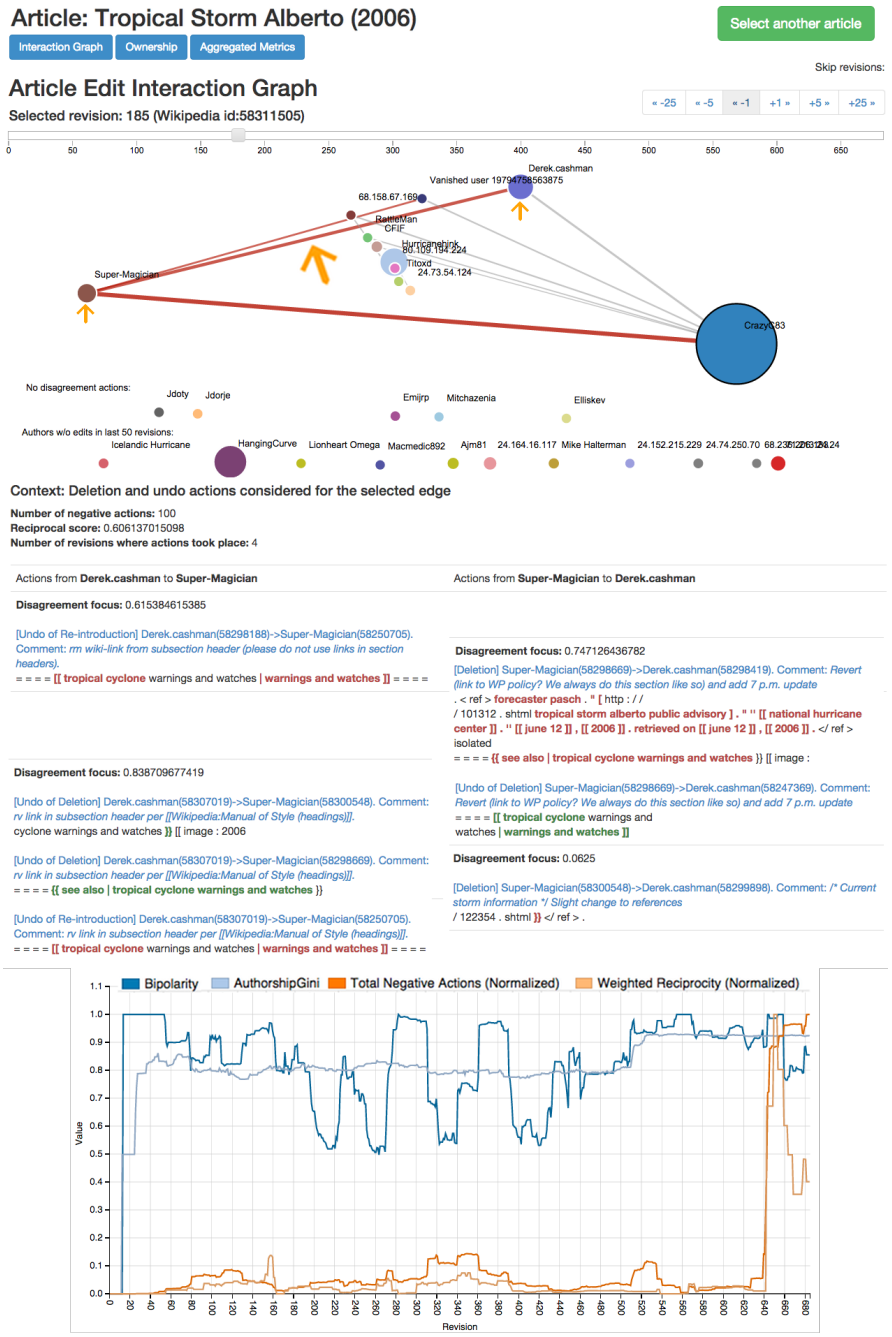
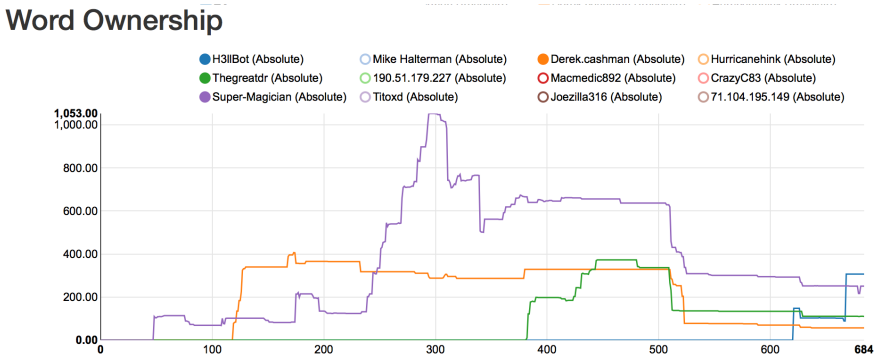
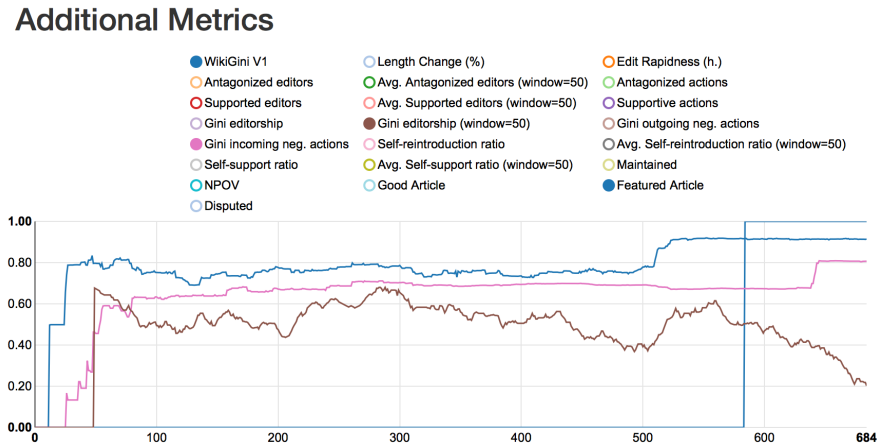


Figure 15: **The main components of whoVIS: the interaction graph (top), edge context (middle) and auxiliary metrics (bottom)**, compressed illustration. Article: "Tropical Storm Alberto (2006)". Inspection of the edge (highlighted via added orange arrows) between Super-Magician and Derek.cashman via edge context reveals a dispute about style policies for headings.

dubbed *edge context*, is – apart from the drawing method – one of the main novel features of whoVIS, and allows edges of the network to be selected to display information about the context of the disagreement, i.e., all the negative actions between editors that were used to



(a) "Word Ownership" view, depicting the amount of words written by certain editors being present in each revision



(b) "Additional Metrics" view with four line graphs enabled

Figure 16: **The two additional views of whoVIS: "Word Ownership" and "Additional Metrics"** for the article "Tropical Storm Alberto (2006)". X-Axes show revisions in the article over time.

construct the edge (see the middle part of Figure 15). This feature adds a qualitative dimension for exploration to the drawn network that enables end-users to gauge the meaning of certain edges and disagreements. Underneath the edge context, line graphs over time for some auxiliary metrics are shown (cf. Section 6.2.3). Apart from the main graph, whoVIS also provides two more view tabs: "Word Ownership" and "Additional Metrics". Once selected, the "Word Ownership" tab shows the absolute amount of words originally authored (owned) by editors u for each revision r_i as a line chart, as shown in Figure 16. Only those editors are selected for display that have been in the top 5 word owners at any given point in time for this article, so as to depict an uncluttered chart and to focus on the main actors. The "Additional Metrics" tab contains another collection of line charts, plotting some of the auxiliary metrics we will describe in Section 6.2.3 as well as indicating with values of 0 and 1 the presence of certain important templates at given revisions, such as "Featured Article" (as in Figure 16), "Disputed" and others.

The principal contribution presented in this section is a working and usable system built on top of the previously presented techniques to mine Wikipedia interactions, which are enriched with new features tailored to this use case. In doing so, we showcase how authorship and interaction mining from revisioned, collaborative writing can be transformed into a useful visual interface for exploring social dynamics and the provenance of content. A demonstration of the system is available online at http://km.aifb.kit.edu/sites/whovis/ at the time of writing. Appendix A.2 gives a much more complete overview of all the whoVIS features.[6]

In the remainder of this section we will elaborate on the technique used to transform the editor graph from Chapter 5 into a temporal representation, explain the construction of the edge context, describe auxiliary metrics we compute and display in the tool, and clarify the concrete implementation of the network graph as a Web application. The subsequent Sections 6.3 and 6.4 will describe application scenarios of whoVIS on concrete article instances.

6.2.1 *Editor-Editor Network over Time*

For the visualization purpose we need to represent the state of the network we defined in Section 5.2.5 at each individual revision over time. This allows us to visualize the editor interaction network graph for each revision of the article to illustrate its evolution. While Brandes et al. [19] propose a visualization approach for this purpose, their technique as presented in [19] reflects only the state of the network at the *last* available revision, conflating all interactions over the whole revision history. Yet, even when drawing the network of every revision r_i in the sequence individually, merely aggregating for all editors all interaction edges stemming from all previous revisions $r_0, ..., r_{i-1}$ is not practical for two reasons: (i) The network graph quickly grows to be cluttered with a high amount of nodes and edges (already at around 70-90 revisions for most articles) and it thus becomes impractical for a user to distinguish interactions between individual editors or see patterns in a sub-graph as nodes and edges highly overlap; (ii) the "current state" of interaction patterns for a revision is not (easily) observable if graph elements generated in older revisions never disappear and thus can be hardly distinguished from recently created elements that might highlight a currently more relevant editing dynamic. For instance, a strong disagreement between two editors u and v generated in revision r_{i-100} would still appear equally as prominent in r_i, although no disagreement between u and v has arisen since and

6 We also recommend opening the graph for "Tropical Storm Alberto (2006)" during the lecture of this section for a better understanding: http://km.aifb.kit.edu/sites/whovis/graph.php?article=Tropical+Storm+Alberto+%282006%29. Please also refer to http://km.aifb.kit.edu/sites/whovis/howto.html

it has consequently no connection to the current editor dynamics at r_i. A more recently prevailing and ongoing disagreement structure for r_i between different editors that emerged, for example, at r_{i-10} might in turn lose salience when displayed alongside the older one. On these grounds, our definition of the editor network over time for revision r_i excludes actions previous to a threshold (window) $r_{i-\omega}$. We used $\omega = 50$ for the current implementation, as a result of experimentation with different values.[7]

Although in Chapter 5, we defined agreement relations between revisions/editors (e.g.,"reintroduction" and "redeletion"), which can be also inferred by our algorithm, only the defined negative (disagreement) interactions will be included in our visualization. The main reason is that the drawing method by Brandes et al. we adopt and build upon in Section 6.2.4 does not natively support negative and positive edges in the same graph but was especially conceived for negative/disagreement edges. Secondly, according to how we defined interactions, negative relations between editors are necessarily much more common than positive ones (agreement edges always follow from disagreements) and hence arguably give a sufficiently complete picture of inter-editor dynamics for this prototype.

In the following, we provide a formal definition of the editor interactions over time, based on the definitions given in previous chapters.

Definition 18 *(Editor-Editor Disagreement Network Over Time.) Given a revisioned document* D *and its history of revisions* $r_1, ..., r_N$*, the edit interaction network over time associated with* D *is defined as an* N*-tuple* $\overline{G'}_\omega = (\overline{G'_1}, ..., \overline{G'_N})$ *where* ω *is the window size and each* $\overline{G'_i}$ *is defined as follows:*

- $\overline{G'_i} = (\overline{V'_i}, \overline{E'_i}, \overline{\alpha'_i}, (\overline{l'_i}, \overline{w'_i}))$ *is the editor-editor network occurring within the window, i.e., between revisions* $r_{i-\omega}$ *and* r_i*.*
- *The nodes in* $\overline{G'_i}$ *correspond to editors* u *that have done at least one edit on* D *between* $r_{i-\omega}$ *and* r_i*, or authors with at least one token originally written by them still present in* r_i*.*
- *A property of all nodes* u *is defined as a function* $\overline{\alpha} : \overline{V'} \to \mathbb{N}_0$*. For each node* $u \in \overline{V'}$*,* $\alpha(r_i)$ *corresponds to the number of tokens in* r_i *that have been originally authored by* u*.*
- *The set of edges* $\overline{E'_i} \in \overline{G'_i}$*; with* $\overline{l'_i}$ *as the negative edit interactions among editors: 'deletion', 'undo-deletion', and 'undo-reintroduction'; and* $\overline{w'_i}$ *representing the number of tokens involved in the interaction.*

7 We also experimented with different decay functions instead of a hard cut-off. One problem with decaying edge weights is, however, that the semantics for how the interactions are translated into thickness of edges change, which might be confusing for the user. Future usability experiments have to determine the optimal function for excluding past interactions.

6.2.2 *Edge Context: Explaining Disagreement*

In existing solutions for depicting editor-interaction in Wikipedia, it is close to impossible for a user to understand *what* exactly editors were (dis)agreeing about from the plain network edges and hence what was the origin of the edges in the first place. We thus introduce *edge context*. When clicking on an edge, all disagreement actions leading to the creation of that edge in the graph will appear below the graph, so as to understand the disagreement in better detail. The context lists all revisions that contained the *'deletion'*, *'undo-deletion'* and *'undo-reintroduction'* interactions the selected edge is based on, from node u to node v and vice versa (listed left and right, cf. middle of Figure 15). Each token being target of a specific action is highlighted and depicted with the closest four tokens to the left and to the right as seen by the editor at the time she took the action. If the direct neighbor tokens of two affected tokens overlap, they are merged. Removals of tokens (*'deletion'*, *'undo-reintroduction'*) are highlighted in red, adding of tokens (*'undo-deletion'*) in green. The edit comment and the source and target revisions for the action are displayed, and a link is given to the Wikipedia diff for the revision.[8]

6.2.3 *Auxiliary Metrics*

We define several metrics that can help to guide a user by (i) providing additional information about the editor relations and patterns explorable in the interaction graph, to better understand their meaning and (ii) by highlighting potentially interesting phases in the development of the article for target-oriented navigating of the sequence of network states per revision. These metrics are displayed partly in the line graph view under the main interaction network graph (cf. Figure 15 bottom) and in the Additional Metrics view (Figure 16). Some, such as reciprocity, are also used in drawing the network graph, as we will see in Section 6.2.4.1.

Given a multigraph $\overline{G'_i} = (\overline{V'_i}, \overline{E'_i}, \overline{\alpha'_i}, (\overline{l'_i}, \overline{w'_i}))$ of the editor-editor network, we define:

1. *Bipolarity*: A degree of how well editors can be divided into two poles of opinion. According to Brandes et al. [19], it has "originally been defined to assess whether political conflict networks decompose into two opposing groups and to visualize conflict networks that have high bipolarity. [...] It estimates to what extend the set of authors can be partitioned into two subsets such that disagreements are more frequent between members of different clusters than between members of the same cluster. If this is the case, the clusters are likely to represent groups of authors

8 Text-diff by Wikiwho and Mediawiki can differ in some instances, which does *not* imply one of the methods being objectively wrong.

that have contradicting opinions. [...] The bipolarity lies between minus one and plus one. It equals plus one if the graph is bipartite, i.e., edges connect only members from different groups and, therefore, the division into opposing groups is perfect. The bipolarity equals zero if the graph is complete (i. e., all pairs of actors are connected) and all edges, including loops, have the same weight. In this case, the partition into two groups is completely arbitrary and does not represent opposing groups." Brandes et al. also propose a concrete calculation of bipolarity for the disagreement network case between Wikipedia editors, which we have adopted for our implementation. Bipolarity is shown as a metric in the line graph under the main network graph.

2. *Authorship Gini Coefficient*: Measures in an interval of 0 to 1 how equal the authorship of tokens is distributed over the editors that have contributed to the content of revision r_i. The closer to 0 the value of the coefficient, the more equally distributed are all written words over all editors that authored any words, while a coefficient closer to 1 signals a ownership of most words to only very few editors. Let c_j with $j = 1...n$ be the sequence of the n editors that own at least one token in r_i, indexed in non-decreasing order of authorship at revision r_i ($\overline{\alpha'_i}$) , we define:

$$\mathtt{authorship\_gini}_i = \frac{2 \cdot \sum_{j=1}^{n} j \cdot \overline{\alpha'_i}(c_j)}{\sum_{j=1}^{n} \overline{\alpha'_i}(c_j)} - \frac{n+1}{n}$$

The authorship Gini coefficient is shown as a metric in the line graph under the main network graph.

In analogous way, we also computed the Gini coefficient for edits. Taking as a base all unique editors ever active in the article, this measure shows how equally all edits made in the article are distributed over the editors. I.e., if most or all of the edits were made just by a few editors, the Gini coefficient will converge to 1. If the edits were made at equal proportions by all editors, it will converge to 0. The edit Gini coefficient is included in the "Additional Metrics" view.

3. *In-/Outgoing Disagreement Gini Coefficient*: Based on the interaction network, these two metrics show for the previous 50 revisions how equally distributed the in- and outgoing disagreement edges are over the editors. They are computed analogous to the authorship Gini coefficient. I.e., if most or all of the negative actions target just a few editors – or (for outgoing) came from just a few editors – the Gini coefficient will converge to 1. If the disagreement actions equally came from (go to) all editors, it will converge to 0. Both coefficients are included in the "Additional Metrics" view.

4. *Number of Total Disagreement Actions*: Corresponds to the total number of negative actions, computed as the sum of all the values in $\overline{w'_i}$. The metric is shown in the line graph under the main network graph in a normalized form in an interval of 0 to 1 (normalized by its own minimum and maximum values over the article history).

5. *Disagreement Focus*: High values of this metric indicate that the negative actions performed in r_i by u are particularly targeting editor v; calculated as:

$$\text{focus}_i(u,v) = \frac{\overline{w'_i}(u,v)}{\sum_{z \in V_i} \overline{w'_i}(u,z)}$$

   I.e., the value would be 1 if all disagreement actions of u are targeting v's content and approach zero if u's actions are targeting a broad scope of content by several different editors. Disagreement focus is mainly used as a metric for Reciprocity (below), but is also displayed next to each action in the *edge context* to give background information to the end-user if a disagreement she inspects was targeted only at the shown antagonist in the dyad.

6. *Reciprocity*: We compute the reciprocity as an aggregated value of the mutual disagreement between editors within a window. This allows to detect whether the mutual conflicts between editors are recurrent over time. The reciprocity between editors u and v in r_i denoted as $\text{reciprocity}_i(u,v)$ is modeled as a weighted function (with weight $\phi \in [0.0; 1.0]$) of portion of mutual disagreed content between u and v in r_i and the average disagreement focus between u and v in the window:

$$\text{reciprocity}_i(u,v) = \phi \cdot \overbrace{\frac{\min(\{\overline{w'_i}(u,v), \overline{w'_i}(v,u)\})}{\max(\{\overline{w'_i}(u,v), \overline{w_i}(v,u)\})}}^{\text{Portion of mutual disagreement}} + (1-\phi) \cdot \underbrace{\text{avg}_{i-\omega \leqslant j \leqslant i}(\{\text{focus}_j(u,v), \text{focus}_j(v,u)\})}_{\text{Average disagreement focus in the window } \omega}$$

7. *Miscellaneous Metrics:* Some additional metrics less central to the use of the tool are included in the "Additional Metrics" view. *Length change* gives the percentage of article growth or shrinkage in byte length, compared to the previous revision. *Edit rapidness* measures the time in hours that has elapsed since the previous revision. *Template indicators* show for several important templates when they were present (value=1) or not present (value=0) in the article (or its talk page, if template does not appear on main page). *Average editors disagreed with (window=50)* shows how many editors' content was affected on average per revision by the actions in the last 50 revisions (i.e., how many "feet were stepped on", on average, in the last 50 revisions).

6.2.4 *Visualization Implementation*

The whoVIS interface (Figure 15) is implemented using the D3 JavaScript library.[9] After selecting an article, users visualize the editor network and a plot of the metrics from Section 6.2.3.

6.2.4.1 *Drawing the Editor Interaction Graph*

Given a revision r_i, the ratio of a node $u \in \overline{V'_i}$ in the graph is proportional to the percentage of words in r_i that were authored by u, i.e., $ratio_i(u) = \overline{\alpha'_i}(u) / \sum_{v \in V_i} \overline{\alpha'_i}(v)$. A minimum size in pixel is defined to make nodes visible even if the editor did not author any text. Every node is assigned a color; this allows for easily tracking nodes when their position changes over time. The node of the editor of r_i is highlighted with a dark-colored border. Nodes can be dragged to clear any potential overlap of graphical elements; hovering over a node will highlight all connected nodes.

The coordinates of nodes are computed with the approach by Brandes et al. [19]. This technique is the most fitting as nodes are unvaryingly placed more to the center of the graph if they are neutral to each other or only do small corrections, while high disagreement nodes get "pushed out" to the periphery and farther apart from each other the more they disagree. As most network visualization approaches place nodes *closer* the stronger their ties are, the method put forward by Brandes et al. conforms best with our use case. We also experimented with force-directed layouts with a gravity center and stronger repulsion of nodes with increasing edge weight between them, but found the results to not represent the actual disagreement structure of the article, as, e.g., nodes with very little disagreement would not be placed in the center, but the periphery of the graph, not allowing for a coherent interpretation of positions of the node in the graph with respect to the disagreement occurring in the article.

To determine the position of the nodes, the algorithm computes the eigenbasis of the matrix A, $A(u, v)$ being the disagreement between editors u and v, and vice versa. We build a matrix A_i for each r_i, with $A_i(u, v) = \sum_{j=i-\omega}^{i} (\overline{w'_j}(u, v) + \overline{w'_j}(v, u))$, where (u, v) and (v, u) are arcs in $\overline{E'_i}$. Then, the two most negative eigenvalues of A_i and their eigenvectors x_i and y_i are computed, resulting in the x- and y-coordinates of nodes in r_i, respectively. Editors in window ω that did not cause a disagreement edge (e.g., by just adding content) are not displayed in the main graph but on a separate "non-disagreeing editors" line, to keep track of recently active editors. An additional row lines up all nodes that represent those authors of any content in r_i which did not edit in the window ω at all.

9 http://d3js.org/

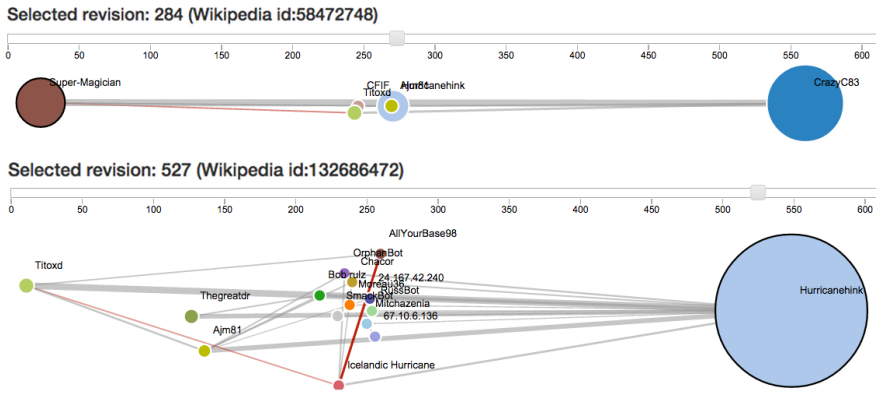
Edges $e' \in \overline{E'_i}$ between nodes u and v are drawn as a single grey line with width proportional to the value $A_i(u, v)$. The edge coloring is changed to red if the disagreement is mutual, i.e., values $\overline{w'_i}(u, v)$ and $\overline{w'_i}(v, u)$ are both > 0. The opacity of the red color starts with minimal value and increases according to the reciprocity metric.

An illustrative example can be given with the lower graph in Figure 17a. The green node to the very left (user Titoxd) is placed on the complete opposite side of the graph than the large blue node on the right (user Hurricanehink). This is firstly due to Hurricanehink having disagreed with many actions of Titoxd (cf. the high weight of the edge between them). Yet, Titoxd has not responded with disagreement actions against Hurricanehink, such that the value of *reciprocity* is close to 0 and the edge is not colored red. Secondly, the disagreement relations of Hurricanehink also include users that have shown no or little disagreement with Titoxd, in some cases placing them closer to Titoxd in the graph (e.g., Thgreatdr and Ajm81). This as well is taken into account when placing the nodes. The editors in the center of the graph exhibit weak or no disagreements towards either of the sides, therefore placing them in the central "neutral" area.

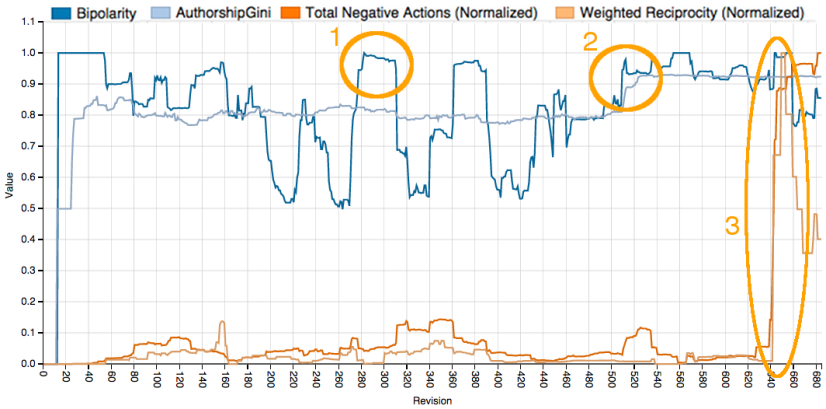
6.3 USE CASE 1 – WHOVIS: EXPLORING "TROPICAL STORM ALBERTO (2006)"

To explicate the operating principle of whoVIS, we first look at the article "Tropical Storm Alberto (2006)", given by Brandes et al. [19] as an example of a "featured" (i.e., high quality) article with low bipolarity, meaning that the aggregated network of the last revision exhibits multipolar disagreement structures instead of, e.g., two dominant disagreeing groups (cf. Figure 2 in [19]).[10] Exploring the history of interactions in whoVIS (Figure 17), we can first see in the line chart under the network graph that the *bipolarity* of the network can be very high at times, even when the aggregated graph for the last revision exhibits low bipolarity (as Brandes et al. suggest). This can be explained by the fact that over time, we see ephemeral bipolar disagreement "camps" of different editor combinations emerge and disappear, even when in the aggregate no such bipolarity manifests. For instance, the high bipolarity spikes at SrevID $\approx$ 280 (SrevID = "sequential revision id", assigned by whoVIS for this article) and SrevID $\approx$ 360 indicate disagreements between editors CrazyC83 and Super-Magician, while the spike at, e.g., SrevID $\approx$ 520 shows a disagreement between Hurricanehink vs. Thegreatdr and Titoxd (see Figure 17). The revision-wise exploration hence allows us to *deconstruct and better understand the aggregate*

10 Again, we would invite the reader to explore the online use case for better understanding: http://km.aifb.kit.edu/sites/whovis/graph.php?article=Tropical+Storm+Alberto+%282006%29



(a) Two examples of temporal bipolar graph structures (i.e., a point in the evolution of the article with two strongly disagreeing actor groups). They correspond to the highlighted mark-up #1 and #2 in the bipolarity metric line in Figure 17b below.



(b) Auxiliary metrics, marked at the jumps of the bipolarity measure (#1 and #2), authorship concentration (#2) and total negative actions/reciprocal disagreement (#3) help finding changed editing dynamics after "featured article" status is reached at SrevID = 584 (cf. featured article indicator from "additional metrics" in Figure 16).

Figure 17: **Examples of bipolar graph structures and auxiliary metrics for "Tropical Storm Alberto (2006)".** The auxiliary metrics graph is located under the network graph in the application, giving valuable context and aiding in pinpointing interesting developments over time (such as the orange example markings given here).

disagreement network in terms of its dynamic evolution by identifying temporal sub-structures of disagreement.

Going through the network graph chronologically, we can see that after the foundation of content by CrazyC83, a phase of indicated disagreement between several editors follows, – mainly dominated by three actors, CrazyC83, Super-Magician and HangingCurve – starting at about SrevID $\approx$ 80. We can see mutual disagreement mainly between CrazyC83 and Super-Magician, with several editors entering into a "disagreement triangle" with them before the dissent dies down towards SrevID $\approx$ 280 (Figure 15 shows an intermediate step). We observe this development mirrored in the average *reciprocity* and *total nega-*

tive actions charts. Inquiring into the (mostly highly reciprocal) disagreement edges via *edge context* in this phase reveals that the mutual editing of the actors, for the most part, concerns updates relating to recent developments of the titular "Storm Alberto" rather than a major clash of subjective viewpoints. This can be gleaned from the actions performed (largely date-related updates), the comments ("7 p.m. update") and the high edit rapidness (via the corresponding line chart in the "Additional Metrics" view). Yet, mixed into this "live reporting" spurt are genuine opinion clashes about how to write the article, e.g., the disagreement edge emerging between Derek.cashman and SuperMagician at SrevID $\approx$ 184, arguing whether to include links in section headers and citing the pertaining Wikipedia policies, as illustrated in Figure 15. Later, we see disputes about, e.g., the veracity of a report of Weatherfreak111 vs. Ajm81 and Hurricanehink (SrevID $\approx$ 470); and vandalism fighting, as surfacing at SrevID $\approx$ 648 between the IP 190.51.x.x and several registered users amidst other, content-related disagreements, which develop (quite literally and visually) orthogonal to the vandalism fight (cf. Crisco1492 vs. Juliancolton around the same time).

These examples showcase, with the help of the *edge context* and revision-wise exploration, that disagreement – modeled as text-deletes and reintroductions – can have highly different meanings in specific situations and in fact moves on a spectrum between mere "corrections", "profound disagreement" and "outright conflict", a distinction that can be easily overlooked when boiling down real human editor interactions into statistical graph representations. These different disagreement types can overlap, co-exist in parallel or appear at different points in time. *Edge context hence enables a crucial qualitative assessment of editor interactions by augmenting the information captured in the network graph.*

Another interesting observation in the article is the development towards "featured article" status, which it reaches at SrevID = 584 (cf. featured article indicator from "additional metrics" in Figure 16). The *Authorship Gini* curve shows a significant increase in authorship concentration before that event, at SrevID $\approx$ 510 (Figure 17), which, upon inspection of the word ownership of the top authors in the respective whoVIS tab, can be attributed to a large "writing sprint" by user Hurricanehink. This editor contributes a vast amount of content with some deletion/rewriting of authors Titoxd, Thegreatdr and Ajm81, but without being antagonized, and mostly adding new material of his own. After reaching featured article status, however, we see a burst of disagreement following SrevID $\approx$ 635, which is caused by many new editors doing small corrections but also partly due to vandals appearing, e.g., at SrevID $\approx$ 648 (IP 190.51.x.x). By tracking authorship concentration, the top editors' authored content and other metrics over time, as well as monitoring important Wiki-templates in the article,

whoVIS can inform a targeted inspection of editing interaction dynamics in the network by indicating significant phases in the article lifecycle through aggregate metrics over time.

6.4 USE CASE 2 – WHOVIS AND WHOCOLOR: EXPLORING "GAMERGATE CONTROVERSY"

An example of an article for which it is especially hard to attain the full picture of all parties involved in its creation, and which content they have been arguing about, is "Gamergate Controversy".[11] It is a highly controversial article on the English Wikipedia that has recently garnered even the attention of prominent media outlets, when eleven editors were sanctioned by Wikipedia's "Arbitration Committee" (ArbCom), mainly including 1-year topic-bans on the article and related topics, with most participants disciplined for "uncivilized and "battleground" behavior.[12] The sole involvement of the ArbCom, arguably Wikipedia's "Supreme Court" when it comes to quarrels between editors, plus the scope of the sanctions shows the gravity of the dispute this article has been subject to.[13]

The conflict surrounding the "Gamergate" phenomenon – going far beyond Wikipedia – had already become of significant societal relevance, as seen by the coverage in many major media outlets like The Washington Post, The New Yorker and the New York Times [78, 115, 166]. After "Gamergate" was discussed in social media and news media, the Wikipedia article "Gamergate Controversy" was created and specifically the related editing dispute and resulting editor bans came into focus of the news media as well, as indicated by the coverage in media outlets like The Guardian, The Washington Post and Slate [70, 40, 104].

The article deals mainly with alleged corruption in gaming journalism and the following reported sexism and harassment of (mostly female) individuals through the "anti-corruption" or "pro-Gamergate" faction. All in all, the article is a good example of how Wikipedia coverage of a topic can achieve high societal relevance and how it is therefore important for readers to understand the underlying motivations and agendas of editors fighting for control over the article *and* the effect of those disagreements on the eventual output in form of the content presented to the readers. Only this transparency enables a critical and informed consumption of the information therein. In a case like this, moreover, the number of editors involved and the amount of content changes they have been applying over time is so

11 https://en.wikipedia.org/wiki/Gamergate_controversy

12 Find the full ArbCom case here: https://en.wikipedia.org/wiki/Wikipedia:Arbitration/Requests/Case/GamerGate

13 Usually users in the "administrator" role police articles and implement sanctions. ArbCom is only appealed to if those sanctions fail repeatedly to restore order to the editors' behavior.

vast that the resulting patterns of editor interactions and content development are not viable to understand in their entirety for an average reader (or, e.g., a journalist) just by looking at the revision history provided in the MediaWiki software or reading the talk page. As journalist Amanda Marcotte puts it in an article for the magazine "Slate" in direct reference to the Wikipedia article: "[...] piecing together the story of what really happened amid the cacophony of finger-pointing and recrimination is nearly impossible [...]" [104].

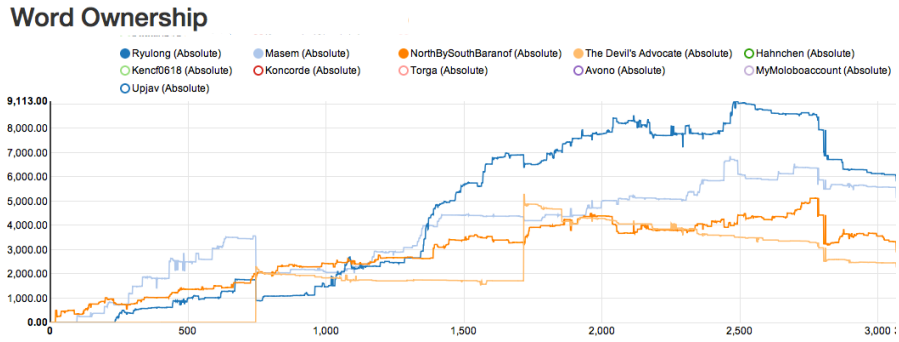
When reading about the dispute concerning "Gamergate controversy" in news media, on Wikipedia meta pages or other external sources, it is routinely portrayed as being carried out between two factions: "pro-gamergate" against "feminists"; or at least the situation is outlined as a clear-cut edit war.[14] The lecture of the ArbCom page on the case gives a vague impression of who the opponents in the dispute were: looking at the list of banned editors and consulting third party websites and the article talk page, one is prone to believe that the "pro-feminist" or "anti-gamergate" faction comprised 5 now-banned editors (sometimes even called "The Five Horsemen").[14] On the other side we seem to have another – although even less clearly defined – "pro-gamergate", "anti-feminist" group of 6-9 now-banned editors and several unnamed users. Yet, this vague picture is most likely a very strong simplification of the actual editing dynamics and actors in the article.

The specific *disputed contents* of the conflict are even harder to pinpoint, apparently ranging from wording disputes over using expressions like "misogyny", "harassment", etc., to arguing about whether certain factual claims are correct, to disagreement about whether certain statements, quotes or sources belong in the article at all. But which exact formulations in the article are changed from what to what, which ones are most disputed and between which editors these disagreements actually took place is very hard to discern only from the article itself, the associated talk page or third sources. To get an unbiased, first-hand picture ex-post, in the following we explore the Wikipedia article about the "Gamergate controversy" using the proposed tools whoVIS and whoCOLOR.

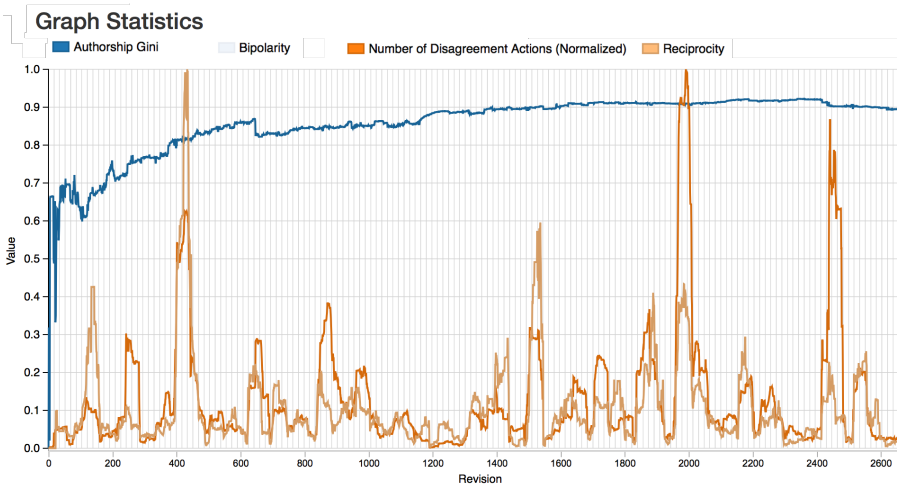
6.4.1 *Exploring with whoVIS*

The "Word Ownership" view of whoVIS provides first of all insights into who were the *most influential editors in terms of written words*. As shown in Figure 18a, four editors mainly coined the narrative of the article by having authored most of the text, especially in the hot phases of the debate (cf. disagreement spikes in Figure 18b): Ryulong, Masem, NorthBySouthBaranof and The Devil's Advocate. Three of

14 Cf., e.g., http://thinkprogress.org/culture/2015/03/06/3629086/wikipedia-gamergate-war/



(a) Line graphs of the amount of words owned in a specific revision, over all revisions, by the top four contributors of content to "Gamergate controversy": Ryulong (dark blue), NorthBySouthBaranof (dark orange), The Devil's Advocate (light orange), Masem (light blue). All the other editors' own word shares are notably lower than those depicted and therefore hidden here. These four editors have also been heavily involved in content disputes.

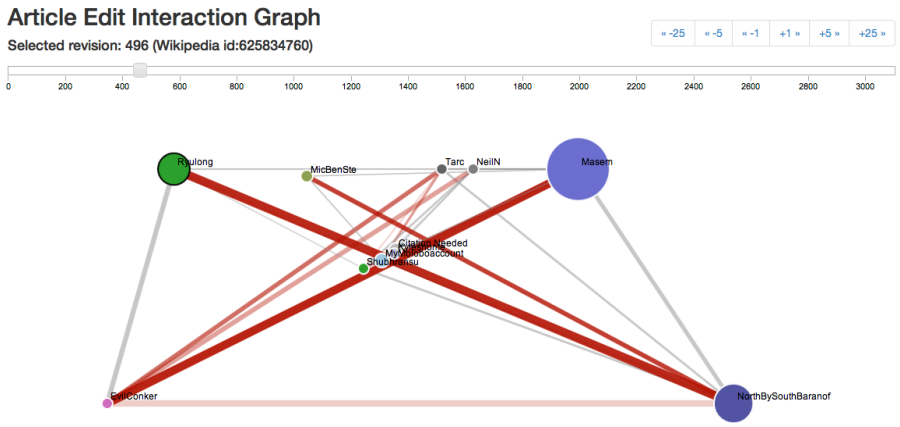


(b) The Gini coefficient for authorship, the disagreement actions between editors and the mutual disagreement (reciprocity), over the whole revision history. We can see a constant increase of the authorship Gini (=increase in concentration) and clear spikes of disagreement and reciprocity at several time points.

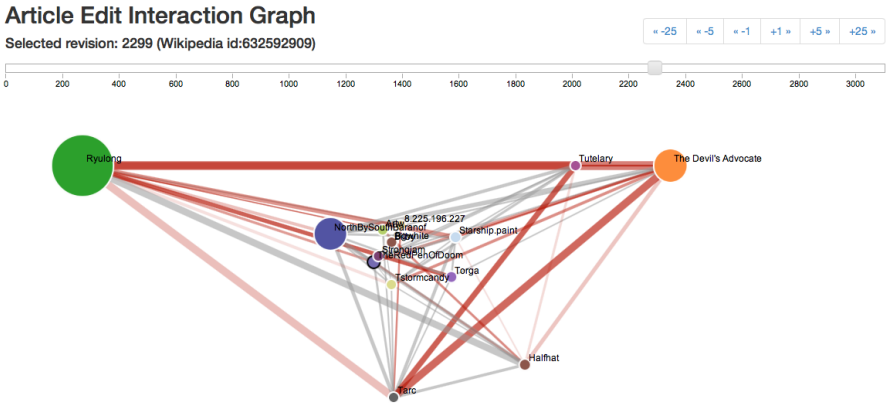
Figure 18: **Additional metrics provided by whoVIS for "Gamergate controversy"** regarding word ownership of the top editors (top) and conflict development and word ownership Gini coefficient (bottom).

those editors were later banned.[12] The "Additional Metrics" view of whoVIS (not depicted) also shows that the article was at least semi-protected almost its entire life and has seen frequent full-protection periods.

We can glean from whoVIS that these users wrote significantly more content on the page than other editors (not shown in Figure 18a) and did so increasingly over time. The Gini coefficients measuring (i) the concentration of ownership of words (shown as the blue line in Figure 18b) and (ii) how concentrated the distribution of edit



(a) **Disagreements between four main editors at SrevID 496.** Masem, NorthBySouthBaranof, Ryulong and EvilConker show major disagreements, alongside various minor disagreements. Editor EvilConker's content changes will be undone before the conflict ends.



(b) **A spike of disagreements towards the later third of the article (cf. Fig. 18b) involves** The Devil's Advocate in disagreement with one of the dominant editors, Ryulong . Also shown: disagreements between these two editors and Tarc and Tutelary, respectively, two editors that were also later banned.

Figure 19: **whoVIS disagreement network graphs** at different development stages of the English Wikipedia article "Gamergate controversy".

actions is over all active editors (not depicted, very similar trend) are in accordance with this *apparently increasing dominance of just a few editors*. Both curves are showing a steady incline of words owned and edits applied, that levels off in the last third of the article at a high value.

Just towards the end of the recorded revision history, we see unusual, distinct and synchronized drops in the amount of words owned for all four individuals (Figure 18a); upon inspection of the Wikipedia diffs corresponding to those revisions and associated comments, we learn that apparently, *the community has started a separate draft article* which was at these revisions merged into the original (hence removing or replacing much of the original content). The need for the page to be fully protected at times and Wikipedians to start a parallel arti-

cle draft *at all* is a strong indicator that the article climate up to this point was too unwelcoming for many editors. A reason for this could be the dominance of some authors in the article as well as the ongoing conflicts, depicted by the number of spikes in disagreed-on words and mutual disagreements in Figure 18b.

We therefore take a look at the disagreement network graph provided by whoVIS. The basic pattern visible starts at an early stage, from approximately SrevID 400 (of approximately 3100 as of writing).[15] *Three main actors seem to dominate the stage, often strongly disagreeing with other editors and each other*: Ryulong, Masem and NorthBySouthBaranof (cf. Figure 19a). Frequently, other editors are involved in these disagreements, but never for equally long periods as these main actors, who are almost constantly in disagreeing relations. One example is an intense mutual disagreement of user EvilConker with Masem (Figure 19a) at around SrevID 490 about how the introducing "Background" section should be written (indicated as well by the first major spike in Figure 18b). Eventually, the content by EvilConker is reverted back to the version before his intervention and the user ceases the conflicting interaction. Several of these short-lived, intense conflicts with various editors can be observed. Yet, some distinct antagonists emerge – although often only active temporarily – as e.g. users Torga and Diego Moya at around SrevID 510, or user Titanium Dragon at SrevID 640, to give just a few of many examples.

Although we will not dive into the finer details extractable with the whoVIS tool here, certain patterns in these interactions become salient. *(i) There are constant challenges of the content written by the main three authors. (ii) The main three authors challenge each other significantly as well*, especially Ryulong and NorthBySouthBaranof. (iii) *The challenges of content by less-dominant editors seem rarely to result in their own content to be accepted in the article*, while the main three actors increase the amount of owned words, as we have seen in Figure 18a.

The only major exception to this rule seems to be editor The Devil's Advocate, who, starting at around SrevID 740, begins rewriting and adding much content in the article, consequently entering into major disagreements with other editors about his/her changes. The Devil's Advocate is henceforth very active in disputes, as for example a major dispute starting at SrevID 2250 (cf. Figure 18b, second-to-last major spike). Nonetheless, the editor can gradually increase influence in form of originally written words in the article, as seen in Figure 18a.

As first conclusions of this very preliminary analysis, one could infer that *(i) while certain camp-like behavior exists* – e.g. many edit comments, especially in mutual disagreements, are trivializing edits and content by putting them into the dichotomous categories of either "pro-gamergate" or "anti-gamergate" – *the reality of the edit interactions is much more complex than one would think from reading about clear-cut*

15 The respective Wikipedia revision-IDs can be gleaned from the whoVIS tool.

pro-gamergate and anti-gamergate editor camps. Somewhat clearer is the insight that *(ii) it presumably early on became very hard and unwelcoming for "average" editors to sustainably contribute to the article without possessing a high degree of boldness and endurance*, such as editor The Devil's Advocate. Yet having only a group of such very bold (and possibly strongly opinionated) editors be the main actors and writers of an article might deter more moderate contributions, and moderating voices, and might have self-reinforced the climate that eventually caused the banning of many participants. As a conclusion for the article reader, the revisions of the article up to January 2015 should be read with the clear awareness about remnants of these – sometimes very intense and possible biased – editor disputes in the article. Even today a lot of content as a result from these disputes still persists, as we will see in the following section.

6.4.2 *Exploring with whoCOLOR*

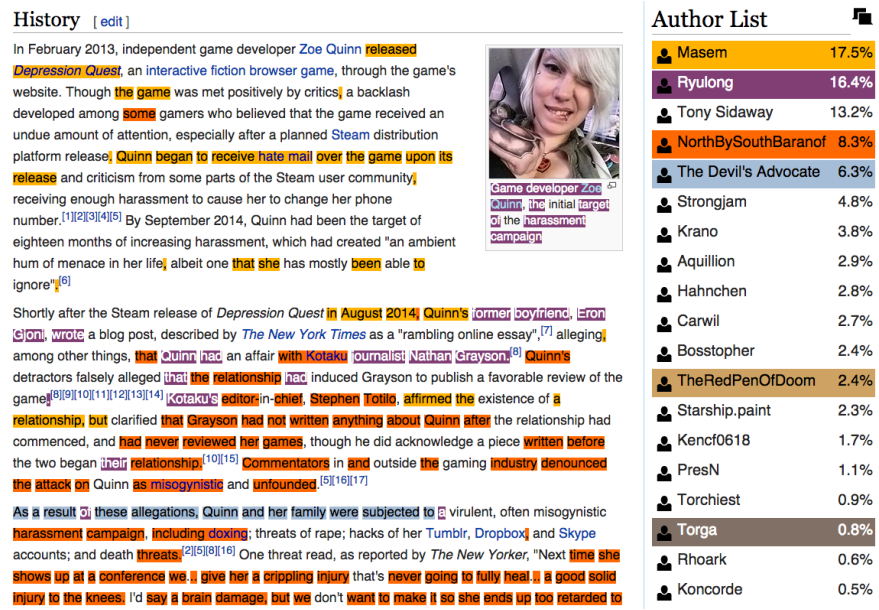
While with whoVIS we can explore the editor interactions and the contested content attached to them, this approach might be too abstract or complicated for a casual reader who is simply interested in which content is controversial or where it is coming from in the current article revision she/he is reading. With whoCOLOR, the reader can retrieve information about the author and provenance of a word easily in the browser as an intuitive annotation of the text while reading the article.

We see in the "Provenance View", as shown exemplarily in Figure 20, that the merging of the draft article, bans and the activity of new editors seem to have had a diversifying effect. While before the imposed topic bans and intervention of new editors, some sections were written almost entirely by the previously discussed dominant editors (Figure 20a), currently[16], *(i) the overall share of words written by these users has dropped dramatically* (although still high) and *(ii) sections like "History" contain now content written by many different users* (Figure 20b). While this is not necessarily a sign of quality in Wikipedia, it might be interpreted as such here, as it can be presumed that more points of view on the topic now found their way into the article.

Via the "Conflict View", the user can also explore which the most contested parts of the content have been so far. In Figure 21 we see an example of a paragraph that was heavily disputed in the article. It involves a statement, to paraphrase, about "what description of their movement Gamergate supporters have taken issues with". The inspection of this word sequence via the "Word History" feature further shows when the main dispute about those words happened, how long it lasted and who was involved (cf. Figure 22). We see that some previously mentioned main actors in the article were most involved

16 As of 30.03.2015

(a) Before the merging of the draft article and editor bans put into effect (22.11.2014) we see (i) a clear overall dominance by a small group of authors (cf. section 6.4.1, esp. Figure 18a) in written words and (ii) in some sections, like "History", an almost complete dominance of these authors' content.



(b) Today, much of the content of the previously dominating editors has been replaced (as of 30.03.2015), as well as content by other, banned editors. The content of the previously dominating editors has (i) decreased overall, as the comparison to Figure 18a shows. (ii) Some sections, like "History", are now much more diverse in terms of authorship (non-marked words in the above screenshot were, e.g., written by 22 distinct editors)

Figure 20: **Screenshots of the "History" section of "Gamergate controversy"** with whoCOLOR markup on the text. Except Masem and Torga, only editors that were later banned have been selected.

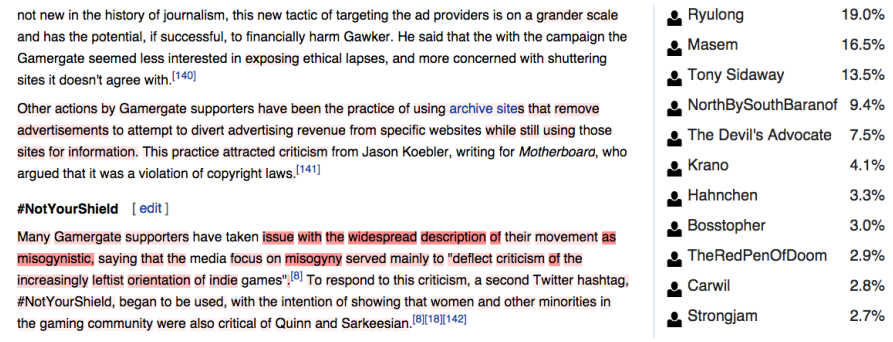


Figure 21: **A paragraph with heavily controversial content in a recent version of the article, as seen in the whoCOLOR Conflict View** (as of 09.02.2015). Shades of red are relative to each other, being more intense the more conflicted a word is.

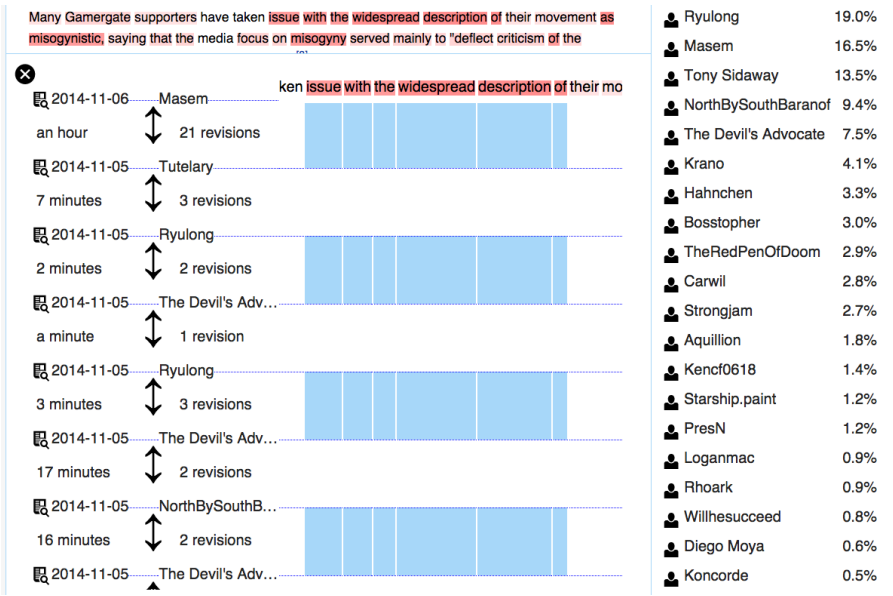


Figure 22: **The "Word History" feature of whoCOLOR is used to inspect the most controversial words from Figure 21.** Shown: a time period (descending from most recent to older) where the marked-up words were heavily contested.

in the dispute about whether or not to include those exact words in the article. This is also the case for many other disputes still remaining. By clicking on the arrow icons, one can navigate to the Wikipedia diffs of those edits to also see which words were proposed as an alternative.

Examining the whole article, the main controversial words still present are concentrated in individual paragraphs, mainly in the sections "History" and "Misogyny and antifeminism", and distributed over the whole document in sequences about harassment, threads and alleged statements of individuals. A reader using whoCOLOR is

alerted to those controversies and can hence interpret them accordingly.

6.5 RELATED WORK

The following visualization and exploration software tools exist that can potentially be used to shed light on the development of the article by an end-user.

- **Contropedia** [18] highlights most controversial elements in a Wikipedia article, and when and why there was dispute about them. Two main views are the entry point to inspect activity around a specific topic: the "Layer View" and the "Dashboard View". The "Layer View" provides an overlay for the original article, highlighting controversial elements, similar to the "Controversy View" of whoCOLOR. The "Dashboard View" presents a ranking of the most controversial elements together with a timeline, showing when each element underwent most dispute activity and the users involved. The tool however only tracks changes to, and disputes about, *internal Wiki links* in an article, not all words.

- **WikiDashboard** [139] visualizes edits over time by contributors to an article in a graph above the article content, but does not track the actual content changed or conflicts. The service is no longer officially available.

- **WikiTrust** [41] provides a browser add-on that superimposes an overlay on Wikipedia articles to display estimated trustworthiness of content, according to the longevity of introduced changes. It provides word provenance information (with low accuracy), but not interactions of editors. The API providing the trust mark-up was discontinued.

- **History Flow** [153] creates a layer-like visualization of the different parts of the article written by distinct editors, over the revision history. In this way it helps to follow content changes and moves over time, although the concrete content or disputes are not visible. Content attribution is performed on sentence, not word granularity, often leading to crave misinterpretations of authorship (e.g., if one word is changed, to whole sentence is reattributed).

- **Wikireplay**[17] (or "re Edit") is a community-built Web application that allows users to select a Wikipedia article and a starting revision. It then displays the look of the HTML view of the

17 http://cosmiclattes.github.io/wikireplay/player.html, by Wikipedia user Jeph Paul

article at the given time and sequentially visualizes all single additions and deletions that took place in a video-like animation.

6.6 ENVISIONED ADVANCEMENTS IN VISUALIZATION TOOLS

An optimal visual tool aimed at enabling article development transparency, and possibly social mechanisms behind it, should make several aspects available to the user in an intuitive way, without overburdening her with information: (i) the interaction patterns of the editors *with the content* (e.g. prominent editing or writing sprints, even if only by a single editor) and *with each other*, in terms of disagreement and how it is resolved (e.g. conflicts/edit wars and resulting controversial content); (ii) the development of the content over time (which content was there first, deleted/reintroduced, replaced by which other content, when was it disputed); (iii) the overall "climate" of the article, given by meaningful aggregate metrics about editor behavior; and lastly, (iv) tools to focus on the most important (e.g., controversial) (a) content, (b) users, (c) interactions and (d) time periods (or events), so that the end-user does not have to explore the complete potential space of edit information. This last point is crucial to enable the user to filter the data and *make sense* out of it, but also hard to achieve as this selection of what is "most important" is often ambiguous.

In Section 6.4 we have seen how the tools presented each allow distinct but complementary insights into the analyzed article. Some of their main features could hence be combined to benefit from this complementarity. Still, our prototypic tools are not yet offering all of the aspects listed above and hence encourage further development.

6.6.1 *Integrating Visualization Tools*

We envision an integrated platform that brings together the functionalities of the two tools presented above and additional features, potentially such of other platforms (e.g., those listed in Section 6.5).

Such a service would offer an "annotated content" view with several modes as a primary interface, with additional views to further explore the history and mechanisms behind selected content fragments or the whole article.

6.6.1.1 *Annotated Content View*

The most intuitive perspective for an average user of Wikipedia is arguably the view of the native article content, annotated with additional information. As such, it would be the main entry point and view of the envisioned platform.

- The *Provenance Mode* of this view would be akin to the whoCOLOR authorship mark-up (with article content and mark-up stored on the platform), alongside the whoCOLOR authorship list.
- The *Conflict Mode* (cf. whoCOLOR conflict mark-up) of the text would be provided as another view, highlighting all controversial words. Here, a more elaborate approach for calculating controversy scores would have to be tested, as well as controls for the users to define which level of controversy should be shown (only above a certain level, only recent disputes, etc.). It might also be beneficial to offer a listing option similar to that of Contropedia [17], showing the most conflicted words and sequences over time even if they are not present anymore.
- Further, a *Word Age Mode* would be highlighting the most recently added vs. the oldest sequences still present. This might especially be interesting in cases of dates and other facts that need to be updated frequently.
- Lastly, a more sophisticated approach for quality assessment of individual passages could be implemented in a fourth *Quality Mode*. Such an annotation could be based on similar metrics used in WikiTrust [3] or newer approaches, e.g., based on inter-editor "reviewing", as proposed by de La Robertie et al. [37] (all of which are easily calculable on top of WikiWho data). It would provide an amalgamate of several more complicated metrics, which makes it less intuitive to interpret than the formerly presented modes [101] – and also more ambiguous, as the selection of metrics for such a view has always to be subjective to some extent. Yet, this mode would capture the deeper underlying dynamics of editing by using not only relatively simplistic metrics of age, etc., but incorporate for instance both inter-editor relations plus longevity of content (cf. [37]).

6.6.1.2 *Historic Line Graphs*

For each mode in the Annotated Content View, a custom line graph with selected metrics of time can be provided. I.e., for instance in the Provenance Mode, a line graph with the Gini coefficient of word ownership can be integrated horizontally on top or bottom of the tool interface, including as well the top 10 authors over time (both metrics are already included in the whoVIS tool, cf. Section 6.2). For the Conflict Mode, the metrics displayed could be overall controversy score, mutual disagreements or similar (cf. again whoVIS' additional metrics). Integrating such metrics over time with the directly corresponding annotation modes makes them relatively easy to interpret as they can be directly set into relation to the annotations displayed

on the same page for the content. They can also provide an entry for navigation to older revisions of the page in the same mode (e.g. by selection older points in time in the line graph).

6.6.1.3 *Word History View*

Orthogonal to the Annotated Content View modes, the Word History View, as presented in Section 6.1, can be provided in the envisioned platform as well. An extension would be to switch from the inspection of a small number of selected words to a more extended view that displays the survival and replacements of text passages in the manner of the *HistoryFlow* (cf. Section 6.5) visualization, returning the user back to the Annotation View once closed. WikiWho provides the data needed to implement this view on word level granularity (in comparison to sentence-level granularity of the original implementation).

6.6.1.4 *Editor-Editor Network View*

As the last view of the integrated platform, an interactive editor-editor disagreement network based on the whoVIS visualization could be provided. It would be accessible via the Conflict Mode and serve to further investigate any disputes of editors over content. Additionally to the full article network, one option here would be to display only disputes regarding a specific selected section or paragraph in the editor graph. Another extension could be to further filter out disagreements that are apparently simply updates (e.g. time words or numbers only replaced once, or more sophisticated methods to classify edits into "corrections", "disagreements", etc.) and to let the users select which level of disagreements should be regarded (e.g., at least two mutual disagreement actions in 24 hours, etc.).

6.6.1.5 *Integrating Talk Page Data*

Finally, integrating data from Wikipedia Talk pages (the discussion space attached to each article) in a structured format would lend additional context to changes and conflicts. While post-response-structures of single threads can be extracted, the matching of discussion entries to specific edits or revert disputes is not trivial; additionally, talk page data is relatively sparse for most articles. Yet, for highly disputed and edited articles, this approach might be worth investigating further.

6.6.2 *Evaluation*

End-user studies – like the work by Lucassen and Schraagen on *Wikitrust* [101] – have shown that an average reader cannot easily interpret the information that sophisticated computational methods present as annotation to content, even if that information could be objectively

deemed "correct". Hence it is paramount (i) that our tools and the envisioned platform are tested and iteratively developed further with a pool of test users that employ the platform for actual reading or editing related task in Wikipedia, to gauge potential for improvement and barriers for usage; and in a subsequent step (ii) controlled experiments should be carried out to assess the actual usability and the effect on users. We hope to enable at least a better understanding of the social dimension and the decision-making process behind the concrete content parts of an article, as in such examples as the "Gamergate Controversy", so that engaged readers and editors can achieve a better "frame of interpretation" for what they are reading and better judge if it might need improvement or at least has to be taken with a grain of salt.

6.7 CONCLUSIONS

We have presented two interactive visualizations to showcase how the provenance and interaction data generated by WikiWho can be used to better understand the dynamics of the content building and negotiation process in an article.[18] We did so on two concrete article examples, one of which ("Gamergate controversy") clearly demonstrated the need for better software tools, especially for casual users, to cope with the complexity of understanding all the intricacies entailed by numerous actors trying to reach a consensus of how an article should be written. Such tools are currently not offered by the MediaWiki software or any extension or third party tool we are aware of.

Of course, a comprehensive evaluation of the utility of our interactive visualizations "in production" for readers and frequent editors (or even journalists) is still to be delivered and in our future plans. These visualizations serve, however, a second purpose, namely aiding researchers to (i) understand the data that we offer through WikiWho (and the API service we are currently expanding) and (ii) to gain a first, exploratory look at the data with interfaces that are tailored to it, which might be of interest for research applying qualitative as well as quantitative approaches.

Lastly, we outlined what an integrated platform for exploring editing dynamics could look like and how it could incorporate the advances made by research and the community up to now and leverage their potential in an integrated environment. We think that such an integrated visual data analytics service is crucial in the reduction of complexity when it comes to making transparent and understanding

18 We would also like to emphasize that the efficient calculation of the needed data by WikiWho reduces runtime and storage requirements for analogous services significantly, such that it becomes feasible to offer them on top of on-demand, up-to-date data.

collaboration data in Wikipedia; and it might serve the same purpose for other CWS.

7

CONCLUSIONS

Motivated by the need for more transparency in the collaborative writing process of digital documents, this thesis has contributed novel tools and perspectives for the research of social dynamics in CWS. Firstly, via an analysis of related work on social phenomena in Wikipedia, and employing the theoretic framework of social mechanisms, we have argued which systematically appearing cause-effect relationships might occur among groups of editors writing an article together and what kind of data is needed to study them. We thus found an answer to our first research question (cf. Section 1.1):

> **RQ1**: *In collaboratively writing and editing specific digital documents together, what systematically appearing social mechanisms can be identified that have the potential to influence the quality of the eventual document produced and what methods do we need to model and detect them?*

Although these suspected phenomena of territoriality, ownership behavior, social proof and eristic argumentation are surely not the only social mechanisms at work in Wikipedia or CWS in general, they are in our view the most salient when perusing the related empirical research literature on Wikipedia and matching them with applicable sociological theory. As such, they open a first avenue for theorizing about specific recurring micro- to meso-level patterns in such systems and precipitate a closer look at the digital traces of collaborative writing.

Additionally, the exploration of the information needed for modeling such potentially existing mechanisms revealed that available algorithmic methods were not suited to deliver the required high-accuracy modeling of provenance and editor interaction and were moreover not particularly efficient in providing such or similar data. This gave rise to our second research question:

> **RQ2**: *How do models and algorithmic methods have to be designed to help us extract (a) provenance and (b) interactions of editors with the content and with each other in a way that is (i) efficient and (ii) produces accurate data models of the reality of the socio-technical environment in digital collaborative writing platforms?*

This question was answered by developing the precise and efficient algorithmic approach WikiWho to extract provenance and word change data from revisioned text data by closely observing the reality

of collaborative writing (in Wikipedia). Further, we determined how to more realistically model the disagreement (revert) relationships between users to transform edit interaction with the text into a meaningful social network graph by formally defining different antagonistic and cooperative interactions that can take place when editors change existing content. We integrated this model in our WikiWho algorithm, thereby producing the most efficient and accurate algorithm to date to extract editor-editor interactions from revisioned text data on word level. The development of our algorithms was evaluated through studies with Wikipedia editors and crowdsourcing workers.

With the extracted data at hand, we further asked how it could enable more transparency of the collaborative writing process:

> **RQ3**: *Which novel end-user tools, especially visualizations, can be built on top of the extracted data that provide casual readers as well as editors and scientist with a low-threshold, intuitive way to explore and understand the collaboration and content provenance dynamics in Collaborative Writing Systems?*

As a response, we developed and presented two interactive visualization tools to explore provenance, word histories, disputes and editor interactions. We showed how those tools can benefit an end-user in better understanding the complex interactions of editors with each other and the text and how they might be a first step towards more elaborate visual analytics tools that even lay researchers (interested readers, editors, journalists) can use to better comprehend the processes of CWS and the evolution of single articles or parts of articles. Through building these visualization tools and the lessons learned in the process, we found answers to our third research question; however, as showcased by our future plans in Section 6.6, this line of investigation bears much more potential for future interfaces between complex social interaction data and end-users.

Lastly, by making relevant, quality-assessed data available in an efficient way, we have paved the way for studying two further, manifest research questions within large-scale CWS:

> **RQ4:** *Can we systematically detect the [previously described] behavioral dynamics through revision history data with statistical models?*

and

> **RQ5:** *Can we empirically link these patterns to specific quality changes in the content?*

Future research on these (and potentially many other) questions can be founded on the data that our algorithms are able to harvest at large scale, and that we made available under free licenses. Yet, we

aim to facilitate this development in research even further – as well as the construction of new visual analytics tools – by offering our data in more convenient and efficient ways. As such the API prototype for word provenance we have put in place is a first step in this direction.

7.1 FUTURE WORK

We have already outlined in the respective chapters what possible advancements could be made in regard to algorithms and visualizations. We have likewise set an agenda for future research with the proposed research questions **RQ4** and **RQ5**. Below, we further want to point out some remaining potential and challenges in terms of generalizability to other CWS and concrete advanced services we aim to build.

7.1.1 *Generalization to Other Collaborative Writing Systems*

In Section 4.5 we have already outlined why the core algorithm, WikiWho, is likely to be applicable to many different types of CWS, given slight adaptations in particular cases. As our model for editor interactions is built on top of WikiWho, the social network graph can likewise be constructed for other CWS and their documents. And with the underlying data being identical, also our visualization tools and API services could be provided for other platforms. The exact tuning of these approaches has, however, to be in line with the nature of the system to be studied, technology-wise as well regarding the user population and its habits, respecting all the CW settings (cf. Section 2.1.1). While this signifies on the one hand adapting the WikiWho algorithm to correctly detect the origin of changes, it also means, for instance, to consider if the social network between editors should be constructed in exactly the same way. While disagreements between editors in the form of deletions and undos of tokens might be very common and make sense to calculate in the Wikipedia setting, disagreeing changes to specific parts of software code might have to be interpreted and weighted very differently, for instance regarding their specific role and dependencies to other code components. Similarly, for visualizations, it might be much more plausible to highlight certain interactions and omit others, e.g., between certain actors in the network (such as managers or other roles in an organizational Wiki for example) or regarding certain content (such as specific data fields); and while a conflict view makes sense for a tool like whoCOLOR in Wikipedia, it might not bring much additional insight in a business-internal Wiki that has a clear set of rules for writing and editing content, avoiding editing clashes (such a system might, e.g., rather need a view for outdated content). The social mechanisms one can expect to find in CWS different from Wikipedia of course also highly depend on the CW settings. While territoriality regarding content that was

written under high expenditure by an editor (and/or self-perceived expertise) may well be expected in other systems, tighter rules system or managerial supervision might effectively stifle such conduct as well as conflict. Distrust against unknown editors can equally occur in large open platforms but is unlikely to be prevalent in small groups or platforms with systematic user registration and informative profile pages. Social proof leading to less needed changes can on the other hand be a factor in many scenarios, particularly if the right motivation structure is lacking for newcomers to be bold in their editing.

7.1.2 *Services*

The visualization tools we presented in Chapter 6 (and their planned extension/integration) are one part of the services we aim to offer based on the results of this thesis. Another one is the provision of datasets for provenance, token-level change histories and controversy scores as well as the editor interactions for all articles in several language versions of Wikipedia, starting with the English Wikipedia. The API for querying WikiWho-preprocessed provenance data for live article revisions is already running as a prototype service and is currently being extended to other language editions and with further dimensions. First additional outputs will likely be change history data, conflict scores and editor interaction networks.

The target audiences of these services are non-scientific as well as scientific end-users; the latter belonging not only, but particularly to such disciplines that face a higher barrier than Computer Science in accessing and processing the large amounts of textual data in Wikipedia or other popular CWS – such as the Social Sciences. Especially an option to choose from different generation types for networks (time windows ω, definition of (dis)agreement edges, e.g. focussing on edits classified as full reverts and mutual reverts) to produce downloadable and immediately usable datasets for statistical analysis seems to be a promising resource for empirical researchers interested in social network phenomena.

7.2 CLOSING REMARKS

With the new methods and services made possible trough the work carried out in this thesis, we hope to have paved the way for much more intricate analyses of the social dynamics that make large online CWS tick, especially Wikipedia. If the past success of these platforms is any indicator of what is to come in terms of growth, complexity and societal impact, new research tools – as well as the formulation of new research questions based in social theory that these tools help to answer – will be pivotal in understanding and improving CWS.

BIBLIOGRAPHY

[1] M. Abolafia and M. Kilduff. Enacting market crisis: The social construction of a speculative bubble. *Administrative Science Quarterly*, pages 177–193, 1988.

[2] T. Adler and L. Alfaro. A content-driven reputation system for the Wikipedia. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 261–270, 2007.

[3] T. Adler, K. Chatterjee, L. Alfaro, M. Faella, I. Pye, and V. Raman. Assigning trust to Wikipedia content. In *International Symposium on Wikis*, 2008.

[4] T. Adler, L. de Alfaro, I. Pye, and V. Raman. Measuring author contributions to the Wikipedia. In *Proceedings of the 4th International Symposium on Wikis*, WikiSym '08, pages 15:1–15:10, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-128-6.

[5] O. Arazy, O. Nov, R. Patterson, and L. Yeo. Information quality in wikipedia: The effects of group composition and task conflict. *Journal of Management Information Systems*, 27(4):71–98, 2011.

[6] J. Atwood. Mixing oil and water: Authorship in a wiki world. *CODING HORROR blog*, 2009. URL http://blog.codinghorror.com/mixing%2Doil%2Dand%2Dwater%2Dauthorship%2Din%2Da%2Dwiki%2Dworld/.

[7] P. Ayers, C. Matthews, and B. Yates. *How Wikipedia works: And how you can be a part of it*. No Starch Press, 2008.

[8] R. Baggen, J. P. Correia, K. Schill, and J. Visser. Standardized code quality benchmarking for improving software maintainability. *Software Quality Journal*, 20(2):287–307, 2012.

[9] P. Bak, M. Paczuski, and M. Shubik. Price variations in a stock market with many agents. *Physica A: Statistical Mechanics and its Applications*, 246(3):430–453, 1997.

[10] A. V. Banerjee. A simple model of herd behavior. *The Quarterly Journal of Economics*, pages 797–817, 1992.

[11] E. E. Beck. A survey of experiences of collaborative writing. In *Computer supported collaborative writing*, pages 87–112. Springer, 1993.

[12] J. Beck. Doctors' #1 source for healthcare information: Wikipedia. *The Atlantic*, 2014. URL http://www.theatlantic.

com/health/archive/2014/03/doctors%2D1%2Dsource%2Dfor%2Dhealthcare%2Dinformation%2Dwikipedia/284206/.

[13] I. Beschastnikh, T. Kriplean, and D. W. McDonald. Wikipedian Self-Governance in action: Motivating the policy lens. In E. Adar, M. Hurst, T. Finin, N. S. Glance, N. Nicolov, and B. L. Tseng, editors, *ICWSM*. The AAAI Press, 2008.

[14] S. Bikhchandani, D. Hirshleifer, and I. Welch. A theory of fads, fashion, custom, and cultural change as informational cascades. *Journal of political Economy*, pages 992–1026, 1992.

[15] R. P. Biuk-Aghai. Visualizing co-authorship networks in online wikipedia. In *Communications and Information Technologies, 2006. ISCIT'06. International Symposium on*, pages 737–742. IEEE, 2006.

[16] P. Bogdanov, N. D. Larusso, and A. Singh. Towards community discovery in signed collaborative interaction networks. In *Data Mining Workshops (ICDMW), 2010 IEEE International Conference on*, pages 288–295. IEEE, 2010.

[17] E. Borra, E. Weltevrede, P. Ciuccarelli, A. Kaltenbrunner, D. Laniado, G. Magni, M. Mauri, R. Rogers, and T. Venturini. Contropedia - the analysis and visualization of controversies in wikipedia articles. In *Proceedings of The International Symposium on Open Collaboration*, page 34. ACM, 2014.

[18] E. Borra, E. Weltevrede, P. Ciuccarelli, A. Kaltenbrunner, D. Laniado, G. Magni, M. Mauri, R. Rogers, and T. Venturini. Societal Controversies in Wikipedia Articles. In *Proc. CHI*, 2015.

[19] U. Brandes, P. Kenis, J. Lerner, and D. van Raaij. Network analysis of collaboration structure in wikipedia. In *Proceedings of the 18th International Conference on World Wide Web*, 2009.

[20] J. Broughton. *Wikipedia: the missing manual*. " O'Reilly Media, Inc.", 2008.

[21] R. Burns and D. Long. A linear time, constant space differencing algorithm. In *Performance, Computing, and Communications Conference, 1997. IPCCC 1997., IEEE International*, pages 429–436. IEEE, 1997.

[22] B. Butler, E. Joyce, and J. Pike. Don't look now, but we've created a bureaucracy: the nature and roles of policies and rules in wikipedia. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1101–1110, New York, NY, USA, 2008. ACM.

[23] S. Bykau, F. Korn, D. Srivastava, and Y. Velegrakis. Fine-grained controversy detection in wikipedia. In *Proceedings of The International Conference on Data Engineering*. IEEE, 2015.

[24] A. Capocci, V. D. Servedio, F. Colaiori, L. S. Buriol, D. Donato, S. Leonardi, and G. Caldarelli. Preferential attachment in the growth of social networks: The internet encyclopedia wikipedia. *Physical Review E*, 74(3):036116, 2006.

[25] L. Carozza and F. Macagno. The evaluation of emotional arguments: a test run. *Philosophical Research Online*, 2011.

[26] P. Chekroun and M. Brauer. The bystander effect and social control behavior: The effect of the presence of others on people's reactions to norm violations. *European Journal of Social Psychology*, 32(6):853–867, 2002.

[27] S.-C. Chin, W. N. Street, P. Srinivasan, and D. Eichmann. Detecting wikipedia vandalism with active learning and statistical language models. In *Proceedings of the 4th workshop on Information credibility*, pages 3–10. ACM, 2010.

[28] N. Christakis and J. Fowler. The spread of obesity in a large social network over 32 years. *New England journal of medicine*, 357(4):370–379, 2007.

[29] R. Cialdini. *Influence science and practice.* Gardners Books Ltd, [S.l.], 2007. ISBN 9780321188953.

[30] J. Coleman, E. Katz, and H. Menzel. The diffusion of an innovation among physicians. *Sociometry*, pages 253–270, 1957.

[31] "Conservatism". *Merriam-Webster Online Dictionary*. Merriam-Webster Incorporated, 2015. http://www.merriam-webster.com/dictionary/conservatism, Accessed 15.04.15.

[32] T. H. Cormen, C. Stein, R. L. Rivest, and C. E. Leiserson. *Introduction to Algorithms*. McGraw-Hill Higher Education, 2nd edition, 2001. ISBN 0070131511.

[33] L. A. Coser. *The functions of social conflict*, volume 9. Routledge, 1956.

[34] K. Crowston and J. Howison. The social structure of free and open source software development. *First Monday*, 10(2), 2005.

[35] B. Dave and L. Koskela. Collaborative knowledge management - a construction case study. *Automation in Construction*, 18(7): 894–902, 2009.

[36] L. de Alfaro and M. Shavlovsky. Attributing authorship of revisioned content. In *Proceedings of the 22nd international conference on World Wide Web*, WWW '13, pages 343–354, 2013.

[37] B. de La Robertie, Y. Pitarch, and O. Teste. Measuring article quality in wikipedia using the collaboration network.

[38] P. Denning, J. Horning, D. Parnas, and L. Weinstein. Wikipedia risks. *Communications of the ACM*, 48(12):152–152, 2005.

[39] M. Deutsch. Conflicts: Productive and destructive*. *Journal of social issues*, 25(1):7–42, 1969.

[40] C. Dewey. Gamergate, wikipedia and the limits of 'human knowledge'. *Washington Post Online*, 2015. URL http://www.washingtonpost.com/news/the-intersect/wp/2015/01/29/gamergate-wikipedia-and-the-limits-of-human-knowledge/.

[41] B. T. dler, L. de Alfaro, and I. Pye. Detecting wikipedia vandalism using wikitrust. *Notebook Papers of CLEF*, 1:22–23, 2010.

[42] M. D. Ekstrand and J. T. Riedl. rv you're dumb: Identifying discarded work in wiki article history. In *The Fifth International Symposium on Wiki's and Open Collaboration*, Orlando, FL, October 2009.

[43] J. K. Esser. Alive and well after 25 years: A review of groupthink research. *Organizational behavior and human decision processes*, 73 (2):116–141, 1998.

[44] S. Eubank, H. Guclu, V. Kumar, M. Marathe, A. Srinivasan, Z. Toroczkai, and N. Wang. Modelling disease outbreaks in realistic urban social networks. *Nature*, 429(6988):180–184, 2004.

[45] G. Farner. *Literary Fiction: The Ways We Read Narrative Literature*. Bloomsbury Publishing, 2014.

[46] F. Flöck and A. Rodchenko. Whose article is it anyway?– Detecting authorship distribution in Wikipedia articles over time with WIKIGINI. In *Online proceedings of the Wikipedia Academy 2012*. Wikimedia, 2012.

[47] F. Flöck. What web collaboration research can learn from social sciences regarding impairments of collective intelligence and influence of social platforms. *'Harnessing the Power of Social Theory for Web Science' workshop at the ACM WebScience Conference 2013*, 2013.

[48] F. Flöck and M. Acosta. Wikiwho: Precise and efficient attribution of authorship of revisioned content. In *Proceedings of the 23rd International Conference on World Wide Web*, WWW '14, pages 843–854, New York, NY, USA, 2014. ACM.

[49] F. Flöck and M. Acosta. whovis: Visualizing editor interactions and dynamics in collaborative writing over time. In *Proceedings of the 24th International Conference on World Wide Web Companion, WWW 2015, Florence, Italy - Companion Volume*, pages 191–194, 2015.

[50] F. Flöck, D. Vrandečić, and E. Simperl. Towards a diversity-minded Wikipedia. In *Proceedings of the ACM 3rd International Conference on Web Science 2011*, 06 2011.

[51] F. Flöck, D. Vrandečić, and E. Simperl. Revisiting reverts: Accurate revert detection in wikipedia. In *Proceedings of the 23rd ACM Conference on Hypertext and Social Media*, HT '12, pages 3–12, New York, NY, USA, 2012. ACM.

[52] F. Flöck, D. Laniado, F. Stadthaus, and M. Acosta. Towards better visual tools for exploring wikipedia article development–the use case of "gamergate controversy". In *Wikipedia, a Social Pedia - Workshop at the Ninth International AAAI Conference on Web and Social Media*, 2015.

[53] F. Floeck, J. Putzke, S. Steinfels, K. Fischbach, and D. Schoder. Imitation and quality of tags in social bookmarking systems – collective intelligence leading to folksonomies. In *On Collective Intelligence*, volume 76 of *Advances in Intelligent and Soft Computing*, pages 75–91. Springer Berlin Heidelberg, 2011.

[54] A. Forte and A. Bruckman. Why do people write for Wikipedia? Incentives to contribute to open-content publishing. In *Workshop of Sustaining Community: The Role and Design of Incentive Mechanisms in Online Systems. Sanibel Island, FL*, pages 6–9, 2005.

[55] A. Forte and A. Bruckman. Scaling consensus: Increasing decentralization in wikipedia governance. In *HICSS*, page 157. IEEE Computer Society, 2008.

[56] D. Gambetta. 5. concatenations of mechanisms. *Social mechanisms: An analytical approach to social theory*, page 102, 1998.

[57] R. S. Geiger and D. Ribes. The work of sustaining order in wikipedia: the banning of a vandal. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, pages 117–126. ACM, 2010.

[58] J. Giles. Internet encyclopaedias go head to head. *Nature*, 438 (7070):900–901, 2005.

[59] R. Glott, P. Schmidt, and R. Ghosh. Wikipedia survey — overview of results. Technical report, UNU- MERIT, United Nations University, Maastricht, Netherlands, March 2010.

[60] E. Goldman. Wikipedia's labor squeeze and its consequences. *J. on Telecomm. & High Tech. L.*, 8:157, 2010.

[61] J. Gottlieb and C. S. Carver. Anticipation of future interaction and the bystander effect. *Journal of Experimental Social Psychology*, 16(3):253–260, 1980.

[62] M. Granovetter and R. Soong. Threshold models of diffusion and collective behavior. *Journal of Mathematical sociology*, 9(3): 165–179, 1983.

[63] A. Halfaker, A. Kittur, R. Kraut, and J. Riedl. A jury of your peers: quality, experience and ownership in Wikipedia. In D. Riehle and A. Bruckman, editors, *Int. Sym. Wikis*. ACM, 2009.

[64] A. Halfaker, R. S. Geiger, J. T. Morgan, and J. Riedl. The rise and decline of an open collaboration system: How wikipedia's reaction to popularity is causing its decline. *American Behavioral Scientist*, 2012.

[65] K. A. Hansen, M. Neuzil, and J. Ward. Newsroom topic teams: Journalists' assessments of effects on news routines and newspaper quality. *Journalism & Mass Communication Quarterly*, 75 (4):803–821, 1998.

[66] D. Hasan Dalip, M. André Gonçalves, M. Cristo, and P. Calado. Automatic quality assessment of content created collaboratively by web communities: a case study of wikipedia. In *Proceedings of the 9th ACM/IEEE-CS joint conference on Digital libraries*, pages 295–304. ACM, 2009.

[67] A. Head and M. Eisenberg. How college students use the web to conduct everyday life research. *First Monday*, 16(4), 2011. ISSN 13960466. URL http://firstmonday.org/ojs/index.php/fm/article/view/3484.

[68] P. Hedström. Rational imitation. *Social mechanisms: An analytical approach to social theory*, pages 306–327, 1998.

[69] P. Hedström and R. Swedberg. Introduction. *Social mechanisms: An analytical approach to social theory*, 1998.

[70] A. Hern. Wikipedia votes to ban some editors from gender-related articles. *The Guardian*, 2015. URL http://www.theguardian.com/technology/2015/jan/23/wikipedia%2Dbans%2Deditors%2Dfrom%2Dgender%2Drelated%2Darticles%2Damid%2Dgamergate%2Dcontroversy.

[71] B. M. Hill. Cooperation in parallel: a tool for supporting collaborative writing in diverged documents. Master's thesis, Massachusetts Institute of Technology, 2007.

[72] B. Hilligoss and S. Y. Rieh. Developing a unifying framework of credibility assessment: Construct, heuristics, and interaction in context. *Information Processing & Management*, 44(4):1467–1484, 2008.

[73] T. Iba, K. Nemoto, B. Peters, and P. A. Gloor. Analyzing the creative editing behavior of wikipedia editors: Through dynamic social network analysis. *Procedia-Social and Behavioral Sciences*, 2 (4):6441–6456, 2010.

[74] I. L. Janis. *Groupthink: Psychological studies of policy decisions and fiascoes*. Houghton Mifflin Boston, 1982.

[75] R. Jesus, M. Schwartz, and S. Lehmann. Bipartite networks of wikipedia's articles and authors: a meso-level approach. In *Proceedings of the 5th international symposium on Wikis and open collaboration*, page 5. ACM, 2009.

[76] J. Jones. Patterns of revision in online writing a study of Wikipedia's featured articles. *Written Communication*, 25(2):262–289, 2008.

[77] O. Kamm. Wisdom? more like dumbness of the crowd. *The Times*, August 2007. URL http://www.timesonline.co.uk/tol/comment/columnists/guest_contributors/article2267665.ece),453-462,April2007.

[78] S. Kaplan. With gamergate, the video-game industry's growing pains go viral. *The Washington Post*, 2014. URL https://www.washingtonpost.com/news/morning-mix/wp/2014/09/12/with%2Dgamergate%2Dthe%2Dvideo%2Dgame%2Dindustrys%2Dgrowing%2Dpains%2Dgo%2Dviral/.

[79] B. Keegan, D. Gergle, and N. Contractor. Staying in the loop: structure and dynamics of wikipedia's breaking news collaborations. In *Proceedings of the Eighth Annual International Symposium on Wikis and Open Collaboration*, page 1. ACM, 2012.

[80] K.-S. Kim, E. Yoo-Lee, and S.-C. Joanna Sin. Social media as information source: Undergraduates' use and evaluation behavior. *Proceedings of the American Society for Information Science and Technology*, 48(1):1–3, 2011.

[81] A. Kittur and R. E. Kraut. Beyond Wikipedia: Coordination and conflict in online production groups. In *Proceedings of the 2010 ACM conference on Computer supported cooperative work*, CSCW '10, pages 215–224, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-795-0.

[82] A. Kittur, E. H. Chi, B. A. Pendleton, B. Suh, and T. Mytkowicz. Power of the few vs. wisdom of the crowd: Wikipedia and the rise of the bourgeoisie. In *Proceedings of the 25th Annual ACM Conference on Human Factors in Computing Systems (CHI 2007)*, 2007.

[83] A. Kittur, B. Suh, B. A. Pendleton, and E. H. Chi. He says, she says: Conflict and coordination in Wikipedia. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '07, pages 453–462, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-593-9.

[84] A. Kittur, B. A. Pendleton, and R. E. Kraut. Herding the cats: the influence of groups in coordinating peer production. In D. Riehle and A. Bruckman, editors, *Int. Symposium on Wikis '09*. ACM, 2009. ISBN 978-1-60558-730-1.

[85] A. Kittur and R. E. Kraut. Harnessing the wisdom of crowds in wikipedia: quality through coordination. In *CSCW '08: Proceedings of the ACM 2008 conference on Computer supported cooperative work*, pages 37–46, New York, NY, USA, 2008. ACM. ISBN 978-1-60558-007-4.

[86] A. Kittur, B. Suh, and E. H. Chi. Can you ever trust a wiki?: impacting perceived trustworthiness in wikipedia. In *Proceedings of the 2008 ACM conference on Computer supported cooperative work*, pages 477–480. ACM, 2008.

[87] A. Kittur, E. H. Chi, and B. Suh. What's in wikipedia?: mapping topics and conflict using socially annotated category structure. In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1509–1512. ACM, 2009.

[88] T. Kuran. The tenacious past: Theories of personal and collective conservatism. *Journal of Economic Behavior & Organization*, 10(2):143–171, 1988.

[89] S. K. Lam, A. Uduwage, Z. Dong, S. Sen, D. R. Musicant, L. Terveen, and J. Riedl. Wp:clubhouse? an exploration of wikipedia's gender imbalance. In *WikiSym 2011*, Mountain View, CA, 10/2011 2011. ACM, ACM.

[90] R. N. Langlois and G. Garzarelli. Of hackers and hairdressers: Modularity and the organizational economics of open-source collaboration. *Industry and Innovation*, 15(2):125–143, 2008.

[91] D. Laniado and R. Tasso. Co-authorship 2.0: Patterns of collaboration in wikipedia. In *Proceedings of the 22nd ACM conference on Hypertext and hypermedia*, pages 201–210. ACM, 2011.

[92] P. F. Lazarsfeld, B. Berelson, and H. Gaudet. The people's choice: how the voter makes up his mind in a presidential campaign. 1968.

[93] D. Lemire, S. Downes, and S. Paquet. Diversity in open social networks. Technical report, University of Quebec, Montreal, CA, October 2008.

[94] K. Leung. Some determinants of conflict avoidance. *Journal of Cross-Cultural Psychology*, 19(1):125–136, 1988.

[95] M. Linares-Vasquez, K. Hossen, H. Dang, H. Kagdi, M. Gethers, and D. Poshyvanyk. Triaging incoming change requests: Bug or commit history, or code authorship? In *2012 28th IEEE International Conference on Software Maintenance (ICSM)*, pages 451–460, 2012.

[96] J. Liu and S. Ram. Who does what: Collaboration patterns in the wikipedia and their impact on article quality. *ACM Trans. Manage. Inf. Syst.*, 2(2):11:1–11:23, July 2011. ISSN 2158-656X. URL http://doi.acm.org/10.1145/1985347.1985352.

[97] Z. Liu. Perceptions of credibility of scholarly information on the web. *Information Processing & Management*, 40(6):1027–1038, 2004.

[98] D. Loshin. *Enterprise knowledge management: The data quality approach*. Morgan Kaufmann, 2001.

[99] P. B. Lowry, A. Curtis, and M. R. Lowry. Building a taxonomy and nomenclature of collaborative writing to improve interdisciplinary research and practice. *Journal of Business Communication*, 41(1):66–99, 2004.

[100] T. Lucassen and J. M. Schraagen. Trust in wikipedia: How users trust information from an unknown source. In *Proceedings of the 4th Workshop on Information Credibility*, WICOW '10, pages 19–26, New York, NY, USA, 2010. ACM. ISBN 978-1-60558-940-4.

[101] T. Lucassen and J. M. Schraagen. Evaluating wikitrust: A trust support tool for wikipedia. *First Monday*, 16(5), 2011.

[102] T. Lucassen and J. M. Schraagen. Propensity to trust and the influence of source and medium cues in credibility evaluation. *Journal of information science*, 38(6):566–577, 2012.

[103] S. Maniu, B. Cautis, and T. Abdessalem. Building a signed network from interactions in Wikipedia. In *Databases and Social Networks*, pages 19–24, 2011.

[104] A. Marcotte. On wikipedia, gamergate refuses to die. *Slate*, 2015. URL http://www.slate.com/blogs/xx_factor/2015/03/06/the_gamergate_wars_over_wikipedia_show_that_wikipedia_s_neutrality_measure.html.

[105] K. Marks. Power laws and blogs, 2003. URL http://homepage.mac.com/kevinmarks/powerlaws.html.

[106] A. Mawson. Understanding mass panic and other collective responses to threat and disaster. *Psychiatry: Interpersonal and biological processes*, 68(2):95–113, 2005.

[107] A. Mawson. *Mass panic and social attachment: the dynamics of human behavior*. Ashgate Publishing Company, 2007.

[108] R. K. Merton. The self-fulfilling prophecy. *The Antioch Review*, pages 193–210, 1948.

[109] R. K. Merton. *On sociological theories of the middle range*. na, 1949.

[110] S. Milgram and E. Van den Haag. Obedience to authority, 1978.

[111] J. Mossong, N. Hens, M. Jit, P. Beutels, K. Auranen, R. Mikolajczyk, M. Massari, S. Salmaso, G. Tomba, J. Wallinga, et al. Social contacts and mixing patterns relevant to the spread of infectious diseases. *PLoS medicine*, 5(3):e74, 2008.

[112] M. Newman. Spread of epidemic disease on networks. *Physical Review E*, 66(1):016128, 2002.

[113] E. Noelle-Neumann. The spiral of silence: a theory of public opinion. In *Journal of Communication*, volume 24, pages 43–51, 1974.

[114] E. M. Nussbaum. Argumentation, dialogue theory, and probability modeling: Alternative frameworks for argumentation research in education. *Educational Psychologist*, 46(2):84–106, 2011.

[115] S. Parkin. Zoe quinn's depression quest. *New Yorker*, 2014. URL http://www.newyorker.com/tech/elements/zoe-quinns-depression-quest.

[116] D. Perrin. Shaping the multimedia mindset: Collaborative writing in journalism education. *Perspectives on Writing*, page 389, 2012.

[117] C. M. Pilato, B. Collins-Sussman, and B. W. Fitzpatrick. *Version control with subversion*. O'Reilly Media, Inc., 2009.

[118] D. Porter and V. Smith. Stock market bubbles in the laboratory. *Applied Mathematical Finance*, 1(2):111–128, 1994.

[119] M. Potthast, B. Stein, and R. Gerling. Automatic vandalism detection in wikipedia. In *Advances in Information Retrieval*, pages 663–668. Springer, 2008.

[120] C. R. Prause. Maintaining fine-grained code metadata regardless of moving, copying and merging. In *Source Code Analysis and Manipulation, 2009. SCAM '09. Ninth IEEE International Working Conference on*, pages 109–118, 2009.

[121] R. Priedhorsky and L. Terveen. Wiki grows up: arbitrary data models, access control, and beyond. In *7th International Symposium on Wikis and Open Collaboration*, pages 63–71, Mountain View, CA, 10/2011 2011. ACM.

[122] R. Priedhorsky, J. Chen, S. T. K. Lam, K. Panciera, L. Terveen, and J. Riedl. Creating, destroying, and restoring value in Wikipedia. In *GROUP '07: Proceedings of the 2007 international ACM conference on Supporting group work*, pages 259–268, New York, NY, USA, 2007. ACM. ISBN 978-1-59593-845-9.

[123] L. L. Putnam. Productive conflict: Negotiation as implicit coordination. *International Journal of Conflict Management*, 5(3):284–298, 1994.

[124] F. Rahman and P. Devanbu. Ownership, experience and defects: a fine-grained study of authorship. In *33rd International Conference on Software Engineering (ICSE)*, pages 491–500, 2011.

[125] R. Rivest. The MD5 message-digest algorithm. United States, 1992. RFC Editor, MIT and RSA Data Security, Inc.

[126] M. E. Roloff, D. E. Ifert, and S. Petronio. Conflict management through avoidance: Withholding complaints, suppressing arguments, and declaring topics taboo. *Balancing the secrets of private disclosures*, pages 151–163, 2000.

[127] J. Rydgren. The logic of xenophobia. *Rationality and Society*, 16 (2):123–148, 2004.

[128] R. D. Sack. *Human territoriality: its theory and history*, volume 7. CUP Archive, 1986.

[129] S. Schulz-Hardt, M. Jochims, and D. Frey. Productive conflict in group decision making: Genuine and contrived dissent as strategies to counteract biased information seeking. *Organizational Behavior and Human Decision Processes*, 88(2):563–586, 2002.

[130] H. Sepehri Rad, A. Makazhanov, D. Rafiei, and D. Barbosa. Leveraging editor collaboration patterns in wikipedia. In *Proceedings of the 23rd ACM conference on Hypertext and social media*, pages 13–22. ACM, 2012.

[131] P. Shachaf and N. Hara. Beyond vandalism: Wikipedia trolls. *Journal of Information Science*, 36(3):357–370, 2010.

[132] M. Sharples, J. Goodlet, E. E. Beck, C. C. Wood, S. Easterbrook, and L. Plowman. Research issues in the study of computer supported collaborative writing. In *Computer supported collaborative writing*, pages 9–28. Springer, 1993.

[133] R. Shiller, S. Fischer, and B. Friedman. Stock prices and social dynamics. *Brookings Papers on Economic Activity*, 1984(2):457–510, 1984.

[134] M. Spence. Signaling in retrospect and the informational structure of markets. *American Economic Review*, 92(3):434–459, 2002.

[135] D. Spinellis and V. Giannikas. Organizational adoption of open source software. *Journal of Systems and Software*, 85(3):666–682, 2012.

[136] F. Stadthaus. User interfaces for tracing social editing dynamics in wikipedia. Bachelor's thesis, Karlsruhe Institute of Technology, 2015.

[137] A. Strauss. The literature on panic. *The Journal of Abnormal and Social Psychology*, 39(3):317, 1944.

[138] B. Suh, E. H. Chi, B. A. Pendleton, and A. Kittur. Us vs. them: Understanding social dynamics in wikipedia with revert graph visualizations. In *Visual Analytics Science and Technology*, pages 163–170, 2007.

[139] B. Suh, E. H. Chi, A. Kittur, and B. A. Pendleton. Lifting the veil: Improving accountability and social transparency in Wikipedia with Wikidashboard. In *Proc. CHI*, 2008.

[140] B. Suh, G. Convertino, E. H. Chi, and P. Pirolli. The singularity is not near: slowing growth of Wikipedia. In *Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, pages 8:1–8:10, New York, NY, USA, 2009. ACM.

[141] R. Sumi, T. Yasseri, A. Rung, A. Kornai, and J. Kertész. Edit wars in wikipedia. *arXiv preprint arXiv:1107.3689*, 2011.

[142] C. R. Sunstein. *Republic.com 2.0*. Princeton University Press, 2009.

[143] J. Surowiecki. *The wisdom of crowds*. Anchor, 2005.

[144] Y. Suzuki. Quality assessment of wikipedia articles using h-index. *Journal of information processing*, 23(1):22–30, 2015.

[145] Y. Suzuki and M. Yoshikawa. Mutual evaluation of editors and texts for assessing quality of wikipedia articles. In *Proceedings of the Eighth Annual International Symposium on Wikis and Open Collaboration*, page 18. ACM, 2012.

[146] A. Swartz. Who writes wikipedia? 2006. URL http://www.aaronsw.com/weblog/whowriteswikipedia.

[147] S. Taneja and M. Taneja. A comparison of cloud based office productivity suites. *International Journal of Computer Science and Mobile Applications*, 2(6):1–11, June 2014.

[148] D. Tapscott and A. D. Williams. *Wikinomics: How mass collaboration changes everything*. Penguin, 2008.

[149] J. Thom-Santelli. *Expressing territoriality in online collaborative environments*. PhD thesis, Cornell University, 2010.

[150] J. Thom-Santelli, D. Cosley, and G. Gay. What's mine is mine: territoriality in collaborative authoring. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1481–1484. ACM, 2009.

[151] J. Thom-Santelli, D. Cosley, and G. Gay. What do you know?: experts, novices and territoriality in collaborative systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1685–1694. ACM, 2010.

[152] L. Van Dyne and J. L. Pierce. Psychological ownership and feelings of possession: three field studies predicting employee attitudes and organizational citizenship behavior. *Journal of Organizational Behavior*, 25(4):439–459, 2004.

[153] F. B. Viégas, M. Wattenberg, and K. Dave. Studying cooperation and conflict between authors with history flow visualizations. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, CHI '04, pages 575–582, New York, NY, USA, 2004. ACM. ISBN 1-58113-702-8.

[154] J. Voss. Measuring wikipedia. In *Proceedings of the International Conference of the International Society for Scientometrics and Informetrics (ISSI)*, Stockholm, 2005.

[155] B. Vuong, E. Lim, A. Sun, M. Le, and H. W. Lauw. On ranking controversies in wikipedia: models and evaluation. In *WSDM '08: Proceedings of the international conference on Web search and web data mining*, pages 171–182, New York, NY, USA. ACM.

[156] J. Wales. Jimmy wales talks wikipedia, December 2005. URL http://writingshow.com/?pageid=91.

[157] M. Warncke-Wang, D. Cosley, and J. Riedl. Tell me more: an actionable quality model for wikipedia. In *Proceedings of the 9th International Symposium on Open Collaboration*, page 8. ACM, 2013.

[158] M. Warncke-Wang, V. R. Ayukaev, B. Hecht, and L. G. Terveen. The success and failure of quality improvement projects in peer

production communities. In *Proceedings of the 18th ACM Conference on Computer Supported Cooperative Work & Social Computing*, pages 743–756. ACM, 2015.

[159] S. Wessely. Mass hysteria: two syndromes? *Psychological medicine*, 17(01):109–120, 1987.

[160] A. G. West, S. Kannan, and I. Lee. Detecting wikipedia vandalism via spatio-temporal analysis of revision metadata? In *Proceedings of the Third European Workshop on System Security*, pages 22–28. ACM, 2010.

[161] A. G. West, J. Chang, K. Venkatasubramanian, O. Sokolsky, and I. Lee. Link spamming wikipedia for profit. In *Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, pages 152–161. ACM, 2011.

[162] S. Whittaker, L. Terveen, H. Hill, and L. Cherny. The dynamics of mass interaction. In *Proceedings of the 1998 ACM conference on Computer supported cooperative work (CSCW98)*, 1998.

[163] D. M. Wilkinson and B. A. Huberman. Assessing the value of cooperation in wikipedia. *First Monday*, 12(4), 2007.

[164] W. W. Wilmot and J. L. Hocker. *Interpersonal conflict*. McGraw-Hill New York, 2001.

[165] E. Wilson and D. Sherrell. Source effects in communication and persuasion research: A meta-analysis of effect size. *Journal of the Academy of Marketing Science*, 21(2):101–112, 1993.

[166] N. Wingfield. Feminist critics of video games facing threats in 'gamergate' campaign. *Slate*, 2014. URL http://www.nytimes.com/2014/10/16/technology/gamergate-women-video-game-threats-anita-sarkeesian.html.

[167] "Xenophobia". *Merriam-Webster Online Dictionary*. Merriam-Webster Incorporated, 2015. http://www.merriam-webster.com/dictionary/xenophobia, Accessed 15.04.15.

[168] T. Yasseri, R. Sumi, A. Rung, A. Kornai, J. Kertész, and A. Szolnoki. Dynamics of conflicts in wikipedia. *PLoS ONE*, 7(6), 2012.

[169] T. Yasseri, A. Spoerri, M. Graham, and J. Kertész. The most controversial topics in wikipedia: A multilingual and geographical analysis. *Global Wikipedia: International and Cross-Cultural Issues in Online Collaboration, Fichman P., Hara N., eds., Scarecrow Press*, 2014.

[170] K. Yew Wong. Critical success factors for implementing knowledge management in small and medium enterprises. *Industrial Management & Data Systems*, 105(3):261–279, 2005.

[171] G. K. Zipf. Human behavior and the principle of least effort. 1949.

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ACRONYMS AND ABBREVIATIONS

API Application Programming Interface

ArbCom Arbitration Committee

CSCW Computer-Supported Collaborative Work

CW Collaborative Writing

CWS Collaborative Writing Systems

DIFF Text Difference Algorithm

diffing Applying the Text Difference Algorithm

FLOSS Free/Libre Open Source Software

HIT Human Intelligence Task

HTTP Hypertext Transfer Protocol

revID Revision Identifier

SrevID Sequential Revision Identifier

SIRD Simple Identity Revert Detection

turkers Amazon Mechanical Turk workers

URL Uniform Resource Locator

Wikitext Text written in the Wiki markup language

WWW World Wide Web

APPENDIX

A.1 CROWDSOURCING TASKS FOR ACCURACY EVALUATION OF WIKIWHO

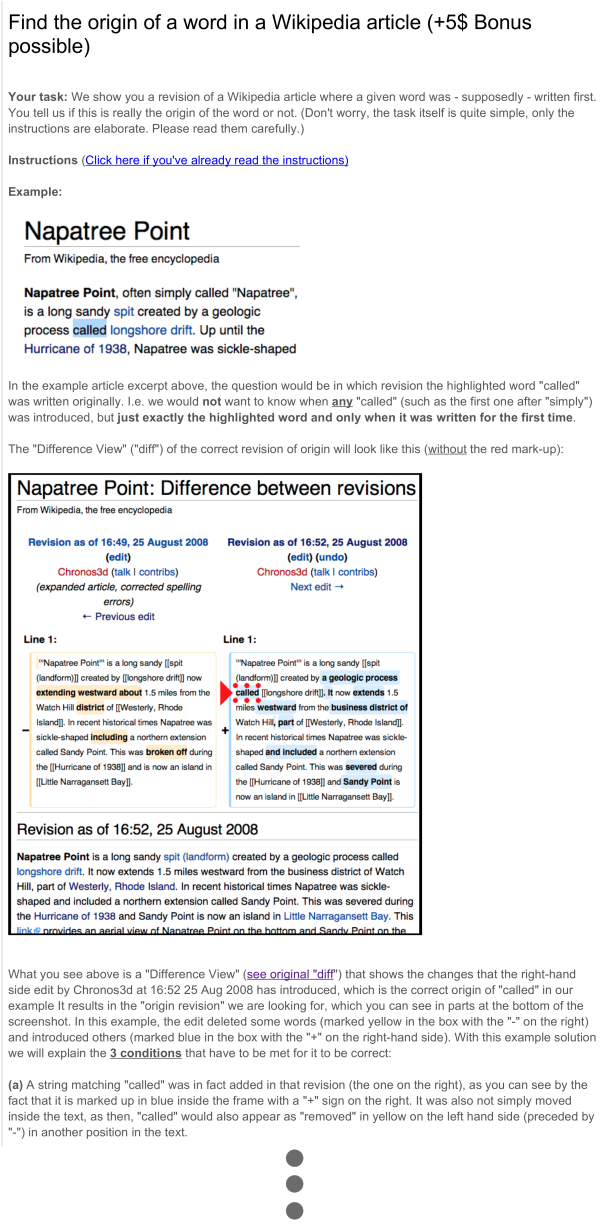


Figure 23: **Task for step 2 (part 1)** as presented to the Mechanical Turk workers for evaluating the accuracy of a revision of origin. It shows the whole page visible to workers, including the detailed instructions. (Online version: `http://people.aifb.kit.edu/ffl//wikiwho/hit_step3.html`). Continued in Figure 24.

(b) It seems to be the same "called" we were looking for, i.e. it is not the first "called" after "simply" that was added here, but the second one. However, it's up to you to decide and interpret if this is actually the same word or just an identical string of letters. Hints could be words in it's vicinity (e.g. "process" and "[[longshore") or embeddedment in the same sentence.

Hint: It helps to use the search function of your browser (usually ctrl+F) to spot the word in question inside the Diff (and maybe other instances of it that we are not looking for, to rule them out).

(c) The third prerequisite is that this "called" was actually added in the shown revision for the first time and not just re-introduced after being deleted before. This cannot be checked directly with the information in the shown screenshot.
To check this, you would have to click on "Previous edit" to navigate back trough older changes or search in the "Revision history" of the article to see of you find a earlier instance of it being introduced (find an elaborate instruction here if needed). Still, you can also look for hints in the comment of the right-hand revision (in our example there is none), such as the editor saying something like "reverted to.." or "reintroduced.." or similar, which however should *not serve as definite proof*.

Ask us if you've got any further questions.

(- end of instructions -)

Task for this current HIT:

For this given word or series of characters (highlighted in yellow*)
is this revision (whose actions are displayed in the differences view)
the correct revision of origin, where it was written for the first time?

(*Sometimes, the word will be a highlighted part of a string of words in a yellow box. This is because it is written in Wiki Syntax (simple markup language for Wikis) and you would otherwise not see it. You will however see it in the Diff, if it is there.)

Choose one and fill the textboxes where applicable. Return the HIT if you are not sure, we employ control HITs to spot and reject sloppy answers.

1. **Yes,** this is the correct revision of origin for this word, meeting conditions (a)+(b)+(c).

2. **No,** because condition (a) is not met (and therefore neither are (b) or (c)).

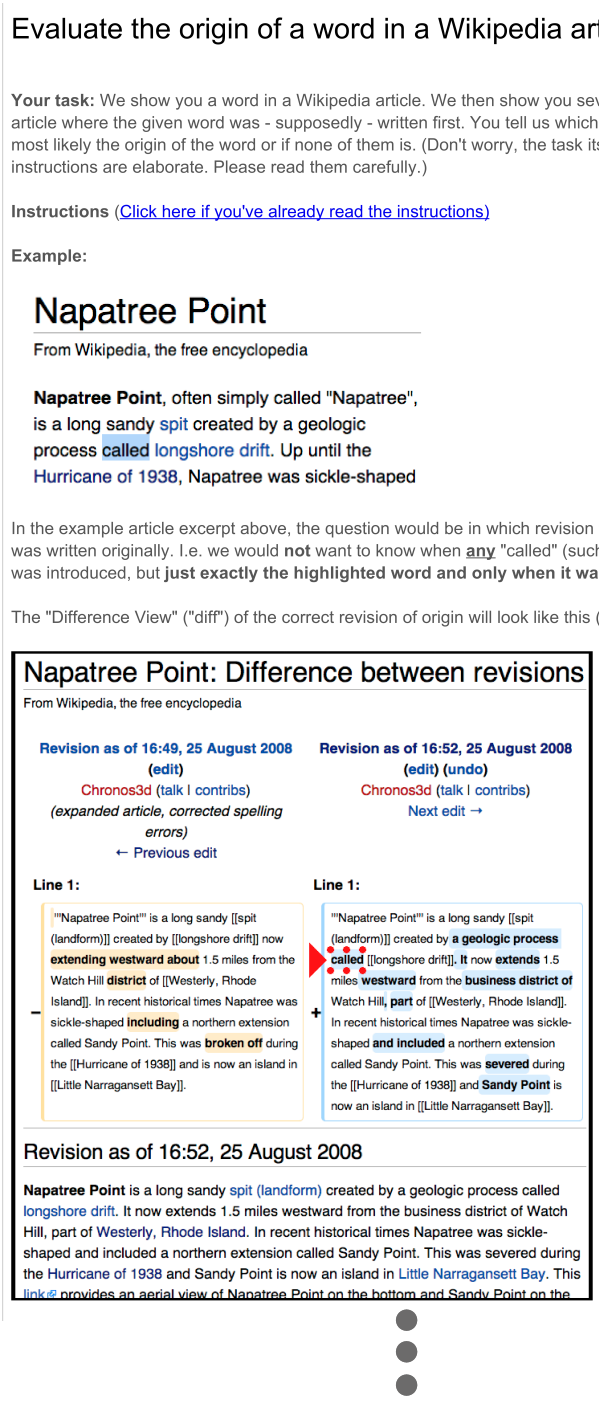
3. **No,** because condition (a) is met, but condition (b) isn't (and therefore neither is (c)).

4. **No,** (a)+(b) are met, but (c) isn't met. Still, you don't know the correct revision of origin. Explain below how you infer that (c) isn't met:

5. **No,** at least (c) is not met and you found the earlier, real origin revision of the word, which meets conditions (a)+(b)+(c). (For instance because our given solution is just a reintroduction). You thus claim the **5$ bonus**.
Enter the URL of the revision you found below:

(Important note regarding **option 5**: If you choose this option (see how to search for the original revision here) and the revision provided by you indeed proves to be a better fit than our suggestion, you will receive a **bonus of 5 Dollars.** For your solution, conditions (a) and (b) must be **strictly** met. The URL pointing to your suggested revision must look like this:
http://en.wikipedia.org/w/index.php?title=SomeArticleName&oldid=12345
, where the number after "oldid=" will be the revision ID where the word has been introduced. The underlined parts will vary. E.g. for our "called" example above, it would be this URL: http://en.wikipedia.org/w/index.php?title=Napatree_Point&oldid=234165074, see see diff here)

Figure 24: **Task for step 2 (part 2).** Continuation of the screenshot from Figure 23



Evaluate the origin of a word in a Wikipedia article

Your task: We show you a word in a Wikipedia article. We then show you several older revisions of the same article where the given word was - supposedly - written first. You tell us which of the presented solutions is most likely the origin of the word or if none of them is. (Don't worry, the task itself is quite simple, only the instructions are elaborate. Please read them carefully.)

Instructions (Click here if you've already read the instructions)

Example:

Napatree Point

From Wikipedia, the free encyclopedia

Napatree Point, often simply called "Napatree", is a long sandy spit created by a geologic process called longshore drift. Up until the Hurricane of 1938, Napatree was sickle-shaped

In the example article excerpt above, the question would be in which revision the highlighted word "called" was written originally. I.e. we would **not** want to know when **any** "called" (such as the first one after "simply") was introduced, but **just exactly the highlighted word and only when it was written for the first time**.

The "Difference View" ("diff") of the correct revision of origin will look like this (without the red mark-up):

Napatree Point: Difference between revisions

From Wikipedia, the free encyclopedia

Revision as of 16:49, 25 August 2008 (edit) Chronos3d (talk | contribs) *(expanded article, corrected spelling errors)* ← Previous edit

Revision as of 16:52, 25 August 2008 (edit) (undo) Chronos3d (talk | contribs) Next edit →

Line 1:

'''Napatree Point''' is a long sandy [[spit (landform)]] created by [[longshore drift]] now **extending westward about** 1.5 miles from the Watch Hill **district** of [[Westerly, Rhode Island]]. In recent historical times Napatree was sickle-shaped **including** a northern extension called Sandy Point. This was **broken off** during the [[Hurricane of 1938]] and is now an island in [[Little Narragansett Bay]].

Line 1:

'''Napatree Point''' is a long sandy [[spit (landform)]] created by **a geologic process called** [[longshore drift]]. **It** now **extends** 1.5 miles **westward** from the **business district of** Watch Hill**, part** of [[Westerly, Rhode Island]]. In recent historical times Napatree was sickle-shaped **and included** a northern extension called Sandy Point. This was **severed** during the [[Hurricane of 1938]] and **Sandy Point** is now an island in [[Little Narragansett Bay]].

Revision as of 16:52, 25 August 2008

Napatree Point is a long sandy spit (landform) created by a geologic process called longshore drift. It now extends 1.5 miles westward from the business district of Watch Hill, part of Westerly, Rhode Island. In recent historical times Napatree was sickle-shaped and included a northern extension called Sandy Point. This was severed during the Hurricane of 1938 and Sandy Point is now an island in Little Narragansett Bay. This link provides an aerial view of Napatree Point on the bottom and Sandy Point on the

Figure 25: **Task for step 3 (part 1)** as presented to the Mechanical Turk workers for evaluating the accuracy of a revision of origin. It shows the whole page visible to workers, including the detailed instructions. (Online version: `http://people.aifb.kit.edu/ffl//wikiwho/hit_step3.html`). Continued in Figure 26.

What you see above is a "Difference View" (see original "diff") that shows the changes that the right-hand side edit by Chronos3d at 16:52 25 Aug 2008 has introduced, which is the correct origin of "called" in our example. It results in the "origin revision" we are looking for, which you can see in parts at the bottom of the screenshot. In this example, the edit deleted some words (marked yellow in the box with the "-" on the left) and introduced others (marked blue in the box with the "+" on the right-hand side). With this example solution we will explain the **3 conditions** that have to be met for it to be correct:

(a) A string matching "called" was in fact added in that revision (the one on the right), as you can see by the fact that it is marked up in blue inside the frame with a "+" sign on the right. It was also not simply moved inside the text, as then, "called" would also appear as "removed" in yellow on the left hand side (preceded by "-") in another position in the text.

(b) It seems to be the same "called" we were looking for, i.e. it is not the first "called" after "simply" that was added here, but the second one. However, it's up to you to decide and interpret if this is actually the same word or just an identical string of letters. Hints could be words in it's vicinity (e.g. "process" and "[[longshore") or embeddedment in the same sentence.

Hint: It helps to use the search function of your browser (usually ctrl+F) to spot the word in question inside the Diff (and maybe other instances of it that we are not looking for, to rule them out).

(c) The third prerequisite is that this "called" was actually added in the shown revision for the first time and not just re-introduced after being deleted before. This cannot be checked only with the information in the shown screenshot. However, the time stamp of the revision can give a hint when comparing several possible candidate revisions for the word origin.

When you compare the 3 different solutions below, carefully check which of these fulfills all three of the conditions. If several revisions fulfill (a) and (b), compare their dates to see which came first and therefore most likely fulfills condition (c).

Ask us if you've got any further questions.

(- end of instructions -)

Task for this current HIT:

For this given word or series of characters (highlighted in yellow*)
which of the following revisions is most likely to be the correct revision of origin?

(*Sometimes, the word will be a highlighted part of a string of words in a yellow box. This is because it is written in Wiki Syntax (simple markup language for Wikis) and you would otherwise not see it. You will however see it in the Diff, if it is there.)

Revision A

Revision B

Revision C

Choose the correct revision of origin below. We employ control HITs to spot and reject sloppy answers.

- Revision A
- Revision B
- Revision C
- None of them (Please state the reason below)

Figure 26: **Task for step 3 (part 2).** Continuation of the screenshot from Figure 25

A.2 EXPLANATION OF WHOVIS FUNCTIONALITIES

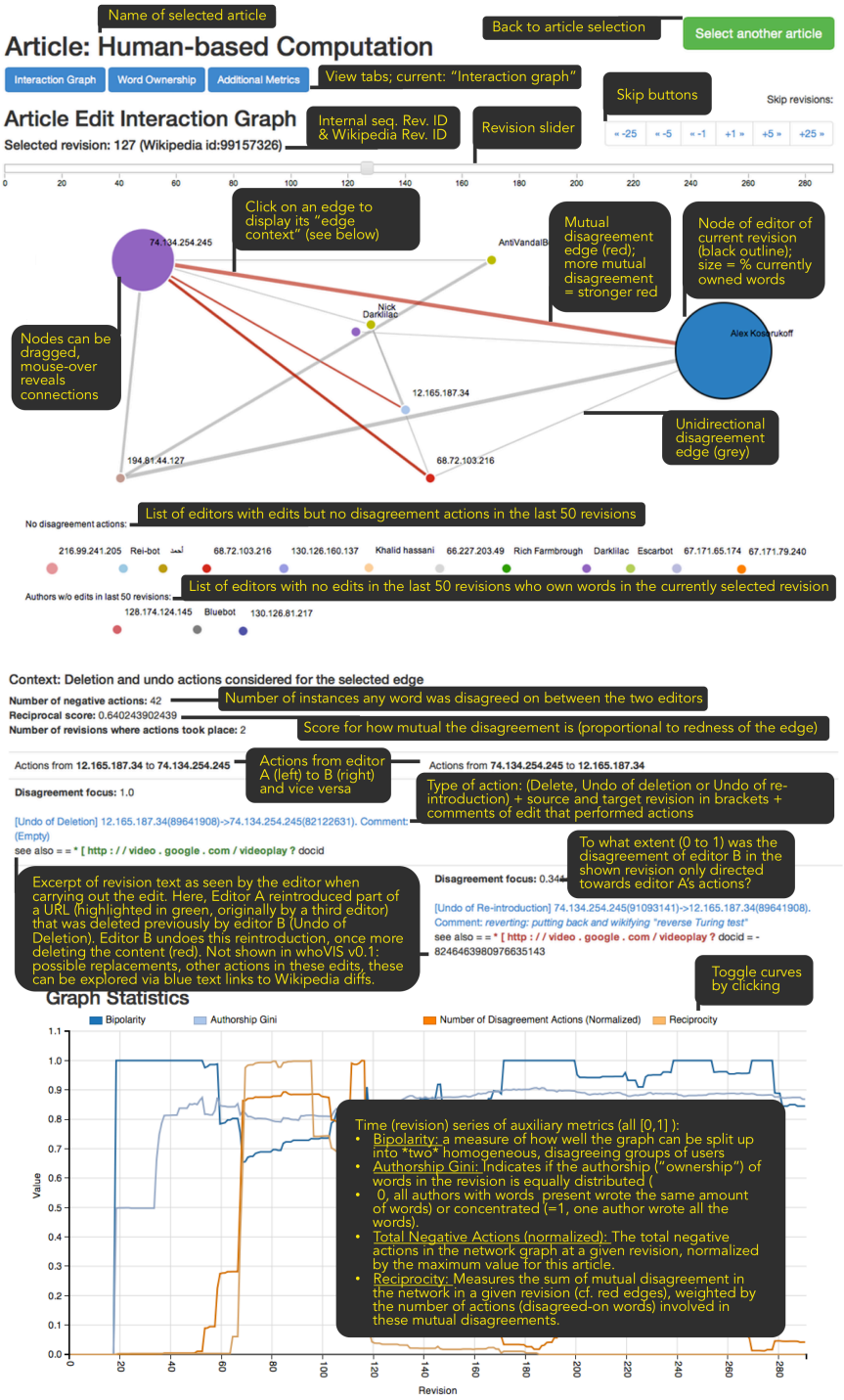


Figure 27: **The "How To" for the whoVIS main view**, annotating the main features, also available at `http://km.aifb.kit.edu/sites/whovis/howto.html`.

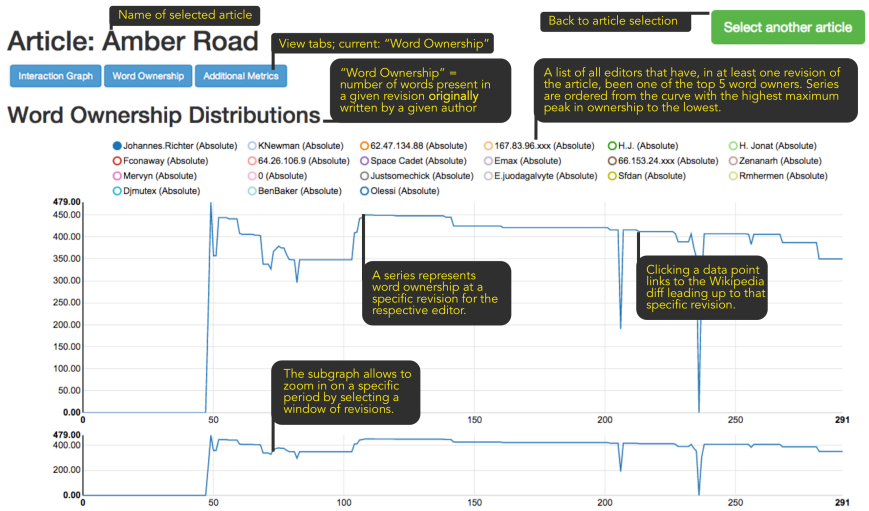


Figure 28: **The "How To" for the whoVIS "Word Ownership" view**, annotating the features, also available at `http://km.aifb.kit.edu/sites/whovis/howto.html`.

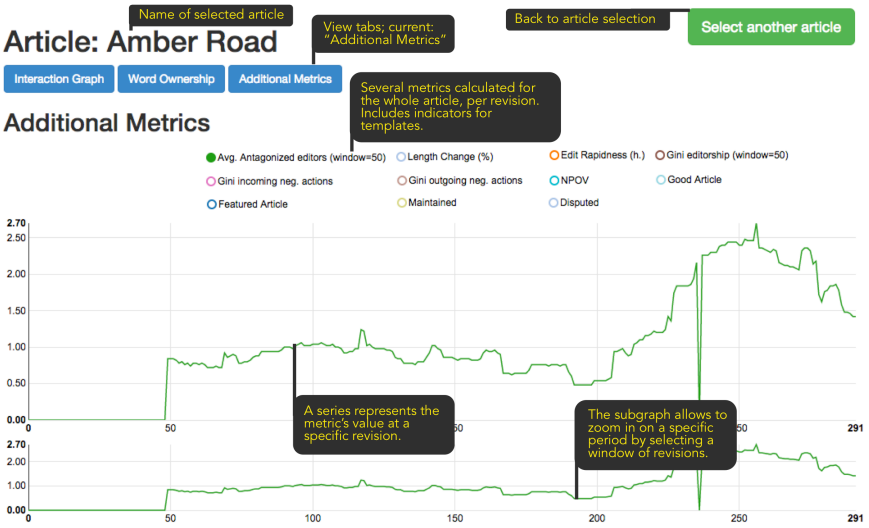


Figure 29: **The "How To" for the whoVIS "Additional Metrics" view**, annotating the features, also available at `http://km.aifb.kit.edu/sites/whovis/howto.html`.