2012

Trust in Collaborative Web Applications

Andrew G. West
University of Pennsylvania, westand@cis.upenn.edu

Jian Chang
University of Pennsylvania, jianchan@cis.upenn.edu

Krishna Venkatasubramanian
University of Pennsylvania, vkris@cis.upenn.edu

Insup Lee
University of Pennsylvania, lee@cis.upenn.edu

Recommended Citation

Andrew G. West, Jian Chang, Krishna Venkatasubramanian, and Insup Lee, "Trust in Collaborative Web Applications", *Future Generation Computer Systems* 28(8), 1238-1251. January 2012. http://dx.doi.org/10.1016/j.future.2011.02.007

Based in part on UPENN MS-CIS-10-33 http://repository.upenn.edu/cis_reports/943/

Trust in Collaborative Web Applications

Abstract

Collaborative functionality is increasingly prevalent in web applications. Such functionality permits individuals to add - and sometimes modify - web content, often with minimal barriers to entry. Ideally, large bodies of knowledge can be amassed and shared in this manner. However, such software also provide a medium for nefarious persons to operate. By determining the extent to which participating content/agents can be trusted, one can identify useful contributions. In this work, we define the notion of trust for *Collaborative Web Applications* and survey the state-of-the-art for calculating, interpreting, and presenting trust values. Though techniques can be applied broadly, Wikipedia's archetypal nature makes it a focal point for discussion.

Keywords

Collaborative web applications, trust, reputation, Wikipedia

Comments

Based in part on UPENN MS-CIS-10-33 http://repository.upenn.edu/cis_reports/943/

Trust in Collaborative Web Applications

Andrew G. West, Jian Chang,
Krishna K. Venkatasubramanian, Insup Lee

Department of Computer and Information Science
University of Pennsylvania – Philadelphia, PA

{*westand, jianchan, vkris, lee*}*@cis.upenn.edu*

Abstract

Collaborative functionality is increasingly prevalent in web applications. Such functionality permits individuals to add – and sometimes modify – web content, often with minimal barriers to entry. Ideally, large bodies of knowledge can be amassed and shared in this manner. However, such software also provide a medium for nefarious persons to operate. By determining the extent to which participating content/agents can be trusted, one can identify useful contributions. In this work, we define the notion of trust for *Collaborative Web Applications* and survey the state-of-the-art for calculating, interpreting, and presenting trust values. Though techniques can be applied broadly, Wikipedia's archetypal nature makes it a focal point for discussion.

Keywords: Collaborative web applications, trust, reputation, Wikipedia

1. Introduction

Collaborative Web Applications (CWAs) have become a pervasive part of the Internet. Topical forums, blog/article comments, open-source software development, and *wikis* are all examples of CWAs – those that enable a community of end-users to interact or cooperate towards a common goal. The principal aim of CWAs is to provide a common platform for users to share and manipulate content. In order to encourage participation, most CWAs have no or minimal barriers-to-entry. Consequently, anyone can be the source of the content, unlike in more traditional models of publication. Such diversity of sources brings the trustworthiness of the content into question. For Wikipedia, ill-intentioned sources have led to several high-profile incidents [41, 52, 59].

Another reason for developing an understanding of trust in CWAs is their potential influence. CWAs are relied upon by millions of users as an information source. Individuals that are not aware of the pedigree/provenance of the sources of information may implicitly assume them authoritative. For example, journalists have erroneously considered Wikipedia content authoritative and reported false statements [53]. While unfortunate, tampering with CWAs could have far more severe consequences – consider Intellipedia (a *wiki* for U.S. intelligence agencies), on which military decisions may be based.

Although many CWAs exist, the most fully-featured model is the *wiki* [45] – a web application that enables users to create, add, and delete from an interlinked content network. On the assumption that all collaborative systems are a reduction from the *wiki* model (see Sec. 2.1), we use it as the basis for discussion. No doubt, the "collaborative encyclopedia", Wikipedia [10], is the canonical example of a *wiki* environment. Although there is oft-cited evidence defending the accuracy of Wikipedia articles [34], it is negative incidents (like those we have highlighted) that tend to dominate external perception. Further, it is not just *possible* to 'attack' collaborative applications, but certain characteristics make it *advantageous* to attackers. For example, content authors have access to a large readership that they did not have to accrue. Moreover, the open-source nature of much *wiki* software makes security functionality transparent.

Given these vulnerabilities and incidents exploiting them, it should come as no surprise that the identification of trustworthy agents/content has been the subject of many academic writings and on-*wiki* applications. In this paper we present a survey of these techniques. We classify our discussion into two categories: (1) *Trust computation*, focuses on algorithms to compute trust values and their relative merits. (2) *Trust usage*, surveying how trust values can be conveyed to end-users or used internally to improve application security. The combination of effective trust calculation and presentation holds enormous potential for building trustworthy CWAs in the future.

The remainder of this paper is structured as follows. Sec. 2 establishes the terminology of collaborative web applications, attempts to formalize the notion of 'trust', and discusses the granularity of trust computation. Sec. 3 describes various trust computation techniques, and Sec. 4 examines their relative strengths and weaknesses. Sec. 5 discusses how trust information can be interpreted and used for the benefit of end-users and the collaborative software itself. Finally, concluding remarks are made in Sec. 6.

2. Background & Terminology

In this section, we standardize terminology for the remainder of this work. Further, we examine/define the meaning of 'trust' in a collaborative environment and claim that regardless of the entity for which trust is calculated, the notion is transferrable to other participating entities.

2.1. Defining a Collaborative Web Application

Put simply, a *collaborative web application (CWA)* is one in which two or more *users* or *contributors* operate in a centrally shared space to achieve a common goal. Taken as a whole, the user-base is often referred to as a *community.* Most Web 2.0 applications such as *wikis* (*e.g.,* Wikipedia), video sharing (*e.g.*, YouTube), and social networking (*e.g.*, Facebook) have sufficient collaborative functionality to be considered CWAs. A distinguishing factor between CWAs

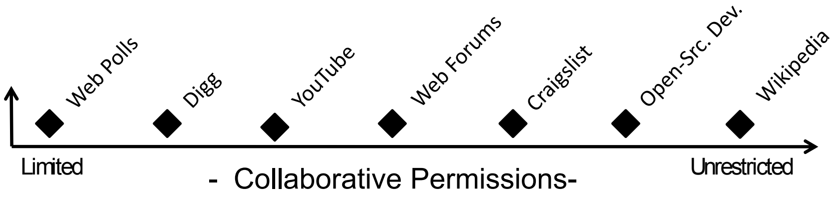


Figure 1: Accessibility among popular CWAs

and traditional web properties is their *accessibility*, that is, the extent to which read/write/create/delete permissions are extended to the community. Content in CWAs is driven by the user-base, and the host's primary function is only to facilitate the sharing of information.

CWAs can be classified along several dimensions:

- Barrier to Entry: In order to ensure information security and/or disincentivize disruptive participants, it is often necessary to *define the community* for CWAs. Thus, barriers-to-entry are introduced. Many CWAs such as *wikis* have effectively no barrier to entry (allowing anonymous editing). Many other well known communities have minimal barriers, such as CAPTCHA solves or required (but free) registrations. At the other extreme are corporate SVN repositories and Intellipedia, which are not open communities, but limit access to members of some organization.

- Accessibility: Accessibility of a CWA is defined by *the permissions available to the community.* One of the most constrained examples of CWA accessibility is a "web poll", where users can select among pre-defined options and submit a response which is stored and displayed in aggregate fashion (*e.g.,* a graph). A more permissive example are "blog comments", where readers can append content of their choosing on existing posts. At the extreme of this continuum lies the "*wiki* philosophy" [45], which in its purest[1] form gives its users unrestricted read/write/create/delete permissions over all content. Figure 1 visualizes some well-known CWAs with respect to their varying accessibility levels.

- Moderation: As CWAs generally have low barriers-to-entry, some form of moderation may be required to uphold certain standards. Thus, moderation in a CWA is defined by *who has permissions outside of those available to the general community.* On the video-sharing site YouTube, for instance, it is the hosts who intervene in resolving copyright issues. In contrast, moderators on Wikipedia are community-elected, with the parent organization, the Wikimedia Foundation, rarely becoming involved.

[1] Wikipedia is not a *wiki* in the purest sense. The realities of operating a web presence of that magnitude have led to the installation of minimal protections.

In this paper, we are primarily concerned with CWAs with (1) low to minimal barriers-of-entry, (2) comprehensive accessibility permissions, and (3) moderated by community consensus. A CWA meeting these requirements is Wikipedia [10], which is the most well-known application of the *wiki* model. It is reasonable to assume that all other types of CWAs must operate within a sub-space of its capabilities. Further, Wikipedia is a *de facto* standard in evaluating collaborative methodologies. For these reasons, our discussion moving forward will focus heavily on the *wiki* model and Wikipedia, in particular.

We believe that an in-depth case-study of Wikipedia trust is preferable to examining trust issues across the breadth of CWAs. A single point of focus permits coherent and subtle discussion – all of which is applicable to reductions of the *wiki* model (*i.e.,* all other CWAs).

2.2. Related Work

Before focus shifts to the *wiki* model, however, we believe it helpful to highlight related works regarding trust calculation for CWAs outside the *wiki* realm.

For example, much related work resides in the *web services* domain, where atomic tasks/data can be performed/obtained over a network protocol. More interesting is when these services are composed into *service-oriented architectures* or *mash-ups* to provide higher-level services. Whenever a service is built from components spanning organizational boundaries, trust becomes an issue. Much as multiple Wikipedia authors might contribute to an article, multiple service providers are collaborating towards a single result. Maximilien and Singh [46] were among the first to describe trust in such environments, although work by Dragoni [30] provides a more state-of-the-art survey. Meanwhile, Yahyaoui [67] discusses game-theoretic trust models for such settings.

The notion of *grid computing* is analogous to such service-driven architectures – except that it is raw computational resources which are being amassed and shared. The need for grid-trust has been acknowledged by [31]. Papaioannou and Stamoulis [51] observe that it is not easy to decompose the individual contributions that form such collaborative efforts. Thus, it is difficult to identify free-riding or low-performing agents in a collaborative grid environment. To this end, the authors' develop and evaluate a reputation-based mechanism enabling the grid-service broker to monitor such dynamics. The proposed scheme allows the broker to provide recommendations about which agents should be utilized.

There are also notable works focusing specifically on *collaborative knowledge grids.* Targeting the problem of content-quality assessment, Zhuge and Liu [70] proposes a fuzzy collaborative strategy, combining objective and subjective assessment. The authors' approach integrates the criteria used in website assessment, knowledge organization, and expert-agent cooperation. At a different level, CFMS [60] provides a data management solution for collaborative knowledge grids. CFMS allows a user to navigate the history and relationships between "knowledge files" using a web browser – and to check-in/check-out files in a similar fashion. Focusing on knowledge grids for geo-spatial information, [27] presents an architecture to improve the availability of geo-spatial data resources and also to ease their deployment into information infrastructures.

Having briefly explored these alternative CWAs, our focus now returns to the *wiki* model, on which the remainder of this paper focuses. Because the *wiki* model is the purest collaborative paradigm, knowledge garnered through discussion of *wikis* should be broadly-applicable for all CWAs.

2.3. Wiki Terminology

Given our focus on *wiki* environments, it is helpful to standardize their terminology. A *wiki* consists of a set of content *pages* or *articles.* Content in articles evolves through a series of *revisions* or *edits*, which taken in series form a *version history*, $R = \{V_0, V_1, V_2 \ldots V_n\}$. Though it is possible to browse an article's version history, it is the most recent version, V_n that is displayed by default. A special form of edit called a *revert* or *undo* creates a new version, but simply duplicates the content of a previous one. Such edits are of interest because they are often used to restore content after damage.

An edit is made by exactly one *editor* or *contributor* or *author*, belonging to the *community* of users. Authors may be assigned persistent identifiers that allow their contributions to be tracked through time. Alternatively, some systems permit authors to edit in a more transient fashion (see Sec. 4.1).

Individual pages within a *wiki* can be interconnected via hyperlinks, and such links are termed *internal links* or *wikilinks.* These should be distinguished from hyperlinks which lead users to destinations outside the wiki environment, known as *external links.*

2.4. Defining Collaborative Trust

As Jøsang [39] notes, the meaning of *trust* varies dramatically throughout research literature. The collaborative domain is no exception. We first examine how existing literature approaches the notion of trust. Then, we propose our own definition which improves upon the *status quo.*

It should be emphasized that we are primarily interested in trust as it pertains to the content of a collaborative system (and the participants who generate it). The infrastructure which enables this is not a point of focus. This does not mean that trust values are not influential in the software design process. On the contrary, these values permit designers to make software changes which enhance application security and the end-user experience (see Sec. 5.2).

Distinction should also be made between the broad notion of trust and the very specific notion of *trust management* as introduced by Blaze *et al.* [21]. Trust management refers to a formal access-control model built upon delegation of permissions between trusted entities, often using cryptographic fundamentals. While trust values can be used for access-control, their dynamic, quantifiable, and predictive properties permit a wider range of use-cases.

2.4.1. Existing Definitions in Literature

Existing writings often handle the notion of trust generically, giving readers little insight into precisely what properties the calculated values are meant to quantify. The need for a rigorous and objective definition is usually side-stepped

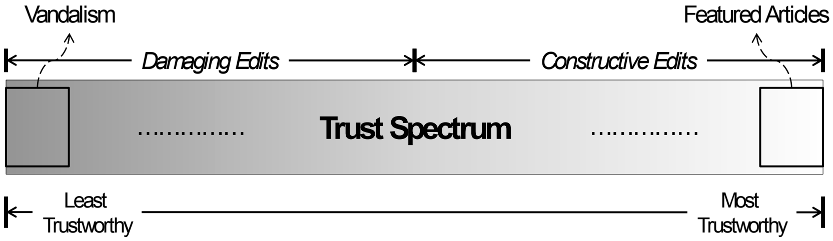


Figure 2: Collaborative trust spectrum

with the choice of evaluation criteria. Typically, evaluation concentrates on the most objective subset of the trust spectrum, (*e.g.,* vandalism, see Fig. 2); or divides the spectrum at coarse granularity.

For Wikipedia-based analysis, vandalism detection is the most prominent example of the first technique. *Vandalism* is defined to be any edit which exhibits ill-intentions or gross negligence on the part of an editor. Vandalism is the least trustworthy of all content and lends itself to objective labeling. The second evaluation strategy divides the trust spectrum at coarse granularity. For example, "Featured Articles", those tagged to be of high quality, will be compared against those known to be of poor quality.

While these two evaluation techniques represent the current state-of-the-art, they are less than ideal. First, vandalism detectors operate on a subset of the trust problem, so it remains to be seen if the same metrics are meaningful at the far-right of the trust spectrum (Fig. 2). That is, can the same criteria be applied to distinguish mediocre edits from quality ones? Indeed, it would seem a holistic measurement of trust might be more complex.

Second, treating trust as a two-class problem seems inappropriate as it captures no subtleties. It is unsurprising that good articles are usually longer than poor ones. However, article length may be a poor comparator among a set of reasonable articles. Lastly, both approaches are able to rely on community-based labeling, allowing author's to side-step the need for precise definitions regarding how content should be tagged.

2.4.2. A Proposal for Defining Trust

Given these deficiencies, we now propose a more rigorous definition of trust. We define trust in collaborative content as *the degree to which content quality meets the community's requirements*. Thus, trust must be measured through the subjective lens of the community consensus on which it resides.

In order to reason about trust in CWAs, it must be formalized. Our formalism of content trust builds on two properties: (1) the measurement of information quality, and (2) the subjective interpretation of information quality by a community. Consequently, we identify trust as an 8-dimensional vector:

$$\begin{gathered}[\textsc{Scope}, \textsc{Accuracy}, \textsc{Source}, \textsc{Volatility}, \textsc{Cohesiveness}, \\ \textsc{Comprehensiveness}, \textsc{Timeliness}, \textsc{Neutrality}]\end{gathered} \tag{1}$$

Before we describe each of these properties in greater detail, some general commentary is necessary. For the discussion herein, where Wikipedia is a primary focus, we believe it to be the case that *all* eight properties are appropriate measures. For other CWAs, it is the community expectation that defines which metrics are *relevant*, and of those, the *polarity* of their interpretation.

For example, for a fictional book being collaboratively authored, notions like accuracy and timeliness might have little bearing. Further, even when a community believes that a measure is relevant, it may be the case that "poor" measures are desirable. For example, consider Encyclopedia Dramatica [3], a *wiki* which parodies Wikipedia by encouraging biased and offensive content. There, the most "trustworthy" contributions are those which are *not* accurate. Similarly, a politically-grounded *wiki* might trust content which is *not* neutral.

With this in mind, we now describe the eight properties in greater detail:

1. Scope: Content should appropriately scoped. For example, Wikipedia is an online encyclopedia that enforces notoriety requirements for topics.
2. Accuracy: If content intends to be factual, then it should be rooted in truth, without misinforming or deceiving readers.
3. Source: If content intends to be factual, then claims should be referenced and verifiable via reliable and high-quality sources.
4. Volatility: The extent to which content is stable.
5. Cohesiveness: Quality of writing and presentation style.
6. Comprehensiveness: The *breadth* and *depth* of topic examination.
7. Timeliness: The currency of the content (*i.e.,* "is it up-to-date?").
8. Neutrality: The degree of bias in presentation.

The metrics of this vector are drawn from information-quality literature and are quite *qualitative* in nature. To calculate actual trust values with mathematical rigor, it becomes necessary to *quantify* these properties. Existing literature [62, 69] demonstrates how quantification can be performed, although it is difficult to assess the performance of those attempts. The statement of precise mathematical definitions for our metrics is beyond the scope of this work. However, on the assumption such quantification can take place, our trust vector is given greater attention in Sec. 3.1.

2.5. On the Associativity of Collaborative Trust

The methodologies we will examine in the coming section calculate trust values for either (1) articles, (2) article fragments, (3) revisions, or (4) contributors. We assume that these entities have an associative trust relationship. That is, if one has trust values for any one of these sets, than this is sufficient to calculate trust values for the other three types. For example, the trust values of all edits made by a particular author should speak to the trust of that author.

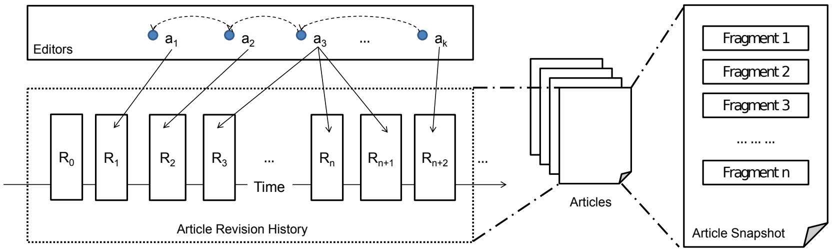


Figure 3: Relationships between *wiki* entities

Similarly, the trust of all text fragments of an article should speak to the trust value of that article. Thus, all collaborative trust systems are calculating at the same granularity and can be treated as comparable. Figure 3 visualizes the relationship between these different entities.

What is not precisely defined are the mathematical functions that define these associative relationships. Occasionally, systems define these in an application-specific manner. On the whole, we consider this to be outside the scope of this work and an open research question.

3. Collaborative Trust Algorithms

In this section, we overview trust computation proposals from literature. In particular, we emphasize four domains of research, and choose to highlight a seminal paper in each domain:

1. Content Persistence (PERSIST): Building on [68], Adler *et al.* [17, 18] propose a system whereby the persistence of an author's content determines his/her reputation (trust value). In turn, author reputation can speak to the quality/trust of new content authored by that contributor.
2. Natural-Language Processing (NLP): Akin to the use of NLP in email spam detection [58], the proposal of Wang *et al.* [64] uses language features to distinguish damaging edits from quality ones.
3. Metadata Properties (META): Just as the SNARE system [36] did for email spam, Stvilia *et al.* [62] identify poor contributions by looking at the *metadata* for an edit – properties unrelated to the linguistics of the content (*e.g.,* article size, time-stamp, account age, *etc.*).
4. Citation Quantity (CITE): Based on well-known algorithms for search-engine ranking [42, 50], the work of McGuinness *et al.* [47] proposes that pages with a large number of incoming links (internal or external of the *wiki*) are likely to be reliable resources.

A motivating factor in the creation of this taxonomy was its exhaustive coverage of related works. To the best of our knowledge, there are no *wiki*-centric trust proposals available (at the time of this writing) that cannot be classified in this

Approach		Strength	Weakness
Content-persist		Implicit feedback mechanism holds f-back providers accountable	Difficulty with Sybil and new users. Reliant on hindsight
NLP	Lexical	Regexps easy to implement, modify, and understand	Evadable by obfuscating or avoiding poor language
	n-gram	Find unusual or bad text w/o manual rules	Processing topic-specific corpora is CPU expensive
Metadata-based		Size/diversity of available feature space	Properties are “a level removed” from content
Citation-based		Calculation breadth makes evasion difficult	Unclear if citation action actually speaks to article trust

Table 1: Signature strengths and weaknesses of approaches

scheme (see Fig. 11). While future proposals may fall outside these bounds, we believe it sufficient and complete insofar as this is a survey work.

Moving forward, we will begin by describing how each of the four trust systems fulfill the multi-dimensional trust definition proposed in Sec. 2.4.2. Then, we will summarize the algorithmic function of each proposal, before comparing these proposals based on their relative merits. Table 1 summarizes the characteristic strengths and weaknesses of each approach (and later, Figure 11 summarizes the related works and research timeline for each approach).

3.1. Existing Systems and Proposed Trust Definition

Given that the algorithms we are about to overview were authored prior to our proposed definition in Sec. 2.4, we believe it important to identify how these techniques map to our definition.

Let $M = [m_1, m_2, ...m_8]$ be the set representing the eight metrics of information quality. Trust computation algorithms identify a set of quantitative values $Q = [q_1, ...q_n]$ and a subjective mapping Δ such that $\Delta : Q \rightarrow M'$, where $M' \subset M$. That is to say, for some metric(s) in M', there is at least one quantitative value in Q that speaks to it. Note that the mapping Δ has not been explicitly defined in the original papers, and Table 2 presents our own subjective attempt at doing so.

3.2. Trust Computation Algorithms

3.2.1. Content-driven Trust

Approach: As detailed by Adler *et al.* [17, 18], content-persistence trust is built on the intuition that the survival/removal/restoration of text fragments in subsequent revisions speaks to the trust of that fragment and to the reputation of its author. Content which survives future revisions, especially those of reputable authors, is likely to be trustworthy. Content which is removed but eventually restored is also trustworthy, but content which remains deleted speaks poorly of that content and its contributor.

Metric	PERSIST	NLP	META	CITE
Scope	✓		✓	✓
Accuracy	✓	✓	✓	✓
Source	✓		✓	✓
Volatility	✓		✓	✓
Cohesiveness	✓	✓	✓	✓
Comprehensiveness			✓	✓
Timeliness	✓		✓	✓
Neutrality	✓		✓	✓

Table 2: Mapping of existing systems to proposed trust metrics

Two quantities are used to define the notion of persistence. First, *text-life* is the percentage of space-delimited words added in some revision, r_i, which persist after a subsequent edit, r_j. The second is *edit-distance*, which measures the extent to which reorganization and deletions are preserved. The authors' develop a specialized `diff` algorithm to quantify the latter quantity.

Assume author A has made edit r_n on some 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-life of r_n relative to r_{n+1}, (3) the edit-distance 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 r_{n+10} is reached. The reputation of A speaks directly to the trustworthiness of A's content, which is especially useful in judging new contributions of A which are yet to be vetted by subsequent editors.

Figure 4 helps exemplify the content-persistence algorithm. Assume authors A_1, A_2, and A_3 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, and the statement "Mr. Franklin flew a kite" gains reputation, as well as the authors who endorsed this view (A_1 and A_3) – and this process would continue to some specified depth (Adler uses $depth = 10$).

The measurement of content-persistence cleverly circumvents much of the problem with a lack of a precise trust definition. It assumes that an edit will only persist if it is deemed trustworthy by the community. This allows the technique to implicitly fulfill seven of the eight metrics of the proposed trust vector (see Table 2), failing only to speak to 'comprehensiveness'.

Related Works: Adler's system is both a formalization and refinement upon the informal proposal made in [26] by Cross, which suggests that text-age may be indicative of fragment trust. Whereas Cross would treat restored text as

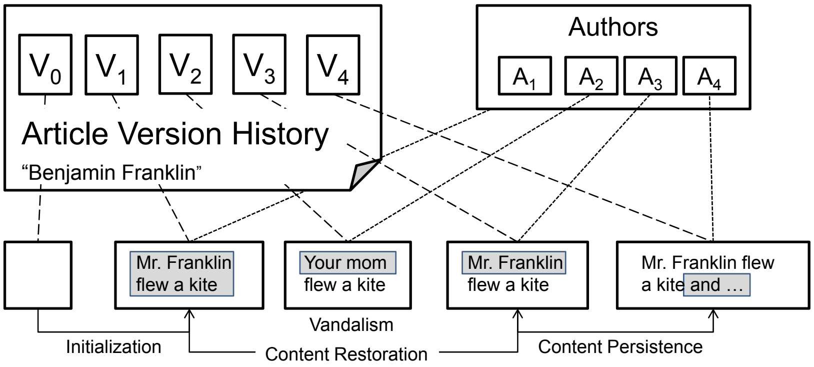


Figure 4: Example content-persistence calculation

new and potentially untrustworthy, Adler investigates the transience of content through greater revision depth.

The system most related to Adler's is that of Zeng *et al.* [68] who used Dynamic Bayesian networks and the Beta probability distribution to model article quality. Zeng's system takes both author reputation and `diff` magnitude as inputs when calculating article trust. Whereas Adler computes predictive author reputation, Zeng uses pre-defined *roles* (*e.g.,* administrator, registered, anonymous, *etc.*) as predictors of author behavior.

Wöhner *et al.* [66] take a similar approach by measuring content persistence and transience rates throughout an article's lifespan. They find that quality articles are defined by a stage of high editing 'intensity', whereas low quality articles tend to have little of their initial content modified as they mature.

The notion of author reputation was also investigated by West *et al.* [65]. Rather than doing fine-grained content analysis of Adler, West detects an administrative form of revert called *rollback* to negatively impact the reputations of offending editors. Reputations improve only via the passage of time and this lack of normalization is problematic because rarely-erroneous prolific editors may appear identical to dedicated but small-scale vandals.

Live Implementation: The proposal of Adler has been implemented as a live Wikipedia tool, WikiTrust [16]. WikiTrust colors text fragments to display the associated trust values (see Sec. 5.2.1).

3.2.2. NLP-based Trust

Approach: Distinct from content-persistence (Sec. 3.2.1) which treats words as meaningless units of content, natural-language processing (NLP) techniques analyze the language properties of tokens. The techniques are varied; from simple text properties (*e.g.,* the prevalence of capital letters), obscenity detection (via regular expressions), to text similarity and predictability (*n*-gram analysis). We

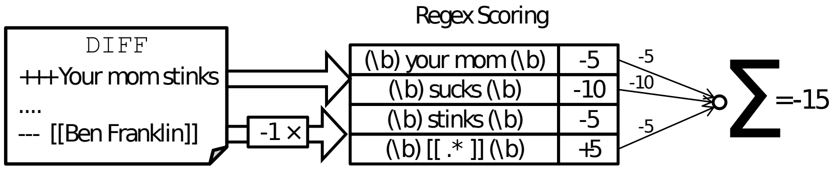


Figure 5: NLP - Example of lexical analysis

choose the recent work of Wang *et al.* [64] to be representative of this domain due its breadth of techniques.

Wang (and practically all NLP-based works) produce a feature-vector over which traditional machine-learning techniques are applied. In particular, Wang *et al.* divide their feature-set into three different NLP-driven categories: (1) lexical, (2) semantic, and (3) syntactic.

Lexical features are straightforward and are generally implemented via regular expressions. For all content added in a revision Wang implements a measure of, (i) vulgarity, (ii) slang (*e.g.,* 'LOL' or 'p0wned' – phrases which are not obscene, but improper in formal English), and (iii) improper punctuation (*e.g.,* the repetitive usage of question or exclamation marks). Figure 5 shows an example of lexical analysis being performed over an edit `diff`.

The *syntactic* and *semantic* categories are more complex. For syntactic analysis, Wang performs n-gram analysis using only part-of-speech (POS) tags. That is, using some corpus (general or topic-specific) one computes the probability of all POS sequences of length n. Then, when an edit is made, the probabilities of new POS sequences are calculated. Improbable POS sequences are likely indicative of a damaging edit. Wang's semantic analysis also uses n-gram analysis but uses unique words instead of POS tags.

Figure 6 shows an example analysis using semantic unigrams (*i.e.,* $n = 1$). Related sources are amassed to build a dictionary of words common in discussion of the article under investigation, "Benjamin Franklin." When words added to the article elicit a high "surprise factor" (*i.e.,* have not been seen in the corpus), there is good probability of suspicious activity. Indeed, Ben Franklin never flew a jet, and the revision is vandalism.

NLP-based approaches satisfy few of the proposed trust metrics, as shown in Table 2. The lexical models are only effective in detecting inappropriate use of language, a 'cohesiveness' issue. Syntactic and semantic models can be useful to determine the 'accuracy' of content.

Related Works: The work of Wang is recent to this writing and incorporates many ideas from earlier literature. Many such works investigated the predictive nature of n-gram analysis. One of the first was Smets *et al.* [61], utilizing Bayesian analysis (initially shown useful in email spam detection [58]) and Probabilistic Sequence Modeling. Similarly, [23] used a generic predictive analysis, while Itakure *et al.* [37] leveraged dynamic Markov compression. While different in technique, these techniques calculate roughly equivalent probabili-

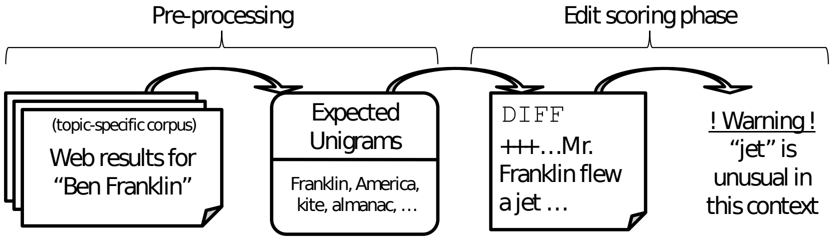

Pre-processing
Edit scoring phase
(topic-specific corpus)
Web results for "Ben Franklin"
Expected Unigrams
Franklin, America, kite, almanac, …
DIFF
+++…Mr. Franklin flew a jet …
! Warning !
"jet" is unusual in this context

Figure 6: NLP - Example of semantic analysis

ties. However, the work of Wang is unique in that probabilities are generated from web-based corpora (*i.e.,* the top k search-engine results), whereas earlier literature used only the (narrower) Wikipedia article or a generalized corpus.

Distinct from predictive techniques are those of Potthast *et al.* [54] which tend to focus on aggregate-count measures. For example, Potthast includes simplistic features such as (i) ratio of upper-case characters, (ii), longest word length, and (iii) pronoun frequency. Along the same lines, Rassbach *et al.* [57] use an un-described set of "about 50 features" from an NLP toolkit.

Also in the NLP realm would be the 'readability' measures (*e.g.,* Flesch-Kincaid, SMOG) incorporated into some trust systems [57, 62]. Though collaborative literature provides little insight regarding their function or usefulness, these systems produce a measure of text complexity by examining sentence lengths and syllable counts.

Live Implementation: NLP techniques are being applied in real-time on Wikipedia by an autonomous script called ClueBot [1], which automatically reverts trivially offensive edits. Due to a low tolerance for false-positives, ClueBot operates using a conservative and manually-authored set of regular expressions. ClueBot has been well studied [33, 61, 64] and exemplifies that lexical measures need not be strictly punitive. For example, regexps capturing advanced *wiki*-markup can increase edit trust.

3.2.3. Metadata-based Trust

Approach: If we consider article versions to be the *data* in a *wiki* system, *metadata* is then any property which describes that data. We divide metadata into two sets: content-exclusive and content-inclusive.

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

[2] An optional text field where an editor can briefly summarize changes.

Content-Exclusive Features	
Editor	**Time-stamp**
· Anonymous/registered	· Local time-of-day
· Time since first edit	· Local day-of-week
· User edit count	· Time since article edited
Article	**Revision Summary**
· Num. edits in history	· Comment length
· Article age	· If edit marked 'minor'
Content-Inclusive Features	
· Article length	· Revision `diff` size
· Num. external links	· Num. images

Table 3: Example metadata features [22, 54, 62, 65]

Meanwhile, *content-inclusive* measures permit summarization of the article or `diff` text. For example, this could be a measure of article length or the number of images in an article. Indeed, some degree of text-parsing would be required to extract these properties. Thus, we believe such properties may verge on being lexical NLP ones (like those of Potthast [54]). In general, we prefer language-driven features of this kind to be classified in the NLP domain and structurally-driven ones considered metadata[3].

Regardless, systems of this kind proceed by identifying multiple metadata-based indicators and producing predictive measures via machine-learning. Table 3 lists several example features of each type. Incorporating many of these features is the work of Stvilia *et al.* [62], which we choose to be representative of metadata-based approaches.

Rather than simply identifying metadata indicators, Stvilia takes an information quality (IQ) approach. IQ metrics [63] are properties like completeness, informativeness, consistency, and currency which generally define document quality (even outside of collaborative environments [69]). Stvilia's contribution is the quantification of these metrics for Wikipedia via the use of metadata features. For example, a measure of *completeness* considers the article length and the number of internal links. *Consistency* considers an article's age and the percentage of edits made by administrators. This IQ-based approach seems a more intuitive and elegant use of metadata than simply pushing raw-features to a machine-learning framework.

[3] By definition, properties we have separated out as entirely different techniques (*e.g.,* content persistence) could also be considered content-inclusive metadata. For consistency, reader's that believe these categories to be in conflict should consider only content-exclusive properties to be part of a metadata-based approach.

We believe that the metadata-based approach is general enough to capture *all* the trust metrics proposed in Sec. 2.4.2. While we believe metadata-driven formulations exist for each metric, literature has only defined a subset of them.

Related Works: The work most similar to Stvilia's is that of Dondio *et al.* [28]. Dondio begins by formally modeling the Wikipedia infrastructure and identifying ten "propositions about trustworthiness of articles" which are essentially IQ metrics. However, only two metrics are developed (fulfilling three of the propositions), *leadership* and *stability*. These "domain-specific expertise" metrics are shown to marginally improve on cluster analysis over 13 raw metadata features (*e.g.,* article length, number of images).

Meanwhile, inspired by the use of metadata to combat email spam [36], West *et al.* [65] concentrate on a set of content-exclusive metadata features based on spatio-temporal properties. Simple properties include the time when an edit was made, the length of the revision comment, *etc.*. More novel are reputations generated from metadata-driven detection of revert actions. Article and author reputations are straightforward, but *spatial reputations* for topical-categories and geographical regions are novel in their ability to have predictive measures available for new entities.

Almost comical compared to the complexity of these approaches, Blumenstock [22] claims that a single metric – word count – is the best indicator of article quality and significantly outperforms other discussed strategies.

Live Implementation: Metadata properties are being used to evaluate Wikipedia edits in a live fashion. The STiki anti-vandalism tool [8] is built on the logic of West's approach. It calculates trust scores which are used to prioritize human-search for damaging edits (see Sec. 5.2.2).

3.2.4. Citation-based Trust

Approach: Borrowing from citation-based algorithms commonly used in search-engine retrieval ranking such as HITS [42] and PageRank [50], McGuinness *et al.* [47] propose a *link-ratio* algorithm.

First, consider an article, a_n on Wikipedia (*e.g.,* "Benjamin Franklin"). The title of a_n can then be treated as an *index term* and full-text search can be conducted on all other *wiki* articles (*i.e.,* $\forall a_i, i \neq n$), counting the number of occurrences of that term (*e.g.,* articles like "Philadelphia" or "bifocals" are likely to have occurrences of "Benjamin Franklin").

Each of these occurrences are then labeled. Occurrences formatted to be internal wiki-links (*i.e.,* the index term is a hyperlink to the matching article) are termed *linked*, whereas occurrences where this is not the case (*i.e.,* the term appears as plain-text) are *non-linked.* The ratio of linked occurrences to all occurrences is the *link-ratio*, the metric of interest. McGuinness argues that high link-ratios are indicative of trusted articles, as the decision to cite another article is an implicit recommendation of that article's content.

An example of McGuinness' algorithm is visualized in Figure 7 (using our "Benjamin Franklin" example) – note that the `[[...]]` syntax is common

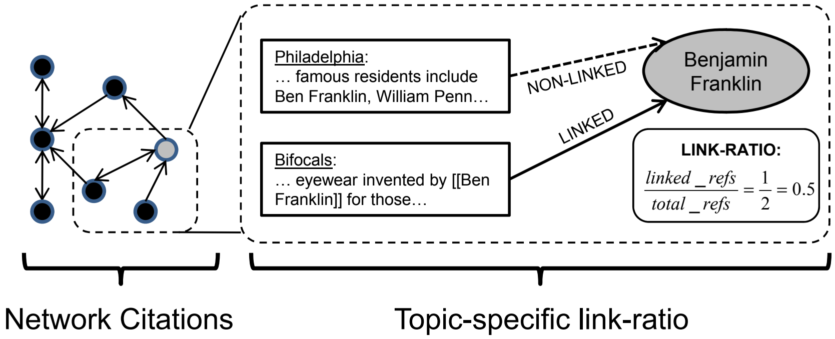

Philadelphia:
… famous residents include
Ben Franklin, William Penn…
Bifocals:
… eyewear invented by [[Ben
Franklin]] for those…
NON-LINKED
LINKED
Benjamin
Franklin
LINK-RATIO:
$\frac{linked\_refs}{total\_refs} = \frac{1}{2} = 0.5$
Network Citations
Topic-specific link-ratio

Figure 7: Example link-ratio calculation

wiki markup for internal links. To give some idea of the scale at which such algorithms operate, the actual "Ben Franklin" article has over 4000 incoming citations as of this writing.

The design to cite content is an implicit approval regarding everything about that content. Thus, we believe that citation-based approaches are capable of fulfilling *all* of the proposed trust metrics (see Table 2). Despite its wide-coverage of metrics, flaws of the approach make it less desirable than readers might expect (as we discuss later in Sec. 4).

Related Works: In the course of their evaluation, McGuinness *et al.* compared their link-ratio algorithm to results using the PageRank algorithm [50]. Earlier, Bellomi *et al.* [20] performed internal network analysis using both the PageRank and HITS [42] algorithms. The major difference between the link-ratio and search-inspired strategies is the extent of normalization.

For example, if an index term appears just once in the full-text of the *wiki*, and that once instance is linked, than the term will have a perfect link-ratio. Thus, to increase an article's trust value, one need only convert existing plain text references to linked ones. In contrast, PageRank and HITS perform more complex (non-normalized) graph analysis.

Live Implementation: To the best of our knowledge, there is no live implementation calculating citation-based trust for Wikipedia. However, Google's browser toolbar [5] exposes the PageRank values calculated by the search-engine provider, which could be interpreted comparatively.

Wikipedia does identify *orphaned* articles – those with few or no incoming links. While Wikipedia provides such lists [7] to encourage the strengthening of network connectivity, citation-based strategies contend these would be articles of least trustworthiness.

Comparison Criteria	PERSIST	NLP	META	CITE	Sec.
Persistent IDs critical	Yes	No	Feature dependent	No	§ 4.1
Human involvement	Implicit Feedback	Corpus Building	Corpus Building	Implicit Feedback	§ 4.2
Integration of ext. data	No	n-grams	Yes	Yes	§ 4.3
Efficiency	Sufficient	Variable	Good	Sufficient	§ 4.4
Portability	See Table 5				§ 4.5

Table 4: Comparative summary for techniques

4. Comparing Trust Algorithms

In the previous section, we introduced different techniques for trust calculation. Now, we examine these methods comparatively. Our dimensions for comparison are not intended to be exhaustive. Instead, we choose attributes which highlight the strengths/weaknesses of each approach and reflect the design decisions commonly faced by collaborative trust systems. Table 4 summarizes the comparative merits of these algorithms.

4.1. User Identifier Persistence

For systems that include some notion of author trust (which per Sec. 2.5 should be *all* systems), it is desirable that identifiers be persistent so one's contributions may be tracked throughout time. However, due to (1) anonymous editing, and (2) ill-intentioned users – this is not always the case.

Wikipedia allows users to edit *anonymously*, whereby their IP addresses become used as identifiers. In such cases, it is unreliable to assume there is a 1:1 mapping between an IP address and an editor. A single public computer may have many users, and a single user may use computers in multiple locations. Further, a single computer may have a dynamic IP such that its addressing is not constant over time. Thus, it seems unreasonable to praise or punish IP identifiers for fear of collateral damage.

Even so, there exists a tension between anonymous and *registered* users (those with a persistent username/password). Nearly 80% of vandalism is committed by anonymous users [65], who contribute only 31% of all article edits [15]. Goldman [35] notes that anonymous users are sometimes treated as "second class citizens" and that their edits undergo additional scrutiny.

An obvious suggestion is to make all community members register, which is problematic for two reasons. First, Wikipedia (and its parent, the Wikimedia Foundation) is adamant in supporting anonymous editing, as it provides both convenience and privacy. Second, malicious users can still manipulate registered accounts to their benefit.

For example, one of the most common abuses leveraged at trust systems is the Sybil attack [29]. New users must be given an initial trust value, and if the trust value of an account ever falls below that threshold, then it may be easier for an attacker to create a new account rather than repairing the trust value of the existing one. Wikipedia's barrier-to-entry – a CAPTCHA solve – seems ineffective in this regard since it has been shown that such protections can be evaded cheaply and at scale [49]. As a result, trust systems must set initial values extremely low. Thus, new or casual users may be perceived just as anonymous users – "second-class" participants.

Choosing to be a registered editor does have benefits. Notably, the IP addresses associated with registered accounts are treated as private information[4], which may hamper some analysis. For example, the WikiScanner tool [14] detects conflicts-of-interest based on IP geo-location (*e.g.,* Edits from an IP from Redmond, Washington to the "Microsoft" article might warrant extra scrutiny). Similarly, [65] computes geographical reputations based on geo-location that prove effective in predicting the behavior of new users. Such analysis is not possible when IP addresses are not available.

So what do these issues mean for trust systems? Certainly, systems that arrive at user-reputations associatively (citation-based, NLP-based) are less affected than those that compute user reputations directly (content-driven, metadata-based). For the latter class, it is important that mechanisms are in place to evaluate users in the absence of history (for example, the spatial reputations of [65]). Secondly, if trust values are used to incentivize good behavior (see Sec. 5.2.4), then users will be rewarded for creating and maintaining persistent identifiers, lessening the severity of the issue.

4.2. Degree of Autonomy

We next examine the degree of *autonomy* at which each of the proposed techniques operates. That is, what role do humans play in the computation of trust values? We divide the space into three divisions: (1) *Corpus-driven*, (2) *Explicit-feedback*, and (3) *Implicit-feedback*.

Corpus-driven: First, we consider models which require no human intervention to evaluate a revision at the time it is committed. This includes NLP-based and metadata-driven strategies – precisely those which employ machine-learning and are corpus-driven. Whether knowingly or implicitly, humans have aided in labeling the corpora used to construct scoring models. Since models are pre-computed, they can be readily used to quantify revision quality. However, there are start-up costs associated with such approaches since corpora must be amassed for this purpose.

[4] Wikipedia does retain such data and makes it visible to a small set of extremely trusted users (`checkusers`). IP addresses are only investigated when it is suspected that abuse is being conducted via multiple accounts under the control of one individual.

Explicit-feedback: Second, are systems which require human involvement external of normal *wiki* actions in order to produce trust values. In our survey, we consider no systems of this type because they are uncommon, intrusive, prohibit automatic trust calculation, and have marginal cost. Nonetheless, such systems do exist in literature [44] and are in active use [13].

Such systems often manifest themselves as dialog boxes which allow a user to rate the quality of an article from an enumerated set of options. In other words, such systems collect *feedback*, subjective observations which form the basis for trust value computation [38, 40].

Implicit-feedback: Most interesting are the content-driven and citation-based techniques which non-intrusively produce feedback by monitoring typical *wiki* behavior. For example, Adler's [17, 18] content-driven approach considers the removal of content to be an implicit *negative* feedback against that content and its authors. Citation-algorithms consider the citation of an article to be an implicit *positive* feedback about article quality.

Thus, these approaches can use well known feedback-aggregation strategies to produce behavior-predictive values. Beyond this, many systems have leveraged properties of collaborative environments to overcome complications typical of trust management. For example, Adler's approach succeeds in holding feedback *providers* accountable – a challenge in traditional systems. Consider that an editor B who removes all the additions of A in an attempt to discredit him will be jeopardizing his own reputation, since if A's contribution is restored, it will be B who is punished. Similarly, B cannot simply praise the edits of A. Instead, B must actually edit the article, and then both the edits of A and B will be judged by subsequent editors. Further, since edit-magnitude is a factor, ballot-stuffing attacks are averted. However, many reputation systems are vulnerable to the "cold-start problem" (and thus, Sybil attacks, see Sec. 4.1) since multiple feedbacks may be required before meaningful values can be computed for an entity. West [65] overcomes this issue by broadening the entity under evaluation, leveraging the sociological property of homophily[5]. [48].

Implicit-feedback approaches have additional drawbacks as well, the most significant of which is *latency.* With content-persistence, multiple subsequent revisions are the best measure of a previous revision's quality. Thus, it may take considerable time for rarely edited articles to get their content vetted. Such latency could be significant in small communities where there are few feedback providers (see the 'intra-magnitude' portion of Sec. 4.5).

Latency is far worse for citation-based approaches. The decision to cite an article can speak to quality *only* when the citation was made. It is unreasonable to assume that the citation network evolves as dynamically as the underlying content (*i.e.,* such metrics are poor for vandalism detection).

Latency aside, the primary criticism of citation approaches is whether or not

[5] Homophily is the tendency of individuals to share behavioral characteristics with similar others. Spatial adjacency (both geographical and abstract) is one means of defining similarity.

a citation actually constitutes a subjective feedback. That is, do *wiki* citations occur because individuals actually trust the page being cited, or is convention simply being followed? Wikipedia does specify linking conventions [12] which would skew the calculation of link-ratio and PageRank-like metrics. For example, the policy states one should "...link only the first occurrence of an item" on an article and that "...religions, languages, [and] common professions ..." should generally not be cited. Even the link-ratio authors recognize that proper nouns tend to be linked more frequently than well understood concepts (*e.g.,* love) [47]. These factors seriously challenge the extent to which citation-based metrics are measuring trust.

4.3. Integration of External Data

A *wiki* environment, in and of itself, provides a wealth of information which enables the calculation of trust values. However, several techniques distinguish themselves in that they are able to use data *external* to the wiki for on-wiki evaluation. The advantages of using external data are numerous. First, such data is outside the immediately modifiable realm, making it difficult for malicious users to manipulate. Additionally, smaller *wiki* installations may have sparse data, which external information could bolster.

Citation-based strategies can utilize external data by expanding the scope of their network graph. Rather than considering the internal hyperlink structure of the *wiki*, HITS/PageRank could measure incoming citations from outside the *wiki*. In other words, the algorithms would be used precisely as they are for search engine ranking – by crawling the entire Internet and processing citations. Then, the scores for articles could be interpreted comparatively. Indeed, an external citation of a *wiki* article seems to be a stronger endorsement of article trust than an internal one (per Sec. 4.2).

Only the most recent NLP-based works have utilized external data, in particular that of Wang [64] in their syntactic and semantic n-gram analysis. Whereas previous works pre-computed n-gram probabilities using a general corpus or the article itself as a topic-specific corpus – Wang uses the top-50 search engine results for an article title as the corpus for that article's probabilities. Scalability issues aside, external data succeeds in increasing the breadth of such corpora. Further, one could imagine that web-corpora make n-grams more adaptable than other types. For instance, breaking news events may cause a revision to deviate from an article's typical scope. While typical corpora would cause such an addition to be viewed as unusual or deviant – Internet sources would likely have updated information and a constructive revision would be marked as such.

Finally, metadata approaches provide the richest opportunities for the use of external data. The number of `JOINS` between metadata fields and external data seems limitless, although few have been investigated in literature. As an example, consider the IP address of an editor (a metadata field). In turn, that IP address could be used to: geo-locate the editor (and determine their local time-of-day or day-of-week), determine the editor's ISP, investigate the blacklist status of the IP address, or scan for open ports to determine if the IP is a proxy.

The sheer size of feature-space available to researchers is undoubtedly one of the strongest assets of the metadata approach. However, critics may argue that metadata-feature are "a level removed" from what is really of interest – the content. Rather than encouraging contributors to author quality content, metadata-based features introduces other variables into the evaluation process. Furthermore, there is the possibility of collateral damage and introducing disincentives to participation. Imagine a rule like "if an editor is from region x the trust in their edits should be reduced by y." Though it may be based on evidence, such a rule may discourage innocent editors from the same region.

4.4. Computational Efficiency

Although theoretical advancements are useful, for a trust calculation system to actually be useful it needs operate efficiently at the *wiki* scale. Certainly, English Wikipedia suggests this may be computationally non-trivial. As of this writing, Wikipedia averages 1.5 edits/sec. in English, and 4 edits/sec. across all language editions [15] – and it is reasonable to assume peak loads may exceed these rates by an order of magnitude or more.

In the literature, we are aware of two works which cite concrete throughput figures. The NLP approach of Potthast [54] states it can handle 5 edits/sec., while the metadata technique of West [65] claims 100+ edits/sec[6]. While WikiTrust [16] (content-persistence) cites no explicit throughput numbers, its live implementation suggests it is capable of sufficient scalability. Similarly, Cluebot [1] speaks to the scalability of lexical NLP techniques. Thus, significant scalability questions remain about (1) citation-based and (2) predictive NLP (*i.e.,* n-grams), and we examine each in turn.

It would seem that no matter the citation-based algorithm, a considerable amount of pre-processing is required to construct the initial network graph. However, once this is complete, the link-ratio algorithm of McGuinness could trivially update values incrementally (as each index term has a value independent of all others). Probability-based citation algorithms like PageRank/HITS are more complex, given that an evolving network structure could alter probabilities for a large number of nodes. Nonetheless, incremental update techniques have been developed for PageRank, and the Wikipedia network is orders of magnitude smaller than the Internet-scale space these algorithms were designed to process. Further, since citation-based trust is ineffective for short-term assessments (*e.g.,* vandalism), some delay in trust value calculation is acceptable.

Predictive NLP techniques also require a large amount of pre-processing to have n-gram probabilities ready to compare against new ones in an edit `diff`. The distinguishing factor is *when* this pre-processing can be performed. If one uses a large and general-purpose corpus, there is little issue in having probabilities readily available at edit-time. However, research has shown that

[6] Latency is not believed to be a significant issue. Although production systems make API calls [11] to Wikipedia, adding latency, such approaches could conceivably run on the Wikimedia servers if they were deemed sufficiently important.

Approach		Intra-Language	Intra-Purpose	Intra-Magnitude
Content-persist		✓	✓	
NLP	Lexical			✓
	n-gram	✓		✓
Metadata-based		✓		✓
Citation-based		✓	✓	

Table 5: Portability of trust approaches

domain-specific probabilities are advantageous. This means, at a minimum (supposing the previous article version is treated as a corpus), probabilities would need to be re-calculated for each article after every edit. In the worst case are dynamic web-based corpora like those proposed by [64], who used the top-50 web results for an article's title as the training corpus. Such a massive amount of text-processing (and bandwidth) seems problematic at scale.

4.5. Technique Portability

Though our analysis herein is focused on the English Wikipedia, it is important to realize there are many *wiki* installations across the Internet. For instance, Wikipedia has 273 language editions and nearly a dozen sister projects (and their language editions). Additionally, `wikia.com` – a centralized *wiki* hosting service – supports over 100,000 *wikis* [9]. These examples likely constitute only a trivial fraction of installations on the Internet. It is likely that most of these communities lack the tools, vigilance, and massive user-base that enables English Wikipedia to thrive.

Thus, automatic calculation of trust values seem especially useful in such installations. We consider three dimensions of portability for our trust techniques: (1) *intra-language* (*e.g.,* as English Wikipedia relates to French Wikipedia), (2) *intra-purpose* (*e.g.,* as Wikipedia relates to Encyclopædia Dramatica), and (3) *intra-magnitude* (*e.g.,* as Wikipedia relates to a small-scale installation). Table 5 indicates which algorithms can be transitioned between dimensions with no/trivial modification to their approach.

Intra-language: First, we address the portability of techniques across different natural languages. Intuitively, such a transition is most problematic for NLP-based measurement, but to a surprisingly small extent. Lexical techniques (*e.g.,* bad-word regexps) would need to be localized, but semantic and syntactic measures (*e.g., n*-gram probabilities) can be used so long as they are calibrated over corpora in the language of interest. Meanwhile, content-persistence techniques require only that the natural language be delimited in some way (presumably at word or sentence granularity). It is reasonable to assume most natural languages have this characteristic.

Intra-purpose: Second is the issue of intra-purpose portability. Are trust mechanisms tuned for Wikipedia's encyclopedic expectations, or do these expectations hold for content in general? Both NLP and metadata-based approaches seem challenged by such a transition. The biggest unknown for NLP is how predictive measure (*i.e., n*-grams) might operate when *novel* content is being generated (*e.g.,* imagine collaboratively authoring a fiction novel), rather than summarizing some existing body of knowledge (as with an encyclopedia). Similarly, metadata-based IQ metrics would also be sensitive to change, as they were manually crafted for encyclopedic use by [62] (though versions do exist for generalized web documents [69]).

Intra-magnitude: Finally, we consider the magnitude of the *wiki* under investigation and in particular how smaller wikis might affect trust computation. Content-persistence methods are dependent on the implicit feedback made by subsequent editors to an article. Such assessments may be considerably latent in a *wiki* with a low edit volume. Citation-driven approaches could also be hampered. Consider that a *wiki* with few editors is unlikely to generate much content, and in turn, the citation graph is likely to be sparse. Such graphs are not ideal for calculating link-ratios or internal PageRank/HITS scores.

4.6. Comparative Effectiveness

Perhaps the most obvious question regarding the varied techniques is, *"which works best?"* – and unsurprisingly, a definitive answer is not possible. Most satisfying is the recent vandalism corpus and detection competition of Potthast *et al.* [55]. The corpus is composed of 32,000 revisions, labeled by crowd-sourced annotators. For the detection competition (which withheld labels for half the corpus), 9 different schemes were submitted, encompassing 55 different features, all of which are discussed in the competition summary [55].

Three of our methodologies, (1) content-driven, (2) NLP-based, and (3) metadata-based were well represented in the competition (only citation-based is not, which does not apply well at revision-granularity). An NLP approach based on [54] won the competition, with WikiTrust [16] (content-persistence) finishing second. We believe these results should be interpreted cautiously, as each system is only a single, non-comprehensive, point of reference into a domain. Further, the competition only gauged how well systems apply in the domain of vandalism detection and not across the entire trust spectrum.

Most importantly, [55] reports that a meta-classifier built from all competition entries significantly outperforms the single winning classifier. Thus, differing strategies capture unique sets of vandalism, validating that trust is an issue best approached via multiple methodologies. Fine-grained analysis of one such meta-classifier was conducted in [19], which examined the contributions of various feature types (NLP, metadata, and reputation-driven).

5. Usage of Trust Values

Assuming we have calculated a trust value – we must examine how it can be *interpreted* and *utilized* to benefit the end-user and application security. First,

Approach	**Granular.**	**Tasks** (Sec. 5.2)
Content-persist	Fragment, Author	Fragment trust, Revision selection, User privileges
NLP	Revision	Anti-vandalism
Metadata-based	Article, Revision	Article trust, Anti-vandalism
Citation-based	Article	Article trust

Table 6: Describing the "preferred granularity" of each approach – and the tasks that computed values are most useful at optimizing.

Sec. 5.1 talks about the interpretation of trust values. Then, Sec. 5.2 describes some prominent and/or potential use-cases. Finally, in Sec. 5.3 we provide some cautionary remarks on how/why the use of trust values could be a *bad* idea.

5.1. Interpreting Trust Values

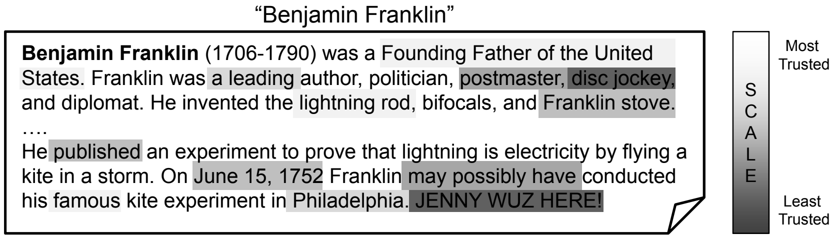
If we have computed a quantitative (*i.e.,* numerical) trust value, it cannot be effectively presented until we understand the semantics of that value (*i.e.,* its interpretation). Although it may the case that trust is defined along multiple-dimensions (as with our own proposal), we assume a reduction can be performed so that trust is defined along a single dimension.

Examining the output of the techniques surveyed, we find that they all meet this criterion. However, none of the systems are capable of computing values that can be read in an absolute capacity – that is, they must be *relatively interpreted*. As a result, no definitive statements can be made about what is 'good' and 'bad' and comparative analysis becomes necessary. Comparative values are not ideal. Unlike in search-engine retrieval, it seems unlikely that a *wiki* user would need to determine which of two documents is most trustworthy. It is more likely that they would wish to know the trust of an article in isolation.

Two strategies attempt to impart meaning onto values: (1) Treating values as a classification problem and applying thresholds based on empirical evidence, and (2) Normalizing values to make them presentation-friendly. The first approach, as discussed in Sec. 2.4.1, requires training corpora to be amassed. While simple to build for certain subsets of the trust spectrum (*i.e.,* vandalism), this is a difficult approach for more fine-grained analysis. Further, thresholds are often drawn based on a tolerance for false-positives, not the need for accuracy.

The second approach, normalization, is often used for presentation purposes. For example, trust values on the range $[0, 1]$ are more human-friendly than raw values. Of course, normalized values are arbitrary and perhaps even deceptive to human users. For example, articles on a poor quality *wiki* could have full normalized trust because they are the "best among the worst."

Alternatively, one can simply embrace the relative nature of trust values and ignore mapping attempts. Such is the approach of *intelligent routing systems*,


"Benjamin Franklin"

Benjamin Franklin (1706-1790) was a Founding Father of the United States. Franklin was a leading author, politician, postmaster, disc jockey, and diplomat. He invented the lightning rod, bifocals, and Franklin stove.
....
He published an experiment to prove that lightning is electricity by flying a kite in a storm. On June 15, 1752 Franklin may possibly have conducted his famous kite experiment in Philadelphia. JENNY WUZ HERE!

Figure 8: Example using text-coloring to display trust

such as [65], which we discuss further in Sec. 5.2.2. The feasability of calculating values with an absolute interpretation remains an open research question.

5.2. Use-cases for Trust

Having seen how trust values can be interpreted, we next examine the application of these values to tasks on Wikipedia. For each task, we first describe how the Wikipedia community currently performs the task (*i.e.,* the status quo). Then, we demonstrate how the application of trust values may optimize that task, making it more efficient, accurate, or intuitive. Table 6 summarizes the approaches which excel at each task (often due to a preference for calculating trust at a specific granularity). Our choice of tasks is not intended to be comprehensive, but reflect some of the most prominent proposals in the literature.

5.2.1. Visual Display of Trust

Status Quo: Perhaps the most straightforward use of trust values is to present them directly to the end-user, adjacent to the article or text fragments they describe. The Wikipedia software has no functionality for this purpose at present – though it has been a popular proposal among researchers.

Trust Application: Several authors [17, 26, 47] propose the colorization of fragment text as an intuitive and non-intrusive way to present trust values. A live browser plug-in utilizing the technique has been developed [16]. Figure 8 displays an example of the proposed output.

More simply, a suggestion has been to simply display the numerical trust value of the article, on the article itself[7] [44]. Of course, public exposure of trust values can lead to problems with interpretation (Sec. 5.1) or encourage evasion (Sec. 5.3) and may be a reason such proposals are yet to gain traction.

5.2.2. Damage Detection

Status Quo: Broadly, three strategies are currently used to detect and undo damaging edits (*i.e.,* vandalism). First, is the automatic reversion of poor edits

[7] While a valid proposal, we note that the cited system relies on explicit user-provided feedback and is not capable of automatic trust calculation. Thus, the display of these numerical values side-steps earlier issues involving relative interpretation of trust values.

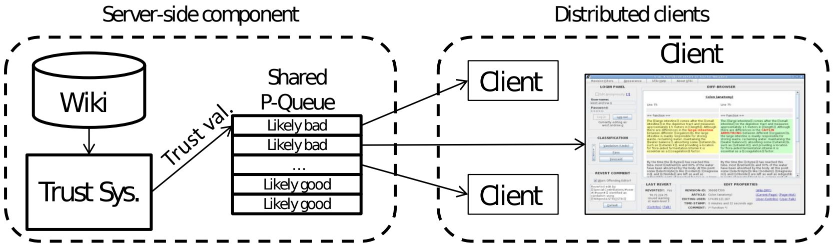


Figure 9: Example intelligent routing tool (STiki [8])

by autonomous *bots* – of which the NLP-based ClueBot [1] would be the characteristic example. Second, is the use of software assistants to present edits to human users and asks them to make determinations. Huggle [6] is the most popular example, which prioritizes edit display using a simple and manually-authored rule set (*e.g.,* show anonymous-user edits before those of registered ones). Finally, there are damaged edits discovered purely by human chance or brute-force. For example, editors often monitor changes to articles they are interested in via customized *watchlists*, or do brute-force patrol by watching the "recent changes" feed.

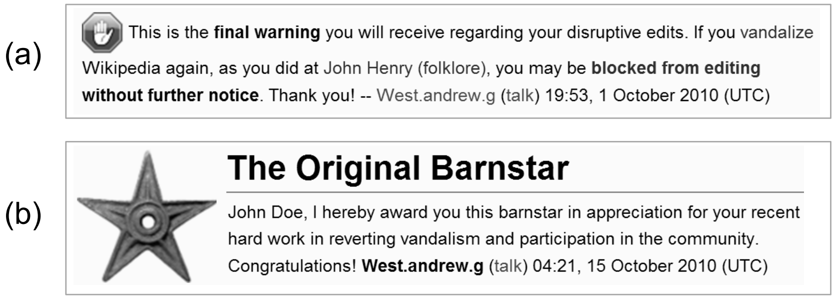
Trust Application: We first address the creation of smarter Wikipedia bots. Bots are attractive since they act quickly and at zero marginal cost. However, community standards are such that there is minimal tolerance for false-positives. Thus, in the current state-of-the-art such bots can only address the most "low hanging fruit." The comparison of detectors by Potthast [55] showed that only one system (a lexical NLP one) was capable of nearly false-positive free performance, and it was only capable finding 20% of damage at such high accuracy.

Given this, we believe software-assisted human detection should be a point of focus. Relative trust values can be well leveraged to build *intelligent routing tools* [25], which direct humans to where their efforts are most needed (*i.e.,* probable damage). At present, this technique is best leveraged by the STiki tool [8], which has a shared priority queue, and is visualized in Figure 9.

5.2.3. Revision Selection

Status Quo: While vandalism detection focuses on determining if the last edit to an article was damaging, *revision selection* tackles the more general problem of determining which version in an article's history is 'best.' The selected version can then be the default displayed under certain criteria or used to build trusted snapshots for other purposes.

On Wikipedia, such functionality is leveraged by a software extension called `FlaggedRevs` [4]. One use-case of the extension – "Pending Changes" – is currently active on several foreign language editions and under trial on the English Wikipedia [24]. The system prevents the revisions of anonymous editors from being publically displayed (on certain protected pages) until they have been



This is the **final warning** you will receive regarding your disruptive edits. If you vandalize Wikipedia again, as you did at John Henry (folklore), you may be **blocked from editing without further notice**. Thank you! -- West.andrew.g (talk) 19:53, 1 October 2010 (UTC)

The Original Barnstar

John Doe, I hereby award you this barnstar in appreciation for your recent hard work in reverting vandalism and participation in the community. Congratulations! **West.andrew.g** (talk) 04:21, 15 October 2010 (UTC)

Figure 10: User Trust: (a) warnings, and (b) barnstars

approved by a trusted community member (a `reviewer`).

Trust Application: As far as Pending Changes is concerned, trust values could be used to reduce reviewer workload by not requiring approval for highly trusted revisions. However, more interesting than its anti-vandalism use is how `FlaggedRevs` might be purposed to 'flag' revisions that occured long in the past.

For example, projects are creating static snapshots of Wikipedia for use by schools and for DVD/print distribution. Clearly, it is desirable that such snapshots contain the 'best' versions of an article possible – and that part of that definition should include 'damage-free.' Content-persistence trust is well suited for this task since it can evaluate revisions using the benefit of hindsight. However, 'currency' is also like to be a factor in what defines the 'best' revision. The most recent edits – those which likely include the most current information – are precisely those which we know the least about under content-persistence. Metadata or NLP techniques could prove helpful in this regard, but how to best weigh these factors remains an open research question. Regardless, any automation of the process is likely to be an improvement over the manual inspection currently employed as a safe-guard.

5.2.4. User Privileges

Status Quo: Editing *privileges* on Wikipedia include not just the advanced permissions delegated to trusted participants, but also the privilege to simply edit the encyclopedia which is sometimes revoked from troublesome users.

Wikipedia has a semi-formal mechanism by which users can lose trust and privileges. Editors committing damaging edits will be communicated increasingly stern warnings (see Figure 10a), which if ignored, will eventually lead to blocks/bans [33]. Individual accounts, single IP addresses, and IP ranges can be blocked/banned as needed to stop abuse.

In contrast, there is little formality in the way trust is amassed. While prolific editors may advance to `administrator` status and have extensive personal interaction histories, the vast majority of editors likely reside in a vast gray area where informal measures dominate. For example, *edit count* is sometimes viewed as a measure of trust, though [32] observes this to be a poor measure.

Further, *barnstars* – personalized digital tokens of appreciation (see Figure 10b) – are sometimes awarded between users [43].

Trust Application: As Adler *et al.* [18] note, the integration of user-level reputations into a *wiki* setting is important because it can *incentivize* constructive behavior. Unfortunately, Wikipedia has seemed to take the opposite approach by simply punishing miscreants.

Wikipedia has long championed the open-editing model, with minimal hierarchy among contributors and few restrictions. However, Goldman [35] notes that Wikipedia's labor shortage may force new built-in protections (*e.g.,* locking articles, pending changes, *etc.*) to mitigate poor behavior. With these protections comes the need for *new permissions* to manage them (or be exempt from them) will be inevitable. User trust could provide a means to automate the delegation and revocation of such rights, while providing a degree of robustness[8].

5.3. Cautions for Value Usage

Though the application of trust values in *wiki* settings is primarily viewed a a benefit, we briefly discuss the potential drawbacks of integrating trust values into collaborative software. These drawbacks are not intended to discourage the use of collaborative trust, but rather to highlight some design decisions about which developers should be cautious.

First, automatic tools and prioritization mechanisms may lead to a false sense of security and over-confidence. For example, if the STiki [8] anti-vandalism tool poorly classifies an edit, it will receive low priority, and may never be reviewed by a human. Tools like STiki and Huggle [6] have reduced the numbers of editors doing brute-force vandalism patrol, though the affect this has on anti-vandalism efforts is unknown.

Second, the exposure of trust values may provide malicious users insight into how trust values are calculated, permitting evasion. The most prominent example of this is Wikipedia's Edit Filter [2], which uses a manually generated rule set and can prevent edits from being committed. If an edit is disallowed, the reader will be informed of such – encouraging them to re-shape their edit into something slightly more constructive (or evasive). Thus, profanity may be obfuscated to evade the filter. Not only will this evade the Edit Filter, but it may also evade downstream mechanisms (*e.g.,* bots) which could have caught the original edit. Fortunately, those who damage articles seem poorly motivated. Priedhorsky *et al.* [56] observes that 71% of damaging edits exhibit 'nonsense' or 'offensive' attributes. However, [35] indicates that Wikipedia's growing popularity will invite motivated malicious users, such as spammers, who have financial incentive to evade protections.

[8] Wikipedia has a psuedo-permission called `autoconfirmed`, to which registered users *automatically* advance after 10 edits and 4 days (post-creation). `Autconfirmed` users need not solve CAPTCHAs and have other minor benefits. Clearly, given the ease of manipulating a metric like "edit count", this could be a vector for abuse.

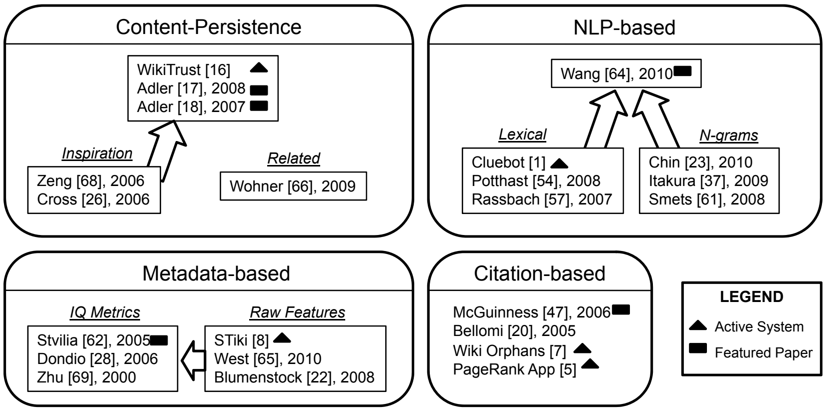


Figure 11: Relationships among publications (Wikipedia-centric)

Finally, the exposure of trust in individual users presents a unique set of challenges. Adler [18] advocates the display of user trust values, arguing that public values will incentivize users to behave well. Nonetheless, there are counterarguments. Wikipedia encourages an open-editing model where everyone is free to edit the work of others. User trust values could create a fine-grained hierarchy of editors which would create a barrier-to-entry and less democratic collaboration. Public display of trust values may also lead editors to overemphasize the importance of their own values. This may lead to editors doing solely what is best for *themselves* as opposed to the *encyclopedia*. For example, under content-persistence, editors may avoid editing breaking news topics, as their contributions are likely to be undone as the story evolves (regardless of their accuracy at the time of editing).

6. Conclusions

Herein, we have surveyed four different classes of calculating trust for collaborative content and discussed how these trust values can benefit the cooperative process. As Figure 11 shows, these works are supported by a large body of prior literature and related research. Each proposal has its relative merits and has been shown successful via evaluation, yet there is evidence that the state-of-the-art still has many challenging, open research questions.

Though it is evident that these systems are computing meaningful values (per their performance), it is not always clear to what extent these values speak to the actual *trust* one should place in an entity. Of course, this is complicated by the many definitions of trust in literature and the fact that few of them make for easy quantification. To side-step this issue, most authors focus on the most trivially untrustworthy of edits (*i.e.,* vandalism) to gain traction on the problem. It remains to be seen if these vandalism-centric values are capable of meaningfully quantifying contributions across the entirety of the trust spectrum.

One of the most encouraging aspects of the differing approaches is that they capture *unique* poor behaviors. As a recent vandalism-detection competition showed, meta-detectors significantly outperformed individual systems. Thus, understanding how these approaches can interact to produce higher-order classifications will be an important advancement.

Moving forward will also involve study of *wiki* environments other than Wikipedia. While Wikipedia is a large entity with available data, its community dynamics may be far different than those elsewhere online. Understanding how trust systems can work in generic collaborative environments is important to their application elsewhere. Further, most *wikis* rely on text-based content. Adapting the techniques to collaborative systems based on non-textual content (*e.g.,* images and data) is an interesting question to explore.

Regardless, the potential for trust systems in collaborative systems is large. For established systems like Wikipedia, they may ease maintenance concerns and allow editors to focus on content development. For emerging systems, trust can allow the community to measure its progress and highlight content which may best serve readers. On the whole, protecting readers from mis-information is crucial as society becomes increasingly reliant on collaborative knowledge.

Acknowledgements: This research was supported in part by ONR MURI N00014-07-1-0907 and NSF CNS-0931239. A preliminary version was published as University of Pennsylvania Technical Report MS-CIS-10-33.

References

[1] ClueBot. `http://en.wikipedia.org/wiki/User:ClueBot.`

[2] Edit filter. `http://en.wikipedia.org/wiki/WP:Edit_filter.`

[3] Encyclopædia Dramatica. `http://www.encyclopediadramatica.com.`

[4] FlaggedRevs. `http://www.mediawiki.org/wiki/Extension:FlaggedRevs.`

[5] Google toolbar. `http://www.google.com/toolbar/.`

[6] Huggle. `http://en.wikipedia.org/wiki/Wikipedia:Huggle.`

[7] Orphaned articles. `http://en.wikipedia.org/wiki/CAT:ORPH.`

[8] STiki. `http://en.wikipedia.org/wiki/Wikipedia:STiki.`

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

[10] Wikipedia. `http://www.wikipedia.org.`

[11] Wikipedia API. `http://en.wikipedia.org/w/api.php.`

[12] Wikipedia manual of style. `http://en.wikipedia.org/wiki/WP:LINK.`

[13] Wikipedia's public policy initiative. `http://outreach.wikimedia.org/wiki/Public_Policy_Initiative.`

[14] WikiScanner. `http://wikiscanner.virgil.gr/.`

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

[16] WikiTrust. `http://wikitrust.soe.ucsc.edu/.`

[17] B. Adler, J. Benerou, K. Chatterjee, L. de Alfaro, I. Pye, and V. Raman. Assigning trust to Wikipedia content. In *Proc. of WikiSym 2008*, 2008.

[18] B. Adler and L. de Alfaro. A content-driven reputation system for the Wikipedia. In *WWW '07: Proc. of the World Wide Web Conference*, 2007.

[19] B. T. Adler, L. de Alfaro, S. M. Mola-Velasco, I. Pye, P. Rosso, and A. G. West. Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *CICLing '11: Proc. of the 12th Intl. Conf. on Intelligent Text Processing and Computational Linguistics*, 2011.

[20] F. Bellomi and R. Bonato. Network analysis for wikipedia. In *WikiMania '05: The First International Wikimedia Conference*, 2005.

[21] M. Blaze, J. Feigenbaum, and J. Lacy. Decentralized trust management. In *IEEE Symposium on Security and Privacy*, pages 164–173, 1996.

[22] 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).

[23] 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.

[24] N. Cohen. Wikipedia to limit changes to articles on people. *New York Times*, page B1, August 25, 2009.

[25] 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.

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

[27] L. Diaz, C. Granell, M. Gould, and J. Huerta. Managing user-generated information in geospatial cyberinfrastructures. *Future Generation Computer Systems*, 27(3):304–314, 2011.

[28] 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, 2006.

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

[30] N. Dragoni. A survey on trust-based web service provision approaches. In *DEPEND '10: Proc. of the 3rd Intl. Conference on Dependability*, 2010.

[31] C. English, S. Terzis, W. Wagealla, H. Lowe, P. Nixon, and A. McGettrick. Trust dynamics for collaborative global computing. In *WETICE '03: Proceedings of the Workshop on Enabling Technologies: Infrastructure for Collaborative Enterprises*, 2003.

[32] 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.

[33] 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.

[34] J. Giles. Internet encyclopedias go head to head. *Nature*, 438:900–901, December 2005.

[35] E. Goldman. Wikipedia's labor squeeze and its consequences. *Journal of*

Telecommunications and High Technology Law, 8, 2009.

[36] 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 *18th USENIX Security Symposium*, 2009.

[37] K. Y. Itakura and C. L. Clarke. Using dynamic Markov compression to detect vandalism in the Wikipedia. In *SIGIR '09*, 2009. (Poster paper).

[38] A. Jøsang, R. Hayward, and S. Pope. Trust network analysis with subjective logic. In *Proc. of the Australasian Comp. Science Conference*, 2006.

[39] 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.

[40] S. D. Kamvar, M. T. Schlosser, and H. Garcia-molina. The EigenTrust algorithm for reputation management in P2P networks. In *Proceedings of the Twelfth International World Wide Web Conference*, 2003.

[41] D. Kaplan. Hackers use German Wikipedia to spread malware. `http://www.scmagazineuk.com/hackers-use-german-wikipedia-article-to-spread-malware/`.

[42] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *Journal of the ACM*, 46(5):604–632, 1999.

[43] T. Kriplean, I. Beschastnikh, and D. W. McDonald. Articulations of wikiwork: Uncovering valued work in Wikipedia through barnstars. In *CSCW '08*, pages 47–56, 2008.

[44] T. Lefévre, C. D. Jensen, and T. R. Korsgaard. WRS: The Wikipedia recommender system. In *IFIPTM '09: Trust Management III*, 2009.

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

[46] E. M. Maximilien and M. P. Singh. Toward autonomic web services trust and selection. In *ICSOC '04: Proceedings of the 2nd International Conference on Service Oriented Computing*, 2004.

[47] 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.

[48] 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.

[49] 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.

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

[51] T. G. Papaioannou and G. D. Stamoulis. Reputation-based estimation of individual performance in collaborative and competitive grids. *Future Generation Computer Systems*, 26(8):1327–1335, 2010.

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

[53] S. Pogatchnik. Student hoaxes world's media on Wikipedia. http://www.msnbc.msn.com/id/30699302/.

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

[55] M. Potthast, B. Stein, and T. Holfeld. Overview of the 1st Intl. competition on Wikipedia vandalism detection. In M. Braschler and D. Harman, editors, *Notebook Papers of CLEF 2010 LABs and Workshops*, 2010.

[56] 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 ACM Conference on Supporting Group Work*, 2007.

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

[58] 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.

[59] J. Seigenthaler. A false Wikipedia 'biography'. http://www.usatoday.com/news/opinion/editorials/2005-11-29-wikipedia-edit_x.htm.

[60] R.-K. Sheu, Y.-S. Chang, and S.-M. Yuan. Managing and sharing collaborative files through WWW. *Future Generation Computer Systems*, 17(8):1039–1049, 2001.

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

[62] 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.

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

[64] 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 Conf. on Computational Linguistics*, 2010.

[65] 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*, 2010.

[66] T. Wöhner and R. Peters. Assessing the quality of Wikipedia articles with lifecycle based metrics. In *Proceedings of WikiSym '09*, 2009.

[67] H. Yahyaoui. Trust assessment for web services collaboration. In *ICWS '10: Proceedings of the International Conference on Web Services*, 2010.

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

[69] X. Zhu and S. Gauch. Incorporating quality metrics in centralized/distributed information retrieval on the World Wide Web. In *SIGIR '00: Research and Development in Information Retrieval*, 2000.

[70] H. Zhuge and J. Liu. A fuzzy collaborative assessment approach for knowledge grid. *Future Generation Computer Systems*, 20(1):101–111, 2004.