DAMAGE DETECTION AND MITIGATION IN OPEN COLLABORATION APPLICATIONS

Andrew G. West

A DISSERTATION

in

Computer and Information Science

Presented to the Faculties of the University of Pennsylvania

in

Partial Fulfillment of the Requirements for the

Degree of Doctor of Philosophy

2013

Supervisor of Dissertation

Insup Lee

Professor

Co-Supervisor of Dissertation

Oleg Sokolsky

Research Associate Professor

Graduate Group Chairperson

Val Tannen

Professor

Dissertation Committee

Luca de Alfaro, Professor (University of California, Santa Cruz)

Andreas Haeberlen, Assistant Professor

Sampath Kannan, Professor

Jonathan Smith, Professor

DAMAGE DETECTION AND MITIGATION
IN OPEN COLLABORATION APPLICATIONS

Acknowledgments

This dissertation was possible only with the technical and personal guidance of many advisors, professors, professional acquaintances, peers, and family. First and foremost I am indebted to my advisor, Insup Lee, who transformed me from a liberal-arts undergraduate into someone capable of rigorous computer science research. He has given me the freedom to pursue topics I am passionate about, doing so with an oversight that has forced me to think independently and learn from my mistakes. When we first met in 2007 I had a quarter-page C.V. and had never flown on an airplane; the dramatic professional and personal transformations of the past years are a testament to his guidance and research program.

I hold all my committee members – Luca de Alfaro, Andreas Haeberlen, Sampath Kannan, Jonathan Smith, and Oleg Sokolsky – in similarly high esteem, and I thank them for their broad support of my research and their guidance on this dissertation. I have been fortunate to co-author with a number of my UPenn colleagues, but Adam Aviv, Jian Chang, and Krishna Venkatasburamanian deserve special mention for our ongoing collaborations and friendships. Also at UPenn, Robert Terrell (legal), Mary Westervelt (technical writing), and our IRB contacts (ethics) have provided valuable perspective in their respective domains.

External to Penn, Angelos Keromytis and Wenke Lee helped me find direction early in my career. Stefan Savage and Nick Feamster are two individuals I have only briefly met but have had a profound influence through their publications and

approaches to security problems. Going back to my undergraduate work at Washington & Lee University, Rance Necaise, Jacob Siehler, and Thomas Whaley are all thanked for encouraging some of my earliest research endeavors.

Then there are those individuals specific to the domains of collaborative and Wikipedia research. Bo Adler and Luca de Alfaro, despite programming in OCaml, have been excellent colleagues in this space. Martin Potthast should be commended for his over-arching coordination of vandalism research efforts. There are also the Wikimedia/Wikipedia communities themselves. Countless individuals have provided practical perspective on the modus operandi by which these properties operate, acting as technical and policy liaisons. In addition, hundreds of users have downloaded, supported, and/or promoted my tools. Though I have never met most of these individuals their contributions to open-source knowledge and my projects are admirable ones. In particular, Jake Orlowitz ("Ocaasi") has been a long-standing advocate.

Financially, ONR-MURI N00014-07-1-0907 has been a source of support throughout my graduate career. Additionally, the Wikimedia Foundation has generously provided travel grants to participate in their annual conferences.

At a more personal level, I thank my family, including my brothers (who I infamously omitted from a high school graduation speech). Laura has been my best friend and a rock of stability, despite my inclinations towards a high stress "go hard or go home" approach to life and research. Finally, I dedicate this to my puppy Maxwell. He has sat patiently on my lap through the entire process and missed many walks but never complained or asked "when are you going to graduate?".

ABSTRACT

DAMAGE DETECTION AND MITIGATION
IN OPEN COLLABORATION APPLICATIONS

Andrew G. West

Insup Lee

Oleg Sokolsky

Collaborative functionality is changing the way information is amassed, refined, and disseminated in online environments. A subclass of these systems characterized by "open collaboration" uniquely allow participants to *modify* content with low barriers-to-entry. A prominent example and our case study, English Wikipedia, exemplifies the vulnerabilities: 7%+ of its edits are blatantly unconstructive. Our measurement studies show this damage manifests in novel socio-technical forms, limiting the effectiveness of computational detection strategies from related domains. In turn this has made much mitigation the responsibility of a poorly organized and ill-routed human workforce. We aim to improve all facets of this incident response workflow.

Complementing language based solutions we first develop content agnostic predictors of damage. We implicitly glean reputations for system entities and overcome sparse behavioral histories with a spatial reputation model combining evidence from multiple granularity. We also identify simple yet indicative metadata features that capture participatory dynamics and content maturation. When brought to bear over damage corpora our contributions: (1) advance benchmarks over a broad set of security issues ("vandalism"), (2) perform well in the first anti-spam specific approach, and (3) demonstrate their portability over diverse open collaboration use cases.

Probabilities generated by our classifiers can also intelligently route human assets using prioritization schemes optimized for capture rate or impact minimization. Organizational primitives are introduced that improve workforce efficiency. The whole of these strategies are implemented into a tool ("STiki") that has been used to revert 350,000+ damaging instances from Wikipedia. These uses are analyzed to learn about human aspects of the edit review process; properties including scalability, motivation, and latency. Finally, we conclude by measuring practical impacts of our work, discussing how to better integrate our solutions, and revealing outstanding vulnerabilities that speak to research challenges for open collaboration security.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Collaborative, user generated, and crowdsourced functionalities are becoming increasingly prevalent in Web applications. One finds these capabilities in blog/article commenting systems, social networks, web forums, question/answer services, and centralized content hosts. The ability of ordinary users to contribute knowledge, opinions, and other content has allowed sites to aggregate massive amounts of data at minimal marginal cost. One can distinguish among these applications based on their *accessibility*, *i.e.,* the barriers-to-entry they present, and the *permission set* they extend to participants. While many environments spur participation by being accessible most are also "append only" in nature, only allowing users to contribute to monotonically growing discussions or repositories.

Remarkable in this space are "open collaboration" applications [57] that are accessible while allowing users to freely *modify* the content of others. The prototypical example of open collaboration is the online encyclopedia Wikipedia [14], our case study in analyzing the vulnerabilities these types of environments face. In this dissertation (and briefly in this introduction) we argue that open collaboration applications redefine how security abuses can manifest and be mitigated (Sec. 1.1), spurring novel computational and human-driven mechanisms to mitigate their ill-effects (Sec. 1.2).

In the course of exploring this hypothesis and developing practical defense mechanisms we make significant contributions towards the security of Wikipedia and the entire open collaboration paradigm (Sec. 1.3).

1.1 Novel Security Considerations

Our case study, Wikipedia, well represents the security challenges that face open collaboration. Well-publicized inaccuracies have proven detrimental to Wikipedia's public image [107, 119], but these only hint at the larger fact that 7-10% of Wikipedia edits are blatantly unconstructive [110]. Some 400,000 abusive edits each month occur on Wikipedia's English edition alone [22], and as the 6^{th} most trafficked Internet site [2] there are a tremendous number of end users to consume this damage. These are problems of significant scale that manifest in varied ways. In the aggregate such behavior is termed *vandalism*, a set predominately characterized by offensive speech, narcissistic behavior, and unjustified deletions. At times we will concentrate on the *link spam* subset when focus is needed on more acute, subtle, and presumably sophisticated attack vectors. In extreme examples contributions can even incur liability for the host site or be a privacy threat to individuals.

The fact these abusive behaviors occur is unsurprising. Wikipedia has virtually no barriers-to-entry and allows anyone to participate. Given such behaviors the objective is to mitigate the damage as accurately and efficiently as possible. When this does not occur there are negative consequences for the host site (*e.g.,* Wikipedia), its participants, and the collaborative paradigm on the whole. Consider that: (1) When damage is public facing it can erode confidence in the site's information accuracy, (2) if damage must be manually located, this wastes participant time, (3) detection latency is proportional to the utility its perpetrators might derive, and (4) perceptions of a site's viability might affect participant retention and recruitment. These are not effects to which Wikipedia is immune. Authors describe bureaucratic policy [38],

increasing traffic [22], and a declining labor force [63] that threaten Wikipedia's security scalability. While Goldman [63] contends that barriers-to-entry will need to be heightened moving forward, we argue that increasingly accurate and efficient security functionality is a realistic and preferable alternative.

Existing work suggests two complementary approaches: (1) using computational methods to autonomously detect and undo damage, and (2) increasing the efficiency of the human actors who are responsible for the remainder of the problem space. Operating with no latency and zero marginal cost, anti-vandalism classifiers have been an area of research emphasis. A majority of these lie in the natural language processing (NLP) domain: using lexical, syntactic, and semantic methods to identify token patterns that are indicative of damaging behavior. Others have attempted to compute simple reputations for authors. Despite being the state-of-the-art, these techniques can autonomously mitigate less than half of vandalism instances at tolerable false positive rates. This places a significant burden on human participants who only have simple tools at their disposal to expedite the edit review process.

We demonstrate the challenges that underlie the detection/mitigation task by conducting a series of measurement studies that characterize damage and highlight the shortcomings of existing approaches. These reveal that modification semantics are central to the problem's difficulty. Socially, editing in a shared space broadens those behaviors that are considered damaging. Technically, small edit payloads yield little data for methods to analyze. Reputation methods are hampered by sparse participation histories. Similar challenges pervade acute subsets of vandalism (*e.g.,* link spam) that have no wiki-specific work from which to draw. Low barriers to entry also enable attackers exhibiting diverse and sometimes ambiguous intentions.

Human mitigation is also influenced by the modification permission, which enables ordinary participants to fulfill security roles. This gives rise to a complex distributed mitigation ecosystem, one currently characterized by disorganization and brute force. Platforms provide insufficient primitives to facilitate the task, and as a

result, some edits are redundantly reviewed while other damage goes undiscovered. Recognizing these challenges our approach intends to improve both automated and human-driven mechanisms.

1.2 Improving Mitigation

Given that computational methods are more efficient than human mitigation our approach begins by improving the coverage of autonomous classifiers. Complementing the NLP methods of related work we contribute *reputation* and *metadata* features that quantify aspects external to the content payload.

Reputation is the notion that prior behavior is indicative of future action, one achieved by aggregating past behavioral observations [84]. Previous attempts on Wikipedia [27, 29] suffered due to sparse participation histories, a problem exacerbated in OCA environments and broadly termed the "cold start problem". To counter this we utilize the principle of homophily [96] to assess entities based on their spatial adjacency to previously observed ones [138]. For example, if little is known about an editor, the behavior of prior participants in the same IP range can provide considerable predictive intelligence. We describe a generic model that elegantly computes reputation by combining evidence from varying spatial granularity. In addition to applying this technique over users we are novel in associating reputations to system artifacts (and their spatial categorizations) on the presumption that certain topics tend to attract more damaging behavior.

In a similar fashion we use *metadata* features to boost classifier performance. Motivated by the effective use of a small set of spatio-temporal metadata features against email spam [75] we sought to identify parallels in the OCA domain. In total we identify 25+ metadata features that capture participatory dynamics, content evolution, and spatio-temporal properties. Many of these are generic to open collaboration use cases, not tied to Wikipedia's idiosyncrasies or encyclopedic content.

Both reputation and metadata features are evaluated against a vandalism corpus using machine learning techniques. When combined with existing NLP approaches, the resulting classifier significantly advances anti-vandalism benchmarks. Extending the reputation/metadata notions into the URL space we also assemble the first anti-spam classifier for wiki environments. Demonstrating the portability of our content agnostic feature set we describe its application in copyright violation detection, foreign language Wikipedias, and collaborative code repositories.

Regardless the application, computational methods tend to be easily evaluated using standard information recall metrics over offline corpora. In shifting focus to human mitigation we define more dynamic notions such as damage longevity and incident exposure. In developing a strategy to minimize these impacts we describe organizational primitives such as prioritization queues, locks for edit review, and the explicit annotation of innocent edits. These aim to eliminate the redundancy and inefficiencies found in our measurement studies while incorporating computational methods to intelligently route a resource constrained workforce. We integrate these notions into a software tool, "STiki" [136], that English Wikipedia users have utilized to locate and expedite the removal of 350,000+ damaging contributions.

Apart from this tremendous direct impact, the tool's 1.1+ million classification actions are a means to study the manual inspection/mitigation process. We use this data set to reason about the latency, motivations, and scalability of the volunteer workforce. We find that selfish competition and high user turnover are not uncommon. While we combat these with strategies to improve user bandwidth and retention, the underlying trend further reinforces the need to limit dependency on human security actors and to better scale available assets.

1.3 Contributions

This dissertation makes both practical and theoretical progress towards security for open collaboration applications. Practically our work has focused on Wikipedia, facilitating nearly half-a-million damage reverts and helping to maintain low incident longevity and impact measures in the face of exponentially growing traffic. It has also sought to extend these benefits to the entire open collaboration paradigm through greater understanding of modification semantics, participatory dynamics, and distributed mitigation. We hope to enable not just the survival of sites like Wikipedia but permit open collaboration to expand into increasingly sensitive and resource constrained use cases.

To restate and establish our organization moving forward: Open collaboration applications (Chap. 2) redefine how security abuses can manifest and be mitigated. Existing work (Chap. 3) proves insufficient (Chap. 4), spurring novel computational (Chap. 5) and human-driven (Chap. 6) mechanisms. Our findings spur discussion about our practical impacts, security tradeoffs, and outstanding vulnerabilities (Chap. 7). Finally, concluding remarks are made (Chap. 8). Towards proving this hypothesis and accommodating its consequences this writing makes 5 contributions:

1. Through a series of measurement studies we characterize damaging contributions in open collaboration applications. We find the social and technical semantics of *modifying* content differ from those seen in comparable domains. These give rise to novel dimensions for damage to manifest, rendering related work insufficient at the detection task.

2. A model is developed for the spatial overlay of entity granularity reputation algorithms. This leverages the sociological notion of "homophily" and elegantly combines evidence from multiple spatial granularity/contexts. We demonstrate its ability to produce predictive values in spite of sparse behavioral histories (the "cold start problem") by applying it over collaborative users and artifacts.

3. Metadata features are identified that indicate damaging behavior by quantifying participatory dynamics and spatio-temporal properties. We utilize these alongside reputation features to effectively augment computational predictors of contribution quality. The portability of this content agnostic approach is confirmed via application in diverse use cases.

4. Complementing autonomous mechanisms, the notions of damage longevity and incident exposure are defined and leveraged to intelligently route a distributed workforce of security enabled participants. Further improving the defense ecosystem we develop organizational primitives not provided by the base platform that coordinate actors and eliminate redundant work.

5. Practically implementing our suggestions of contribution #4, we create a tool that has enormous direct impact by reverting nearly half-a-million damaging contributions on English Wikipedia. Passively, the tool is a feedback mechanism to learn about human patrol behaviors, latency, and intrinsic motivation. This data permits reasoning about defense scalability and strategies to improve human participation and throughput.

Chapter 2

Background

In this chapter preliminaries are established for the remainder of the work. We begin by defining "open collaboration" applications (OCAs) and constraining this definition to refine the scope of this document (Sec. 2.1). Attention then turns to OCA capable platforms and the practical use cases they enable, with wikis and Wikipedia featuring prominently (Sec. 2.2). We then standardize the terminology for discussing these environments (Sec. 2.3) before making simplifying assumptions about OCA operation and the threat model (Sec. 2.4).

2.1 Open Collaboration

Crucial to this dissertation is the existence of accessible online communities that permit participants to modify others' content. While simply stated, Suber [126] observes that the notion escapes a descriptive, versatile, and standardized label. This writing prefers "open collaboration" (as have others [48, 57, 72]) but recognizes that "openness for authoring", "open content curation", "mass collaboration", and other terms are reasonable analogues. Forte and Lampe [57] offer a four part definition for OCAs, as in italics below. Following each criteria we provide our own strict interpretation for the purposes of this document. Such constraints allow focused

discussion on prototypical examples that have interesting security properties. With that in mind, we consider an OCA to be one that:

1. *Enables the collective production of an artifact(s)*: This is the most fundamental and restrictive of conditions. If we interpret an artifact as a *file* then "collective production" is equivalent to all collective members having *write permissions* over that file (*i.e.,* the ability to edit it in full). This notion should be pervasive. That is, *atomic artifact(s)* must be collectively produced; any meta-artifact must itself be composed of collectively produced elements.

   Should one choose to exercise these permissions their affect should be *direct* and *immediate*; they should not be privy to review except in hindsight. This does not prevent the establishment of multiple collectives (perhaps hierarchical), each having access to different artifact set (see #3).

2. *Is a technologically mediated collaboration platform*: The platform must provide an interface supporting collective artifact production. These collaborations must be digital, although offline sociological and economic perspectives on collaboration [108] parallel many of the behavioral observations made herein.

3. *Has low barriers for entry and exit*: Access should be public (*e.g.,* Internet enabled) and implicitly granted unless explicitly revoked. Sanity checks such as required registration or CAPTCHA solves do not constitute high barriers. Entry is only a guarantee of access to *some* collective, and we stipulate that this initial collective must produce one or more public facing artifacts.

4. *Supports the emergence of persistent but malleable social structures*: There must be functionality for community coordination and discussion. Haythornthwaite [77] examines such social connectivity and governance in greater depth.

We proceed by discussing familiar examples that are *not* OCAs. Append only and monotonically growing content/discussion repositories fail to qualify because they

are not collectively produced at any granularity. This includes applications like YouTube, Flickr, forums, and blog/article comments regardless of the fact their content is user generated (these are aggregated *independent* artifacts). Collaborative filtering applications like Reddit, Digg, and Slashdot are also insufficient. Therein, community voting determines the acceptance and/or prominence of individual content items ("posts") towards composing a public facing artifact. These fail in two dimensions: (1) Voting is an append only action, and (2) supposing participants could fully "edit" the ordering, this presentation is nonetheless a meta-artifact of independent posts – failing the atomicity constraint.

Even a platform/software that is OCA capable can be configured such that it is not in compliance with constraint #3. A tremendous amount of open collaboration platforms run internal to corporate networks as groupware/enterprise tools [94] or are Internet-enabled with private access. Others are more subtle in allowing entry level users to participate but initially preventing or quarantining access to public facing artifacts. For example, on the software development host GitHub the "fork and pull" model permits new users to immediately suggest code changes but still requires project owners to proactively moderate them [45].

Applications that fail one or more of the criteria lack the full dynamism of security problems we wish to describe. This does not imply they are not OCAs under all interpretations, nor that they cannot benefit from this work. Prior research [140] makes explicit the functional overlap between OCAs and an expansive set of web applications, suggesting how our findings might be broadly applied.

2.2 Practical Use Cases

The need to digitally manage artifact evolution (*i.e.,* version control) is not a recent one. The origins of modern version control systems (VCS) such as CVS, SVN, and Git trace back to at least to 1972. It is convenient to think of VCS as middleware

that provides the generic primitives and interfaces needed for group collaboration. OCAs build on these primitives and are rarely distinguished by their back end capabilities (we suspect there is a functional equivalence among all OCA/VCS class systems [146]). Existing OCAs are best organized by the artifact type their front end interface is optimized to support. We now describe use cases focusing on unstructured text (*i.e.,* wikis) (Sec. 2.2.1), structured text (Sec. 2.2.2), and geographical mapping (Sec. 2.2.3). We then speculate on potential use cases and the paradigm's struggle with certain content types (Sec. 2.2.4).

2.2.1 Wikis and Unstructured Text

A *wiki* [90] is a browser-based OCA capable platform optimized for natural language artifacts. The system consists of a set of *documents* and the editing interface is a simple text editor. A markup language provides text formatting and hyperlinking between documents is an oft used functionality.

The collaborative encyclopedia Wikipedia [14] is likely the most well known wiki instance. With rich community governance and vast editing privileges available to even unregistered users, it is considered the archetype of open collaboration applications. Using English Wikipedia as a case study this dissertation will examine its security properties in great detail. However, Wikipedia also has 285+ active language editions that receive far less research attention (see Sec. 5.4.2). Wikipedia is supported by the non-profit Wikimedia Foundation [12] (WMF), which also hosts many other popular wikis: Wiktionary (dictionary), Wikibooks (open source textbooks), and Wikinews (citizen journalism). The content cultivation in these environments, and virtually all OCAs, is done a purely volunteer basis. While monetization might not prove difficult (*e.g.,* via ad revenue), its fair distribution, avoidance of gamesmanship, and affect on participation is an unexplored topic.

Wikipedia's success has been a formative model, with many wikis across the

Internet employing similarly low barriers to participation. Assuming an open directory of 10,000+ wiki instances [10] is accurate and representative, 92.5% fit our OCA definition. Of these 62.5% allow unregistered editing and the other 30% present minimal barriers to entry. Browsing that same directory [10] a reader can view the breadth of topics that wikis cover. Many wikis provide encyclopedic coverage of a subject beyond the depth appropriate for Wikipedia but in a similar style. Literature, video games, and TV series are all popular wiki themes with every character/location/episode having a dedicated document. Regional business listings and technical documentation are also popular use cases. Most wikis (and OCAs) serve as reference works, not by chance, but for reasons described in Sec. 4.1.1. Moreover, installations like AskDrWiki [3] for medical professionals demonstrate that wikis are pervading domains with significant information security ramifications.

There are 100+ different software implementations of the wiki model, but evidence shows that 95%+, including Wikipedia, use the Mediawiki engine [6, 10]. User familiarity and a considerable quantity of plug-ins/extensions no doubt influence this statistic. *Wikifarms* provide central wiki hosting as a service, the most prominent being Wikia [9] with a further 300,000+ communities running Mediawiki software. This homogeneity means that both beneficial tools and harmful attacks that build atop Mediawiki's API have a tremendous range of portability.

2.2.2 Structured Text

While Wikipedia and other wikis hold a tremendous amount of data their natural language format does not lend itself to easy information extraction or machine readability. Sites like Freebase and DBpedia have been moderately successful in parsing ontologies and relational models from unstructured wiki data. However, it is more intuitive to explicitly annotate these relationships when creating content: the objective of *semantic wikis*. Simple software extensions such as Semantic Mediawiki provide this functionality and have seen moderate deployment.

Wikidata, launched in late 2012, seeks to bring these tools to the forefront. The objective is to build a centralized knowledge repository from which Wikipedia's 200+ language editions (and other projects) will query, eliminating the redundant and multilingual duplication of facts [132]. The security challenges of such a shift are not yet known but certainly suggests less reliance on natural language detection. Similar conclusions can be drawn regarding increasing wiki support for interpreted language content (everything from inline specification of GNUplot graphs to environments like XWiki that support server-side script authoring/execution).

2.2.3 Geographical Mapping

One OCA domain where non-textual interfaces have proven successful is collaborative geographical mapping [65]. OpenStreetMap [7] (OSM) "The Free Wiki World Map" is perhaps the most popular example, followed by WikiMapia and Google Map Maker. Traditional wiki software does not power these instances but their reuse of "wiki" vocabulary demonstrates how that term has grown to represent an entire collaborative philosophy and ethos. OSM supports multiple artifact types (*e.g.,* nodes and routes) that reside on map tiles and are added/modified using a rich mapping interface. Intentional malicious damage has been observed in these environments. Autonomous methods for its detection are in their infancy [48, 101], some building on the techniques presented in this dissertation.

2.2.4 Application Challenges

With textual and map interfaces being the predominant OCA use cases one might wonder why the paradigm has not expanded to other file types. Binary files prove particularly challenging. The Wikimedia Commons [11] serves as a collaborative repository for 16+ million media files. However, file modifications are external to the interface, new versions replace prior copies, editing is non-compositional, and there is no diff functionality between historical versions. A majority of these files are

the work of just one author.

This suggests that for meaningful collaboration to take place artifacts need to have a compositional structure, *i.e.,* a grammar/language. Presumably this language needs to be complex enough that it demands new interfaces instead of reusing wiki's textual functionality. In mapping examples, the interface enables a visual representation and avoids the handling of precise geocoordinates. So while binary images might not be ideal, one could imagine the shape-driven composition of flowcharts or organizational diagrams as in presentation software. Metavid [47] is striving for the collaborative transcription of videos, which involves both text and temporal cues. Additionally, musical notation has been suggested as an area ripe for exploration.

2.3 Terminology

While this work intends to speak broadly about OCA security we prefer to use wiki terminology in doing so because: (1) English Wikipedia is our case study, (2) there is a sparsity of non-wiki examples, and (3) the wiki approach has been influential on subsequent non-wiki environments. It is our intention to avoid the vast and specialized vocabulary of many Wikipedia specific discussions [32, 37, 38].

A *wiki* is an OCA composed of *document* artifacts, which Wikipedia calls *articles* in its encyclopedic context. These documents are interlinked using hyperlinks, termed *internal links* or *wikilinks*. These are distinct from links with a destination outside the wiki: *external links*. Documents may be organized into *namespaces* that are used for scoping purposes. For example, a document "`A`" may reside in the *main* namespace, while document "`Talk:A`" exists to discuss that article (in the "talk" namespace). Different namespaces may imply different editing conventions, and it is common to see these for purposes of discussion, help, and policy.

Every document, d, has an associated *version history*. The initial version, d_0, is empty. The most recent version is that displayed by default. The transition

$d_{n-1} \rightsquigarrow d_n$ is termed an *edit* or *revision*, the fundamental action by which all changes are made. A special kind of edit, a *revert* or *undo*, restores the content of a previous version, *i.e.,* if version $d_n = d_{n-2}$ then d_{n-1} was reverted. A revision has an associated *edit summary* where the editor can concisely describe/justify the change.

Every revision has a single *editor*, *author*, or *contributor*. Editors are assigned some type of identifier (*i.e.,* a username) although they are not necessarily persistent. Wikipedia allows *unregistered*[1] editing using one's (possibly dynamic) IP address as an identifier. Regardless the registration status one's identifier is generally referred to as an *account*. Some accounts may have *administrative privileges* that pertain to the blocking of users and the locking of artifacts. In addition to administration and editing simply *reading* wiki content can be interpreted as a form of participation. Taken as a whole the user base is referred to as a *community*.

2.4 Threat Model

In examining the security of open collaboration we need to be explicit about the threats we aim to mitigate. This includes those behaviors considered damaging (Sec. 2.4.1) and the capabilities attackers have in mounting such offenses (Sec. 2.4.2).

2.4.1 Scope of Damage/Defense

Damage: Herein we considered *damage* to include any act that blatantly compromises the integrity of an open collaboration application. To be *blatant* a change must egregiously decrease the value of an artifact as objectively determined by community members over application specific objectives. For example, Wikipedia is a reference work that values factual accuracy, and therefore newly contributed text is

[1]Note that this is sometimes erroneously termed *anonymous* editing. In fact, opaque identifiers reveal far less information than an IP address that could be geolocated or associated to some organization. Generally speaking the IP addresses that operate registered accounts are treated as private data and revealed only for security investigations with probable cause.

predominately judged along these terms. Obviously false statements are treated as blatant damage, while a statement with any degree of truth but ill-formed otherwise is treated more favorably (*e.g.,* removed without force, moved into discussion space, or modified into compliance). Establishing such distinctions is terminologically and practically challenging, with the latter relying heavily on precedent. In this dissertation we rely on Wikipedia experts to make these distinctions for us, by mining their implicit actions to generate corpora. This approach allows detection models to capture community conventions no matter how difficult they may be to codify.

A large percentage of blatantly unconstructive actions are *vandalism*, a term implying *intentional* damage in both Wikipedia and non-digital definitions. The ability to determine intent, especially in a semi-anonymous Internet scale system, is dubious at best. For this reason we prefer to ignore this constraint and consider all blatant surface level damage as vandalism. We recognize that portions of vandalism may be non-intentional and therefore labels like *attacker* and *malicious* need to be carefully used. Regardless, our primary focus is the persistent and dynamic threats posed by intentional actors.

Vandalism consists of a diverse set of behaviors which manifest in various ways. Others have created taxonomies describing this space [114, 131] and we offer our own characterizations in Sec. 4.1.1. One subset is of particular interest in this work, link spam: blatant violations of external link policy. Another interesting set we consider has intersection with vandalism: edits redacted over privacy and liability concerns.

This latter case makes evident there can be blatant damage which is *not* vandalism. Copyright violations are often fundamentally good content that cannot be included due to the provenance of that information (*i.e.,* the concern is not "surface level"). Such cases also make clear that "blatant" does not necessarily mean "trivial to detect". Making a damage determination might require subject expertise or external evidence not held by all users. However, we assume there would be no debate over a "blatant damage" classification if all such information were available.

Defense task: Our goal is the efficient mitigation of qualifying damage instances. We are concerned foremost with *zero-delay detection*, assessing edits using only evidence available at the time they are committed. The task of *historical detection* [28, 143] is relevant in building hindsight based reputation metrics (Sec. 5.2.2), but we not consider its broader implications as we have discussed in [28, 143].

It is our objective to mitigate damage without altering wiki infrastructure. We seek efficient ways to reuse existing functionality rather than suggesting mechanisms that limit open collaboration semantics (these practical security tradeoffs are touched on in Sec. 7.1.3). Finally, we aim to secure only the main namespace of content development, not the adjacent discussion/social/policy spaces.

2.4.2 Attacker Capabilities

Having established what qualifies as damage we now outline the vectors available to actors in mounting such attacks. First, damage must manifest via editing actions. For example, DDOS attacks on artifacts or software compromises are not considered. This damage must occur in public-facing and plain text content. This means the addition/inclusion of inappropriate images on Wikipedia articles is outside our scope. Moreover, the impact of this damage must be immediate. One cannot externally link to a webpage under their control and purposefully change the content at that link at some later point in time (*e.g.,* a TOCTTOU attack).

We assume attackers have editing permissions over the entire main namespace and that they can modify any aspect of those documents. These changes could be as subtle as re-organization of existing content or the addition/removal of a single token. Conversely, tactics can be blatant, *e.g.,* the blanking of article content. An attacker may distribute his/her changes over the course of multiple edit commits. The articles being modified are expected to be independent in nature with content or infrastructure not being transcluded/imported from centralized templates.

We assume human actors perform the editing actions, doing so through the normal editing interface. Editing scripts and direct API access are not considered. The speed at which an attacker can edit is limited only by human and network constraints. The IP address(es) from which an attacker operates is that (are those) natively assigned to the machine(s) they are using. An attacker is free to use any/all machines or IP addresses at their immediate disposal but cannot use proxy, cloud, or botnet services to obtain great degrees of IP agility. Using these an attacker is free to create multiple/Sybil accounts, which may be registered, anonymous, or any combination thereof. These accounts can be operated in a deceptive manner, perhaps performing some quantity of constructive edits before placing damage. However, attackers shall not attain advanced or administrative privileges.

This threat model does make simplifying assumptions but it still captures the vast majority of attacks we have observed against Wikipedia in the wild. In Sec. 7.2 we relax some of these criteria to describe potentially problematic, but to this point, hypothetical attack scenarios. While these present challenging security burdens such strategies also require tremendous attacker investment, the security economics of which remain an unexplored question.

Chapter 3

Related Work

In this chapter we examine related work pertaining to damage discovery and mitigation in online collaborative environments. We begin with technical work specific to wikis and Wikipedia (Sec. 3.1) before looking more broadly to other collaborative applications, *e.g.,* forums, blogs, and social networks (Sec. 3.2). Then, social factors in the collaborative security process are examined; those pertaining to perpetrators of damage as well as those who mitigate it (Sec. 3.3).

3.1 Technical Wiki Defense

As the prototypical open collaboration application there has been much Wikipedia specific research on damage discovery and quality assessment. Authors have approached the problem with differing intentions and granularity, and we begin by synthesizing these varying perspectives (Sec. 3.1.1). Then, the three main technical approaches are described: language (Sec. 3.1.2), content persistence (Sec. 3.1.3), and metadata (Sec. 3.1.4). For each we provide an operational overview and enumerate individual works in the domain. In the next chapter we consider the performance and shortcomings of these techniques when they are considered alongside measurement studies of Wikipedia damage.

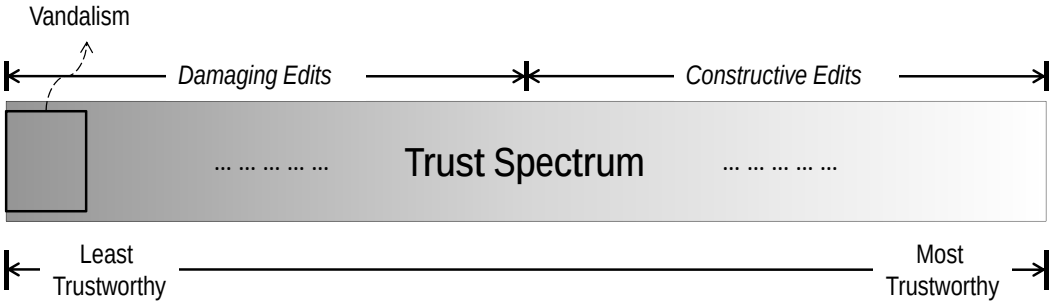


Figure 3.1: Collaborative trust spectrum

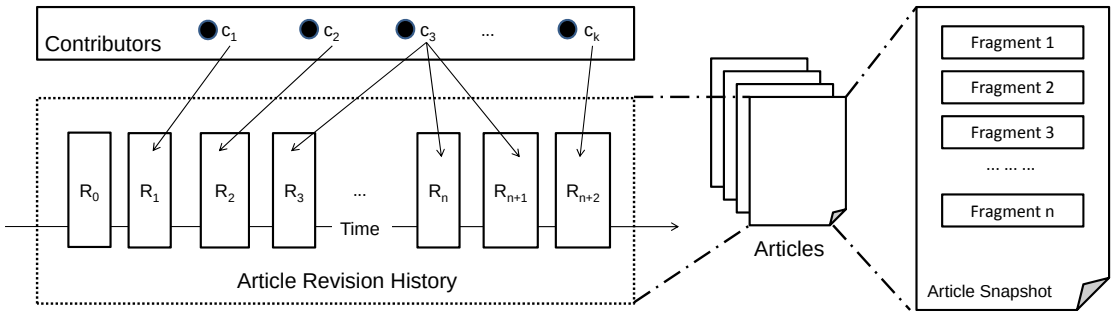


Figure 3.2: Relationships between wiki entities

3.1.1 Contextualizing Literature

When examining Wikipedia literature one will find that systems claim to identify or measure different qualities, among them "vandalism" [42, 80], "trust" [51, 95, 155], and "quality" [35, 125, 152]. Such terminological distinctions are irrelevant given this writing's focus on blatant damage, a more coarse grained problem than the detailed assessments claimed by "trust" and "quality" methods. Virtually all systems in literature are trained and evaluated as two class problems drawn from the extremities of the quality continuum of Fig. 3.1. For this reason we express doubt in the ability to make fine grained distinctions. While expert manual analyses have been performed regarding Wikipedia's accuracy [62, 116], computational parallels seem elusive given that notions such as "trust" are ill-defined and subjective [85].

Systems also assess different wiki entities, focusing on (1) articles, (2) article

fragments, (3) revisions, or (4) editors. We assume that these entities have associative relationships, as visualized in Fig. 3.2. That is, if one can assess any one of these granularity, then this assessment can be mapped to the other three entities. For example, a system that examines document fragments could be run over every fragment written by some author and then an aggregate function applied to make an assessment of that author. This does not imply that approaches work equally well at all granularity. In this dissertation our solutions are optimized for operation at the revision level. Given that edits are the atomic unit of interaction on wikis this seems most intuitive. However, this does not preclude evidence about other entities from being included in a revision level assessment. Note that an *assessment* is simply the output of an anti-damage system. Most often this is a behavior predictive real number that speaks to the probability some entity is damaging.

3.1.2 Language Approaches

Approach: Natural language processing (NLP) analyzes the textual content of additions/removals during a revision on the presumption that damage exhibits different lexical, syntactic, or semantic properties compared to constructive text:

- *Lexical* features are those drawn from the surface level properties of text. These can include token lists or regular expressions that capture vulgarity and slang (indicating damage) or identify advanced editing syntax (likely a benign user). Alternatively, they may include statistical measures that search for repeated characters, calculate word lengths, or calculate alphanumeric character ratios. Fig. 3.3 shows an example of simple lexical analysis being performed over an edit diff using handwritten scoring rules.

- *Syntactic* analysis examines text structure, most often achieved by examining part-of-speech (POS) sequences. Using a POS tagger one can compare the

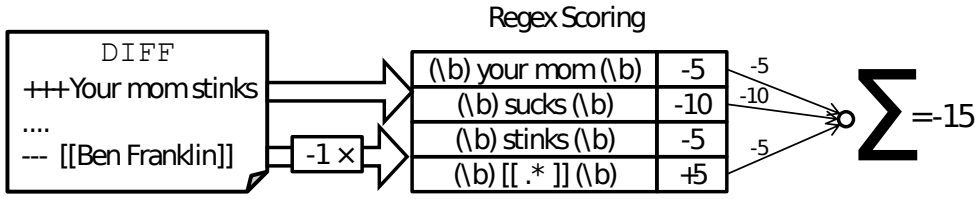


Figure 3.3: Example of edit scoring using lexical features

likelihood of contributed sequences to pre-computed probabilities. Improbable POS sequences are indicative of improper grammatical constructions and possibly damaging contributions.

- *Semantic* features capture the meaning of text. For example, using a tagged corpus of vandalism and innocent edits one can perform Bayesian bag-of-words (BOW) analysis [117] to perform document classification. Given that certain sequences of tokens (*i.e.,* n-grams) appear more often in one class than the other one can compute the product of these probabilities to determine the most likely class for an unlabeled edit's token sequences.

Associated Works: Lexical techniques were among the first brought to bear on Wikipedia's anti-damage problem. The works of Potthast *et al.* [112] and later, Velasco [130, 99], have well-explored this space. Similarly, Rassbach *et al.* [115] used a set of "about 50 features" from a lexical NLP toolkit. These early works found features capturing offensive language, excessive capitalization, and repeated characters to be most indicative of damaging/vandal behavior. Practically speaking, from from mid-2007 to late-2010, "ClueBot" formed the bulk of Wikipedia's autonomous anti-vandal defense using ≈100 handwritten regular expressions capturing many of the above patterns. Leveraging lexical word/sentence lengths, readability metrics (*e.g.,* Flesch-Kincaid, SMOG) have also been shown moderately effective [115, 125].

Syntactically, Wang *et al.* [134] were straightforward in their use of POS tagging and probabilities. Harpalani *et al.* [76] went one step further with their use

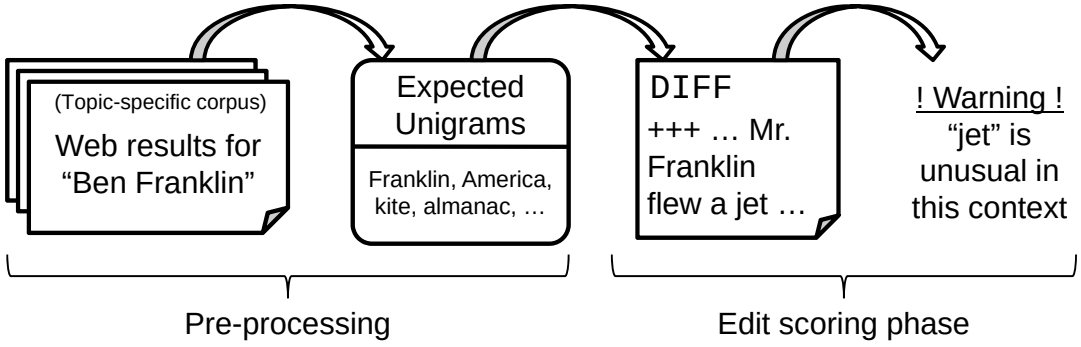


Figure 3.4: Example of topic-specific semantic modeling

of context free grammars (CFGs) using deep stylometric features. Typically used for author identification, the authors showed the methods could identify stylistic elements atypical of Wikipedia's constructive language patterns.

Probabilistic semantic approaches have also been seen in various forms. Chin *et al.* [42] used a generic predictive analysis, Smets *et al.* [121] used Probabilistic Sequence Modeling, and Itakure *et al.* [80] leveraged dynamic Markov compression. These methods use tagged vandalism corpora produced exclusively over Wikipedia edits to produce probabilities. That same approach is put into practice by "ClueBot NG" [36] which uses a neural network classifier. That system has been English Wikipedia's primary means of autonomous anti-vandal defense since late-2010. Rather than using Wikipedia derived corpora Wang *et al.* [134] uses topic specific learning via a search engine's top-k results when queried for an article title. As performance improvements over generic methods demonstrate, these web documents considerably extend the classifiers base of "topic appropriate" token sequences. Fig. 3.4 shows a simplified unigram ($n = 1$) usage of the technique.

3.1.3 Content Persistence Approaches

Approach: Content persistence approaches are built on the intuition that the survival of text through subsequent revisions speaks favorably about: (1) the quality of

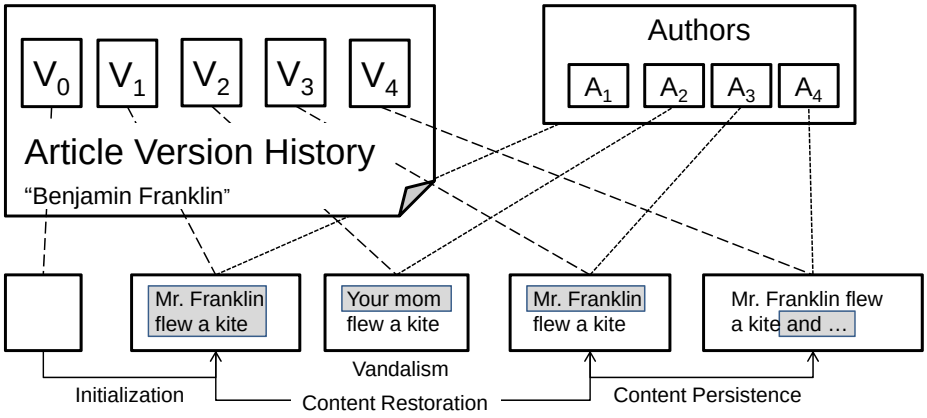


Figure 3.5: Example content persistence calculation

the text fragment and (2) the reputation of its author. Content surviving (or restored by) future revisions, especially those of reputable authors, is likely trustworthy.

Assume author A has made edit r_n on an article and some time later author B edits the same article committing version r_{n+1}. At this point, the reputation of author A can be updated proportional to four factors: (1) the size of A's contribution, (2) the text survival of r_n relative to r_{n+1}, (3) the edit distance (capturing organizational changes) of r_n relative to r_{n+1}, and (4) the reputation of B. The reputation of A will be further updated at each subsequent edit until a specified depth is reached. The reputation of A speaks directly to the trustworthiness of A's prior authorship, which is especially useful in judging new contributions of A which are yet to be vetted by subsequent editors.

Figure 3.5 visualizes an example run of the content persistence algorithm. Assume all authors are equally trusted and author A_1 initializes the "Benjamin Franklin" article with content to form version V_1. The actions of editor A_2 in version V_2 challenge the veracity of A_1, since he modifies content from V_1. However, when A_3 restores the content of A_1/V_1, it is A_2's reputation which is punished. When V_4 is committed, A_2's reputation is further reduced, A_1 continues to accrue reputation for his content's survival, and A_3 is rewarded for the persistence of his revert action.

Associated Works: The work of Adler *et al.* [27, 29], the practical implementation of which is called "WikiTrust", has been definitive in the development of content persistence reputation. It is both a formalization and refinement upon the informal proposal made in [44] by Cross, which suggested that text age may be indicative of fragment trust. The system most related to Adler's is that of Zeng *et al.* [155] who used Dynamic Bayesian networks to model article quality. Whereas Adler computes predictive author reputation, Zeng uses predefined *roles* (*e.g.,* administrator, registered, anonymous, *etc.*) as an input to his reputation system. Finally, Wöhner *et al.* [152] measure content persistence and transience throughout an article's lifespan. He finds that quality articles are defined by periods of high editing intensity, whereas low quality articles tend to undergo little modification as they mature.

3.1.4 Metadata Approaches

Approach: If we consider revisions to be the fundamental building blocks of a wiki system then *metadata* is any property which describes those revisions. We divide metadata into two sets: (1) content-exclusive and (2) content-inclusive:

- Content-exclusive: These properties consider descriptors external of article text. For example, each edit has a: (1) timestamp, (2) editor, (3) article title, and (4) edit summary. These can then be directly quantified, aggregated (for example, to compute the number of unique editors in an article's history), or combined with external information (*i.e.,* off-wiki resources).

- Content-inclusive: These measures permit inspection of the article or diff text (*e.g.,* document length or the number of images in that document). Indeed, some degree of text parsing is required to extract these properties. We prefer language driven features of this kind to be classified as lexical NLP signals and structurally driven ones considered metadata.

Content-Exclusive Features	
Editor	**Article**
· Registration status	· Edits in history
· Special permissions	· Authors in history
Revision Summary	**Timestamp**
· Comment length	· UTC hour/day
Content-Inclusive Features	
· Article length	· Revision `diff` size
· Num. external links	· Num. images

Table 3.1: Example metadata features [35, 112, 125]

Metadata indicators capture varied and often subtle aspects regarding participants, content evolution, and the social context in which they operate. Table 3.1 lists several example features of each type.

Associated Works: Existing metadata systems tend to operate at article granularity. For example, Stvilia *et al.* [125] aggregate multiple metadata features to produce measures of information quality (IQ). IQ metrics [133] are properties like completeness, informativeness, and consistency that define document quality (even outside of wikis [156]). Stvilia's contribution is the quantification of these metrics via the use of wiki metadata. For example, a measure of *completeness* considers the article length and the number of internal links. Similar is the work of Dondio *et al.* [51]. Dondio begins by formally modeling the Wikipedia infrastructure and identifying ten "propositions about trustworthiness of articles", but only develops two metrics in full, realizing just three of the propositions. Surprising compared to the complexity of these approaches, Blumenstock [35] claims that a single metadata metric – article word count – is the best indicator of document quality.

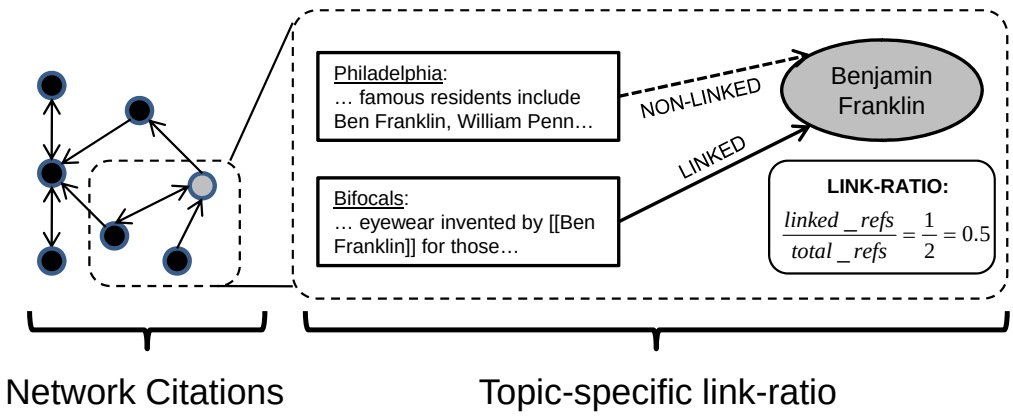


Figure 3.6: Example link-ratio calculation

One subset of content-inclusive metadata pertains to *link-ratio* algorithms. Similar to the way PageRank [106] computes website reputation based on the Internet scale hyperlink graph McGuinness *et al.* [95] and Bellomi *et al.* [33] mine Wikipedia's wikilink graph. The authors argue the decision to wikilink another article (rather than just leaving the subject in plaintext) is an implicit recommendation of that article (see Fig. 3.6), permitting the technique to identify quality articles. In our assessments linking behaviors appear to be driven more by convention than quality, and regardless, the evolution of the link graph is far too slow to be useful for efficient damage discovery. The strategy is not one we pursue further.

3.2 Perspective Outside Open Collaboration

The modification semantics of open collaboration give rise to a rather expansive interpretation of what is considered damaging behavior (Sec. 4.1.1). Most other collaborative applications are concerned with a narrow subset of this space, with the technical detection of "cyberbullying", offensive content, and personal insults in online communities receiving sparse attention [50, 122, 153].

While Wikipedia may be the definitive test bed for vandalism research the same cannot be said for link spam behaviors. Our characterizations in Sec. 4.1.2 and

[137, 139] were the first to address spamming behaviors in a wiki-specific fashion. Thus, we draw from related collaborative fields in trying to develop a baseline anti-spam strategy. The anti-spam strategies of commenting functionality and social networks are well surveyed by Heymann *et al.* [79]. While straightforward lexical and syntactic features are effective [34], more popular is the notion of "language model disagreement". Because comment spam tends to be machine authored (using tools like XRumer [23, 120]) spam posts often are not topic relevant to the content being commented on [34, 98]. Additional work has examined the participatory dynamics of spam users, capturing their bursty and atypical contribution patterns in metadata features [69, 79]. Others have observed that link spam tends to originate from certain IP ranges, suggesting the use of reputation style metrics [26].

An alternative school of thought is to focus on the URL payload included with a spam comment/post by obtaining the source document at that address. Spam *landing sites* tend to show linguistic and structural evidence of commercial intention and/or search engine optimization (SEO) strategies [46, 103]. Additionally, Niu *et al.* [102] found the mere process of obtaining that document was often predictive with spam sites using cloaking and redirection to prevent domain blacklisting.

Practical anti-spam for comments and forums tends to be carried out by proprietary systems such as Akismet [1] and Defensio [4] which easily interface with popular publishing platforms. We presume these integrate many of the aforementioned techniques with the added benefit of global perspective over attacks in progress.

3.3 Social Aspects

Even as computational damage detection techniques advance it is inevitable that classifiers will produce true negatives, borderline cases, and struggle over certain content types. The burden of mitigating these cases falls to human users. For smaller online communities, brute force methods performed by a small set of permissioned

moderators might suffice. Massive content repositories like Facebook and YouTube require more scalable solutions. In practice these sites use low cost outsourced labor to manually review incidents flagged by software or end users [40, 41, 124]. Those citations also describe the psychological toll on these employees who spend hours each day reviewing hate speech, pornography, and violent images.[2] The unrewarding nature of review work suggests it may be difficult to get volunteer users to fulfill such roles. Moreover, the complex image classification guidelines Facebook distributes to its reviewers [41] offer an interesting perspective into the difficulty of codifying objective enforcement policies over user generated content.

Open collaboration applications make mitigation more dynamic in that there are no defined security actors with formal commitments to the task. Besides characterizing their tremendous workload and dedication, little analysis has been done on Wikipedia's (or any comparable) workforce. Geiger and Ribes [60] hint at the coordination and tools available to these users, an ecosystem we describe in greater detail in Sec. 4.2. Wikipedia's mitigation is becoming an increasingly mechanized and structured process; widely interpreted as beneficial technical progress. Halfaker *et al.* [72, 73] counters this with social evidence that robotic actors and template driven warnings are negatively impacting recruitment efforts. Recall that not all damage is the result of malicious actions (Sec. 2.4.1) and well intentioned but ignorant users can easily run afoul of Wikipedia's litany of policies [38].

Information about the demographics and motives of online vandals is also sparse, but there are interesting parallels to criminal justice studies of the physical phenomenon [64]. In the online forum domain some perspective has been offered into behavioral strategies to curb incentives for disruptive "flamers" and "trolls", primarily by ignoring their inflammatory tactics [78, 89]. Meanwhile, the monetary motives of link spammers are far more intuitive, a fact that makes it easier to model their behavior, yet also suggests they will be evasive of any protections put in place.

[2]We have been advised by our University's Office of the General Counsel to avoid non-textual analyses for legal reasons, primarily those surrounding child pornography [149].

Chapter 4

Characterizing Defense Shortcomings

In order to create/improve anti-damage solutions one must first understand the incidents themselves. We conduct measurement studies that characterize the task and highlight the weaknesses of current computational solutions (Sec. 4.1). Then we focus on the deficiencies of human mitigation strategies by modeling that ecosystem and demonstrating that it lacks organizational primitives that could prevent redundant labor and latent damage response (Sec. 4.2).

4.1 Measurement Studies

When conducting measurement studies over damage we use corpora built from the actions of Wikipedia experts. For each damage type we: (1) motivate investigation into the particular type, (2) characterize/quantify the behaviors that qualify as damage, (3) analyze the performance shortcomings of related work, and (4) understand how modification semantics give rise to novel challenges. Vandalism (Sec. 4.1.1) has been characterized in prior literature and the deficiencies of existing anti-vandal systems can be shown via empirical evaluation. Being the first to describe spam

(Sec. 4.1.2) and privacy/liability (Sec. 4.1.3) concerns in open collaboration, we devote considerable space to their characterization and measurement. Then we use this to speculate about the applicability of related work from non-OCA environments.

4.1.1 Vandalism

Motivations: Our investigation into vandalism is foremost motivated by the fact it is an extremely prevalent problem. Over 7%+ of English Wikipedia edits are vandalism, generating over a quarter-million vandal edits per month as of this writing [22, 110]. These are most often acts of malice; the single most edited page on Wikipedia is that used to block persistent vandals where over 500,000 accounts and IP addresses have had their privileges revoked. The large amount of academic and practical research brought to bear on the task (Sec. 3.1) also give some indication of the challenges involved. As our measurements will reveal and prior work has suggested these are solutions not scaling adequately [63]. When instances come to the attention of media outlets and/or a person of influence it can result in considerable criticism for Wikipedia and the open collaboration paradigm. For example, fake biographies [119] and death hoaxes [107] have not been uncommon and occasionally these errors are re-reported by traditional media [109].

Characterization: Other authors have quantified taxonomies of anti-vandal behaviors [114, 131] and there is no need to duplicate those efforts here. Further, a Wikipedia policy page highlights 30+ specific characterizations [21]. For the benefit of readers we convey examples of the most common behaviors seen in the authors' personal experience of reviewing 50,000+ potential vandalism instances:

- Profanity and vulgarity: Inappropriate language is extremely common and rarely accompanies value-adding text.
- Personal opinions: Statements such as "Benjamin Franklin is no fun" are unsourced and without merit. Especially common in rivalry situations such as

politics, sporting teams, and performing artists.

- Unjustified blanking: The removal of large blocks of text without justification (especially long lived and citation inclusive text).
- Narcissism: Vandals mark their presence (*e.g.,* "Alice was here!"), include themselves as "notable alumni" on institutional articles, and/or make themselves part of history (*e.g.,* replacing "Ben Franklin" with their own name).
- Proof of edit-ability: Others are captivated by wiki functionality and are eager to report such (*e.g.,* "I can actually change this?!").
- Random insertions and non-sense: Offenses characterized by randomness (*e.g.,* "asdf"), repeated characters, smilies, and extreme use of copy-paste.
- Attempts at comedy: Despite being damaging, some revisions' attempts at entertainment prove quite successful.

Language shortcomings: Language based detection of vandalism has been popular in both research and practice. One of the most common and standardized methods, Bayesian document classification [117], can autonomously reject less than 50% of vandalism at tolerable false positive rates [36] (see Sec. 5.1 for more about that threshold). To understand this poor performance we compare the output of one such algorithm [36] with a human labeled set (the vandalism corpora described in Sec. 5.2.1). We focus attention on the true negatives, *i.e.,* vandalism instances that evaded language based detection. We find several trends to be common.

First, many edits add information which is blatantly untrue but do so in an encyclopedic style. That is, they are syntactically well formed, lexically well formatted, and semantically appropriate – but the payload itself is categorically false. The interpretation of mis-truth as damage appears to be one unique to open collaboration and a social outgrowth of collective production, *i.e.,* granting the modification permission. Consider the overarching social objective of OCAs: to cultivate a definitive/authoritative artifact. Our survey of OCA use cases in Sec. 2.2 showed that in

purpose they overwhelmingly have some notion of "correctness", often serving as reference works or documentation. This contrasts with the role of simple "append only" collaborative applications that often support opinion sharing and artistic interpretation. The condition that factual inaccuracies can be damaging imposes a significant, if not fundamental, computational hurdle – given the lack of an oracle that can assess the veracity of a statement. Wang's [134] suggestion of using web queries to supplement language models is an improvement over simple Bayesian approaches but can only ensure contributions have context appropriate language. Indeed, most grossly inaccurate statements that are non-offensive and well formatted will evade current vandalism/language filters. It appears the naïvety of the average vandal is one reason computational methods achieve even moderate success.

Second, we observe that small payloads prove difficult for language based methods to assess. Most vandalism consists of less than 1KB of diff data. In some cases only a few characters are available for analysis. Consider that these changes need not "stand alone" and instead build off the context of existing content. Such difficulties are rooted in the technical semantics and representation of editing operations. Append only models have artifacts which are submitted in an independent and whole fashion: An entire video is submitted to YouTube; a blog comment must be a complete statement. In contrast, OCA modification actions operate in a dependent fashion and damage can appear at arbitrary location(s) inside a document.

In a similar manner, other edits yield *no* language tokens for analysis. Modifications can be purely re-organizational and alterations to numerical data (*e.g.,* intentionally skewed records for sports teams) cannot be modeled as language tokens. Content removals also prove challenging to assess. Techniques like stylometric analysis [76] are explicit that they cannot handle such cases. Others invert the probabilities that would be obtained if the content were added (*e.g.,* the removal of "good" tokens indicates vandalism) and fail to capture aspects of the edit maturation process. Finally, consider that some edits are a complex combination of

additions, removals, and reorganizations and how to best aggregate such actions is an unexplored area of research.

Reputation shortcomings: User reputation gleaned implicitly from content survival is an alternative tactic to detect vandalism. By examining performance of the WikiTrust implementation [27, 29] over vandalism corpora (as described in Sec. 5.2.1) we can learn about the performance shortcomings of that approach.

We find sparse behavioral histories are a severe detriment. When there is little or no behavioral history to mine the algorithms only produce default or null reputation values that have no predictive intelligence (the "cold start" problem). This is a problem exacerbated in OCA environments where:

1. Low participation barriers generate many new/Sybil accounts [52]. Wikipedia has 4.3 million registered accounts and under 40% have made 3+ edits [22].

2. Some configurations do not require persistent identifiers. Roughly 30% of Wikipedia's 350 million article edits were performed by unregistered users [22], with 66% of these contributions being likely vandalism [81]. While unregistered users are 83% of all editing "accounts" the IP addresses used to identify such users can rarely be treated as persistent due to shared use settings (*e.g.,* computer labs) or DHCP addressing.

3. New users perpetrate most damage actions. Dynamism concerns force one to treat most unregistered accounts as new ones, and the median age of damaging registered accounts is just 1.6 hours [142].

As a result of these factors we find a tremendous amount of vandals have null reputation when they perpetrate their damage. Even if an editor has prior edits, realize that reputations have latency given their dependence on subsequent action.

Metadata shortcomings: Metadata approaches to damage detection are not so much a centralized methodology as they are a collection of diverse data points. The information quality metrics of Stvilia [125] were optimized for article quality assessments and translate poorly to revision granularity. Further, the arbitrary and static multi-variable aggregations they utilize are inelegant and unnecessary. Many of the individual metadata components they amass are individually meaningful and indicative of damage. While current metadata features have no fundamental shortcomings, they also suggest unexplored opportunities.

Moving forward: Recognizing the limitations of related work is formative in establishing our anti-vandal approach moving forward. Language approaches are quite mature and such content-centric approaches appear to have fundamental shortcomings. Therefore we prefer to concentrate on complementary methods and ultimately evaluate our improvements in combination with language techniques. Reputation was hampered by the cold start problem, a shortcoming we counter through the development of spatial reputation: judging new entities based on how related/adjacent entities have behaved in the past.

With metadata we continue the development of features that speak to misbehavior and participatory dynamics at the revision level. Ultimately, signals of all types are comparatively and cumulatively evaluated using machine learning techniques. Despite being driven foremost by improving anti-vandalism metrics on English Wikipedia, we are also mindful that it is just a single open collaboration use case. Moving forward we strive to develop features (in contrast to much of related work) that do not make assumptions about content types or capture encyclopedic idiosyncrasies.

4.1.2 Link Spam

Motivations: Wikipedia's objective interests (like those of most OCAs) often conflict with those of marketing departments, ghostwriters, and devotees to any particular institution. Wikipedia has to confront paid consultants [104, 105] and reject/block the contributions of biased parties [24, 145].

Interesting security problems arise when parties knowingly commit damage for self-interest and/or monetization, as we suspected might be common with *external link spam* on Wikipedia. We presume these incentives motivate spammers to exhibit sophistication, evasiveness, resource utilization, and adaptive strategies not seen with simple vandalism. Moreover, Wikipedia has no autonomous protection against link spam besides a reactive blacklist. This is a tremendous burden to human reviewers (or worse, end users) who might encounter malware or questionable content beyond the relative safety of the host platform. Our proposed solutions can not only ease the burden on these reviewers but also diminish the utility of spam actions given that damage survival is likely proportional to attacker yield.

Characterization & measurement: External link spam on Wikipedia must be performed with direct intentions: the goal being for end users to view the spam link (an *exposure*) and choose to click it (a *click-through*) and visit the *landing site* [86]. Attempts to use Wikipedia for search engine optimization (SEO) purposes ceased in early 2007 when HTML `nofollow` was enabled for all outgoing links. Our focus is on external link spam and we investigate only well formed links of this type.[3]

As we describe at greater length in [139], external spam link must violate link policy [15] and transgressions pertain either to a link's *presentation* or *destination*. Link presentation wraps four factors: (1) the accuracy of the hyperlink description, (2) the appropriateness of the link formatting, (3) where in the article the link is

[3]Wikipedia "spam" is broader than well formed external links. For example: (1) Internal wikilinks could be used overzealously to draw attention to a topic, (2) an entire article could be commercially motivated, or (3) publishers/authors might spam on behalf of offline resources.

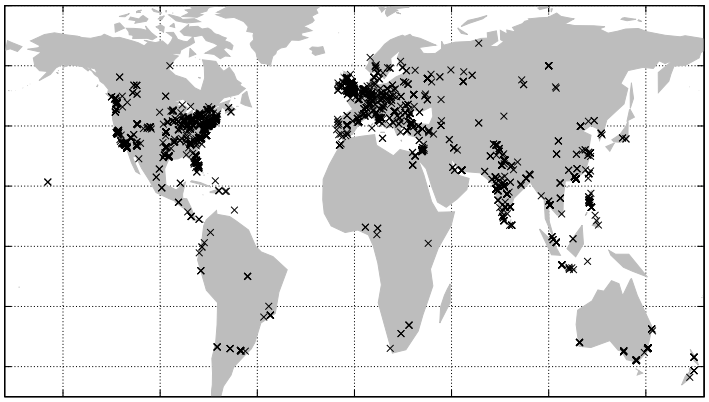


Figure 4.1: Geographical distribution of spamming IP addresses

placed, and (4) the article to which it is placed. Such placement issues are orthogonal to the destination being linked. Though criteria are many [15], it suffices to say that commercial sites, promotional offerings, and narcissistic linking behaviors are prohibited. The practicalities of spam behaviors are quantified via a measurement study, the full details of which are described in [139]. That analysis and our immediate summary utilize the 6,000+ element spam corpus we develop in Sec. 5.3.1, built atop the implicit actions of Wikipedia experts. Here we summarize our most significant findings to be:

- *Atypical spam*: Comparing Wikipedia spam URLs against the Spamhaus DBL blacklist (email spam domains) and Google Safe Browsing project blacklists (phishing/malware) produced virtually no intersection. Similarly, the IPs and geolocation of perpetrators (Fig. 4.1) matched typical editing patterns instead of those identified in measurement studies of "traditional" spamming [139].

- *Little direct commercialism*: Classifying corpus URLs through a link directory taxonomy [5] produces Fig. 4.2, suggesting a diversity of spam types. Extensive manual tagging as per Tab. 4.1 reveals that just 15% of spam links had direct commercial intentions (*i.e.,* products immediately for sale in an online fashion).

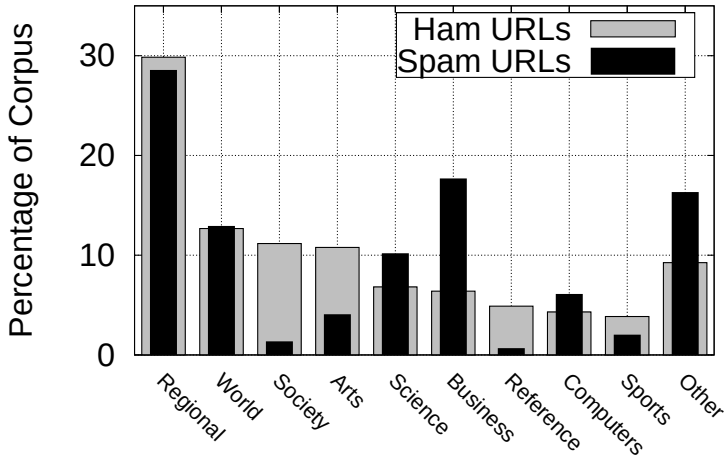


Figure 4.2: Spam corpus URLs classified by genre [5]

- *Indirect commercialism*: Spam for brick-and-mortar businesses was not uncommon, including: (1) small business webpages, (2) bulk regional listings of such businesses (see Tab. 4.1), and (3) "information adjacent services", where a site provides encyclopedic information but also sells a related service. For example, a local restaurant may try to link their website from articles for the city and state in which it resides.

- *Data clearinghouses*: So called "non-authoritative clearinghouses" are common spam destinations: web portals into databases of third party data. For example, a "soccer statistics" clearinghouse may have pages for teams/players, providing the potential for many link points on the wiki. These sites tend to republish information from more official sources and be aggressive in their on-Wikipedia promotion (see Tab. 4.2).

- *Conventional placement*: Spammers tend to follow linking conventions, avoid blanket spamming of context inappropriate articles, and do not abuse style or placement capabilities in order to increase visual prominence. Per Tab. 4.1, only about 2% of spam instances violate these conventions.

DESTINATION PROPERTY	%-age
Commercial storefront	15.5%
Local directory or tourism	7.8%
Social media destinations	2.4%
Foreign language page	1.9%
Adult or offensive link	0.5%

PLACEMENT PROPERTY	%-age
Link uncorrelated w/article	0.9%
Unusual link placement	0.9%
Visual manipulation of link	0.2%

Table 4.1: Characterizing spam properties; these are independent/disjoint properties (thus columns do not sum to 100%) that only aim to capture interesting behaviors in the problem space.

- *Little URL reuse*: Of the nearly 5k spam corpus members, 80% of URLs and 50% of domains are unique. Only 14 domains appear 10+ times and 71% of domains appear just once. In practice one needs to look at the `spam:ham` ratio of a domain to learn about its behavioral patterns, as shown in Tab. 4.2 where three "data clearinghouse" examples are among the worst domains.

- *Problematic users*: Abusive accounts are not uncommon. Our corpus identifies 50+ users adding 10+ spam instances, a majority of which were blocked for their actions. Moreover, the usernames associated with abusive accounts were often indicative of a conflict-of-interest (COI) (*i.e.,* the username matches the domain name, verbatim).

More surprising than the behaviors we do observe are those we do *not*. With English Wikipedia receiving 8.5 billion page views per month we expected to see more blatantly commercial spam,[4] abuse of the modification permission, and use of mechanized scripts. However, it appears that link spammers exhibit sophistication not in technical dimensions but through subtlety and social engineering. By following

[4]Just because a site is not directly commercial does not mean there are no economics, profit, or strong incentive involved. With sufficient traffic, ad revenue could be a significant monetary motivation. Moreover, it is impossible to quantify ideological and narcissistic effects.

DOMAIN	SPAM-#	TOTAL-#	SPAM-%
www.youtube.com	101	3156	3.2%
Area code look-up	72	83	86.7%
www.facebook.com	48	3200	1.5%
Cinematic rankings	41	49	83.6%
www.billboard.com	35	1346	2.6%
Soccer statistics	29	36	80.5%

Table 4.2: Domains w/most spam occurrences in corpus collection; note that non-spam labels are not an endorsement of link quality (Sec. 5.3.1). Some domains are not made explicit because: (1) they are not well known and (2) to avoid additional exposure.

linking conventions they may be hopeful their links will evade (or at least, delay) detection and can persist on articles and derive long term utility.[5] Our insights into the human review of spam in Sec. 5.3.1 and Sec. 6.4 find a conservative labeling bias, *i.e.,* frequent uncertainty over quality result in in link survival. Even if a link is flagged by a human, the fact perpetrators have followed conventions may lessen punitive action or prevent audits of the URL history. Since these subtleties affect human labeling, they also influence our corpus composition of Sec. 5.3.1.

Malicious attackers are able to blend in with those who spam out of policy ignorance, and threads on blackhat and spam forums discuss how to leverage this ambiguity. The mere existence of "ignorant" spammers (*i.e.,* those conducting spam behaviors absent malice and/or an understanding of link policy) speaks to the novel socio-technical configuration of open collaboration applications. It is difficult to imagine a small business owner unknowingly launching a botnet scale email attack. Lower social barriers-to-entry often imply lowered technical barriers as well, enabling misuse by a class of less sophisticated abusers.

Acute instances: External to our corpus and its two month collection, an on-wiki project dedicated to anti-spam efforts is one of the best historical resources regarding

[5]An alternative school of thought is to spam aggressively and maximize resource utilization in the finite window until detection takes place, one we describe further in Sec. 7.2.

persistent attacks of scale [18]. It should be noted that the full scope of many of these incidents (about 100 are reported monthly) is only realized in hindsight.

One case is remarkable for exemplifying the technical vulnerability that Wikipedia does face: "Generic Chinese Knockoff Spam". GCKS is an outfit peddling counterfeit luxury and designer goods, and in 1.5 years it/they have conducted 65+ spam campaigns on English Wikipedia utilizing 300+ domains (see our aggregate report at [148]). The destination sites are template-driven interfaces into a single affiliate transaction processing network (*i.e.,* infrastructure similar to that of sophisticated email spam attacks [91]). More than 3500+ domains have been proactively blacklisted based on the unique signature(s) of this template. The outfit targets multiple language editions of Wikipedia using IP addresses primarily from a single Chinese province. Spam tactics range from blatant to subtle, and anecdotal evidence suggests that this is human-driven "sweatshop spam". In the global landscape of spam behaviors this is not a terribly interesting instance, but it serves as a cautionary example that Wikipedia is not immune to such attacks. However, unlike most other environments, a lack of automated anti-spam defenses for open collaboration means mitigating these recurring instances has been a tremendous human burden.

Our two month corpus collection (see Sec. 5.3.1) does capture multiple GCKS instances (recall the attack has been ongoing for 1.5 years), yet acute strategies such as these form an overwhelming minority of that corpus and the problem space.

Related work shortcomings: While one should remain mindful of the potential for massive commercial attacks like GCKS, they are simply not prevalent in the status quo. However, the absence of direct monetary incentives in no way simplifies the detection task. Quite the opposite, this renders ineffective those approaches that quantify a landing site's "commercial intention" (per Sec. 3.2). Language model disagreement, the most predominant anti-spam mechanism used with commenting functionality, also fails given a lack of blanket spamming tactics and computationally

authored link placement. Remarkably, even when "male enhancement" pharmacies spam Wikipedia they only target those articles relevant to their landing site.

Moving forward: Fortunately, our measurement study suggests novel ways to approach the problem, tactics leveraged in Sec. 5.3.2. Link spam is a subset of vandalism so it is intuitive to reuse the features developed for that purpose. However there is also the opportunity to extend reputation and metadata techniques into the spam/URL space. When historical feedback is sparse internally, Internet scale reputation metrics (*e.g.,* Pagerank [106]) can indicate whether a resource is trusted on a more global scale. Other takeaway properties include: (1) many spam accounts are single purpose in nature, (2) bursty additions of a URL/domain are suspicious, and (3) URLs/domains that that have little author diversity tend to be problematic.

4.1.3 Privacy and Liability

Motivations: When conducting security audits into vandalism and spam instances we discovered that certain events had gone "missing" with their details removed from public view. Subsequent investigation would reveal that such redaction functionality exists to protect the legal and safety interests of Wikipedia and its user base. The need for such actions is not just precautionary, as the encyclopedia has been threatened with litigation for copyright issues [97], accused of hosting child pornography [149], and blacklisted in some regions for content issues [53]. In turn this sparks curiosity about what gets redacted, how often it occurs, how reactive the host is to these threats, and what behaviors most endanger Wikipedia.

Characterization: A functionality available only to Wikipedia administrators is used to redact content from public view (`RevDelete` [19]). We concern ourselves only with actions taken via that tool, where non-administrators can request its use via out-of-band channels. The tool operates at *revision* granularity and is distinct from

Figure 4.3: Example page history w/redaction

the article deletion process [118, 128] (which occurs for less acute reasons). A deleted revision cannot be publicly viewed or diff'ed against, and Fig. 4.3 shows an example redaction in a revision history. Administrators are able to audit the deletions of others and *suppression* functionality elevates this to an even more exclusive set of users. The tool operates only over textual content and while multiple fields can be deleted (see Tab. 4.4) we are concerned primarily with content issues.

Tab. 4.3 and the Wikipedia policy page [19] display the criteria that justify the use of redaction/suppression. These rational are quite opaque and we sought to characterize and quantify actual deletion cases. Using a technique we describe further in [144] we stored every edit to Wikipedia in near real time. Then, public log data and exhaustive API requests reveal revisions which were subsequently deleted. A manual/qualitative analysis allows us to expand on the official "reason for deletion" (RD) criteria and make explicit the liability/privacy issue that each addresses:

- RD1: Copyright violations have obvious legal ramifications. In practice these revisions are exclusively large text insertions where that content has been copy-pasted from another online source.

- RD2: More than simple vandalism, "grossly insulting" deletions tend to address possible libel/slander claims. Revisions often make realistic but unsubstantiated claims regarding promiscuity, pedophilia, and other crimes. Generally an identifiable individual (*i.e.,* full names) is mentioned. Extreme instances of racist and profane hate speech are also removed under this criteria.

ID	DESCRIPTION
RD1	Blatant copyright violation
RD2	Grossly insulting/offensive
RD3	Purely disruptive material
RD4	Revision pending "SUPP"
RD5	Other valid deletion
RD6	Non-contentious housekeeping
SUPP	Privacy violations

Table 4.3: Redaction criteria

REDACTED	NUM	%
`content`	13616	72.0%
`summary`	4082	21.6%
`user`	832	0.8%
combinations	377	5.6%
TOTAL	18907	100.0%

Table 4.4: Fields redacted

- RD3: Disruptive acts could affect system operation and removing them from public view might prevent copycat attacks and/or minimize threats to platform security. In practice these were similar to RD2 cases. No evidence of creative, sophisticated, or large scale vulnerabilities was found in our collection.

- RD4+: RD4 is never cited explicitly (this would draw unnecessary attention to pending "suppression" cases). RD5 and RD6 seem to be precautionary "catch alls" that are unfocused and virtually unused in practice.

- SUPPRESSION: This stronger form of deletion is used to remove "non-public identifying information", *i.e.,* edits containing individual's addresses, phone numbers, contact information, or the IP addresses of registered users.

Our review found no evidence to suggest that redaction was occurring for reasons of administrative self interest or censorship.

Measurement: We now present prevalence and impact statistics for redaction actions, summarizing our prior measurement study [144]. A prominent theme is that copyright violation cases (RD1) are uniquely problematic given their damage is not surface level and extrinsic to the typical review process.

During our year long study some 60,000 revisions were redacted (roughly 45 million edits were made to English Wikipedia in the same period [22]). This is a

MO	RD1	RD2	RD3	RD4+	OTH	**SUM**	**SUPP**
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Jul.	88	718	1695	6	158	2665	658
Aug.	167	840	103	51	313	1474	287
Sep.	129	1846	161	18	193	2347	338
Oct.	252	5067	179	19	165	5682	557
Nov.	1087	535	112	14	215	1963	492
Dec.	338	323	152	84	352	1249	487
YEAR	2146	11021	3853	235	1652	18907	5593

Table 4.5: Redaction prevalence by rationale; suppression cases are counted separately due to differing measurement methodology [144]

lower bound on the prevalence of eligible cases, as many instances have likely been reverted deep into the obscurity of revision histories. Regardless, analyzing redaction in terms of raw revision quantity is not ideal. Imagine r_n introduces dangerous content. Subsequent revisions $r_{n+1} \dots r_{n+x}$ are constructive but do not remove the threat. When the damage is discovered all edits back to r_n will need to be redacted because the threat persists through them, collateral damage of an earlier offense.

This underscores why incident level analysis is more intuitive. Such grouping is straightforward for publicly logged redactions, whereby simultaneously deleted edits are assigned the same identifier. Suppressed portions are privately logged so we assume adjacent suppressed revisions are part of the same incident. For the roughly 60k revision redactions some 24,500 incidents were identified (Tab. 4.5). While 89% of these have just one revision, copyright related incidents average 12.5 revisions. Some copyright incidents persist over several *hundred* revisions. Repairing deep damage is problematic given that one must painstakingly parse out violating text in spite of subsequent modifications. In a late 2010 incident a prolific copyright violator was discovered who had significantly altered 23,000+ articles over several years [25]. Several thousand articles were blanked in lieu of trying to identify offending articles and textual fragments therein.

Tab. 4.5 plots the relative frequency of incidents by reason for deletion, with

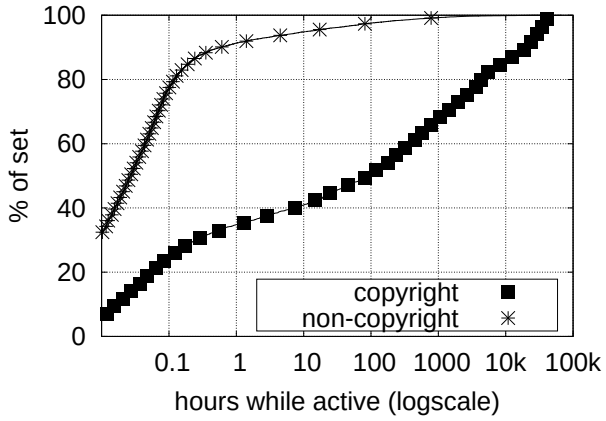


Figure 4.4: Active duration of redacted incidents

insulting/disruptive (61%), privacy cases (23%), and copyright violations (9%) figuring most prominently. Prominence, however, is a poor measure of vulnerability and impact. To this end we consider an incident's *active duration*: the time interval in which the damage was visible in the most recent article version (a metric explored further in Sec. 6.1 and plotted for other damage types in Fig. 6.1). Note that active duration is terminated by a simple revert irregardless of how long it may take an administrator to respond and formally redact the incident. Fig. 4.4 plots the CDF of active duration for copyright incidents against those redacted for other reasons. This confirms the latency of identifying copyright cases. Whereas non-copyright liability/privacy cases are active for only 1.6 minutes at median, copyright ones survive for 3.8 days. The upper quartile of instances have an active lifetime of 110+ days and suggests there might be many instances still live on the encyclopedia.

Related work & moving forward: Our study revealed that nearly 90% of privacy/liability cases are an acute subset of vandalism, exhibiting characteristic properties of the parent class (profanity, disruption, *etc.*). Thus, existing anti-vandal logic along with our improvements of the next chapter are well positioned to revert these cases (although a human is needed for formal redaction). When autonomous

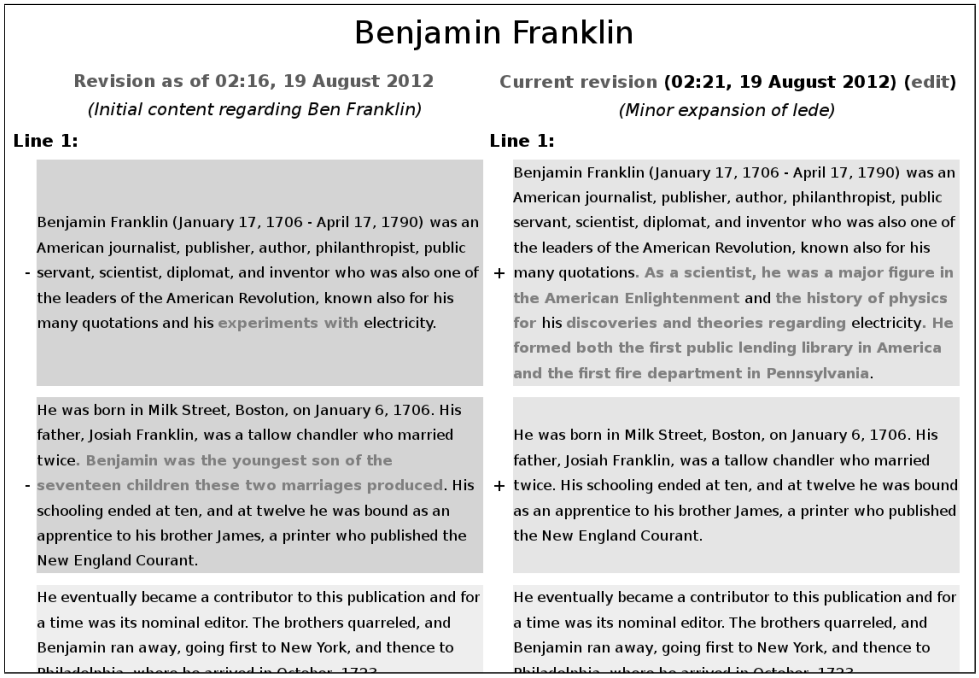


Figure 4.5: Example diff between revisions

methods fall short this is still damage humans can readily identify.

The same cannot be said for the detection of copyright instances, a field for which related literature is extremely sparse. Others do recognize the problem: Cha *et al.* [40] quantifies YouTube's video deletions for copyright reasons, and Google penalizes the search engine ranking of sites that are issued "take down" notices [66]. However, these actions are reactive to the claims of copyright holders. Our work towards proactive discovery in Sec. 5.4.1 and [20] appears academically novel.

4.2 Human Mitigation

Focus now turns to the human mitigation which complements those portions of the problem space not handled autonomously. We begin by describing the "base case" defense model, one that captures how Wikipedia's content security has self-organized using only the core capabilities of the platform (Sec. 4.2.1). This is not a model dictated by software but a fluid ecosystem of users fulfilling security roles at will.

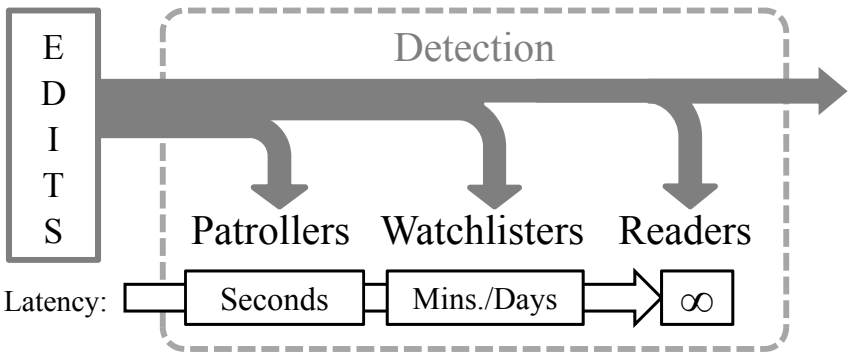


Figure 4.6: Damage detection pipeline for human mitigation

This model is an imperfect one, both in design and practice. We argue that current approaches are the source of redundant and inefficient mitigation work. Moreover, security extensions have done little to address these shortcomings (Sec. 4.2.2).

4.2.1 Base Case Defense Model

The defense ecosystem of Wikipedia is one best described at two granularity. First, we describe how single instances of damage are *discovered*. Then, we discuss how these discoveries *aggregate* into block actions and broader security investigations.

Single instance discovery: Virtually all security these reviews occur by examining the *diff* it produces. Fig. 4.5 shows an example diff, which visually represents all document additions, deletions, and reorganizations. Their use is unremarkable, although there may be social engineering ramifications in how they are interpreted. More interesting is how one selects the revisions they review. We describe this process as a pipeline, visualized in Fig. 4.6:

- PATROLLERS: Assuming an edit is not autonomously blocked, *patrollers* are the first line of human defense. This is a role whereby one performs brute force inspections over a list of *recent changes* without regard for subject matter. These lists often include simple metadata features which patrollers use to

prioritize their reviews, typically focusing on unregistered users or those who do not leave edit summaries. These lists grow/scroll extremely quickly, on the order of several edits per second for English Wikipedia.

- WATCHLISTERS: Users utilize *watchlists* to indicate documents of interest. When an article on one's watchlist is edited they can receive notification in numerous forms. We presume a watchlister has some incentive to ensure these changes are beneficial and therefore serve a reactive security role. Watchlist summaries extend out several days, but it is not unusual for dedicated editors to have watchlists containing 1000+ articles.

- READERS: Damage evading the previous stages is said to be *embedded* on the article. Now, only readers are likely to encounter the damage. A reader may identify but choose not to undo the damage for reasons of apathy or unfamiliarity with the editing system. Anecdotal evidence suggests casual readers are extremely unlikely to fix the problem [31].

When a damaging edit is identified it is removed with a revert action. This revert may itself be audited along the same workflow, although revisions showing this signature (*i.e.,* an edit summary indicating a revert) tend to attract less scrutiny.

Aggregate mitigation: When damage is discovered perpetrators should be issued a warning on their public discussion page. Subsequent warnings escalate in severity as shown in Fig. 4.7. Extensive template based warnings are given to accommodate ignorant users and concerns over editor retention [73]. If these are not heeded a user should file a block request on an administrative noticeboard. Blocks tend to be permanent for registered accounts but temporary for IP-based ones (in case of DHCP). In the base case every stage of the process is manually undertaken.

Damage discovery broader than the account level is less structured. Users may create Sybil [52] or *sock-puppet* accounts to evade blocks. Manual signature detection

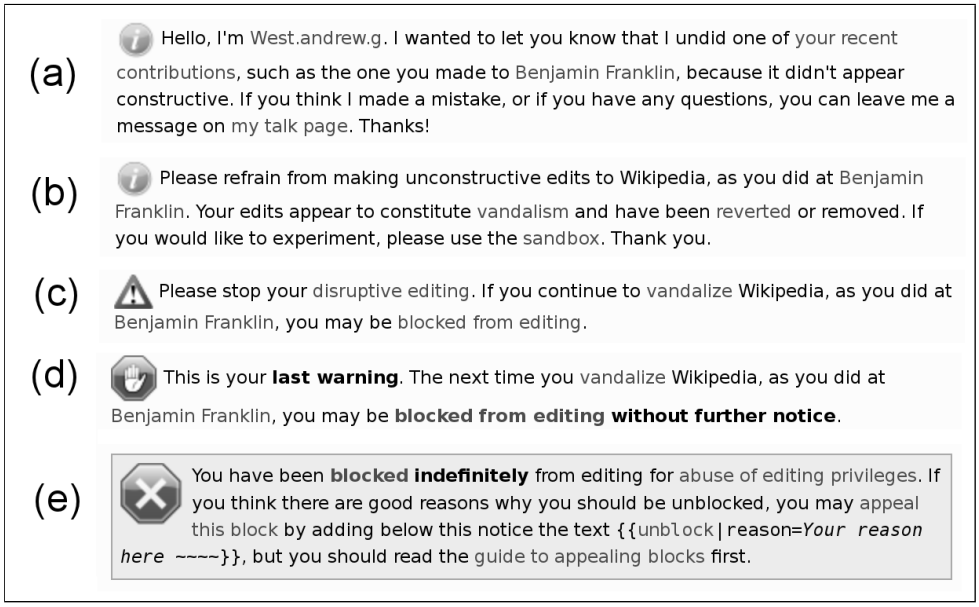


Figure 4.7: Warning hierarchy for damage perpetrators

and probable cause is typically required to access the IP data needed to corroborate these suspicions. When substantiated, single or entire ranges of IP addresses may lose editing and/or account creation privileges (including famously, the Church of Scientology [92] and U.S. Capitol [24]). There is little protection against damaging collectives not confined to a well defined IP range, although a best effort is made to proactively block known proxy and Tor nodes.

All types of damage are mitigated in this fashion. Spam defense adds an administratively controlled URL blacklist and the ability to search by URL.

4.2.2 Inefficiencies and Available Tools

The aforementioned model is not an ideal or efficient one. While we are able to produce quantified metrics that evaluate human response (as done in Sec. 6.1) these are unbounded and relativistic measures that say little about the potential for improvement. Instead, this is a claim we defend by demonstrating logical inefficiencies. The absence of functionality to rectify these shortcomings is influential in the design of

our own human mitigation tool in Chap. 6. While there are software tools extending base platform functionality, these offer only incremental improvements.

Missing functionality: Simple logic combined with our Wikipedia experience allows us to characterize five shortcomings of the human mitigation infrastructure:

1. Redundant review: A single revision can be be needlessly re-reviewed. Poor edits are "explicitly guilty" in that a revert action logs their discovery. However, an "implicit innocence" exists as constructive revisions persist with no annotation of inspection. All active patrollers might review a revision followed by many watchlisters ("Benjamin Franklin" has ≈850 watchers). On popular pages more than 1000 reviews could take place for a single innocent edit. This does not imply a single review can completely vet an edit, but one or two "no action" reviews – if annotated and performed by trusted users – would nearly eliminate the chance the revision is blatant damage.

2. Simultaneous work: A subcase of redundant review occurs when two or more users are reviewing an edit at the same time. This is not uncommon given patroller competition for "low hanging fruit" and can lead to edit conflicts if multiple user's inspections overlap. Such competition can also rush user judgement out of fear someone else will get credit for the discovery/revert.

3. Incomplete coverage: Some edits receive no review whatsoever. With English Wikipedia averaging 1.5 edits/sec. and sustained peak loads of 10+ edits/sec. it is infeasible that any one patroller could exhaustively vet new content [22]. Moreover, decentralized patrollers often apply the same crude prioritization criteria (*e.g.,* focusing on edits by unregistered users), creating non-uniform patrol patterns that are easily gamed. One should also consider there are times where there are no patrollers, allowing even obvious damage to filter to the watchlist level, where some articles have no watchlisters.

4. Naïve prioritization: When a patroller/watchlister decides which edits to review this is done primarily based on the minimal metadata presented alongside edit lists. Meanwhile, there are complex computation mechanisms computing fine grained and multi-variate vandalism probabilities to determine if automatic reversion should take place. The failure to convey these probabilities for human use is clearly a missed opportunity to intelligently route actors and improve capture rates.

5. Compiling human latency: Despite having a relative degree of structure the warning/blocking process is still a manual one. Not only must humans know what action to take (*e.g.,* "readers" who discover damage may be unaware of these mechanisms) but the time they invest in performing them is non-trivial. This would be time better spent reviewing other edits, and the combined latency of these actions extends the lifetime of malicious accounts.

Software editing assistants: In order to improve Wikipedia's mitigation ecosystem several scripts/tools provide functionality beyond that of the base platform (interfacing via the API). Numerous tools exist: Igloo, AutoWikiBrowser, and Twinkle – but most popular and representative among these is Huggle [16, 60]. Traditional patrol involves using a web browser and refreshing the "recent changes" list. Tools like Huggle connect to a live IRC feed of changes and provide an interface to quickly iterate through revision diffs. Damage can be undone in a single click and logic can automatically place the appropriate warning or file a block request.

While this approach improves throughput it only addresses deficiency #5 above. These purely client-side tools lack the centralized or P2P communication needed to eliminate issues #1-3. Moreover, the tools have made only incremental steps towards improved prioritization (#4) by codifying the primitive heuristics used by manual reviewers. Overall, current anti-vandal tools tend to make the act of reviewing quicker for individuals but lack the primitives to coordinate a distributed workforce.

Chapter 5

Improving Computational Detection

The measurement studies of the prior chapter identified shortcomings of existing damage detection approaches. Learning from those we now develop novel techniques that complement or extend that prior work. The goal is a simple one: to classify damaging revisions as accurately as possible. We begin by discussing the metrics used to evaluate effectiveness (Sec. 5.1). Then, methods are described for vandalism (Sec. 5.2) and spam (Sec. 5.3) detection, evaluating them over English Wikipedia. Finally, the portability of these techniques is demonstrated by showing them effective across a diverse set of use cases (Sec. 5.4).

5.1 Evaluation Metrics

The damage classifiers we develop in this chapter are evaluated against labeled corpora of "damage" and "not damage" instances. As such, standard information-retrieval metrics are appropriate for evaluating system performance [61]. Due to class imbalance and other factors of the Wikipedia specific task Potthast [113] and others [68] argue that *precision-recall* (PR) curves are the preferable approach and

we follow this suggestion. Intuitively, *recall* is the portion of the problem space a solution covers (a percentage of damage), where *precision* is how accurate it is over that portion (related to, but not the complement of the *false positive* (FP) rate).[6] As recall increases precision will decrease (*i.e.,* as one wants to identify a greater percentage of damage one must predict more "difficult" cases, and will therefore make more mis-predictions). Looking forward, Fig. 5.4 plots one example. We are interested in two metrics gleamed from a classifier's precision-recall (PR) curve.

First, when considering the autonomous reversion of damage from Wikipedia ("bot operation") one must be sensitive to the number of errors such a system will make. Wikipedia's most prominent anti-vandalism bot [36] has secured community approval to operate at a precision as low as 0.943, corresponding to a 0.25% false positive (FP) rate. Based on this we consider a 0.95 precision to be the threshold for bot operation. That is if a classifier has 0.30 recall at 0.95 precision then 30% of the vandalism problem can be handled without human intervention.

Second, even when the probability/score a classifier generates is not high enough to qualify for automatic reversion, it still has predictive value. To capture this notion over the entirety of the problem space we use the *area under the (precision recall) curve* (PR-AUC) metric. This is a reduction of the entire curve into a single intuitive and quantified measure, whose maximum area is 1.0.

5.2 Vandalism Detection

The first target for computational improvement is the broadest set of damage, vandalism. We establish the corpora used for analysis and evaluation (Sec. 5.2.1) before

[6] As a more formal explanation: Assume an oracle, O, has a set of edits, E, with subsets labeled as damage, E_d, and not-damage, E_n. O gives these edits without labels to a classifier, C, that runs its logic and sorts edits based on prediction confidence. Then, C goes in order (starting with its most confident prediction) broadcasting predictions. After each prediction O can plot two quantities: (1) *Recall*: of all $|E_d|$ edits, the percentage that C has predicted. (2) *Precision*: percentage of the time C has predicted damage and been correct in that prediction.

describing our novel contributions. These focus on reputation (Sec. 5.2.2) and metadata (Sec. 5.2.3) techniques. When this strategy is evaluated in combination with existing techniques its effectiveness is validated given the significant advancement of anti-vandalism benchmarks (Sec. 5.2.4).

5.2.1 Data Sets

Prior to late 2010, corpora for vandalism tended to be extremely small [123], non-representative, and/or had weakly defined notions of "non-vandalism" [142]. Though some of our early feature development is based atop these datasets, our core evaluation uses the PAN-WVC-10 corpus [110] which has become authoritative in the field. The corpus contains ≈32,000 randomly selected English Wikipedia revisions (a 2011 extension adds 10,000 more [111]) with 7% being vandalism, matching research estimates of vandalism prevalence. Tagging was outsourcing to Amazon Mechanical Turk where multiple workers reviewed each edit and only those with strong consensus were included in the final set.

The tool we construct in Sec. 6.3 has also produced 1.1+ million classifications by Wikipedia experts. While not representative for training purposes the breadth of this set does inform observations about vandal behavior.

5.2.2 Reputation Features

Our measurement study analysis (Sec. 4.1.1) of existing reputation mechanisms (Sec. 3.1.3) revealed sparse behavioral histories to be a serious drawback. Towards remedying this we develop the notion of spatial reputation and apply it not just to users, but also to system artifacts.

Gleaning feedback: Reputation algorithms aggregate behavioral observations (*i.e.,* feedback) to compute quantified and predictive values. Adler's work [27, 29] showed

that feedback could be implicitly gleaned based on the survival of individual language/content tokens. This proves unnecessarily fine grained for our analysis, which seeks to aggregate unary observations (*i.e.,* "was this revision damage?"). Producing such feedback is equivalent to the detection of vandalism with *hindsight* [28, 143]and tends to be a refinement on locating previously reverted edits.

A straightforward way to find revert instances is to compute the hash codes of article content. If $\mathsf{hash}(r_n) = \mathsf{hash}(r_{n-2})$ then revision r_{n-1} is an *identity revert* [54]. In [142] we showed that edit summaries could also be parsed to detect reverts with the added benefits of: (1) not having to obtain/hash article content, (2) allowing one to distinguish reverts made for reasons of blatant damage, and (3) trusting the accuracy of those actions. Specialized machinery called *rollback* is available only to trusted users, expressly for expediting non-controversial revert actions. The functionality leaves a standardized edit summary which when cross-checked with user permissions is an efficient and robust means to produce feedback. Prior work has established that such feedback (and therefore reputation) is not terribly latent: at median, revert actions performed by humans come 90 seconds after the reverted edit [142] and are nearly instantaneous for bot reverts.

Spatial reputation: Provided feedback, there are plentiful ways to aggregate those observations into an entity granularity (*i.e.,* editor) reputation [84]. Herein we purposefully select a simple algorithm in order to highlight our contributions in the spatial dimension. As formalized below, every feedback event mapping to an entity results in a penalty for that entity's reputation. The penalty's weight is determined by a time decay function. The overall reputation is the sum of all timed decayed events in the the feedback history.

Such reputations must be interpreted relatively, there is no ceiling on poor behavior. We initialize new users to have zero reputation, although one could imagine seeding new user histories with feedback events to dis-incentivize Sybil attacks (we

capture these notions in other features). Reputation improves only along temporal terms, there is no notion of a "good edit", and no normalization based on edit rate. This prevents ballot stuffing and recognizes that blatant damage is an act committed with intent, not an accidental consequence of performing many edits.

Such methods yield no predictive intelligence for users without feedback history, the "cold start problem". While constructive users may have no feedback events, consistent and long term participation tends to make their reputations irrelevant. Instead, it is the lack of information regarding new users which is the crux of the problem. To counter this we developed the notion of *spatial reputation* [138], an overlay model that makes use of existing reputation algorithms. The idea is to establish spatial *grouping functions* over entities. In a geographical dimension one might imagine functions that given some entity will return: (1) the entity itself, (2) entities in the same city, and (3) entities in the same state. One can compute group reputations by normalizing the reputations of all group members. Then, an aggregate function is applied that can compensate for partial information. If one has much entity specific information then that should be weighted heavily. However, if that proves sparse, then one can default to broader groupings for predictive intelligence. We utilize Subjective Logic [83] atop standard machine learning techniques.

Spatial associations may be defined along geography, graph topologies, membership functions, or anywhere a distance function can be defined. Of course, these groupings/dimensions must capture behavioral tendencies in order to be meaningful. Fortunately, the sociological principal of *homophily* – the notion that those who share characteristics tend to associate – has been found to hold true in a tremendous number of settings [96]. This is not a guarantee that every member shares group properties, only a probabilistic likelihood that leveraging spatial information will outperform random chance. In many ways the technique is a statistical formalization of criminal profiling tactics, also inheriting its ethical contentions. Prior work [138] expands on grouping strategies to prevent gamesmanship.

Formalizing reputation: A feedback history, $H = \{f_1, f_2 \dots f_n\}$ contains feedback events, $f_x = (e_x, t_{fback})$, where e_x is the set of entities to which the feedback maps and t_{fback} is the timestamp of the observed behavior. Note that only a single class of feedback is supported. From this we define two functions:

- $\boldsymbol{fback\_hist(g, H)}$ – given a set of entities, g, and a feedback history, H, the function returns all feedback timestamps $f_x.t_{fback}$ where $f_x \in H$ and $\exists y \in g$ where $y \in f_x.e_x$. Informally, we return the timestamps of all feedback events in which some member of g was involved.
- $\boldsymbol{decay(t, h)}$ – calculate time decay. We use $decay(t) = 2^{-\Delta t/h}$ where $\Delta t = (t_{now} - t)$ and h is the half-life.

Now assume one wishes to calculate the reputation of entity α in the context of a spatial grouping function G such that $G(\alpha) = g$ where g is a set of entities in which α is a member. We can calculate the reputation of this spatial grouping to be:

$$\boldsymbol{rep(g = G(\alpha), h, H)} = \sum_{\substack{t_{fback} \in \\ fback\_hist(g,H)}} \frac{decay(t_{fback}, h)}{|g|} \tag{5.1}$$

All $rep()$ values calculated using the same $G()$ and H are strictly comparable (*i.e.,* can be relatively interpreted). It is expected one will define multiple grouping functions $G_1, G_2 \dots G_n$, thereby enabling the calculation of n reputation values for an entity α, namely $rep(G_1(\alpha), \cdot) \dots rep(G_n(\alpha), \cdot)$ for aggregation purposes.

Application to users: When assigning reputation to Wikipedia users we consider two groupings: (1) the user his/herself (*i.e.,* $G(\alpha) = g = \alpha$, and $|g| = 1$) and (2) the geographic country to which that user's IP address geolocates. The existence of habitual offenders is an underlying assumption of all reputation systems and one previously attempted on Wikipedia [30]. Our single entity *user reputations* perform similarly well and need not be discussed at length.

RANK	COUNTRY	EDITS	%-VAND
1	Italy	116,659	2.85%
2	France	116,201	3.46%
3	Germany	227,037	3.46%
...	...	...	...
12	Canada	989,857	11.35%
13	United States	7,648,075	11.63%
14	Australia	670,483	12.08%

Table 5.1: Normalized vandalism occurence rate by geolocated country of unregistered editors; only countries w/100k+ edits in corpus of [142] are included. Australia is *not* 12% of the total vandalism problem; 12% of edits from Australia are vandalism.

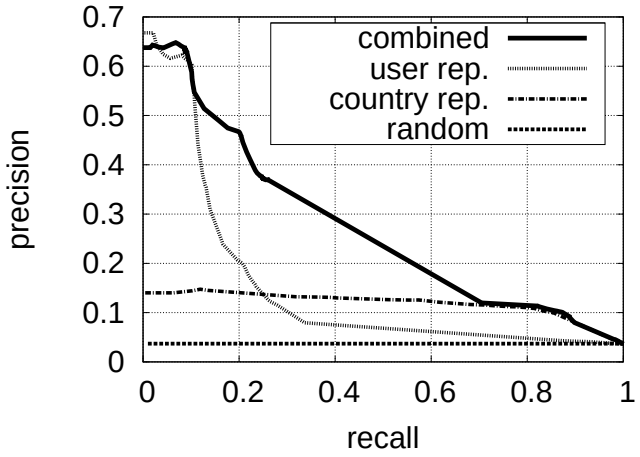


Figure 5.1: Precision-recall graph showing combined and isolated performance of component reputations for Wikipedia users

However, we find roughly 40% of the time a new/unregistered user vandalizes they had $rep(\cdot) = 0$. In these cases spatial expansion into the *user-country* dimension proves helpful (registered users, a small fraction of the vandalism problem, are mapped to their own "country"). Tab. 5.1 shows editors from certain countries tend to show a propensity towards damaging behavior. For example, an Australian edit is 4× more likely to be vandalism than one from Italy. Regardless the social reasoning the statistical gap is significant.

ARTICLE	#VAND	ARTICLE	#VAND
George W. Bush	6546	World War II	1886
Wikipedia	5589	Jesus Christ	1681
Adolf Hitler	2612	George Washington	1648
United States	2161	Bill Clinton	1594

Table 5.2: Most vandalized Wikipedia articles, pre-2010 [142]

While both user and user-country reputation are useful metrics it is their combination to assess a single revision which is most interesting, with Fig. 5.1 displaying the results. For 25% of the recall space it is user specific reputation which is most effective predictor, yet the remaining 75% is better handled by the broader metric. When machine learning with confidence coefficients is brought to bear an aggregate metric ("combined") combines this evidence and significantly outperforms the component measures (using PR-AUC).

Subsequent investigation has revealed intermediate granularity groupings based on IP subnets (*e.g.,* `/24` CIDRs) would also be beneficial. This would allow the capture of DHCP hopping vandals and the educational institutions known to be a significant part of the vandalism problem.

Application to artifacts: Assessing article reputation is an approach not seen previously in literature, and we consider not just articles directly but extend this spatially to include the topical categories they are members of.

Certain topics are inherently controversial and see frequent vandalism (*e.g.,* religious and ethical issues). Others incur temporally variable abuse (*e.g.*, political candidates near elections). *Article reputation* is well equipped to handle both cases. This succeeds because of the non-uniform distribution of damage that Tab. 5.2 makes apparent.[7] When computed, article reputation is 4× higher on average for vandalism

[7]These should not be interpreted as the "most controversial" articles. Administrators can *protect* articles to prevent them from being edited by users of varying permissions, limiting their damage. In practice, such protection is rare and is a practical security tradeoff discussed further in Sec. 7.1.3.

CATEGORY (w/100+ pages)	**PGs**	**VANDs/PG**
World Music Award Winners	125	162.27
Characters of Les Misérables	135	146.88
Former British Colonies	145	141.51
Congressional Medal Recipients	161	121.98

Table 5.3: Most vandalized Wikipedia categories (normalized) [142]

edits than non-damaging ones. Additionally, nearly 85% of vandalism instances have non-zero article reputations compared to 45% for innocent edits [142].

Leveraging topical *category reputation* as a spatial grouping over articles also proves helpful (see the least reputable categories in Tab. 5.3). Identifying 250,000 topical categories, we calculate the reputations for all of an article's category memberships and select the worst reputation as the feature value. Of vandalism edits, 97% have non-zero reputations (compared to 85% in the article case), again demonstrating larger spatial contexts generate more evidence [142]. Manual inspection shows categories most useful when an article experiences a sudden rise to prominence. For example, breaking news stories are often added to a poorly reputed category (*e.g.,* "Deaths in 2013") and this addition foreshadows coming damage that has not yet been experienced at the article level.

5.2.3 Metadata Features

Our measurement study of existing metadata strategies (Sec. 4.1.1) found them underdeveloped, not optimized for revision granularity, and primarily content dependent. Here we develop over two dozen metadata features that address these shortcomings. Three themes are emphasized: (1) spatio-temporal patterns, (2) participatory dynamics, and (3) artifact maturation. In the interest of brevity we concentrate on describing these themes rather than the rote justification of individual features (as done previously [28, 142, 143]). A comprehensive listing of features is found in Tab. 5.4 (the "M" entries) and Tab. 5.5.

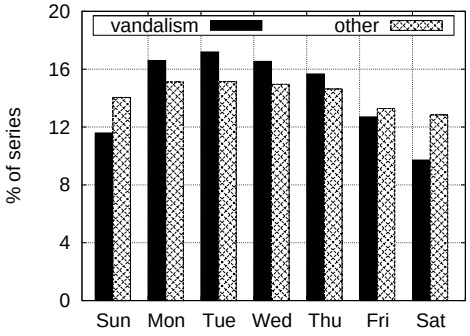


Figure 5.2: Vandalism prevalence by day-of-week

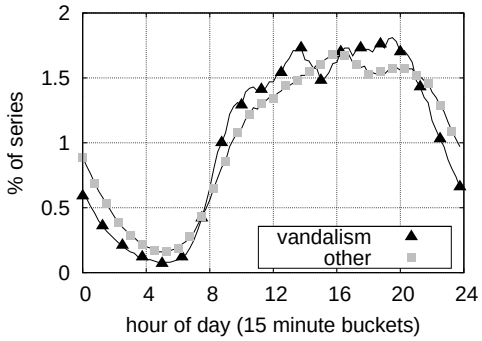


Figure 5.3: Vandalism prevalence by hour-of-day

Spatio-temporal metadata: Just as reputations of the previous section dynamically interpreted spatio-temporal aspects, one can identify static metadata features of the same type. This is motivated by Hao *et al.* [75] who showed spatial (*e.g.,* artifact sizes) and temporal (*e.g.,* time-of-day) notions effective against email spam. Many of these features are also remarkable in their simplicity and predictiveness while remaining content agnostic. We highlight several such features:

- Time-of-day and day-of-week: Using geolocation we determine the GMT offset of an editor and determine the local *time-of-day* (Fig. 5.2) and *day-of-week* (Fig. 5.3) when an edit was committed. Besides following circadian patterns [154] we recognize that most vandalism happens on weekdays during "business hours". We attribute this to primary/secondary school students using institutional access to vandalize [145].

- Size ratio: When comparing the size of a new artifact to the previous version, vandalism edits overwhelmingly come in two forms: (1) massive content removal in the form of section/article blanking, and (2) minor content additions of less than 100 bytes. There is little incentive for vandals to spend considerable effort on their damage given the likelihood it will quickly be undone.

FEATURE	CLS	SRC	DESCRIPTION
IS_REGISTERED	M	[V,A,W]	Whether editor is anonymous/registered (boolean)
COMMENT_LENGTH	M	[V,A,W]	Length (in chars) of revision comment left
SIZE_CHANGE	M	[V,A,W]	Size difference between prev. and current versions
TIME_SINCE_PAGE	M	[A,W]	Time since article (of edit) last modified
TIME_OF_DAY	M	[A,W]	Time when edit made (UTC, or local w/geolocation)
DAY_OF_WEEK	M	[W]	Local day-of-week when edit made, per geolocation
TIME_SINCE_REG	M	[W]	Time since editor's first Wikipedia edit
TIME_SINCE_VAND	M	[W]	Time since editor last caught vandalizing
REP_EDITOR	R	[W]	Reputation for editor via behavior history
REP_COUNTRY	R	[W]	Reputation for geographical region (editor groups)
REP_ARTICLE	R	[W]	Reputation for article (on which edit was made)
REP_CATEGORY	R	[W]	Reputation for topical category (article groups)
WIKITRUST	R	[A]	Multiple features speaking to author's content-persistence
DIGIT_RATIO	L	[V]	Ratio of numerical chars. to all chars.
ALPHANUM_RATIO	L	[V]	Ratio of alpha-numeric chars. to all chars.
UPPER_RATIO	L	[V]	Ratio of uppercase chars. to all chars.
UPPER_RATIO_OLD	L	[V]	Ratio of uppercase chars. to lowercase chars.
LONG_CHAR_SEQ	L	[V]	Length of longest consecutive sequence of single char.
LONG_WORD	L	[V]	Length of longest token
NEW_TERM_FREQ	L	[V]	Average relative frequency of inserted words
COMPRESS_LZW	L	[V]	Compression rate of inserted text, per LZW
CHAR_DIST	L	[V]	Kullback-Leibler divergence of char. distribution
VULGARITY	S	[V]	Freq./impact of vulgar and offensive words
PRONOUNS	S	[V]	Freq./impact of first and second person pronouns
BIASED_WORDS	S	[V]	Freq./impact of colloquial words w/high bias
SEXUAL_WORDS	S	[V]	Freq./impact of non-vulgar sex-related words
MISC_BAD_WORDS	S	[V]	Freq./impact of miscellaneous typos/colloquialisms
ALL_BAD_WORDS	S	[V]	Freq./impact of previous five factors in combination
GOOD_WORDS	S	[V]	Freq./impact of "good words"; wiki-syntax elements
COMM_REVERT	S	[A]	Is rev. comment indicative of a revert? (boolean)

Table 5.4: Listing of features in combined vandalism evaluation [28], by class: M=metadata, R=reputation, L=lexical, S=semantic. Sourcing column is: V(elasco)=[130], A(dler)=[30], W(est)=[142].

- EDIT SUMMARY LENGTH: Whether due to laziness or unfamiliarity with community conventions roughly 40% of vandalism leaves no summary. Comments left with vandalism are 43% the size of those for non-damaging edits.

Participatory dynamics and artifact maturation: Metadata features capturing entity age, interaction density, and evolution can contextualize and complement the behavioral reputations we previously described. Tab. 5.5 contains many of these

FEATURE	DESCRIPTION
USR_IS_BOT	Whether the editor has the "bot" flag (*i.e.,* non-human user)
USR_BLK_BEFORE	Whether the editor has been blocked at any point in the past
USR_PG_SIZE	Size, in bytes, of the editor's "user talk" page
USR_EDITS_*	Editor's revisions in last, $t \in \{hour, day, week, month, ever\}$
USR_EDITS_DENSE	Normalizing USR_EDITS_EVER by account age
USR_PREV_IP	Whether the previous editor of the article was IP/anonymous
USR_PREV_SAME	Whether the previous article editor is same as current editor
ART_AGE	Time, in seconds, since the article was created
ART_EDITS_*	Article revisions in last, $t \in \{hour, day, week, month, ever\}$
ART_EDITS_DENSE	Normalizing ART_EDITS_EVER by ART_AGE
ART_POPULARITY	Number of views the article has recently received (Sec. 6.1)
ART_SIZE	Size, in bytes, of article after the edit under inspection was made
ART_CHURN_CHARS	Quantity of characters added *or* removed by edit
ART_CHURN_BLKS	Quantity of non-adjacent text blocks modified by edit
COMM_HAS_SEC	Whether the comment indicates the edit was "section specific"
COMM_LEN_NO_SEC	Length, in chars., of the comment w/o auto-added section header

Table 5.5: Additional anti-vandalism features introduced in [143]

features (introduced primarily in [143]) and we discuss they distinctions they enable:

- User dynamics: It is intuitive that registered, long term, and active contributors are rarely vandals. Indeed, the median age of registered accounts who vandalize is just 1.6 hours. Unregistered users are frequent perpetrators, but IP addresses showing consistent participation and favorable reputation can be shown less scrutiny. Erratic and bursty participation (captured by usage histograms) from unregistered users is typical of shared use settings (*e.g.,* school computer labs) and these IP based accounts often have block histories and frequent talk page warnings. Isolated mis-behavior in the distant past needs treated more cautiously due to possible DHCP considerations.

- Article evolution: Features gleaned from articles benefit from the persistence of those artifacts. Older and often edited articles tend to have stable content. This makes revisions with large content churn suspicious, especially by new editors. Popular articles are often damaged but this is unsurprising given that every article viewer is a potential vandal. Consider that one's arrival at obscure articles often implies biased interest, as articles for secondary

schools are among the most vandalized after traffic normalization.

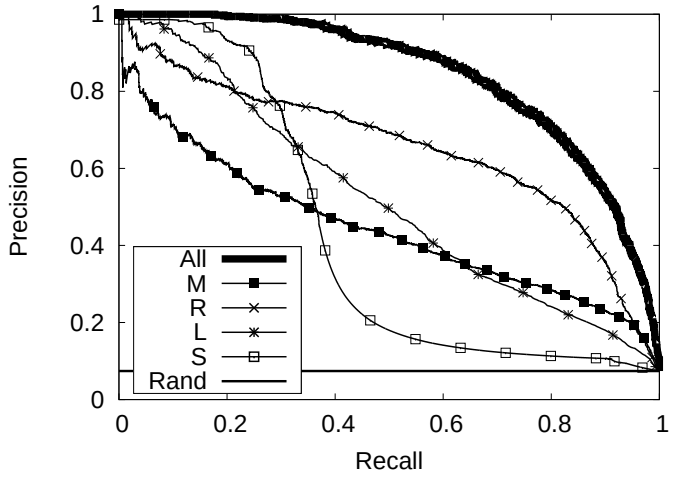
5.2.4 Evaluation

Learning/evaluation methodology: All results of this section were produced using 10-fold cross validation over the PAN-WVC-10 corpus discussed in Sec. 5.2.1. All features are calculated using only prior evidence, with aggregate features drawing from all of Wikipedia history (*i.e.,* features use evidence external to the corpus). Models are trained using the Weka implementation of the Alternating Decision Tree algorithm [58, 74] with Random Forests and boosting optimization. We prefer a decision tree approach because of: (1) strong support for nominal and missing features, (2) human readable output allows for easy auditing of models, and (3) terse models scale well to Wikipedia's high throughput operation.

Combined approach: To evaluate our techniques we combine our novel features with those from existing approaches (particularly those built on language properties) and evaluate them over a single corpus.[8] This enables: (1) investigation into the information gain of individual features, (2) a relative comparison of feature subset performance (*i.e.,* language vs. reputation vs. metadata), and (3) quantification of over arching anti-vandalism benchmarks.

The PAN-WVC-10 corpus of Potthast and an associated anti-vandalism competition [111, 113] spurred our collaboration with other researchers in this space. Aiming to compile a feature set representative of anti-vandalism strategy we combine our reputation/metadata approach [142] with the fine grained content persistence measures of Adler *et al.* [27, 29, 30] and the language approach of Mola-Velasco [99, 130]. The concatenated feature vector has over 60 elements, summarized in Tab. 5.4 (note that

[8]Meaningfully comparing performance against prior literature is difficult. A standardized corpus is relatively recent, with much related work pre-dating its existence. Even still, some practitioners [36] reject this corpus because it was not labeled by Wikipedia experts. Regardless the training/test set, other researchers tend to report only single points along the precision-recall curve and not more complete plots or AUC measures.



Features	PR-AUC
(S)emantic	0.42386
(M)etadata	0.43582
(L)exical	0.51586
(R)eputation	0.59977
M+L	0.68513
L+S	0.73232
M+L+S	0.76124
M+L+R	0.76271
All	0.81829

Figure 5.4: Precision-recall curve and AUC metrics for the combined anti-vandalism effort [28] and subsets thereof

some descriptions wrap multiple features). We group these signals into 4 categorical subsets (M, R, L, S) as follows:

- Metadata (M): Features that do not require inspection of article content, derived primarily from our spatio-temporal efforts of Sec. 5.2.3.
- Reputation (R): Aggregate quantification of behavioral histories. Combines our reputations of Sec. 5.2.2 with the fine grained ones of Adler.
- Lexical (L): Features derived from surface level properties of edit content.
- Semantic (S): These are not pure Bayesian probabilities but static token lists derived from offline Bayesian analysis. The most indicative unigrams have been been manually reviewed/confirmed and categorized.

Combined performance: As visualized in Fig. 5.4 and described further in [28] we see the aggregate approach significantly outperforms component techniques. The overall AUC=0.818 is an anti-vandalism benchmark which as of this writing remains the highest produced to our knowledge. This is by no means a performance ceiling.

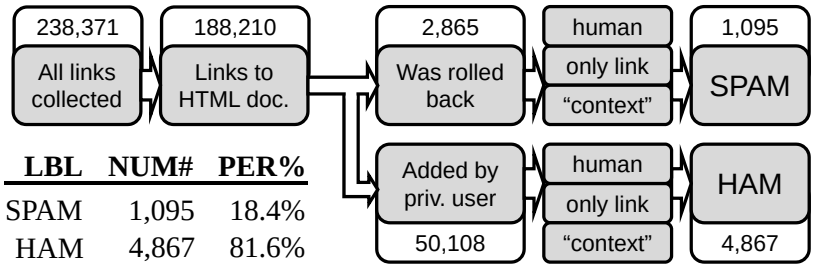
#	Metadata	Reputation	Lexical	Semantic
1	IS_REGISTERED	WIKITRUST	LONG_CHAR_SEQ	VULGARITY
2	TIME_SINCE_REG	REP_EDITOR	NEW_TERM_FREQ	ALL_BAD_WORDS
3	TIME_SINCE_VAND	REP_ARTICLE	UPPER_RATIO_OLD	SEXUAL_WORDS
4	SIZE_CHANGE	REP_COUNTRY	ALPHANUM_RATIO	MISC_BAD_WORDS

Table 5.6: Ranking anti-vandal features by information gain

We have no doubt that encoding additional features would incrementally improve this statistic. Instead, it validates that different approaches capture independent portions of the problem space. At 0.95 precision recall is 38%. This means that our classifier could handle roughly two-fifths of the vandalism problem in a purely autonomous fashion. Conversely, this is also an indication of the ongoing need for human mitigation efforts over large parts of the problem space.

Semantic features actually perform the *worst* overall of any subset (AUC = 0.424) yet they also drive aggregate performance at high precision. A steep performance decline occurs at 30%-40% recall and we attribute this to limited dictionary depth. While certain tokens are very indicative of vandalism, absent such vocabulary there is little predictive capability. Other subsets have more linear dropoffs as recall increases, suggesting their usefulness in routing human efforts beyond those portions handled autonomously. This is a tactic pursued in Sec. 6.3.

Features can also be assessed individually using the information gain metric [61], as Tab. 5.6 displays. This provides insight about vandal behavior and serves as a reference for a live implementation where feature effectiveness must be considered against bandwidth/computational cost. These signals are also a starting point when considering feature development for other damage types, as we now transition to detecting link spam behaviors.



LBL	NUM#	PER%
SPAM	1,095	18.4%
HAM	4,867	81.6%

Figure 5.5: Constructing a corpus of link spam incidents

5.3 Link Spam Detection

Paralleling the organization of our anti-vandalism efforts, we begin by describing the corpus used for link spam detection (Sec. 5.3.1). Features are then developed by focusing on the reuse/extension of metadata and reputation approaches (Sec. 5.3.2). These are then composed and evaluated (Sec. 5.3.3), creating the first anti-spam classifier for wikis, Wikipedia, and open collaboration environments.

5.3.1 Data Sets

Being the first to study link spam behaviors requires us to amass the first link spam corpus, a process described in greater detail in [137, 139]. Summarily, Sec. 5.2.2 described how hindsight could be used to locate blatant damage flagged by Wikipedia experts. The link spam subset is extracted from this by identifying revisions where: (1) exactly one link was added and (2) manual review reveals the link and its immediate context to be the only modifications. Under such criteria we know an expert's decision to revert the revision speaks directly to the inappropriateness of that one link. We also identify a complementary set of *ham* link additions that are constructive in nature. Given that we trust experts to label poor additions, by extension, we should trust the links these users contribute. For consistency these edits are subjected to the same "one link added" and "immediate context" filters.

This tagging technique is biasing, as there are no ham edits contributed by unregistered users. Especial care is taken to ensure that these biases do not become encoded as anti-spam features: We encode no descriptors involving the account that contributed a link. While this criterion makes the corpus non-representative, it permits the construction of a sizable and confidently tagged set. Practically speaking, our research has amassed multiple spam sets [137, 139] but it is a corpus collected over ≈2 months in early 2011 and summarized in Fig. 5.5 which is used for evaluation herein [137]. Uniquely, this collection was built in a live fashion, obtaining the source documents at the URLs being linked.

Our measurement study showed spam instances to be nuanced and ambiguous. The relative latency of spam reverts (ahead in Fig. 6.1) suggests they are made by watchlisters with subject expertise. Thus, attempts to validate our corpus have proven challenging but informative. We had experienced Wikipedia users blindly tag corpus edits/links along a 3-point or 5-point scale. While agreement was strong over ham portions there was a significant conservative bias for spam as users often applied uncertain or intermediate tags (but almost never leaned towards benign labels). In the end only about 25% of gleaned spam portions were consistently and *definitively* labeled as such offline. We argue that focusing only on this most egregious subset would oversimplify the detection task and inflate performance measures. Instead, we contend our implicit method generates a *deep* two class set that leverages the subject expertise of *all* trusted Wikipedia users.

While confident in those revisions the corpus has labeled, these say little about our coverage of the problem space. One drawback of the methodology is that non-trivial portions of link spam might escape identification. Compared to dedicated anti-damage patrollers, subject experts are less concerned/aware of revert semantics (*i.e.,* the machinery we use to identify blatant damage). Only ≈1% of the ≈125,000 links added each month are undone in a manner consistent with blatant damage, while upwards of 10% are undone by softer mechanisms. However, we have no evidence that

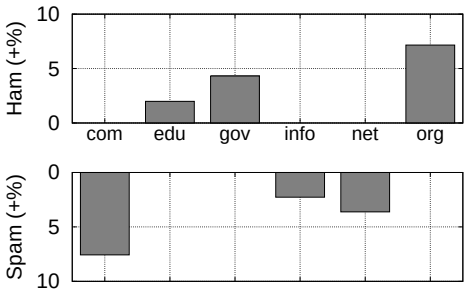


Figure 5.6: Link spam distribution by top-level domain

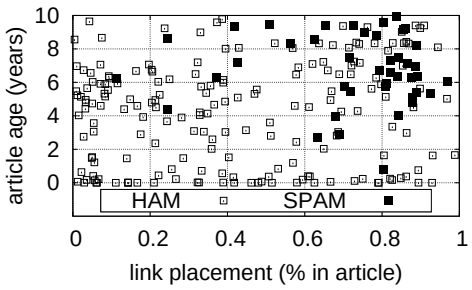


Figure 5.7: Spam distribution by article age × link position

there is bias in those revisions that do go unlabeled (aside from those we introduced). We are primarily unable to reason about the *prevalence* of the spam problem. While such statistics are desirable, our prior corpus validation attempts casts doubt on the use of dedicated taggers to produce them. If revisions were outsourced to a service like Mechanical Turk, we would expect considerable disagreement and very few spam labels due to unfamiliarity with Wikipedia policy. Overcoming these challenges to create a complete corpus for link spam, or any other subtly damaging behaviors, remains an open research challenge.

5.3.2 Anti-spam Features

We organize anti-spam feature discussion into four categories, first concentrating on features internal to the platform with: (1) simple metadata and (2) aggregate features and reputations. Then we leave Wikipedia to: (3) analyze HTML documents at the URL destination and (4) utilize data from third party URL information services. Fig. 5.7 is a comprehensive listing/description of all anti-spam features and also indicates the weight of each feature in our final classifier.

FEATURE	RNK	DESCRIPTION
URL_TLD	••	Top-level domain of the URL (*e.g.,* `*.com` or `*.edu`)
URL_LEN	•	Length (in characters) of the URL being added
URL_IS_DOMAIN	•	If the URL points to a broad domain/folder or specific file
URL_SUBDOMAINS	•	Quantity of subdomains in the URL (*i.e.,* `sub.example.com` = 3)
LINK_IS_CITE	••••	If the link was added per a special reference/citation format
LINK_PLACEMENT	••	Where in the article the link was added (function of article length)
LINK_TEXT_LEN	•	Length (in characters) of the hypertext description of added link
LINK_DISCUSSED	•	Whether the link/URL is found on the article's discussion page
ART_TS_CREATION	••••	Age of the article where link was added (*i.e.,* time-since creation)
ART_REPUTATION	•••	Historical, time-decayed measure of controversy on article [136]
ART_REFERENCES	•	Quantity of citations/references in the article of link addition
ART_LENGTH	-	Length of the Wikipedia article to which the link was added
ART_POPULARITY_*	→	Article visitors in last $t \in \{hour, day, week, month, 6\text{-}mos.\}$ [13]
ART_EDITS_TIME_*	→	Article edits in last $t \in \{hour, day, week, month, 6\text{-}mos.\}$
URL_ADDS_TIME_*	→	Links to URL added in last $t \in \{hour, day, week, month, 6\text{-}mos.\}$
DOM_ADDS_TIME_*	→	Links to domain added in last $t \in \{hour, day, week, month, 6\text{-}mos.\}$
URL_REPUTATION	•••	Historical, time-decayed measure of spam-iness for added URL
URL_DIVERSITY	•••	Of all historical URL links, the % added by the current editor
DOM_REPUTATION	••	Historical, time-decayed measure of spam-iness for added domain
DOM_DIVERSITY	••	Of all historical domain linkings, the % added by the current editor
META_COMM_LEN	••••	Length (in characters) of the revision summary
META_TIME_DAY	•••	Time-of-day when the link was added (UTC locale)
META_DAY_WEEK	-	Day-of-week when the link was added (UTC locale)
SITE_PROFANE	••	Measure of the prevalence of profane language on the landing site
SITE_NUM_IMGS	••	Quantity of images displayed on the landing site
SITE_SIZE	••	Size (in bytes) of the textual content on the landing site
SITE_COMPRESS	••	Ratio of raw content-size to compressed size
SITE_TITLE_LEN	•	Length of the HTML title, in chars. (*i.e.,* `<title>...</title>`)
SITE_NUM_META	•	Quantity of HTML `<meta keywords="`$w_1, w_2, \ldots, w_n$`">` on site
SITE_VOCAB_LEN	•	Average word length of visible textual content on the landing site
SITE_COMMERCIAL	-	Measure of the commercial intent of the landing site
SITE_RELEVANT	-	If landing site is topic-similar to Wikipedia article of addition
ALEXA_BACKLINKS	••••	Quantity of incoming links to landing site, per the crawling by [2]
ALEXA_DELTAS	••••	Meta-feature speaking to site's historical traffic patterns, per [2]
ALEXA_ADULT	•••	If the URL contains adult content, per [2]
ALEXA_SPEED	••	Load time of landing site, as a percentile of all sites, per [2]
ALEXA_AGE	••	Time that the landing site has been online, per the crawling by [2]
ALEXA_CONTINENT	•	Continent to which the `whois` registration of site maps, per [2]
GOOG_MALWARE	-	If URL is active on the Google Safe Browsing "malware" list
GOOG_PHISHING	-	If URL is active on the Google Safe Browsing "phishing" list

Table 5.7: Comprehensive listing of anti-spam features organized by data source. Feature rank/importance was calculated by performing a greedy step-wise comparison over feature subsets [87, 151]. More bullets indicate greater weight in the final classifier. For brevity, rank is omitted for features having multiple variations (indicated by "→").

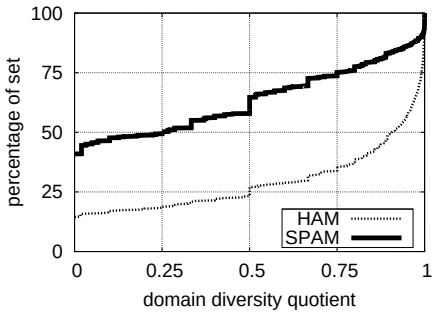


Figure 5.8: CDF for contributor diversity in domain history

FEATURE	UNIT	HAM	SPAM
URL_LEN	chars.	64	38
LINK_PLACEMENT	% art.	41	73
LINK_TEXT_LEN	chars.	26	24
ART_TS_CREATION	mos.	146	192
URL_IS_DOMAIN	bool.	6.3%	37.5%
LINK_IS_CITE	bool.	53.9%	8.3%
LINK_DISCUSSED	bool.	4.5%	2.4%

Table 5.8: Comparing metadata features for link additions; non-percentages presented at median

On-wiki metadata: In addition to reusing some of the most effective features from the anti-vandalism task our metadata set considers the surface properties of the link and how it is presented on the article:

- Reusing Vandalism Features: Similar to our vandalism findings, there is a tendency for naïve spammers not to use edit comments. Additionally, older and often edited content has often matured to a point that these articles are less receptive to new links (Fig. 5.7 and Tab. 5.8).
- URL Properties: Spam URLs are 1.7× shorter and 30% more likely to point to domains (Tab. 5.8). The top-level domain (TLD) of constructive links is skewed towards those with greater administrative control (Fig. 5.6).
- Link Presentation: Citation spamming (using a URL as a reference for some fact) is 6.5× less likely than with general purpose links (Tab. 5.8). Most spam is appended to explicit "external links" sections (by convention), one reason it tends to appear towards the end of articles.

On-wiki reputation and aggregates: The wiki history of a web property is one of the best indicators of its quality. Given the atomic nature of hyperlinks it is straightforward to monitor link survival. The persistence (or lack thereof) for a

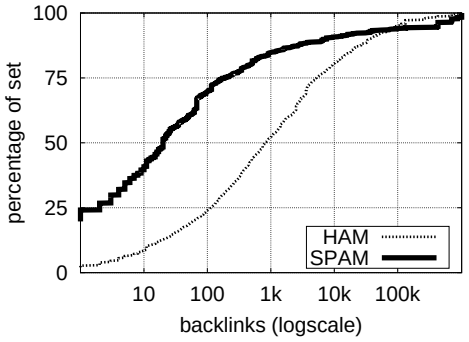


Figure 5.9: CDF for spam/ham landing site backlink quantity

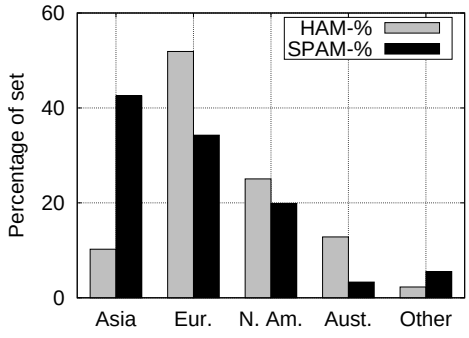
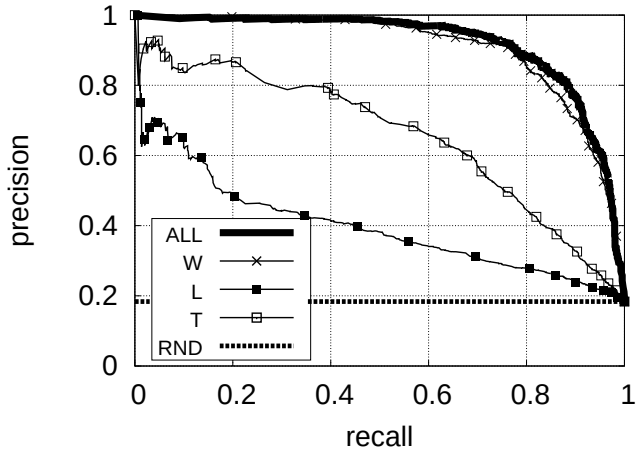


Figure 5.10: Spam distribution by continent of landing site host

URL serves as feedback for a reputation algorithm that can be spatially extended to domain granularity. These reputations can be contextualized via histograms over addition dynamics. Bursty additions for previously unseen domains are often indicative of spam. Accordingly, long term addition history is indicative of link quality: ham domains have 5× the 6-month quantity of spam ones. Encoding editor-link diversity, the CDF of Fig. 5.8 shows 40% of spam links are added by an editor who is responsible for *all* recent links to that domain (versus just 15% for ham).

Landing site analysis: Using lexical language properties (some drawn directly from related work [46, 103]) we attempt to capture commercial intention, inappropriate content, and search engine optimization on landing sites. None of these resulted in particularly impressive information gain and further reinforce the diversity of spam content, the lack of commercial intention, and an absence of SEO strategies.

Third party reputation and metadata: Internet scale measurements are useful when assessing a URL/domain, and we utilize the Alexa API [2] to provide this broad perspective. The quantity of backlinks (*i.e.,* inbound links) a site has, a measure of global reputation, is the most heavily weighted feature in our classifier. At median a ham site has ≈850 backlinks compared to just 20 for spam links (Fig. 5.9). Traffic



Features	PR-AUC
(W)ikipedia	0.909
(L)anding site	0.399
(T)hird party	0.656
W+L	0.902
W+T	0.915
L+T	0.667
ALL	0.917

Figure 5.11: Anti-spam precision-recall curve; feature subsets include: (W)ikipedia, (L)anding site, and (T)hird-party (see Tab. 5.7)

trends function similarly and feature `ALEXA_DELTAS` is the output of a lower order classifier wrapping over 50 data points regarding viewership statistics. Even the continent which hosts a website can be indicative as Fig. 5.10 demonstrates.

5.3.3 Evaluation

Like our anti-vandalism work the anti-spam model is built using ADTrees and evaluated using 10-fold cross validation. The PR curve of Fig. 5.11 best summarizes classifier performance. The complete feature set achieves a 0.917 PR-AUC (versus 0.82 for vandalism). At 0.95 precision recall is ≈67%, indicating two-thirds of spam can be handled autonomously. Given the prior lack of preventative anti-spam infrastructure on Wikipedia a live implementation of this technique would serve tremendous utility. The "on Wikipedia" feature subset drives classifier performance (PR-AUC=0.909) almost to the exclusion of other sets, yet these signals are easily manipulated. Gamesmanship of third party signals would require considerable effort (*e.g.,* expensive TLDs, spamming to amass backlinks, *etc.*) and more heavily weighting these would improve robustness albeit with performance costs [137].

5.4 Broader Applications

Previous sections have made clear the capabilities of reputation and metadata features with respect to English Wikipedia and the anti-vandalism/spam tasks. Now through a series of brief case studies their portability is demonstrated across additional damage types (Sec. 5.4.1), natural languages (Sec. 5.4.2), and content formats (Sec. 5.4.3). Since our original description of these methods we have also observed their reuse in third party open collaboration security research, including vandalism detection in mapping applications [48, 101].

5.4.1 Copyright Violations

In Sec. 4.1.3 we described a class of damage that legally endangers Wikipedia and found copyright violations particularly problematic. Textual copyright violations are usually the result of copy-pasting protected content found elsewhere on the Internet (often promotional) into Wikipedia articles. Since the damage is not surface level, humans are particularly poor at mitigating it, with our work suggesting copyright violations survive orders of magnitude longer than most damage [144] (Fig. 4.4).

This motivated research into autonomous discovery mechanisms [20]. Anti-plagiarism algorithms leveraged over Internet crawled documents form the core of our approach [8]. However, metadata (over participatory dynamics) and reputation (capturing prior involvement in copyright cases) features are also being built from a user labeled corpus of incidents. Similar to our findings for other damage types, signals that speak to community inexperience prove particularly helpful, as do those regarding atypical content evolution (*e.g.,* monolithic content additions are suspicious). Work is underway to mechanize these techniques and secure the community approval needed to leverage them in a live fashion [20].

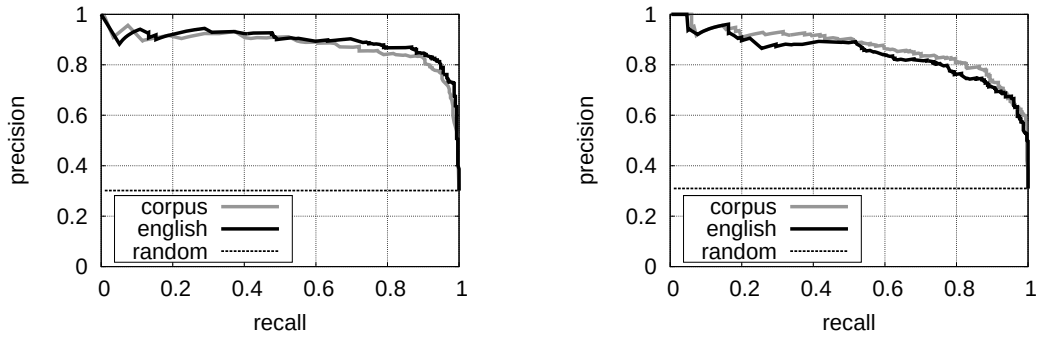


Figure 5.12: Precision-recall curves for German (left) and Spanish (right) Wikipedia test sets, comparing performance of natively and English trained models

5.4.2 Non-English Wikipedias

The vast majority of anti-damage research has been performed on English Wikipedia. Yet just one-third of the 12 million monthly edits to Wikipedia occur on its English version [22]. The PAN-WVC-11 data set [111] makes available both German and Spanish vandalism corpora, compiled similarly to the English version of Sec. 5.2.1. In [143] we use these corpora to investigate a number of properties regarding the cost effectiveness and benefits of language features. The most significant result, and that summarized here, is the portability of our reputation/metadata approach.

One can think about portability in two ways: (1) Feature portability, where signal calculation code can be reused and brought into force after retraining over a new corpus, and (2) classifier portability, where one trains over one labeled set of instances (*e.g.,* English) and brings the resulting model into force over a set from a different environment (*e.g.,* Spanish). This implies that both features and their values/threshold are portable, a very powerful notion. Amassing a representative labeled corpus would be a non-trivial burden for smaller communities. For perspective, Potthast [110, 111] spent $3000+ (USD) on Mechanical Turk labeling for the English corpus. A single *generic* classification model of even moderate performance would have tremendous impact for Wikipedia's 285+ language editions.

To investigate this a set of anti-vandalism features are identified that are not language specific. A model is trained over the English corpus and the Spanish/German corpus is provided as the test set. Fig. 5.12 visualizes the results, comparing them against cross validation output over the equivalent language corpus. In both cases the results are quite comparable: a 7% loss in performance for German (0.811 AUC vs. 0.885) and a 4% AUC gain for Spanish (0.843 vs. 0.805). As a point of comparison recall that 0.82 AUC was our language *inclusive* performance over the English corpus.[9] Such subtleties should not obfuscate the larger point: These initial results are an encouraging first step towards developing completely generic models useful across environments that vary in size and scope. One could imagine such models being part of default security functionality packaged with open collaboration software.

5.4.3 Code Repositories

When considering non-wiki environments that might benefit from our techniques, collaborative code repositories are a good fit. They are version control systems that are OCA-capable, inhibited only by the fact their use cases tend to employ high barriers to entry. Such barriers all but eliminate malicious damage and should challenge our methods' ability to make subtle behavioral distinctions (*e.g.,* distinguish "excellent" participants from "average" ones). Moreover, programming code artifacts represent a use case that is not rooted in natural language.

As a case study we chose to analyze the SVN repository used to develop Mediawiki (the popular wiki software), which has some 170 permissioned users and 117,000 file versions in its primary `/trunk` line of development. In [146] we describe

[9]This makes it appear that finding vandalism on English Wikipedia is a "harder" problem. This is not necessarily the case, observe: (1) The Spanish/German corpora are not representative, as roughly one-third of each corpus is vandalism compared to 7% in the English version. (2) Different Wikipedia's have different security tools/configurations that can be applied proactively (see Sec. 7.1.2) and stop vandalism before it becomes corpus eligible. These same reasons likely explain the performance increase observed over the Spanish test set.

CLEAR ERRORS
"introduced massive breakage . . . "
"revert x . . . trigger errors"
"revert . . . uglier . . . prone to error"
"revert . . . do not remove functions"
AMBIGUOUS QUALITY
"revert x for now . . . needs testing"
"white-space [not per style guide]"

Table 5.9: SVN commit comments associated with reputation loss events

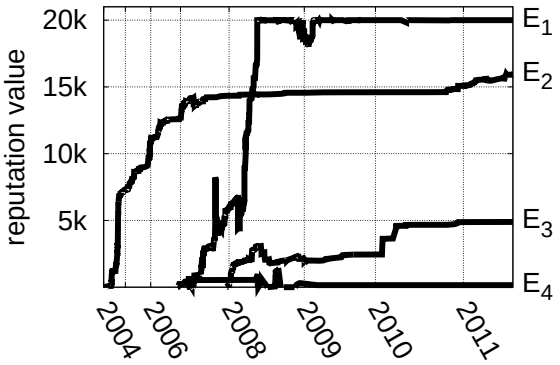


Figure 5.13: Editor reputations in an example SVN repository

how SVN history can be replayed into a wiki format to make reuse of content persistence reputation algorithms. That is, a developer's reputation is based on the survival of the code elements they introduce into the repository. These reputations can be plotted as a function of time, as Fig. 5.13 does for four Mediawiki developers. In a proof-of-concept evaluation we sought to attribute reputation loss events to unconstructive actions using commit comments and bug tracking reports. Reputation decrements were clearly justified in roughly two-thirds of cases, and we were surprised to find a large quantity of revert events in a production repository (see Tab. 5.9). False positives reveal evolutionary differences between interpreted code and natural language content, several of which include:

- *Reorganization*: Whereas wiki artifacts develop internal to a single file, code content is often migrated between files. The single document model treats this as a deletion (hurting the reputation of content authors) and disjoint addition (the "new" code has no provenance information).

- *Need for testing*: Revert actions justified by the need for further testing sometimes follow large commits but do not imply the code is "bad". This results in

massive reputation loss: (1) for the code author, when the code is temporarily removed, and eventually, (2) for whomever requested more testing, because their revert is invalidated when testing is complete and the code is eventually committed in a form similar to the initial attempt. Natural language text is not interpreted/functional and has no equivalent.

- *Non-functional text*: Code comments and whitespace changes are surprisingly prevalent and dynamic. Since this is non-functional text, it could be argued that it should not be included in reputation computation.

Algorithmic adjustments could eliminate many of these shortcomings. For example, code migration could be captured by modeling compile/runtime dependent groups of files as a single artifact (*e.g.,* a very lengthy code file for diff purposes). Language specific tokenization and parsing could create canonical code versions that emphasize functional aspects. These improvements could be integrated with technical [67, 93] and social [39] observations regarding faults/quality in collaborative code.

Developing formal corpora for purposes of evaluation and feature extraction is challenging. Bugs are notoriously hard to track and associate to any single author/commit. Regardless, the proposed reputations could be immediately useful in prioritizing code review. One possible application is the `vehicleforge.mil` [49] project that intends to crowdsource the physical and software design of a military vehicle with unconventionally low barriers to entry. Towards securing access control in that setting, we are exploring the inclusion of content driven reputation alongside traditional quality measures such as test suites and static code analysis.

Chapter 6

Improving Human Mitigation

Humans are a necessary element in the security of open collaboration applications. In the last chapter our state-of-the-art computational mechanisms were only capable of mitigating about half of the problem space. Not just a shortcoming of our approach, our measurement studies found there may be fundamental limits regarding the effectiveness of automated defense. While humans may be required, Sec. 4.2 showed severe inefficiencies in how they were currently being organized and routed. Towards improving this we define evaluation metrics for the task (Sec. 6.1) before discussing our approach for optimizing human security actors (Sec. 6.2). Unlike autonomous modeling done over offline corpora, evaluating and improving human behaviors requires us to develop and deploy software integrating our approach (Sec. 6.3). The resulting tool has been used to review more than one million Wikipedia edits, a data set we mine to learn more about mitigation behaviors (Sec. 6.4).

6.1 Evaluation Metrics

We consider two evaluation metrics that speak to the efficiency of human mitigation: (1) damage survival time and (2) damage impact/exposure:

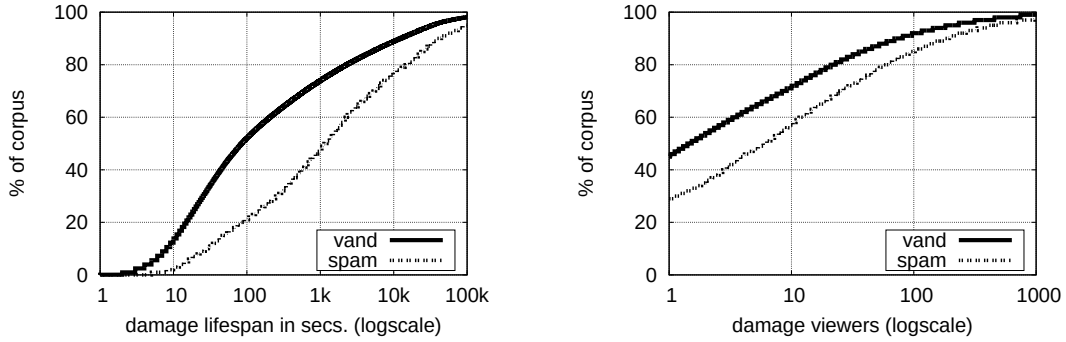


Figure 6.1: CDF of lifespan and exposure for damage events removed by humans; note this excludes damage removed by autonomous bots

Incident longevity: Calculating the survival time of a damage incident is straightforward, as it is the interval between damage being committed and its removal. Lengthy damage survival: (1) Provides more opportunity for the damage to be consumed by end users on the platform, external content scrapers, and search engine crawlers, (2) permits more damage, since the associated accounts are latent in receiving warnings and block actions, and (3) may encourage more damage, as latent detection may be interpreted as success by the perpetrator. Fig. 6.1 plots the survival time for incidents in our corpora of prior chapters, finding human identified vandalism incidents survive for 75 seconds at median and spam instances for 1200 seconds. This difference between damage types substantiates earlier claims about the lack of specialized functionality for anti-spam patrol and the fact most spam is handled at watchlister granularity.

Impact and exposure: Rather than simply measuring the time damage survived one can assess the *impact* incurred during that period. Such a metric might include the quantity of people that consumed the damage (*i.e.,* the *exposure*), measures of severity, and/or prominence. For Wikipedia we are able to interpret an incident's impact as the number of individuals that viewed the damaged article. Fine grained article traffic statistics enable the accurate estimation of this quantity [22, 114, 139,

147]. Damaged views affect external perceptions and speak to the utility an attacker might derive. For example, every link spam exposure is a potential click-through and visit to the spam landing site [86, 141].

While exposure/impact has stronger statistical underpinnings than longevity that does not make it a "better" metric. Consider that a damage instance with low exposure can still have a considerable survival time and incur the ill-effects discussed above. Fig. 6.1 plots the exposure quantity for our damage corpora, with the median human mitigated vandalism instance having 1.6 viewers compared to 6 viewers in the spam case (the difference is 14 vs. 39 views at the third-quartile). Sec. 6.2.1 references the article popularity statistics that underlie this result.

6.2 Routing and Organizational Approach

Improving human mitigation requires addressing the defense shortcomings described in Sec. 4.2.2. Towards eliminating naïve damage patrol we interpret our evaluation metrics as objective functions and use these to prioritize the review process (Sec. 6.2.1). Seeking to minimize simultaneous and redundant work we then coordinate distributed access to this prioritization logic (Sec. 6.2.2).

6.2.1 Review Prioritization

Our goal is to construct an intelligent routing [43] system for damage patrollers. For simplicity assume a set of revisions – some damaging, some not – has been simultaneously committed to Wikipedia. The burden of reviewing these edits falls to a single individual. The order in which he/she chooses to perform the reviews (a *priority queue*) is the only free variable, with performance being measured in an ex post facto fashion using our evaluation metrics. A naïve and poorly performing strategy is to randomly prioritize these edits, one that far too often describes the status quo on Wikipedia. Instead, consider prioritization based on:

#	ARTICLE	Vs/HR.
1	Facebook	3723
2	Wiki	3377
3	Deaths in 2012	2899
4	One Direction	2549
5	The Avengers	2539
6	50 Shades of Grey	2484
7	2012 Phenomenon	2351
8	Dark Knight Rises	2153

Table 6.1: Wikipedia's most popular articles in 2012 by avg. hourly views

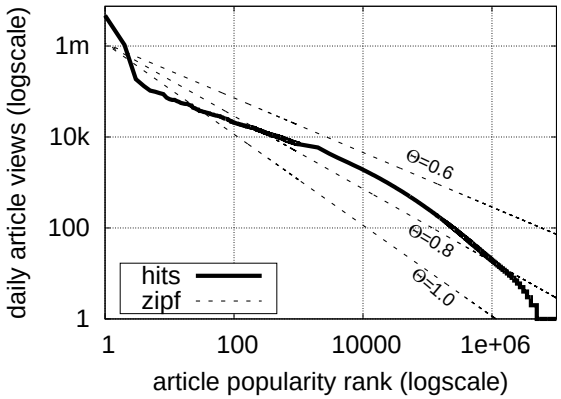


Figure 6.2: log-log plot exhibiting power law distribution of article popularity

Probabilistic strategy: One can use computational methods (like those of the prior chapter) to compute the probability that each edit is damage. This probability becomes the priority for queue insertion. By reviewing the most likely damage first, one minimizes the cumulative survival time of the damage subset. Moreover, at all points during the review process the *recall* is maximized (*i.e.,* as many vandalism cases as possible have been mitigated).

Expected exposure strategy: In order to minimize cumulative damage exposure one needs to know the expected traffic on the corresponding articles. With this one can use the product of the view rate and damage probability to prioritize individual revisions. As Fig. 6.2 visualizes, article popularity follows a power law distribution. This gives rise to interesting queue properties. For example, a revision on the "Facebook" article (the most popular on Wikipedia in 2012, see Tab. 6.1) computed to have a 1% probability of damage will be prioritized above 100% certain damage on 99.99% of articles. Relative to consistently popular articles, traffic spikes [147] can give rise to even more dramatic dynamics (Tab. 6.2).

#	ARTICLE	REASON	DATE	Vs/SEC.
1	Whitney Houston	Death of subject	12 Feb 2012	425.6
2	Amy Winehouse	Death of subject	23 Jul 2011	377.5
3	Steve Jobs	Death of subject	6 Oct 2011	295.5
4	Madonna	Super Bowl halftime	6 Feb 2012	275.9
5	Osama bin Laden	Death of subject	2 May 2011	239.5
6	The Who	Super Bowl halftime	7 Feb 2010	157.8
7	Ryan Dunn	Death of subject	20 Jun 2011	145.1
8	Jodie Foster	Golden Globes speech	14 Jan 2013	125.4

Table 6.2: Peak traffic events on English Wikipedia from 2010-2012, determined at hour granularity ("Vs/SEC." = views per second)

6.2.2 Organizational Primitives

Intelligent prioritization is a first step towards improved mitigation. However, if distributed actors independently calculate and route themselves according to the same logic they will all: (1) simultaneously and/or (2) redundantly review the same edits, to (3) the exclusion of others that may also be damaging (three of the shortcomings identified in Sec. 4.2). Instead one can centrally calculate and maintain a priority queue of revisions. Then, client-server communication can better coordinate patrollers and enforce organizational primitives. For discussion purposes, we make the simplifying assumption that all edit reviews happen internal to our framework.

When a client requests a revision it should be popped the unlocked edit with highest priority. When an edit is given to a client a *revision level lock* should be enforced until released. A lock is released: (1) when the client provides feedback on the edit they were assigned or (2) no feedback is received within some deadline. Feedback must be of least binary specificity (*i.e.,* damage and not-damage) and intermediate granularity are supported.

Edits labeled damaging should correspond to a revert (on Wikipedia) and dequeue action. Critically, non-damaging classifications must be annotated and impact queue structure (establishing the notion of *explicit innocence*). If users are trusted and/or non-guilt is definitive, then a dequeue action may be appropriate. Other situations might dictate reducing the review priority associated with the edit. There

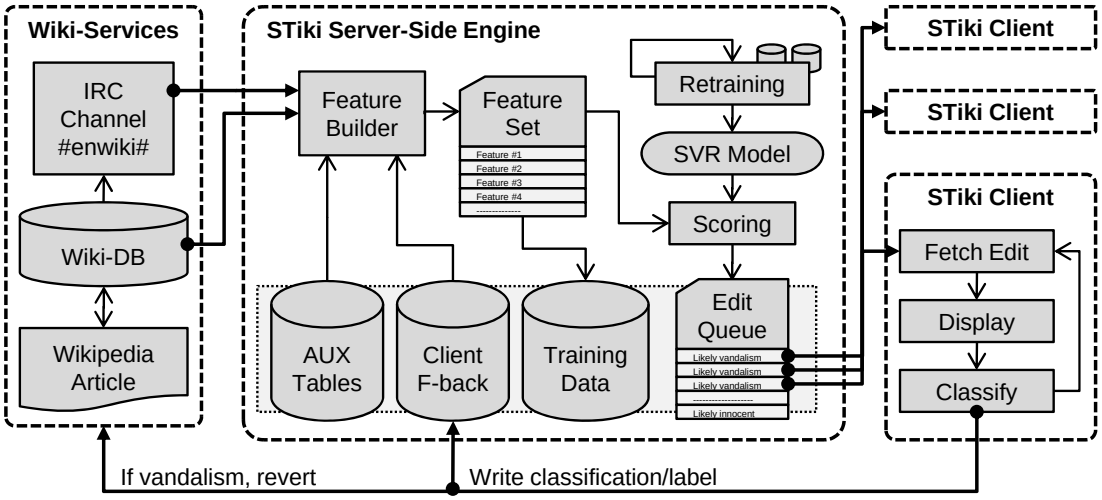


Figure 6.3: Architecture of the STiki framework; here showing the internal calculation for a single scoring/queue system.

are other *queue dynamics* to consider but these are primarily practical considerations addressed in the next section.

6.3 Live Implementation (STiki)

We implement the prioritization and organizational models we just described as a software framework named STiki [136].[10] We begin by describing the client-server architecture of the tool (Sec. 6.3.1) before outlining practical adjustments needed to accommodate live operation, at Wikipedia scale, amongst security actors not bound by our coordinating strategies (Sec. 6.3.2).

6.3.1 System Architecture

STiki's system architecture (Fig. 6.3) consists of (1) multiple revision queues on a centralized server, and (2) a client GUI tool to interface with that logic.

[10]The name STiki is derived from "**S**patio-**T**emporal analysis over W**iki**pedia". The software was initially intended only to support a classifier built off our reputation and metadata features [142]. It has since evolved into a more general purpose tool, so this acronymic significance is now downplayed.

Back end queuing: Three different anti-damage classifiers prioritize revisions:

1. Metadata + Reputation: An implementation of the system described in Sec. 5.2, absent much of the semantic/lexical language functionality.
2. CBNG: Bayesian document classification as calculated by a third party [36].
3. Link spam: An implementation of the system described in Sec. 5.3

In near real time each system calculates a probability on $[0, 1]$ for every revision made to Wikipedia. This is the basis for the insertion priority into two queues: one purely probabilistic and another that integrates recent traffic statistics [22] to sort by expected exposure (yielding 6 queues in total). The behavior of these queues is linked, *e.g.,* a lock in one queue extends to all other queues.

Front end GUI tool: Fig. 6.4 shows the GUI tool that clients use to interface with our queuing logic. While organizational matters are enforced in the background, a user interacts with the tool primarily by reviewing diffs (GUI center) and selecting from four classification options (left-center):

- Damage/Vandalism: The edit is dequeued and reverted. The perpetrator is automatically warned at the appropriate level or a block request lodged.
- Good Faith Revert: The edit is dequeued and reverted.
- Pass: The lock is released on the edit and an annotation is made to ensure the current reviewer will never again see this particular revision.
- Innocent: The edit is dequeued.

No matter the classification the feedback is stored to improve future learning and the GUI automatically advances to the next revision. Pre-fetching and threading are used to minimize transitional latency between reviews. Front end users can choose the priortiy queue from which they draw edits.

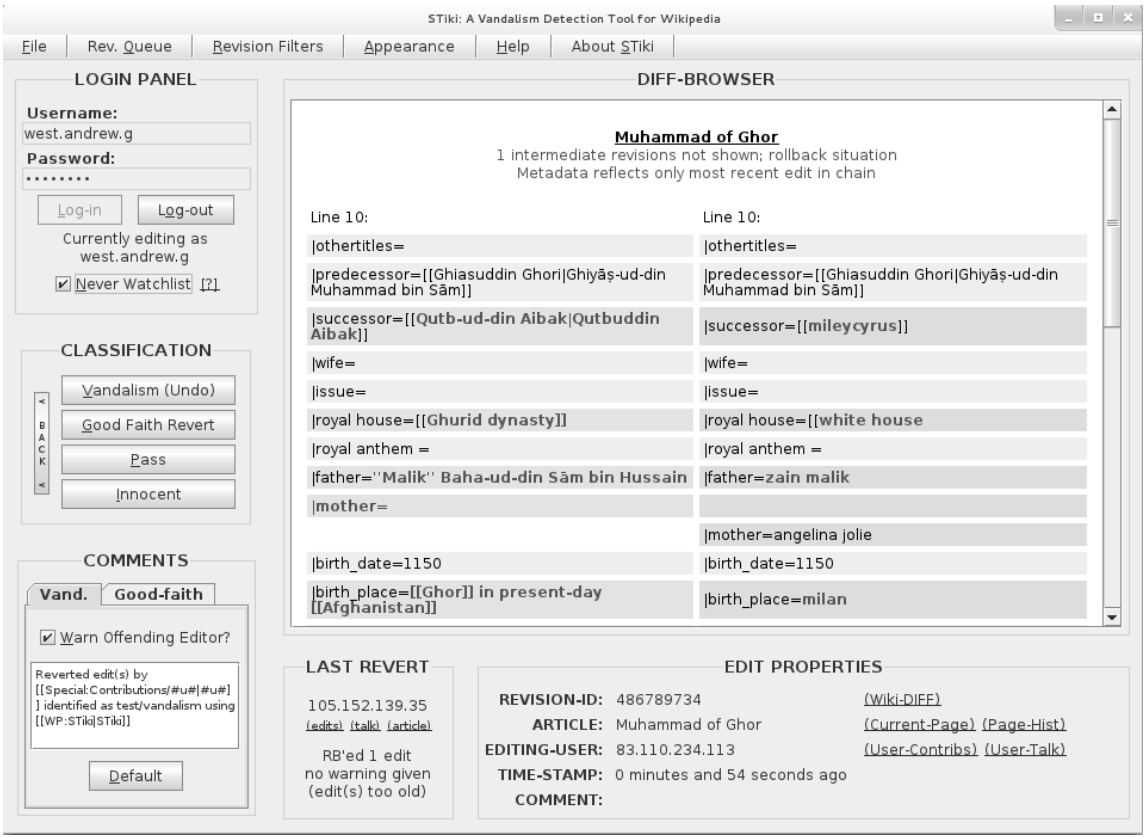


Figure 6.4: STiki user interface showing an instance of vandalism

6.3.2 Accommodating Live Operation

It would be beneficial for Wikipedia if all revision reviews were conducted within the STiki framework.[11] STiki can easily add new queues or damage logic and alternative GUI tools are free to interface with our back end system. In practice a majority of patrol/watchlist reviews occur external to our tool's scope. Combined with the massive scale of the task, these factors dictate design decisions that better accommodate the realities of anti-damage work:

[11]Really, the native Wikipedia platform and other tools just need to speak a standardized language regarding: (1) edits currently under review (for locking) and (2) reviews producing a non-guilty result (to prevent redundant review at patrol granularity).

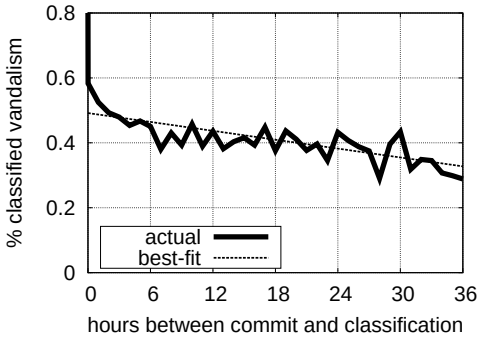


Figure 6.5: Declining accuracy of vandalism probability (here, $\mathsf{prob}(\mathsf{time}_0) = 0.8$) while enqueued

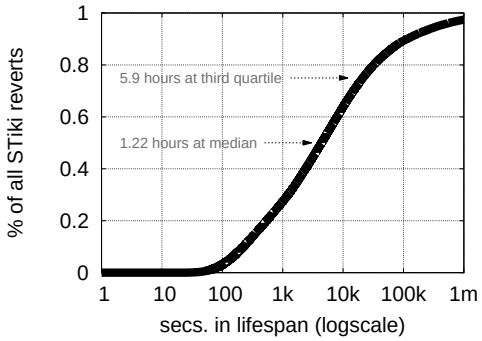


Figure 6.6: CDF of damage lifespan for edits reverted using the STiki GUI classification tool

Review depth: STiki's queues only contain revisions which are most recent on their articles, regardless of whether the previous edit was reviewed or not. A new edit on a page will overwrite and re-prioritize any existing queue entry for that artifact. Deep inspections are best performed by watchlisters with subject expertise; STiki intends only to be a patrol tool for blatant damage.

Immediate competition: Most revision scrutiny, including redundant portions, tends to happen in the moments after a new edit is made. In this period, bots run their analysis (their reverts may take several seconds due to computational and network latency) as traditional and tool enabled patrollers rush to be the first to discover damage [60]. Recall that the median vandalism survival time is just 75 seconds per Fig. 6.1. STiki avoids this competition by only enqueuing edits after they have survived for a minimum of 60 seconds. This is one reason Fig. 6.6 shows STiki has reverted no damage quicker than that threshold.

Stale probabilities: When damage is discovered outside the STiki framework the resulting revert will overwrite the damaging edit in the queues. Edits which are externally reviewed but not reverted remain unchanged in those queues. The longer

a highly prioritized edit remains in the queue, the greater the likelihood that the edit is innocent (*i.e.,* a computational "false positive"), not simply that no one has discovered the damage. Fig. 6.5 makes this explicit over empirical data. We begin with a set of edits computed to have an 80% probability of vandalism (and when tagged offline, we have confirmed ≈80% are damaging). In live operation, a subset classified one minute after they were committed showed a 60% incidence of damage. Even in this short interval patrollers external to our tool had discovered non-trivial parts of the damaging subset, leaving a higher ratio of non-damage in the queues. Edits surviving and classified after 36 hours only exhibited a 33% incidence of vandalism. We are able to compute these rates of decay and are exploring their use to update priorities based on in-queue survival times.

6.4 Understanding Human Response

Since releasing the STiki tool in June 2010 it has been used by the English Wikipedia community to classify over 1.1+ million revisions. Of these, 350,000+ have been identified as damaging and reverted from the encyclopedia. Moreover, STiki's popularity and market share among anti-vandal tools is continuing to grow.

The client-server engineering of the tool has allowed us to extensively log and timestamp every connection request, query, and classification action taken within the tool. These logs and the associated revision metadata are a tremendous resource regarding human mitigation behaviors. We first mine this data for insight on individual reviewers, assessing their motivations for patrolling and characteristics in doing so (Sec. 6.4.1). Then we look at aggregate effects, examining how these reviewers cumulatively affect tool adoption and queuing dynamics (Sec. 6.4.2).

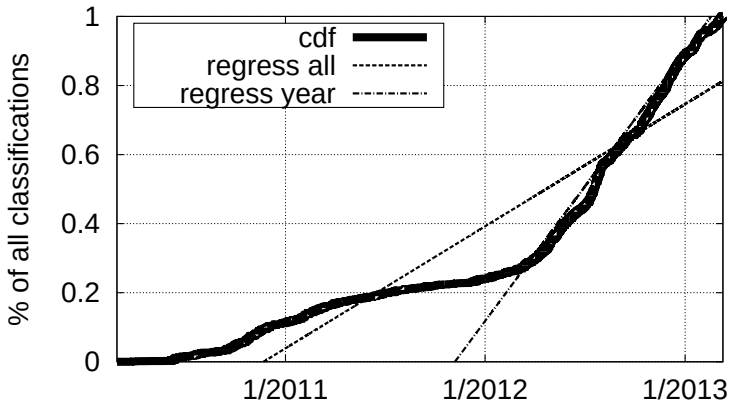


Figure 6.7: STiki tool use as a function of time

6.4.1 Individual Participation and Motivation

Who patrols: Since STiki's creation 640 users have downloaded the tool and classified at least one revision. To prevent tool abuse the Wikipedia community requires that a user have non-trivial anti-damage experience.[12] However, access requests from unqualified new users and recently caught vandals have not been uncommon. Some users are converts from other anti-vandalism tools. Based on volunteered demographic information we know the user base is primarily young (under 30), male, educated, and draws from the United States, England, and India (which is not atypical of the English Wikipedia user base at large [2]).

Extent of participation: A STiki participant spends around 6 seconds reviewing each edit they are assigned. This suggests it would take ≈125 man hours to patrol all the article edits made to English Wikipedia each day. While this is a burden that 20 full time workers could handle, the volunteers who do this work are understandably less dedicated. The average edit "session" [59] consists of about 50 reviews.

We find one factor influential in extending or shortening session length: the *revert*

[12]Per our threat model of Sec. 2.4, attackers are unable to secure the advanced permission that enables STiki use. Vandals cannot dequeue the damage they commit. If STiki were universally enabled it could shift to a "decrement priority" model instead of a "dequeue" one, with the decrement delta for innocent classifications determined by user reputation.

Figure 6.8: Reward "barnstar" to gamify and incentivize tool use

rate that end users experience, *i.e.,* the percentage of edits one classifies as damaging. The tool averages a 33% revert rate across history, but this is an unscientific measure due to environmental and queue dynamics. However, the metric certainly shapes the perceptions of end users. When users have a session with a revert rate over 40% they tend to extend those sessions, with rates below 20% spurring early termination. Seeing edit sessions abandoned at 20% density may also speak to user conditioning; consider that traditional random search will yield a $< 4\%$ revert rate (since about half of the original 7% is taken care of by bots).

The typical 5 minute patrol session is about half the duration of those measured broadly across the encyclopedia [59]. Reviewing is repetitive and monotonous work and this likely affects one's dedication to the task both in individual sessions and as a long term commitment. The median participant uses the STiki tool only for 8 days, though a small set of long term users stretch the average to 88.6 days. Overall there is a high turnover rate in the user base. Users who abandon the tool tend not to stop contributing to Wikipedia altogether. Instead, we find them fulfilling alternative and arguably more complex and rewarding project roles. As Goldman [63] has observed, anti-damage patrol is not a particularly fulfilling task as it is rote work where one frequently encounters immature and offensive content [41, 124].

Reviewer motivations: Understanding why users abandon the tool, we now consider what motivates those who do participate. Incentives for open collaboration participation vary [56] but rarely are there monetary and/or tangible benefits. There

are, however, notions of status and reputation internal to these social communities. Some feel that status is correlated with one's editing rate, an affliction that experienced Wikipedia users sarcastically term "edit-count-itis" [55, 72]. Damage patrol, particularly when well prioritized, is an efficient means to amass many edits.[13]

Regardless one's opinion on edit count inflation, few can deny this damage patrol is a good thing for the encyclopedia when done accurately. Thus we try to harness this trend and increase user throughput. Toward this we have implemented a competitive "scoreboard" of STiki use and recognized/rewarded continued participation with "barnstars" [88] (Fig. 6.8). Both functionalities were initiated in early 2012 and we partially attribute the increase in tool participation at that time to their effects (Fig. 6.7). Even so, motivations driven by edit count are not entirely beneficial. We have observed isolated incidents of aggressive bias in labeling vandalism and hasty reviewing rates. Gamesmanship has been observed via the manipulation of queuing dynamics so a user could inflate his/her revert rate (*i.e.,* labeling "innocent" edits with "pass" so other users must re-review them, as the original reviewer receives fresh edits).[14] Moreover, revert hungry motivations cause users to abandon editing when those rates decline. This has unfortunate consequences for queues prioritized by expected exposure instead of pure damage probability, as improbable damage on extremely popular articles tends to sit atop these queues.

Finally, our above discussion applies only to anti-vandalism behaviors. Usage of anti-spam queues has been minimal (<1% of classifications). Our probabilistic models indicate these queues are ripe with damage, as there are no active anti-spam bots and little competitive review. Yet the average anti-spam review takes over 60 seconds time as one must analyze the landing site. Recall also that our measurement

[13]We do not claim that all damage patrollers in our user base apply this philosophy. However, behaviors suggest the trend is quite prominent, even if it is not entirely intentional.

[14]Such gamesmanship was surprising given our user base is specially permissioned. Because non-guilty classifications like "innocent" and "pass" do not generate on-wiki evidence they can be misused. If edits were redundantly reviewed, one could detect users with divergent labeling patterns. Not content to sacrifice scalability in this fashion we have instead codified expected statistical usage trends that trigger manual audits if violated.

studies found spam behavior to be subtle, requiring greater discretion and policy knowledge on the part of reviewers. Heavy use of the "pass" classification and our personal experience confirms there is often ambiguity over how to label a revision. Many times it is clear that a linking behavior is non-ideal but uncertainty whether it meets the spam threshold. There are also social reasons for such a conservative bias, as it is not unusual for spammers to become confrontational about their link's removal. The average reviewer seems unwilling to make these distinctions. Many of the cases our models label as high probability but STiki users choose not to revert are eventually removed with a latency suggesting watchlister involvement. This fact is evidence there is not a pathological failure of our training set or classification models.

6.4.2 Aggregate Trends

Having assessed individual participation and motivation we now examine the cumulative effects of this workforce on queue dynamics and edit assignment. As of early 2013 the STiki tool averages 30 review sessions daily, classifying 1500–2000 edits, and consuming 2–3 hours of labor. For context, STiki currently rivals Huggle [16] as English Wikipedia's most popular review tool although a majority of diff inspections are still performed via the native interface. While STiki's popularity is increasing (Fig. 6.7) the global quantity of patrollers is decreasing [72], making STiki's intelligent routing all the more necessary.

Just as revert rate influences the duration of review sessions, it appears to do the same with tool adoption and user retention. The size of the user base is inversely correlated to the damage density individual users experience (assuming static user throughput and stable market share). Periods of low tool use have given rise to enthusiastic users and promotional efforts that generated short term popularity spikes. However, the influx of new users that results makes it impossible to meet advertised expectations regarding revert rate. The result is an odd paradox of tool popularity

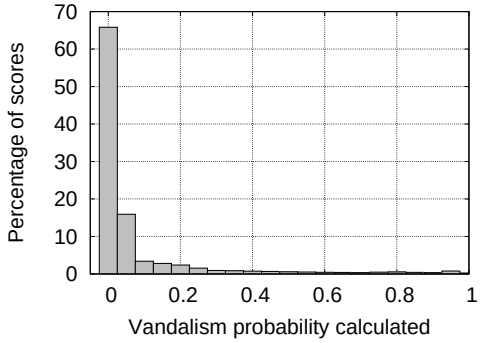


Figure 6.9: Binned distribution of vandalism probabilities computed over all article edits (metadata/reputation strategy)

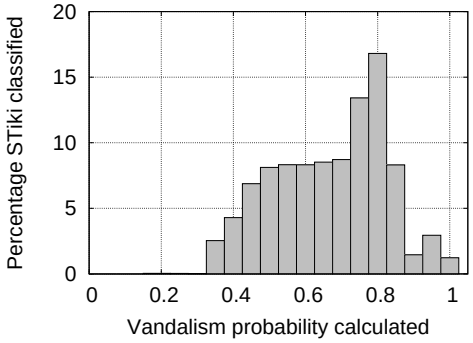


Figure 6.10: Computed vandalism probabilities of edits classified by humans using the probability prioritized GUI tool

vs. user satisfaction, driven by the fact we have crowdsourced the discovery of a finite resource (damage). As a point of reference, recent month long sliding intervals have seen an average of ≈65 users classify using the tool.

It is this aggregate group that influences what edits get reviewed. Fig. 6.9 shows the output of a probabilistic scoring model in live operation, suggesting the vast majority of edits on Wikipedia pose little damage threat (only 8% of edits indicate a 25%+ damage probability). STiki attacks this set in a top-down fashion with Fig. 6.10 showing the probabilities classified by STiki users. STiki rarely displays the most confident damage as this portion is handled by bots and quick acting reviewers (the confidence would imply this is "low hanging fruit"). Conversely, edits below 40% probability are rarely reviewed because: (1) per the previous section, this is the threshold where users tend to lose interest (recall the empirical decay of Fig. 6.5) and (2) it requires a great quantity of classifications in a short window to deplete the queue to such depth given the incoming edit rate.

It is surprising to see that most of the damage undone by STiki is not recent (Fig. 6.6). A median lifetime of 1.2 hours makes it obvious patrollers are incomplete in their coverage. The fact the third-quartile lifetime is 6 hours suggests that were

STiki not in place much of this damage might have become embedded on the article. This evidence suggests that STiki and its philosophy are not just another entry in the race to discover vandalism, but it is helping users locate damage that might *never* have been located otherwise. Incredibly, two edits reverted by STiki were enqueued over a *year* before they were assigned and classified.

Chapter 7

Discussion and Future Work

Complementing the academic contributions of this dissertation (as summarized in Chap. 8), our impact on Wikipedia's security landscape has been significant. We begin by reviewing these advances before offering suggestions on how to improve Wikipedia's practical security with the further integration of our work (Sec. 7.1). However, our research is not without limitations. By discussing the weaknesses of our approaches and overarching vulnerabilities of the collaborative paradigm we identify open questions on which future work can focus (Sec. 7.2).

7.1 Practical Improvements

Portions of our work are already a part of Wikipedia's operations and toolkit, proving themselves to be valuable security assets (Sec. 7.1.1). There is the potential for these developments to play an even larger role if they can be better integrated into the platform (Sec. 7.1.2). Finally, we consider situations where security efforts are insufficient and one must begin restricting collaborative permissions/scope in order to achieve defense goals (Sec. 7.1.3).

7.1.1 Status Quo Impacts

Recent years have seen an increased focus on Wikipedia damage. As of early 2013 the defense ecosystem is built piecemeal atop the efforts of Wikimedia employees, academics, and Wikipedia enthusiasts. Cumulative impacts from recent work are best contextualized by revisiting early Wikipedia research. In 2004 the median damage survival interval was around 2.5 minutes [131]. Currently that number is < 10 seconds since automated mechanisms handle roughly half of the problem space [36]. However, this buries improvements to human response. There were no autonomous anti-damage bots when the 2.5 minute survival figure was produced in 2004 and Fig. 6.1 shows how manually reverted portions fare in the modern day, persisting roughly half the time of the earlier result (from 2.5 $\rightarrow$ 1.3 minutes). Measurements of end user exposure are similar. Currently $\approx$65% of human handled damage has < 1 viewer and the exposure CDF (again, Fig. 6.1) is similar to one produced in 2007 [114]. This stability is impressive considering English Wikipedia's exponential growth in the interval. Page views, article count, and many other metrics grew by about 50$\times$ between 2004 and 2012. Summarily, the past 5 years have seen the emergence of algorithms that can mitigate nearly half of damage and editing tools have helped humans scale reasonably well over the remainder.

Our work has been an important component in this cumulative improvement. Our reputation/metadata classifiers for vandalism (Sec. 5.2) and anti-spam (Sec. 5.3) are implemented and processing all Wikipedia edits. Only bureaucratic issues have prevented these from being leveraged autonomously. Regardless, these scoring engines are being utilized in multiple ways. Most obvious is the STiki tool [136] with 350,000+ reverts on English Wikipedia (Sec. 6.3). Work is ongoing to internationalize that interface and make use of language independent anti-vandalism models (Sec. 5.4.2). Another tool, WikiAudit [135], leverages these scores to report on damaging users/revisions from a user provided set of IP addresses (Fig. 7.1). The tool is useful for network administrators (*e.g.,* auditing contributions from their managed IP

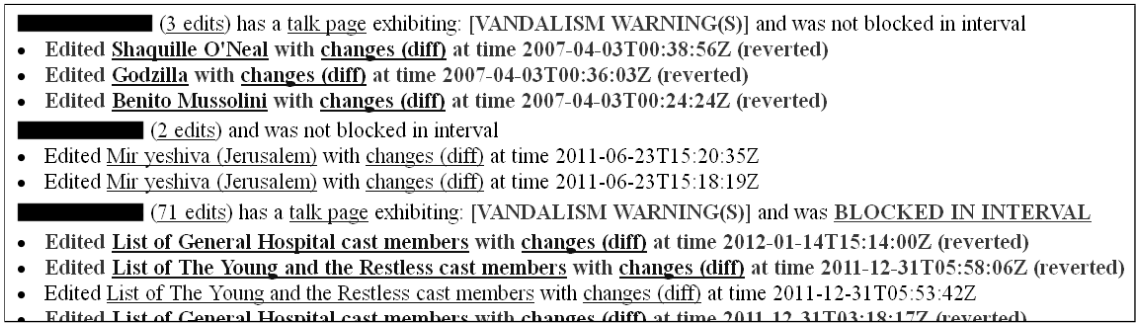


Figure 7.1: Snippet from a WikiAudit [135] report; note that user IP addresses have been redacted for privacy

space) and those conducting investigations into editing bias [145]. Third parties also use our probabilities. For instance, the Snuggle [70, 72, 127] tool uses them to locate newcomers that are *not* damaging and encourage their continued participation.

Minor contributions also abound. Our description of mechanized attack models [139, 141] (upcoming in Sec. 7.2) influenced configuration settings. Additionally, we have developed a tool to expedite the review/labeling of a user provided set of edits. The "Offline Review Tool" [136] has been used by other practitioners for corpus building and user audits. We have also taken the statistics used to predict and measure damage impact and use them to regularly report on Wikipedia's traffic trends [147]. Finally, we have provided perspective on how institutions can help prevent damaging behaviors with a focus on education and internal monitoring [145].

7.1.2 Extending Platform Integration

While the practical contributions of our work are significant their impact could be even greater with cooperation from wiki developers and other tool authors.

Beginning with the prevention of damage, Wikipedia processes edits against an anti-damage rule set before they commit (the "Edit Filter"). If a rule is matched the editor is notified and can alter/abandon the revision, a "scare tactic" to deter vandals. However, the current manually written rules fall short of the capabilities of described machine learning models. The filter should also be configured so it is less

adverse to false positives (typically just warnings that can be overridden). There is little technical value in duplicating subsequent bot logic, and the goal should be getting vandals to abandon damage that would need human mitigation.

This redundancy and poor interconnect between platform security (*e.g.,* the edit filter) and editor contributed logic (*e.g.,* the bot) is a pervasive issue in Wikipedia's decentralized security landscape. While this follows from Wikipedia's community driven philosophies it would seem the mission critical nature of security warrants better integration. At current, complex anti-damage logic is accessible only through independent third party tools. Annotating or overlaying this information directly on the native interface would benefit watchlisters and casual users. Moreover, the functionality could be included with wiki software packages that currently ship with only rudimentary security functions and vandal friendly configurations [145].[15]

Cooperation is also needed to eliminate redundant and incomplete work. While STiki has a notion of "explicit innocence", other tools and the native platform do not generate evidence of non-guilty inspections. Simply knowing what diffs have been rendered would be beneficial. More powerfully, a centralized and real time "anti-damage clearinghouse" could be established to share review information and influence human routing decisions. While this should not replace eventual reviews by subject experts, dedicated and topic agnostic patrollers enable the quick reversion of obvious damage. Such users are a necessary part of timely damage response and it would be worthwhile to explore further gamification and incentives to combat their limited patience and high turnover rates. While focus on patrollers is intuitive, security functionality is also needed to streamline the work of watchlist reviewers. Their dedication to certain topics make them less ephemeral security actors and they have the expertise needed to confidently remove subtle damage.

[15]Recall that our research also suggested the potential for generic anti-damage models built atop participatory dynamics and/or implicit reputation. Simple interpretations of these concepts could compute the probabilities/priorities needed to drive these tools and annotations.

7.1.3 Security Tradeoffs

The system model for this dissertation (Sec. 2.4) assumed "pure" open collaboration where all users can modify any artifact. While Wikipedia strives to uphold this model, practical restrictions have emerged with the project's exponential growth. Also concerning is the potential for more limitations as traffic continues to surge [22] and editor/patroller/administrative populations stagnate or dwindle [63, 71, 127]. Goldman [63] ties the expansion of restrictions directly to Wikipedia's recent and future survival. More generally, these restrictions might be applied to any open collaborative application whose defenses are not adequately scaling.

All of the proposed/enacted restrictions follow from the same principle: restrict the user groups which are allowed to revise some artifact(s). Compared to status quo approaches our developments in this dissertation offer a more autonomous and elegant means to determine these restrictions. Moreover, our suggestions minimize social/participatory impacts for well intentioned users.

The most dramatic suggestion is to eliminate unregistered editing on all artifacts. While unregistered editors are frequent vandals, their *constructive* efforts are still 10% of the project total [81]. Forced registration does not prohibit participation, but research shows even constructive users often abandon actions when presented barriers (*e.g.,* CAPTCHAs) [100]. There is no evidence that obstructions dis-incentivize bad users more than good ones; they have been traditionally applied only to limit automated abuse. We see little justification in implementing a restriction that impacts innocent users as much as guilty ones. Alternative proposals have called for a more fine grained user hierarchy over which to apply restrictions. The codification of new user groups along arbitrary and easily manipulated criteria (*e.g.,* edit count) serve little purpose. If one needs to evaluate a user it is preferable to use the reputations we compute or more robust measures of a contribution history [82].

The need for quantitative measures instead of blanket policies also resonates in the *article protection* scheme deployed on ≈5000 English Wikipedia articles. Therein,

Figure 7.2: On-wiki link placement (left) and landing site (right) for proof-of-concept link spam attacks

articles experiencing frequent damage may request "protection" and be made read-only for unregistered/newly-registered users. The proposed *pending changes* system prefers a non-public quarantine in favor of explicit blocking. Regardless, the decision over which artifacts to protect should not be a manual one. Applying human discretion and bureaucracy to combat situations where manual effort is currently insufficient is paradoxical. Our article reputations capture the most abused pages and protection could be autonomously triggered at certain thresholds. We also advocate the preventative protection of extremely popular articles, and the power law distribution of article views in Fig. 6.2 shows this would protect a tremendous number of exposures while impacting relatively few artifacts.

Autonomous handling also enables protection at revision granularity without the need to explicitly target certain articles for focus. We have demonstrated accurate anti-damage engines whose output probability could dictate revisions being marked for: (1) autonomous reversion, (2) private quarantine, or (3) immediate publication with prioritized ex post facto review. Interfaces like STiki would provide a familiar and rapid means to label quarantined edits. If Wikipedia hopes to maintain its open nature these computational scoring engines must be embraced to further enable platform functionality, not just external third party tools.

7.2 Outstanding Vulnerabilities

Metrics regarding Wikipedia's security performance only speak to the system's handling of status quo attackers. With our measurement studies finding naïve perpetrators plentiful these measures say little about the robustness of open collaboration defense. In trying to identify system weaknesses we begin by describing and demonstrating an aggressive spam attack model. The economic viability of our attempts confirms the need for focus on several vulnerabilities. These complement discussion regarding other system weaknesses gleaned from our community experience and gamesmanship involving our tools. Ultimately these shortcomings speak to deeper challenges and research questions in the domain of open collaboration.

A proof of concept attack: In our link spam measurement study of Sec. 4.1.2 we were surprised to find a lack of aggressive spam strategies as are common in other domains. This sparked curiosity as to whether Wikipedia's damage response was rendering these strategies economically ineffective. In a series of controversial experiments [141] we engineered a payment disabled "male enhancement pharmacy" [86] and prominently linked to that site on Wikipedia (Fig. 7.2). In less than 5 minutes time we were able to post nearly 350 links that garnered 14,000 active views, 6,307 landing site visits, and 8 "purchases" for about $2000USD. The fallout was significant enough to be reported by security media [150]. Crucially, under $20USD were spent mounting these attacks. Core to our strategy were:

- Script driven editing: API based editing scripts rapidly placed links. The rates at which this could occur were seemingly limited only by network latency.

- Popular artifacts: Wikipedia's most popular articles (at the time of launch) were targeted to maximize exposure [129]. As Tab. 6.1 and Tab. 6.2 show, only several seconds of survival on such pages can permit *many* exposures.

- Compiling human latency: With exposures compiling rapidly, defense latency is critical. Our links survived 70 seconds at median, accounts were blocked after 90 seconds, and domain blacklisting took place hours later.

- Privileged accounts: Page protection on popular articles requires registered and experienced accounts. We autonomously attained these by outsourcing CAPTCHA solutions [100] and amassing edits in non-monitored spaces.

- Broad IP agility: Three accounts were used in our experiments, however, many more were brought to privileged status. These were never correlated due to our use of IPs from a large and geographically diverse cloud provider.

- Redundant human effort: Amazingly, 80% of landing site visits were the result of users clicking through *after* the links had been reverted. This speaks to a tremendous number of watchlisters reviewing the effected articles' evolution in an ex post facto fashion.

Mitigation considerations: These proof-of-concept experiments do not intend to make claims about the sustained performance of similar attacks but make apparent Wikipedia's immediate vulnerability. Some of these weaknesses make concrete our earlier claims regarding signature driven detection, focus on popular pages, and the need for efficient and coordinated human routing. Other vectors were leveraged precisely because of the more fundamental challenges they pose.

Automated attack is one such challenge. Assuming an attacker evades computational filters the low/no marginal cost of script driven editing is problematic for a human defense force. Increased barriers-to-entry are not a capable or cost effective deterrent [100]. Rate limiting strategies might force attackers to operate at human-like speed but still do not impact the labor imbalance. Similarly problematic are broadly sourced attacks. In the worst case a botnet could be leveraged to achieve massive IP agility, perhaps in parallel alongside a traditional (email) spam campaign.

Wikipedia and partner operations (recall that most major wikis share an API) could be significantly disrupted at rather low cost. Reinforcing this, Wikipedia's most prolific perpetrators [17] tend to not be characterized by technical novelty, but instead by their ability to obtain many accounts. A massive pool of accounts could be managed deceptively and disrupt ordinary defense processes (*e.g.,* blanking user warnings, flooding administrative spaces, burying damage in revision histories *etc.*).

It is interesting to consider why, if these vulnerabilities exist, they are not being actively exploited. Blackhat software like XRumer [23, 120] already *broadly* targets collaborative functionality. However, this appears optimized for Internet scale SEO rather than targeted attacks against high value targets using wiki specific functionality. Perhaps these targeted environments lack the scale or expected profit margins needed to incentivize capable attackers. This question remains one unanswered by our research findings. Regardless, our approach has taken caution to inform relevant parties of our findings [141]. It is worthwhile to consider that should someone go to such lengths there is likely a desire to monetize their efforts. External hyperlink additions are a minority of revisions yet the source of many security issues. More scrutiny, focus, and restrictions surrounding their placement may be warranted along with investigation into undermining their financial interests (*e.g.,* domain registrations were the most costly portion of our proof-of-concept attacks).

Other susceptibility: Orthogonal to aggressive spamming strategies are those based on subtlety, a tactic not uncommon per our measurement study. One concern is that intelligent attackers are carefully juxtapositioning their behaviors near where the "damage" distinction is drawn. The more benign an edit appears to be (regardless the underlying intention), the more likely it will survive on the encyclopedia, even if this is only a product of social engineering. With humans handling significant portions of damage and most of the "interesting" and borderline cases it is worthwhile to consider the cognitive biases they might apply to the review task.

For example, disguising small quantities of damage in large revision payloads might be a fruitful tactic. Technical solutions should be explored to assist humans more than the standard diff representation.

Collaborative security is also recursive notion. Just as our tools permit one to audit changes to the encyclopedia, we have found ourselves auditing tool users' actions upon suspected misbehavior. In environments that are truly open and accessible even the reviews need reviewed – and manipulation can pervade to the deepest levels. While the threat of gamesmanship is omnipresent in open collaboration settings the naïvety of the typical attacker and payload force one to question the necessity of robust mechanisms. Consider our reputation calculations that only aggregate feedback from trusted users and are not normalized due to ballot stuffing concerns. There might be a net benefit in relaxing these constraints, even if such changes would enable evasion by a narrow class of attackers.

This work has been particularly attentive to evasive and intelligent strategies, *i.e.,* the automated attacks of this section and the "Generic Chinese Knockoff Spam" of Sec. 4.1.2. We have identified features that capture properties of these isolated incidents and our expectations about how hypothetical vulnerabilities might manifest. Because such behaviors are sparse in our training sets these features tend not to be leveraged during model construction. We author static rules to bring them into force and have capability against unseen threats. While frustrating that naïve damage obfuscates focus on the more subtle security properties of the platform, the fact remains that open collaboration presents a broad attack plane. Dealing with this breadth is a challenging defense problem in and of itself.

Chapter 8

Conclusions

In this dissertation we began by describing a class of systems characterized by "open collaboration": accessible platforms granting participants modification permissions to support artifact cultivation. It is this ability to *modify others' content* which distinguishes these applications from simple collaboration (*e.g.,* commenting functionality, web forums) and results in a more dynamic security environment.

Using Wikipedia as a case study we observed this novelty gives rise to unique damaging behaviors. First measuring *vandalism*, a catch-all of damage, we found offensive, opinionated, narcissistic, and deceptive contributions common. Less "open" environments would not consider many of these problematic given that their append only models facilitate artistic expression and opinion sharing. Conversely, modifications over shared artifacts necessitate detailed policies to maintain social order and define expected output. We find these guidelines often include expectations of formality and correctness, with our survey revealing open collaboration commonly supporting documentation and reference works. Consequently there is a broad set of behaviors outside these policies which are considered damaging. In addition to these novel social considerations, the technical semantics of modification actions also present challenges. Revisions are not independent but build on the context of the

artifact they modify. Content can be placed at arbitrary positions and significant impact can be achieved with small payloads. Deletion and re-organization capabilities provide avenues to make unconstructive changes without even adding content.

Much vandalism is immature, so by characterizing the *link spam* subset we sought to learn how incentivized attackers were targeting the platform. Amassing the first link spam corpus we found: (1) less than 10% of spammed sites had direct commercial intentions, (2) spam is almost always context appropriate, (3) spammers tended to follow linking conventions, and (4) links are manually placed in conservative quantities. These are strategies intent on long term survival. Low barriers to entry have enabled a fundamentally new class of attackers – in some cases unaware that they are even perpetrating damage – and in others content to perform ambiguous self-promotion without sophisticated technical infrastructure.

These findings render related work from analogous domains insufficient for accurate damage detection. Language techniques like Bayesian document classification are easily evaded, cannot measure content veracity, and only accommodate content additions. For spam, language model disagreement (*i.e.,* out of context links) and shared incentives (*e.g.,* commercial intentions) were not exhibited by our corpus but are security strategies common elsewhere. Due to this lack of existing computational solutions, a majority of damage on Wikipedia escapes autonomous filters. This means its discovery becomes the responsibility of a distributed and disorganized human workforce. Recognizing these shortcomings we sought to improve all aspects of damage detection and mitigation for open collaboration.

Computational predictors of damage are helpful in both autonomous and human mitigation strategies. We improved on existing language based methods with a content agnostic approach built atop reputation and metadata features. While simple reputation schemes had previously been described, we improved their scalability and extended them spatially. Spatial extensions help overcome the generic "cold start" problem, the fact sparse behavioral histories yield little predictive capability; one

exacerbated by low barriers-to-entry and long tailed participation. For example, the geolocation of a user holds tremendous predictive capability and we described a model that could proportionally integrate this evidence along with user specific feedback. We also uniquely applied these reputations to system artifacts (articles) and the category spaces in which they are members.

While reputations leverage historical aggregation we also found extremely simple metadata features indicative of damage. Features capturing adherence to community conventions and disruption to typical content evolution have proven beneficial ("participatory dynamics"). Cumulatively our work identified 25+ anti-damage features and advanced performance benchmarks when these were brought to bear against a standardized vandalism corpus (from 0.74 to 0.82 per "area under the curve" metrics). We also showed 40% of instances could be autonomously mitigated with tolerable false positives. Content agnostic features have also given models a robustness and portability not present with language only approaches.

Extending our reputation/metadata approach to the link spam problem also proved beneficial. In particular, Internet scale reputation based on backlinks was an extremely indicative quantity. Assembling features into the first wiki specific anti-spam classifier we found 70% of the problem space could be handled autonomously. The atomicity of linking actions and the additional evidence they provide seem to enable the performance increases over anti-vandalism efforts. Regardless, it is content agnostic features that drive performance in both cases. Not just useful for spam and vandalism, we have made these signals portability explicit via cases studies spanning damage types (copyright detection), natural language (foreign language Wikipedias), and content format (source code repositories).

Even as computational methods mature, significant portions of the problem space fall to human actors. To decrease redundant effort by this distributed workforce we described organizational primitives such as edit locking and the explicit annotation of non-damage. We also proposed using computational mechanisms to intelligently

route participants towards potential damage, describing two schemes: (1) a purely probabilistic one maximizing capture rate while minimizing incident survival and (2) one combining this with article popularity to estimate expected impact. Given that current open collaboration platforms (and Wikipedia in particular) lack this organizational and routing functionality we independently developed a practical tool that does (named "STiki"). The STiki tool has proven very popular, streamlining the reversion of over 350,000 English Wikipedia edits en route to 1.1+ million reviews.

We used this massive set to learn more about human factors in the mitigation process, focusing predominately on "patrollers" who form the first line of defenses by reviewing recent changes en masse. We found such users spend only about 6 seconds per edit, are uninterested in performing more time consuming and ambiguous spam reviews, and abandon the tool when revert rates fall below 20%. We attribute portions of this to a user base that correlates personal reputation to the number of editing actions a user performs. Long term participation trends show rapid user turnover. Those who do not quit Wikipedia altogether move on to alternative (and arguably more fulfilling) roles. Damage patrol is rote work, and in Wikipedia's case, one being taken on by a declining number of users even as end user traffic continues to grow. Though experiments in the STiki tool showed gamification could be leveraged to increase review throughput, the trend persists broadly. These patterns make clear the need for our proposed routing and organizational strategies in order to better scale the human assets available.

Cumulatively our impacts on Wikipedia's practical security are undeniable. Our contributions are already – or in the process of – being integrated into an infrastructure that reverts 75% of damage before it is likely consumed by a single end user. While we can identify areas for improvement and potential vulnerabilities, this is nonetheless a tremendous technical and social achievement for a purely volunteer effort that protects 4+ million artifacts. However, as the prototypical open collaboration application Wikipedia's lessons also cascade to the entire class of systems.

Our content agnostic approaches have sought to embrace this generality. For open collaboration to succeed it must allow installations to embrace collaborative benefits that exceed the security burdens that modification freedoms impose. This demands not only the technical capability to detect damage but the practical integration that makes this functionality available. Our efforts herein have sought to advance both aspects, enabling the expansion of open collaboration philosophies, and laying the foundation towards more nuanced quality assessments.

References

[1] Akismet. http://akismet.com/.

[2] Alexa. http://www.alexa.org.

[3] AskDrWiki. http://askdrwiki.com/.

[4] Defensio. http://www.defensio.com.

[5] DMOZ: The Open Directory Project. http://www.dmoz.org/.

[6] Mediawiki. http://www.mediawiki.org.

[7] OpenStreetMap. http://www.openstreetmap.org/.

[8] TurnItIn. http://turnitin.com/.

[9] Wikia. http://www.wikia.com/.

[10] WikiIndex. http://wikiindex.org/.

[11] Wikimedia Commons. http://commons.wikimedia.org/.

[12] Wikimedia Foundation. http://wikimediafoundation.org/.

[13] Wikimedia page-view statistics. http://dammit.lt/wikistats.

[14] Wikipedia. http://www.wikipedia.org.

[15] Wikipedia: External links. http://en.wikipedia.org/wiki/WP:External_links.

[16] Wikipedia: Huggle. http://en.wikipedia.org/wiki/Wikipediapo:Huggle.

[17] Wikipedia: Long-term abuse. http://en.wikipedia.org/wiki/WP:LTA.

[18] Wikipedia: Project Spam. http://en.wikipedia.org/wiki/WP:WikiProject_Spam.

[19] Wikipedia: Revision deletion. http://en.wikipedia.org/wiki/WP:REVDEL.

[20] Wikipedia: Turnitin. http://en.wikipedia.org/wiki/Wikipedia:Turnitin.

[21] Wikipedia: Vandalism. http://en.wikipedia.org/wiki/Wikipedia:Vandalism.

[22] Wikistats: Wikimedia statistics. http://stats.wikimedia.org.

[23] Xrumer. http://www.xrumerseo.com/.

[24] Congressional staff actions prompt Wikipedia investigation. *Wikinews*, January 30 2006. http://en.wikinews.org/wiki/Congressional_staff_actions_prompt_Wikipedia_investigation.

[25] Mass blanking of copyright violations. *Wikipedia Signpost*, September 2010. http://en.wikipedia.org/wiki/WP:Wikipedia_Signpost/2010-09-13/.

[26] S. Abu-Nimeh and T. Chen. Proliferation and detection of blog spam. *IEEE Security and Privacy*, 8(5):42–47, 2010.

[27] B. Adler, J. Benerou, K. Chatterjee, L. de Alfaro, I. Pye, and V. Raman. Assigning trust to Wikipedia content. In *WikiSym '08: Proceedings of the International Symposium on Wikis and Open Collaboration*, 2008.

[28] B. Adler, L. de Alfaro, S. M. Mola-Velasco, P. Rosso, and A. G. West. Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *CICLing '11: Proceedings of the 12th International Conference on Intelligent Text Processing and Computational Linguistics, LNCS 6609*, pages 277–288, 2011.

[29] B. T. Adler and L. de Alfaro. A content-driven reputation system for the Wikipedia. In *WWW '07: Proceedings of the 16th International World Wide Web Conference*, pages 261–270, May 2007.

[30] B. T. Adler, L. de Alfaro, and I. Pye. Detecting Wikipedia vandalism using WikiTrust. Technical report, Lab Report for PAN at CLEF, 2010.

[31] J. Antin and C. Cheshire. Readers are not free-riders: Reading as a form of participation on Wikipedia. In *CSCW '10: Proceedings of the 2010 ACM Conference on Computer Supported Cooperative Work*, pages 127–130, 2010.

[32] P. Ayers, C. Matthews, and B. Yates. *How Wikipedia Works*. No Starch, 2008.

[33] F. Bellomi and R. Bonato. Network analysis for Wikipedia. In *Wikimania '05: The First International Wikimedia Conference*, 2005.

[34] A. Bhattarai, V. Rus, and D. Dasgupta. Characterizing comment spam in the blogosphere through content analysis. In *CICS '09: Proc. of the IEEE Symposium on Computational Intelligence in Cyber Security*, pages 37–44, 2009.

[35] J. E. Blumenstock. Size matters: Word count of a measure of quality on Wikipedia. In *WWW '08: Proceedings of the 17th International Conference on the World Wide Web*, pages 1095–1096, 2008. (Poster paper).

[36] C. Breneman and C. Carter. Cluebot NG. `http://en.wikipedia.org/wiki/User:ClueBot_NG`.

[37] J. Broughton. *Wikipedia: The Missing Manual*. O'Reilly Media, 2008.

[38] 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: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1101–1110, 2008.

[39] M. Cataldo. Sources of error in distributed development projects: Implications for collaborative tools. In *CSCW '10: Proceedings of the Conference on Computer Supported Cooperative Work*, 2010.

[40] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon. I tube, you tube, everybody tubes: Analyzing the world's largest user generated content video system. In *IMC '07: Proc. of the Internet Measurement Conference*, 2007.

[41] A. Chen. Inside Facebook's outsourced anti-porn and gore brigade, where 'camel toes' are more offensive than 'crushed heads'. Gawker.com, `http://gawker.com/5885714/`, February 2012.

[42] S.-C. Chin, W. N. Streeta, P. Srinivasan, and D. Eichmann. Detecting Wikipedia vandalism with active learning and statistical language models. In *WICOW '10: The 4th Workshop on Info. Credibility on the Web*, 2010.

[43] D. Cosley, D. Frankowski, L. Terveen, and J. Riedl. Using intelligent task routing and contribution review to help communities build artifacts of lasting value. In *CHI '06: Proceedings of the SIGCHI Conference on Human Factors in Computing*, pages 1037–1046, 2006.

[44] T. Cross. Puppy smoothies: Improving the reliability of open, collaborative wikis. *First Monday*, 11(9), September 2006.

[45] L. Dabbish, C. Stuart, J. Tsay, and J. Herbsleb. Social coding in GitHub: Transparency and collaboration in an open software repository. In *CSCW '12: Proc. of the Conf. on Comp. Supported Cooperative Work*, 2012.

[46] H. Dai, Z. Nie, L. Wang, L. Zhao, J.-R. Wen, and Y. Li. Detecting online commercial intention (OCI). In *WWW'06: Proceedings on the 15th World Wide Web Conference*, 2006.

[47] M. Dale, A. Stern, M. Deckert, and W. Sack. Metavid.org: A social website and open archive of congressional video. In *DG'09: Proceedings of the 10th Conference on Digital Government Research, track on Social Networks: Making Connections between Citizens, Data and Government*, 2009.

[48] L. De Alfaro, A. Kulshreshtha, I. Pye, and B. T. Adler. Reputation systems for open collaboration. *Comm. ACM*, 54(8):81–87, August 2011.

[49] C. Dillow. DARPA's Vehicleforge.mil aims to crowdsource next-gen combat vehicles. *Popular Science (online)*, 2011. http://www.popsci.com/technology/article/2011-07/darpas-vehicleforgemil-aims-crowd-sourcing-next-gen-combat-vehicles.

[50] K. Dinakar, R. Reichart, and H. Lieberman. Modeling the detection of textual cyberbullying. In *Proceedings of International AAAI Conference on Weblogs and Social Media*, 2011.

[51] P. Dondio, S. Barrett, S. Weber, and J. M. Seigneur. Extracting trust from domain analysis: A case study on the Wikipedia project. In *Autonomic and Trusted Computing*, volume 4158, pages 362–373. Springer Berlin, 2006.

[52] J. Douceur. The Sybil attack. In *1st IPTPS*, March 2002.

[53] L. Edwards. Content filtering and the new censorship. In *ICDS '10: Proceedings of the Conference on Digital Society*, 2010.

[54] F. Flöck, D. Vrandečić, and E. Simperl. Revisiting reverts: Accurate revert detection in Wikipedia. In *HT '12: Proceedings of the 23rd ACM Conference on Hypertext and Social Media*, pages 3–12, 2012.

[55] P. K.-F. Fong and R. P. Biuk-Aghai. What did they do? Deriving high-level edit histories in wikis. In *WikiSym '10: Proceedings of the Sixth International Symposium on Wikis and Open Collaboration*, July 2010.

[56] A. Forte and A. Bruckman. Why do people write for Wikipedia? Incentives to contribute to open-content publishing. In *GROUP '05: Proceedings of the ACM Conference on Supporting Group Work*, 2005.

[57] A. Forte and C. Lampe. Defining, understanding, and supporting open collaboration: Lessons from the literature. *American Behavioral Scientist*, 2013.

[58] Y. Freund and L. Mason. The alternating decision tree algorithm. In *ICML'99: Proc. of the 16th Intl. Conference on Machine Learning*, pages 124–133, 1999.

[59] R. S. Geiger and A. Halfaker. Using edit sessions to measure participation in Wikipedia. In *CSCW '13: Proceedings of the 2013 Conference on Computer Supported Cooperative Work*, pages 861–870, 2013.

[60] R. S. Geiger and D. Ribes. The work of sustaining order in Wikipedia: The banning of a vandal. In *CSCW '10: Proceedings of the ACM Conference on Computer Supported Cooperative Work*, pages 117–126, 2010.

[61] L. Geng and H. J. Hamilton. Interestingness measures for data mining: A survey. *ACM Computer Surveys*, 38(9), 2006.

[62] J. Giles. Internet encyclopedias go head to head. *Nature*, 438, December 2005.

[63] E. Goldman. Wikipedia's labor squeeze and its consequences. *Journal of Telecommunications and High Technology Law*, 8, 2009.

[64] A. P. Goldstein. *The Psychology of Vandalism*. The Springer Series in Social Clinical Psychology. Springer, 1996.

[65] M. F. Goodchild. Citizens as sensors: The world of volunteered geography. *GeoJournal*, 69(4):211–221, 2011.

[66] W. Gordon. Google to lower web sites' ranking based on alleged copyright infringement. *Lifehacker.com*, August 2010. `http://lifehacker.com/5933776/google-now-factors-copyright-infringement-into-their-search-results`.

[67] T. L. Graves, A. F. Karr, et al. Predicting fault incidence using software change history. *IEEE Transactions on Software Engineering*, 26(7):653–661, 2000.

[68] D. Gray, D. Bowes, N. Davey, Y. Sun, and B. Christianson. Further thoughts on precision. In *EASE '11: Proceedings of the 15th Annual Conference on Evaluation & Assessment in Software Engineering*, pages 129–133, 2011.

[69] C. Grier, K. Thomas, V. Paxson, and M. Zhang. @ spam: The underground on 140 characters or less. In *CCS '10: Proceedings of the 17th ACM conference on Computer and Communications Security*, pages 27–37, 2010.

[70] A. Halfaker. Snuggle. `http://en.wikipedia.org/wiki/Wikpedia:Snuggle`.

[71] A. Halfaker. Wikipedia research: Vandal fighter work load. `http://meta.wikimedia.org/wiki/Research:Vandal_fighter_work_load`.

[72] 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*, 2013.

[73] A. Halfaker, A. Kittur, and J. Riedl. Don't bite the newbies: How reverts affect the quantity and quality of Wikipedia work. In *WikiSym '11: Proc. of the 7th Intl. Symposium on Wikis and Open Collaboration*, October 2011.

[74] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. Witen. The WEKA data mining software: An update. *SIGKDD Explor.*, 11(1), 2009.

[75] S. Hao, N. A. Syed, N. Feamster, A. G. Gray, and S. Krasser. Detecting spammers with SNARE: Spatio-temporal network-level automated reputation engine. In *Proceedings of USENIX Security*, 2009.

[76] M. Harpalani, M. Hart, S. Singh, R. Johnson, and Y. Choi. Language of vandalism: Improving Wikipedia vandalism detection via stylometric analysis.

In *HLT '11: Proceedings the 49th Meeting of the Association for Computational Linguistics, Human Language Technologies*, pages 83–88, 2011.

[77] C. Haythornthwaite. Democratic process in online crowds and communities. *Journal of Democracy and Open Government*, 4(2):160–170, 2012.

[78] S. Herring, K. Job-Sluder, R. Scheckler, and S. Barab. Searching for safety online: Managing "trolling" in a feminist forum. *The Information Society*, 18(5):371–384, 2002.

[79] P. Heymann, G. Koutrika, and H. Garcia-Molina. Fighting spam on social web sites: A survey of approaches and future challenges. *IEEE Internet Computing*, 11(6):36–45, 2007.

[80] K. Y. Itakura and C. L. Clarke. Using dynamic Markov compression to detect vandalism in the Wikipedia. In *SIGIR '09: Proc. of the 32nd Intl. ACM SIGIR Conference on Research and Development in Info. Retrieval*, 2009.

[81] S. Javanmardi, Y. Ganjisaffar, C. Lopes, and P. Baldi. User contribution and trust in Wikipedia. In *CollaborateCom '09: Proceedings of the 5th International Conference on Collaborative Computing: Networking, Applications and Worksharing*, pages 1–6, 2009.

[82] C. D. Jensen. Security in wiki-style authoring systems. In *IFIPTM '09: Proceedings of Trust Management III*, pages 81–98, 2009.

[83] A. Jøsang. A logic for uncertain probabilities. *International Journal of Uncertainty, Fuzziness, and Knowledge-Based Systems*, 9(3):279–311, June 2001.

[84] A. Jøsang, R. Ismail, and C. Boyd. A survey of trust and reputation systems for online service provision. *Decision Support Systems*, 43(2), 2007.

[85] A. Jøsang, C. Keser, and T. Dimitrakos. Can we manage trust? In P. Herrmann, V. Issarny, and S. Shiu, editors, *Trust Management*, volume 3477 of *Lecture Notes in Computer Science*, pages 93–107. 2005.

[86] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. Spamalytics: An empirical market analysis of spam marketing conversion. In *CCS '08: Proceedings of the 14th ACM Conference on Computer and Communications Security*, pages 3–14, 2008.

[87] R. Kohavi and G. H. John. Wrappers for feature subset selection. *Artificial Intelligence Journal*, 97(1):273–324, 1997.

[88] T. Kriplean, I. Beschastnikh, and D. W. McDonald. Articulations of wikiwork: Uncovering valued work in Wikipedia through barnstars. In *CSCW '08: Proc. of the Conf. on Computer Supported Cooperative Work*, pages 47–56, 2008.

[89] H. Lee. Behavioral strategies for dealing with flaming in an online forum. *The Sociological Quarterly*, 46(2):385–403, 2005.

[90] B. Leuf and W. Cunningham. *The Wiki Way: Quick Collaboration on the Web*. Addison-Wesley, 2001.

[91] K. Levchenko, N. Chachra, B. Enright, M. Felegyhazi, C. Grier, T. Halvorson, C. Kanich, C. Kreibich, H. Liu, D. McCoy, A. Pitsillidis, N. Weaver, V. Paxson, G. M. Voelker, and S. Savage. Click trajectories: End-to-end analysis of the spam value chain. In *Proc. of the IEEE Symp. on Security and Privacy*, 2011.

[92] K. Linthicum. Wikipedia blocks access from Church of Scientology in L.A. *LA Times*, June 5 2009. `http://articles.latimes.com/2009/jun/05/business/fi-wikipedia-scientology5`.

[93] B. Livshits and T. Zimmermann. Dynamine: Finding common error patterns by mining software revision histories. In *ESEC/FSE*, pages 296–305, 2005.

[94] I. Lykourentzou, F. Dagka, K. Papadaki, G. Lepouras, and C. Vassilakis. Wikis in enterprise settings: A survey. *Enterprise Info. Systems*, 6(1):1–53, 2012.

[95] D. L. McGuinness, H. Zeng, P. D. Silva, L. Ding, D. Narayanan, and M. Bhaowal. Investigation into trust for collaborative information repositories: A Wikipedia case study. In *Proceedings of the Workshop on Models of Trust for the Web*, 2006.

[96] M. McPherson, L. Smith-Lovin, and J. M. Cook. Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27:415–444, 2001.

[97] J. Merante. UK Natl. Portrait Gallery threatens Wikipedia user over public domain images. `http://creativecommons.org/weblog/entry/15764`, July 2009.

[98] G. Mishne, D. Carmel, and R. Lempel. Blocking blog spam with language model disagreement. In *AIRWeb'05: Proceedings of the Workshop on Adversarial Information Retrieval on the Web*, 2005.

[99] S. M. Mola-Velasco. Wikipedia vandalism detection. Master's thesis, Universidad Politecnica de Valencia, September 2011.

[100] M. Motoyama, K. Levchenko, C. Kanich, D. McCoy, G. M. Voekler, and S. Savage. Re: CAPTCHAs - Understanding CAPTCHA-solving services in an economic context. In *USENIX Security '10: Proceedings of the 19th USENIX Security Symposium*, August 2010.

[101] P. Neis, M. Goetz, and A. Zipf. Towards automatic vandalism detection in OpenStreetMap. *ISPRS Intl. J. of Geo-Information*, 1(3):315–332, 2012.

[102] Y. Niu, Y. min Wang, H. Chen, M. Ma, and F. Hsu. A quantitative study of forum spamming using context-based analysis. In *NDSS'07: Proceedings of Network and Distributed System Security Symposium*, 2007.

[103] A. Ntoulas, M. Najork, M. Manasse, and D. Fetterly. Detecting spam web pages through content analysis. In *WWW'06: Proceedings of the 15th World Wide Web Conference*, 2006.

[104] J. Orlowitz. Does Wikipedia pay? The communicator: Phil Gomes. *Wikipedia Signpost*, May 7 2012.

[105] J. Orlowitz. Does Wikipedia pay? The consultant: Pete Forsyth. *Wikipedia Signpost*, April 30 2012.

[106] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical Report 422, Stanford, 1999.

[107] B. Pershing. Kennedy, Byrd the latest victims of Wikipedia errors. *Washington Post*, January 2009. `http://voices.washingtonpost.com/capitol-briefing/2009/01/kennedy_the_latest_victim_of_w.html`.

[108] G. Pisano and R. Verganti. Which kind of collaboration is right for you? *Harvard Business Review*, 86(12):78–86, December 2008.

[109] S. Pogatchnik. Student hoaxes world's media on Wikipedia. *MSNBC News*, April 2009. `http://www.msnbc.msn.com/id/30699302/`.

[110] M. Potthast. Crowdsourcing a Wikipedia vandalism corpus. In *SIGIR '10: Proc. of the 33rd Intl. ACM SIG Info. Retrieval Conference*, 2010.

[111] M. Potthast and T. Holfeld. Overview of the 2nd International competition on Wikipedia vandalism detection. In *PAN-CLEF 2011 Labs and Workshops: Plagiarism, Authorship, and Social Software Misuse*, 2011.

[112] M. Potthast, B. Stein, and R. Gerling. Automatic vandalism detection in Wikipedia. In *Advances in Information Retrieval*, pages 663–668, 2008.

[113] M. Potthast, B. Stein, and T. Holfeld. Overview of the 1st International competition on Wikipedia vandalism detection. In *PAN-CLEF 2010 Labs and Workshops: Plagiarism, Authorship, and Social Software Misuse*, 2010.

[114] R. Priedhorsky, J. Chen, S. K. Lam, K. Panciera, L. Terveen, and J. Riedl. Creating, destroying, and restoring value in Wikipedia. In *GROUP '07: Proc. of the 2007 Intl. Conference on Supporting Group Work*, pages 259–268, 2007.

[115] L. Rassbach, T. Pincock, and B. Mingus. Exploring the feasability of automatically rating online article quality. Unpublished.

[116] N. Reavley, A. Mackinnon, A. Morgan, M. Alvarez-Jimenez, S. Hetrick, E. Killackey, B. Nelson, R. Purcell, M. Yap, and A. Jorm. Quality of information sources about mental disorders: A comparison of Wikipedia with centrally controlled web and printed sources. *Psychol. Med.*, 42(8), 2012.

[117] M. Sahamia, S. Dumais, D. Heckerman, and E. Horvitz. A Bayesian approach to filtering junk email. In *Proceedings of the AAAI Workshop on Learning for Text Categorization*, 1998.

[118] J. Schneider, A. Passant, and S. Decker. Deletion discussions in Wikipedia: Decision factors and outcomes. In *Wikisym '12: Proceedings of the 8th International Symposium on Wikis and Open Collaboration*, August 2012.

[119] J. Seigenthaler. A false Wikipedia 'biography'. *USA Today*, November 2005. `http://www.usatoday.com/news/opinion/editorials/2005-11-29-wikipedia-edit_x.htm`.

[120] Y. Shin, M. Gupta, and S. Myers. The nuts and bolts of a forum spam automator. In *LEET'11: Proceedings of the 4th USENIX Workshop on Large-Scale Exploits and Emergent Threats*, 2011.

[121] K. Smets, B. Goethals, and B. Verdonk. Automatic vandalism detection in Wikipedia: Towards a machine learning approach. In *WikiAI '08: Proceedings of the AAAI Workshop on Wikipedia and Artificial Intelligence*, 2008.

[122] S. O. Sood, E. F. Churchill, and J. Antin. Automatic identification of personal insults on social news sites. *Journal of the American Society for Information Science and Technology*, 2011.

[123] B. Stein. Webis Group Weimar corpora. `http://www.webis.de/research/corpora/`. (Online source for vandalism corpora).

[124] B. Stone. Policing the web's lurid precincts. *The New York Times*, page B1, July 18 2010. `http://www.nytimes.com/2010/07/19/technology/19screen.html`.

[125] B. Stvilia, M. B. Twidale, L. C. Smith, and L. Gasser. Assessing information quality of a community-based encyclopedia. In *Proceedings of the International Conference on Information Quality*, pages 442–454, 2005.

[126] P. Suber. Open for edits. *SPARC Open Access Newsletter*, March 2011. `http://www.arl.org/sparc/publications/articles/open-for-edits.shtml`.

[127] B. Suh, G. Convertino, E. H. Chi, and P. Pirolli. The singularity is not near: Slowing growth of Wikipedia. In *WikiSym '09: Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, 2009.

[128] D. Taraborelli. Beyond notability: Collective deliberation on content inclusion in Wikipedia. In *SASOW '10: Proceedings of the 4th International Workshop on Self-Adaptive and Self-Organizing Systems*, pages 122–125, September 2010.

[129] B. E. Ur and V. Ganapathy. Evaluating attack amplification in online social networks. In *W2SP'09: Workshop on Web 2.0 Security and Privacy*, 2009.

[130] S. M. M. Velasco. Wikipedia vandalism detection through machine learning: Feature review and new proposals. In *PAN-CLEF '10: Notebook Papers on Uncovering Plagiarism, Authorship, and Social Software Misuse*, 2010.

[131] F. B. Viégas, M. Wattenberg, and K. Dave. Studying cooperation and conflict between authors with history flow visualizations. In *CHI '04: Proc. of the SIGCHI Conf. on Human Factors in Computing Systems*, pages 575–582, 2004.

[132] M. Völkel, M. Krötzsch, D. Vrandecic, H. Haller, and R. Studer. Semantic Wikipedia. In *WWW '06: Proceedings of the 15th International Conference on the World Wide Web*, pages 585–594, 2006.

[133] R. Wang and D. Strong. Beyond accuracy: What data quality means to data consumers. *Journal of Management Information Systems*, 12(4):5–34, 1996.

[134] W. Y. Wang and K. McKeown. "Got you!": Automatic vandalism detection in Wikipedia with web-based shallow syntactic-semantic modeling. In *COLING' 10: Proc. of the 23rd International Conf. on Computational Linguistics*, 2010.

[135] A. G. West. WikiAudit. `http://en.wikipedia.org/wiki/Wikipedia:WikiAudit`.

[136] A. G. West. STiki: A vandalism detection tool for Wikipedia. `http://en.wikipedia.org/wiki/Wikipedia:STiki`, 2010.

[137] A. G. West, A. Agrawal, P. Baker, B. Exline, and I. Lee. Autonomous link spam detection in purely collaborative environments. In *WikiSym '11: Proc. of the 7th Intl. Symposium on Wikis and Open Collaboration*, 2011.

[138] A. G. West, A. J. Aviv, J. Chang, and I. Lee. Spam mitigation using spatio-temporal reputations from blacklist history. In *ACSAC '10: Proc. of the 26th Annual Computer Security Applications Conference*, pages 161–170, 2010.

[139] A. G. West, J. Chang, K. Venkatasubramanian, O. Sokolsky, and I. Lee. Link spamming Wikipedia for profit. In *CEAS '11: Proc. of the 8th Collaboration, Electronic Messaging, Anti-Abuse, and Spam Conference*, September 2011.

[140] A. G. West, J. Chang, K. K. Venkatasubramanian, and I. Lee. Trust in collaborative web applications. *Future Generation Computer Systems, special section on Trusting Software Behavior*, 28(8):1238–1251, October 2012.

[141] A. G. West, P. Hayati, V. Potdar, and I. Lee. Spamming for science: Active measurement in Web 2.0 abuse research. In *WECSR '12: Proceedings of the Third Workshop on Ethics in Computer Security Research*, March 2012.

[142] A. G. West, S. Kannan, and I. Lee. Detecting Wikipedia vandalism via spatio-temporal analysis of revision metadata. In *EUROSEC '10: Proceedings of the Third European Workshop on System Security*, pages 22–28, 2010.

[143] A. G. West and I. Lee. Multilingual vandalism detection using language-independent & ex post facto evidence. In *PAN-CLEF '11: Notebook Papers on Uncovering Plagiarism, Authorship, and Social Software Misuse*, 2011.

[144] A. G. West and I. Lee. What Wikipedia deletes: Characterizing dangerous collaborative content. In *WikiSym '11: Proceedings of the Seventh International Symposium on Wikis and Open Collaboration*, pages 25–28, 2011.

[145] A. G. West and I. Lee. Open wikis and the protection of institutional welfare. Research bulletin, EDUCAUSE Center for Applied Research, February 2012.

[146] A. G. West and I. Lee. Towards content-driven reputation for collaborative code repositories. In *WikiSym '12: Proceedings of the Eighth International Symposium on Wikis and Open Collaboration*, August 2012.

[147] A. G. West and Wikipedia editors. Examining the popularity of Wikipedia articles: Catalysts, trends, and applications. *Wikipedia Signpost*, 9(5), 2013.

[148] Wikipedia contributors and A. G. West. Compiling "Generic Chinese Knock-off Spam" reports. http://en.wikipedia.org/wiki/User:West.andrew.g/GCKS.

[149] J. Winter. Wikipedia distributing child porn, co-founder tells FBI. *FoxNews.com*, April 27, 2010. http://www.foxnews.com/scitech/2010/04/27/wikipedia-child-porn-larry-sanger-fbi/.

[150] C. Wisniewski. Wikipedia hacked - Footballers need help in bed? *Sophos Naked Security Blog*, July 2010. http://nakedsecurity.sophos.com/2010/07/13/wikipedia-hacked-footballers-need-help-in-bed/.

[151] I. H. Witten and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, San Francisco, 2nd edition, 2005.

[152] T. Wöhner and R. Peters. Assessing the quality of Wikipedia articles with lifecycle based metrics. In *WikiSym '09: Proceedings of the 5th International Symposium on Wikis and Open Collaboration*, 2009.

[153] G. Xiang, B. Fan, L. Wang, J. Hong, and C. Rose. Detecting offensive tweets via topical feature discovery over a large scale Twitter corpus. In *CIKM '12: Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 1980–1984, 2012.

[154] T. Yasseri, R. Sumi, and J. Kertész. Circadian patterns of Wikipedia editorial activity: A demographic analysis. *PloS one*, 7(1), 2012.

[155] H. Zeng, M. A. Alhossaini, L. Ding, R. Fikes, and D. L. McGuinness. Computing trust from revision history. In *Proceedings of the International Conference on Privacy, Security, and Trust*, November 2006.

[156] X. Zhu and S. Gauch. Incorporating quality metrics in centralized/distributed information retrieval on the World Wide Web. In *SIGIR '10: Proc. of the Conf. on Research and Development in Info. Retrieval*, pages 288–295, 2000.