A Content-Driven Reputation System for the Wikipedia*

B. Thomas Adler
Computer Science Dept.
University of California
1156 High St., Santa Cruz, CA 95064, USA

Luca de Alfaro
Computer Engineering Dept.
University of California
1156 High St., Santa Cruz, CA 95064, USA

ABSTRACT

We present a *content-driven* reputation system for Wikipedia authors. In our system, authors gain reputation when the edits they perform to Wikipedia articles are preserved by subsequent authors, and they lose reputation when their edits are rolled back or undone in short order. Thus, author reputation is computed solely on the basis of content evolution; user-to-user comments or ratings are not used. The author reputation we compute could be used to flag new contributions from low-reputation authors, or it could be used to allow only authors with high reputation to contribute to controversial or critical pages. A reputation system for the Wikipedia could also provide an incentive for high-quality contributions.

We have implemented the proposed system, and we have used it to analyze the entire Italian and French Wikipedias, consisting of a total of 691,551 pages and 5,587,523 revisions. Our results show that our notion of reputation has good *predictive* value: changes performed by low-reputation authors have a significantly larger than average probability of having poor quality, as judged by human observers, and of being later undone, as measured by our algorithms.

Categories and Subject Descriptors

H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces—*Computer-supported cooperative work, Web-based interaction*; K.4.3 [**Computers and Society**]: Organizational Impacts—*Computer-supported collaborative work*

General Terms

Algorithms, experimentation, measurement

Keywords

Wikipedia, reputation, user-generated content

1. INTRODUCTION

The collaborative, web-based creation of bodies of knowledge and information is a phenomenon of rising importance. One of the most successful examples of collaborative content creation, as well as one of the oldest, is the Wikipedia.[1] The Wikipedia is a set of encyclopedias, each in a different language, that cover a very wide swath of knowledge, from history to geography, from math to politics, from pop culture to archeology. Anyone can contribute to the Wikipedia: when an article is displayed, the reader can click on an "edit" button, and is thus able to modify the content of the article. An important feature of the Wikipedia is that for each article, all versions are kept. Users can easily roll back an article to a previous version, undoing the contributions of other users. A fundamental insight behind wiki development is that, if well-intentioned and careful users outnumber ill-intentioned or careless users, then valuable content will predominate, as the undesired contributions are easily undone [4].

We propose to use the information present in article versions to compute a *content-driven* reputation for Wikipedia authors. Authors gain reputation when the edits they perform to Wikipedia articles are preserved by subsequent authors, and they lose reputation when the edits are undone in short order. The lifespan of an edit to a Wikipedia article is inferred from an analysis of the subsequent versions of the article. We call such reputation *content-driven,* since it is computed on the basis of how content evolves, rather than on the basis of user comments or ratings.

A notion of author reputation can serve several purposes in the Wikipedia. Author reputation provides a guide to the value of fresh contributions, which have not yet been vetted by subsequent authors. The reputation of the authors who contributed and vetted the text can be used as a rough guide to the veracity, or *trust,* of the article [19]. Author reputation can also be used for author management. Highly controversial articles could be protected, preventing low-reputation authors from editing them. Alternatively, editors could be alerted when crucial or controversial articles are edited by low-reputation authors. Furthermore, reputation can consitute an incentive for authors to provide high-quality contributions.

1.1 A Content-Driven Reputation System

Most reputation systems are *user-driven:* they are based on users rating each other's contributions or behavior [16, 5]. A famous example is the rating system of Ebay, where buyers and sellers rate each other after performing transactions. In contrast, the reputation system we propose for the Wikipedia requires no user input: rather, authors are evaluated on the basis of how their contributions fare. Sup-

*This is an electronic version of an Article published in the Proceedings of WWW 2007. This research was supported in part by the grant CCR-0132780.

WWW 2007, May 8–12, 2007, Banff, Alberta, Canada.
ACM 978-1-59593-654-7/07/0005.

[1] www.wikipedia.com

pose that an author A contributes to a Wikipedia article by editing it. When another author B subsequently revises the same article, she may choose to preserve some of the edits performed by A. By preserving them, B provides her vote of confidence in these edits, and in author A. Our reputation system will increase the reputation of A in a manner that depends on the amount of preserved edits, as well as on the reputation of B. Specifically, our system measures the following quantities:

- *Text life:* How much of the text inserted by A is still present after B's edit.
- *Edit life:* How much of the reorganization (text reordering and deletions) performed by A is preserved after B's edit.

Author A receives a reputation increment proportional to the size of his contribution, to the text and edit life of the contribution, and to the amount of reputation of B. Author A receives the largest reputation increment when B preserves his contribution in its entirety; conversely, A's reputation is most negatively affected when B rolls back A's contribution. More subtly, the way we compute text and edit life ensures that if B modifies the contribution of A, building upon it, then A still receives a reputation increment. Of the two criteria we use, text life is the most natural. We introduced edit life to account for article reorganizations that do not involve the creation of new text. For instance, if A reorders the paragraphs of an article, little text may be inserted, but edit life enables to give credit to A for this reorganization, if it is preserved by B.

A content-driven reputation system has an intrinsic objectivity advantage over user-driven reputation systems. In order to *badmouth* (damage the reputation of) author B, an author A cannot simply give a negative rating to a contribution by B. Rather, to discredit B, A needs to undo some contribution of B, thus running the risk that if subsequent authors restore B's contribution, it will be A's reputation, rather than B's, to suffer, as A's edit is reversed. Likewise, authors cannot simply praise each other's contributions to enhance their reputations: their contributions must actually withstand the test of time. Finally, the reputation is inferred uniformly and automatically from all edits; the automatic nature of the computation also ensures that the user experience of Wikipedia visitors and authors is minimally affected.

Admittedly, a content-driven reputation system can be less accurate than a user-driven one. Author contributions can be deleted for a variety of reasons, including thorough article rewrites. Our reputation system copes with this in two ways. First, as mentioned above, the way we assign reputation is able to distinguish between reverted and refined contributions. Authors of reverted contributions lose reputation, while authors of subsequently refined contributions do not, even though both kinds of contributions do not survive in the long term. Second, contributions that are appropriate, but that are subsequently rewritten, tend to last longer than inappropriate contributions, which are usually reverted in short order: appropriate contributions thus stand a good chance of receiving at least partial credit.

While it could be arguably interesting to explore combinations of user-generated and content-driven reputation, in this paper we focus on content-driven reputation alone. The computational nature of content-driven reputation enables us to evaluate its effectiveness directly on the Wikipedia: building a separate (and, inevitably, small-scale and low-traffic) test wiki to experiment with the ideas is not necessary. We will present extensive data on the accuracy and performance of our content-driven reputation system, by applying it to the revision history of the Italian and French Wikipedias. The data will show that content-driven reputation can be used to predict the quality of author's contributions.

1.2 Prescriptive, Descriptive, and Predictive Reputation

Reputation is most commonly used for its *prescriptive* and *descriptive* value:

- *Prescriptive value.* Reputation systems specify the way in which users can gain reputation, and thus define, and prescribe, what constitutes "good behavior" on the users' part. Users are strongly encouraged to follow the prescribed behavior, lest their reputation — and their ability to use the system — suffer.
- *Descriptive value.* Reputation systems can be used to classify users, and their contributions, on the basis of their reputation, making it easier to spot high-quality users or contributions, and flagging contributions or users with low reputation.

Ebay's reputation system is an example of a system that has both prescriptive, and descriptive, value. Users are strongly encouraged to accumulate positive feedback: it has been documented that positive feedback increases the probability of closing a transaction, and in the case of sellers, is connected to higher selling price for goods [12]. The original PageRank algorithm [15] constitutes a reputation system for web pages with descriptive intent.

Our reputation system for the Wikipedia has prescriptive and descriptive value. Reputation is built by contributing lasting content; by flagging author reputation, we encourage such contributions. The reputation we compute also has descriptive value: Wikipedia visitors can use the author's reputation as a guide to the trustworthiness of freshly-contributed text.

In addition to prescriptive and descriptive values, we argue that a reputation system that is truly useful for the Wikipedia must also have *predictive* value: an author's reputation should be statistically related to the quality of the author's *future* contributions. To a Wikipedia visitor, it is not generally very useful to know the reputation of an author who made a long-past contribution to an article. If the contribution survived for a long time, its quality is essentially proven: all subsequent editors to the same page have already implicitly voted on the contribution by leaving it in place. Reputation in the Wikipedia is most useful as a guide to the value of fresh contributions, which have not yet been vetted by other authors. Reputation is most useful if it can predict how well these fresh contributions will fare; whether they are likely to be long-lasting, or whether they are likely to be reverted in short order. Furthermore, when reputation is used to grant or deny the ability to edit a page, it is its predictive value that matters: after all, why deny the ability to edit a page, unless there is some statistical reason to believe that there is an increased likelihood that the edits will need to be undone?

1.3 Summary of the Results

In order to measure the predictive value of the proposed reputation, we implemented our reputation system, and we used it to analyze all edits done to the Italian Wikipedia from its inception to October 31, 2005 (154,621 pages, 714,280 versions), and all edits done to the French Wikipedia from its inception, to July 30, 2006 (536,930 pages, 4,837,243 versions).[2] We then measured the statistical correlation between the author's reputation at the time an edit was done, and the subsequent lifespan of the edit. This is a valid statistical test, in the sense that the two variables of the study are computed in independent ways: the reputation of an author at the time of the edit is computed from the behavior of the author *before* the edit, while the lifespan of the edit is determined by events *after* the edit. To summarize the results, we introduce the following informal terminology; we will make the terms fully precise in the following sections:

- *Short-lived edit* is an edit which is 80% undone in the next couple of revisions;
- *Short-lived text* is text that is almost immediately removed (only 20% of the text in a version survives to the next version).
- *Low-reputation author* is an author whose reputation falls in the bottom 20% of the reputation scale.

When measuring the quantity of text or edits, our unit of measurement is a word: this provides a uniform unit of measurement, and ensures that splitting changes in two or more revisions does not affect the results.

Anonymous authors are the largest source of short-lived contributions; as we cannot distinguish their identity, we keep their reputation fixed to the low value assigned to new authors. We first present the results *excluding* anonymous authors from the evaluation. This enables us to measure how well our notion of reputation predicts the quality of contributions by authors with trackable identity. Our results indicate that the fact that an author's reputation is low at the time a contribution is made is statistically correlated with the contribution being short-lived. For the French Wikipedia, the results for edits are as follows:

- While 5.7% of all edits are short-lived, 23.9% of the edits by low-reputation authors are short-lived. Thus, edits by low-reputation authors are 4.2 times more likely than average to be short-lived.
- 32.2% of short-lived edits are performed by low-reputation authors.

For text, the situation is as follows:

- While 1.3% of all text is short-lived, 5.8% of text contributed by low-reputation authors is short-lived. Thus, text by low-reputation authors is 4.5 times more likely than average to be short-lived.
- 37.8% of short-lived text is contributed by low-reputation authors.

Using search terminology, these results can be restated as follows:

- The *recall* provided by low reputation is 32.2% for short-lived edits and 37.8% for short-lived text.
- The *precision* provided by low reputation is 23.9% for short-lived edits and 5.8% for short-lived text.

If we consider also anonymous authors, the results improve markedly:

- The *recall* provided by low reputation, considering also anonymous users, is 82.9% for short-lived edits and 90.4% for short-lived text.
- The *precision* provided by low reputation, considering also anonymous users, is 48.9% for short-lived edits and 19.0% for short-lived text.

While we use search terminology to report these results, we stress that these results are about the reputation's ability to *predict* the longevity of future contributions by an author, rather than *retrieve* past contributions.

The above results may underestimate the recall of our content-driven reputation. We asked a group of volunteers to decide, of the short-lived contributions to the Italian Wikipedia, which ones were of poor quality. The results indicated that short-lived contributions by low-reputation authors were markedly more likely to be of poor quality, compared to similarly short-lived contributions by high-reputation authors. This allowed us to calculate that, excluding anonymous authors, the recall for bad-quality, short-lived edits is 49%, and the recall for bad-quality short-lived text is 79%.

We are unsure of the extent with which prediction precision can be increased. Many authors with low reputation are good, but novice, contributors, who have not had time yet to accumulate reputation. Indeed, an unfortunate effect of the ease with which users can register anew to the Wikipedia is that we cannot trust novices any more than confirmed bad contributors — if we trusted novices more, bad contributors would simply re-register. Furthermore, even authors who contribute short-lived text and edits, and who therefore have low reputation, do not do so consistently, interjecting good contributions among the bad ones.

A basic form of reputation, currently used to screen contributors to controversial pages, is the length of time for which users have been registered to the Wikipedia. We did not have access to this temporal information in our study.[3] As a proxy, we considered the number of edits performed by a user. This is a form of reputation that has no prescriptive value: all it does is encouraging users to split their contributions in multiple, small ones, or even worse, to perform a multitude of gratuitous edits. Indeed, if edit count were used as reputation, it would most likely induce undesirable behaviors by the Wikipedia users. We will show that the predictive value of such a reputation is also inferior to the one we compute, although the difference is not large, and we will discuss why we believe this is the case.

1.4 Related Work

The work most closely related to ours is [19], where the revision history of a Wikipedia article is used to compute a trust value for the article. Dynamic Bayesian networks

[2]These numbers reflect the fact that we consider only the last among consecutive versions by the same author, as explained later.

[3]Wikipedia makes only a limited amount of user information available for download.

are used to model the evolution of trust level over the versions. At each revision, the inputs to the network are a priori models of trust of authors (determined by their Wikipedia ranks), and the amount of added and deleted text. The paper shows that this approach can be used to predict the quality of an article; for instance, it can be used to predict when an article in a test set can be used as a featured article. Differently from the current work, author trustworthiness is taken as input; we compute author reputation as output. Several approaches for computing text trust are outlined in [13]. A simpler approach to text trust, based solely on text age, is advocated in [3].

Document trust, and author reputation, have different applications. Document trust provides a measure of confidence in the accuracy of the document. Author reputation is most useful as an indicator of quality for fresh text, and can be used to manage author activity, for instance, preventing low-reputation authors from editing some pages.

Reputation systems in e-commerce and social networks has been extensively studied (see, e.g., [10, 16, 5, 9]); the reputation in those systems is generally user-driven, rather than content-driven as in our case. Related is also work on trust in social networks (see, e.g., [7, 6]).

The history flow of text contributed by Wikipedia authors has been studied with flow visualization methods in [18]; the results have been used to analyze a number of interesting patterns in the content evolution of Wikipedia articles. Work on mining software revision logs (see, e.g., [11]) is similar in its emphasis of in-depth analysis of revision logs; the aim, however, is to find revision patterns and indicators that point to software defects, rather than to develop a notion of author reputation.

2. CONTENT-DRIVEN REPUTATION

Reputation systems can be classified into two broad categories: *chronological,* where the reputation is computed from the chronological sequence of ratings a user receives, and *fixpoint,* where the reputation is computed via a fixpoint calculation performed over the graph of feedbacks. The Ebay reputation system is an example of a chronological system, while the PageRank and HITS algorithms are examples of fixpoint algorithms [15, 10]. We chose to follow the chronological approach to develop our content-driven reputation for the Wikipedia. The chief advantage of a chronological approach is that it is computationally lightweight. When an author revises a Wikipedia article, we can efficiently compute the feedback to authors of previous versions of the same article, and we can modify their reputation in real-time, with little impact on server response time.

2.1 Notation

The following notation will be used throughout the paper. We assume that we have $n > 0$ versions $v_0, v_1, v_2, \ldots, v_n$ of a document; version v_0 is empty, and version v_i, for $1 \leq i \leq n$, is obtained by author a_i performing a revision $r_i : v_{i-1} \rightsquigarrow v_i$. We refer to the change set corresponding to $r_i : v_{i-1} \rightsquigarrow v_i$ as the *edit* performed at r_i: the edit consists of the text insertions, deletions, displacements, and replacements that led from v_{i-1} to v_i. When editing a versioned documents, authors commonly save several versions in a short time frame. To ensure that such behavior does not affect reputations, we *filter* the versions, keeping only the *last* of consecutive versions by the same author; we assume thus that for $1 \leq i < n$ we have $a_i \neq a_{i+1}$. Every version v_i, for $0 \leq i \leq n$, consists of a sequence $[w_1^i, \ldots, w_{m_i}^i]$ of words, where m_i is the number of words of v_i; we have $m_0 = 0$. A *word* is a whitespace-delimited sequence of characters in the Wiki markup language: we work at the level of such markup language, rather than at the level of the HTML produced by the wiki engine. Given a series of versions $v_0, v_1, \ldots, v_n$ of a text document, we assume that we can compute the following quantities:

- $txt(i, j)$, for $0 < i \leq j \leq n$, is the amount of text (measured in number of words) that is introduced by r_i in v_i, and that is still present (and due to the same author a_i) in v_j. In particular, $txt(i, i)$ is the amount of new text added by r_i.
- $d(v_i, v_j)$, for $0 < i < j \leq n$, is the *edit distance* between v_i and v_j, and measures how much change (word additions, deletions, replacements, displacements, etc.) there has been in going from v_i to v_j.

We will describe in the next section how these quantities can be computed from the series of versions $v_0, \ldots, v_n$.

2.2 Content-Driven Reputation in a Versioned Document

We propose the following method for computing content-driven reputation in a versioned document. Consider a revision $r_i : v_{i-1} \rightsquigarrow v_i$, performed by author a_i, for some $0 < i \leq n$. Each of the subsequent authors $a_{i+1}, a_{i+2}, \ldots$ can either retain, or remove, the edits performed by a_i at r_i. Thus, we consider later versions v_j of the document, for $i < j \leq n$, and we increase, or decrease, the reputation of a_i according to how much of the change performed at r_i has survived until v_j. The greater is the reputation of the author a_j of v_j, the greater is the change in the reputation of a_i. This ensures that high-reputation authors risk little of their reputation when fighting revision wars against low reputation authors, such as anonymous authors. If this were not the case, high-reputation authors would be wary of undoing damage done by low-reputation authors, including spammers, for fear of reprisal.

We use two criteria for awarding reputation: the survival of text, and the survival of edits. Text survival is the most natural of these two criteria. Adding new text to a Wikipedia article is a fundamental way of contributing — it is how knowledge is added to the Wikipedia. However, text survival alone fails to capture many ways in which authors contribute to the Wikipedia. If a user rearranges the content of an article without introducing new text, the contribution cannot be captured by text survival. Similarly, reverting an act of vandalism does not result in the introduction of new text.[4] Edit survival captures how long the re-arrangements performed by an author last in the history of a Wikipedia article, and captures all of the above ways of contributing.

2.3 Accounting for Text Survival

If text introduced by r_i is still present in v_j, for $0 < i < j \leq n$, this indicates that author a_j, who performed the revision $r_j : v_{j-1} \rightsquigarrow v_j$, agrees that the text is valuable. To reflect this, we increase the reputation of a_i in a manner that is proportional both to the amount of residual text, and to the reputation of a_i. Thus, we propose the following rule.

[4]Our algorithms distinguish between new text, and the reintroduction of previously deleted text.

RULE 1. **(reputation update due to text survival)** *When the revision r_j occurs, for all $0 < i < j$ such that $j - i \leq 10$ and $a_j \neq a_i$, we add to the reputation of a_i the amount:*

$$c_{scale} \cdot c_{text} \cdot \frac{txt(i,j)}{txt(i,i)} \cdot (txt(i,i))^{c_{len}} \cdot \log(1 + R(a_j, r_j)),$$

where $c_{scale} > 0$, $c_{text} \in [0,1]$, and $c_{len} \in [0,1]$ are parameters, and where $R(a_j, r_j)$ is the reputation of a_j at the time r_j is performed.

In the rule, $txt(i,j)/txt(i,i)$ is the fraction of text introduced at version v_i that is still present in version v_j; this is a measure of the "quality" of r_i. The quantity $\log(1+R(a_j, r_j))$ is the "weight" of the reputation of a_j, that is, how much the reputation of a_j lends credibility to the judgements of a_j. In our reputation system, the reputations of many regular contributors soar to very high values; using a logarithmic weight for reputation ensures that the feedback coming from new authors is not completely overridden by the feedback coming from such regular contributors. The parameters c_{scale}, c_{text} and c_{len} will be determined experimentally via an optimization process, as described later. The parameter $c_{len} \in [0,1]$ is an exponent that specifies how to take into account the length of the original contribution: if $c_{len} = 1$, then the increment is proportional to the length of the original contribution; if $c_{len} = 0$, then the increment does not depend on the length of the original contribution. The parameter c_{scale} specifies how much the reputation should vary in response to an individual feedback. The parameter c_{text} specifies how much the feedback should depends on residual text (Rule 1) or residual edit (Rule 2, presented later).

To give feedback on a revision, the rule considers at most 10 successive versions. This ensures that contributors to early versions of articles do not accumulate disproportionate amounts of reputation. A rule using exponential decay, rather than a sharp cutoff at 10, would probably have been preferable, at a slightly larger computational cost.

2.4 Accounting for Edit Survival

To judge how much of a revision $r_i : v_{i-1} \rightsquigarrow v_i$ is preserved in a later version v_j, for $0 < i < j \leq n$, we reason as follows. We wish to rate r_i higher, if r_i made the article more similar to v_j. In particular, we wish to credit a_i in proportion to how much of the change r_i is directed towards v_j. This suggests using the formula:

$$ELong(i,j) = \frac{d(v_{i-1}, v_j) - d(v_i, v_j)}{d(v_{i-1}, v_i)}. \qquad (1)$$

If d satisfies the triangular inequality (as our edit distance does, to a high degree of accuracy), then $ELong(i,j) \in [-1,1]$. For two consecutive edits r_i, r_{i+1}, if r_i is completely undone in r_{i+1} (as is common when r_i consists in the introduction of spam or inappropriate material), then $ELong(i, i+1) = -1$; if r_{i+1} entirely preserves r_i, then $ELong(i, i+1) = +1$. For more distant edits, $ELong(i,j)$ is a measure of how much of the edit performed during r_i is undone (value -1) or preserved (value $+1$) before r_j.

Note that $ELong(i,j) < 0$, i.e., j votes to lower the reputation of i, only when $d(v_{i-1}, v_j) < d(v_i, v_j)$, that is, when r_j is closer to the version v_{i-1} *preceding* a_i's contribution, than to v_i. Thus, the revision performed by a_i needs to have been undone at least in part by a_{i+1}, a_{i+2}, ..., a_j, for a_i's reputation to suffer. This is one of the two mechanisms we have to ensure that authors whose contributions are rewritten, rather than rolled back, still receive reputation in return for their efforts. The second mechanism is described in the next section. We use the following rule for updating reputations.

RULE 2. **(reputation update due to edit survival)** *When the revision r_j occurs, for all $0 < i < j$ such that $j - i \leq 3$, we add to the reputation of a_i an amount q determined as follows. If $a_j = a_i$ or $d(v_{i-1}, v_i) = 0$, then $q = 0$; otherwise, q is determined by the following algorithm.*

$$q := \frac{c_{slack} \cdot d(v_{i-1}, v_j) - d(v_i, v_j)}{d(v_{i-1}, v_i)}$$

if $q < 0$ **then** $q := q \cdot c_{punish}$ **endif**

$$q := q \cdot c_{scale} \cdot (1 - c_{text}) \cdot (d(v_{i-1}, v_i))^{c_{len}} \cdot \log(1 + R(a_j, r_j))$$

In the algorithm, $c_{punish} \geq 1$, $c_{slack} \geq 1$, $c_{scale} > 0$, $c_{text} \in [0,1]$, and $c_{len} \in [0,1]$ are parameters, and $R(a_j, r_j)$ is the reputation of a_j at the time r_j is performed.

The rule adopts a modified version of (1). The parameter $c_{slack} > 1$ is used to spare a_i from punishment when $r_i : v_{i-1} \rightsquigarrow v_i$ is only slightly counterproductive. On the other hand, when punishment is incurred, its magnitude is magnified by the amount c_{punish}, raising the reputation cost of edits that are later undone. The parameters c_{slack} and c_{punish}, as well as c_{scale}, c_{text} and c_{len}, will be determined via an optimization process. To assign edit feedback, we have chosen to consider only the 3 previous versions of an article. This approach proved adequate for analyzing an already-existing wiki, in which authors could not modify their behavior using knowledge of this threshold. If the proposed content-driven reputation were to be used on a live Wiki, it would be advisable to replace this hard threshold by a scheme in which the feedback of v_j on r_i is weighed by a gradually decreasing function of $j - i$ (such as $\exp(c \cdot (i - j))$ for some $c > 0$).

2.5 Computing Content-Driven Reputation

We compute the reputation for Wikipedia authors as follows. We examine all revisions in chronological order, thus simulating the same order in which they were submitted to the Wikipedia servers. We initialize the reputations of all authors to the value 0.1; the reputation of anonymous authors is fixed to 0.1. We choose a positive initial value to ensure that the weight $\log(1+r)$ of an initial reputation $r = 0.1$ is non-zero, priming the process of reputation computation. This choice of initial value is not particularly critical (the parameter c_{scale} may need to be adjusted for optimal performance, if this initial value is changed). As the revisions are processed, we use Rules 1 and 2 to update the reputations of authors in the system. When updating reputations, we ensure that they never become negative, and that they never grow beyond a bound $c_{maxrep} > 0$. The bound c_{maxrep} prevents frequent contributors from accumulating unbounded amounts of reputation, and becoming essentially immune to negative feedback. The value of c_{maxrep} will be once again determined via optimization techniques.

Wikipedia allows users to register, and create an *author* identity, whenever they wish. As a consequence, we need to make the initial reputation of new authors very low, close to the minimum possible (in our case, 0). If we made the initial reputation of new authors any higher, then authors, after committing revisions that damage their reputation,

would simply re-register as new users to gain the higher value. An unfortunate side-effect of allowing people to obtain new identities at will is that we cannot presume that people are innocent until proven otherwise: we have to assign to newcomers the same reputation as proven offenders. This is a contributing factor to our reputation having low *precision:* many authors who have low reputation still perform very good quality revisions, as they are simply new authors, rather than proven offenders.

3. TEXT LONGEVITY AND EDIT DISTANCE IN VERSIONED DOCUMENTS

In this section, we propose methods for tracking text authorship, and computing edit distances, in versioned documents. We developed our algorithms starting from standard text-difference algorithms, such as those of [17, 14, 1]. However, we adapted the algorithms in several places, to enable them to better cope with the versioned nature of the Wikipedia, and with the kind of edits that authors perform.

3.1 Tracking Text Authorship

Given a sequence of versions $v_0, v_1, \ldots, v_n$, we describe an algorithm for computing $txt(i, j)$ for $0 < i \leq j \leq n$. Superficially, it might seem that to compute $txt(i, j)$, we need to consider only three versions of the document: v_{i-1}, v_i, and v_j. From v_{i-1} and v_i we can derive the text that is added at revision $r_i : v_{i-1} \rightsquigarrow v_i$; we can then check how much of it survives until v_j. This approach, however, is not appropriate for a versioned document like a wiki page, where authors are allowed to inspect and restore text from any previous version of a document. For example, consider the case in which revision $r_{i-1} : v_{i-2} \rightsquigarrow v_{i-1}$ is the work of a spammer, who erases entirely the text of v_{i-2}, and replaces it with spurious material; such spam insertions are a common occurrence in open wikis. When author a_i views the page, she realizes that it has been damaged, and she reverts it to the previous version, so that $v_i = v_{i-2}$. If we derived the text added by r_i by considering v_{i-1} and v_i only, it would appear to us that a_i is the original author of all the text in v_i, but this is clearly not the case: she simply restored pre-existing text. To compute the text added at a revision $r_i : v_{i-1} \rightsquigarrow v_i$, we keep track of both the text that is in v_{i-1}, and of the text that used to be present in previous versions, and that has subsequently been deleted.

Our algorithm proceeds as follows. We call a *chunk* a list $c = [(w_1, q_1), \ldots, (w_k, q_k)]$, where for $1 \leq j \leq k$, w_j is a word, and $q_j \in \{1, \ldots, n\}$ is a version number. A chunk represents a list of contiguous words, each labeled with the version where it originated. The algorithm computes, for each version v_i, $1 \leq i \leq n$, its *chunk list* $C_i = [c_0^i, c_1^i, \ldots, c_k^i]$. The chunk c_0^i is the *live* chunk, and it consists of the text of v_i, with each word labeled with the version where the word was introduced; thus, if $v_i = [w_1^i, \ldots, w_{m_i}^i]$, we have $c_0^i = [(w_1, q_1), \ldots, (w_{m_i}, q_{m_i})]$, for some $q_1, \ldots, q_{m_i}$. The chunks $c_1^i, \ldots, c_k^i$ are *dead* chunks, and they represent contiguous portions of text that used to be present in some version of the document prior to i. Given the chunk list $C_i = [c_0^i, c_1^i, \ldots, c_k^i]$ for document v_i, we can compute $txt(j, i)$ via

$$txt(j, i) = \left|\left\{(u, j) \mid \exists u.(u, j) \in c_0^i\right\}\right|,$$

for $1 \leq i \leq j \leq n$.

To compute C_i for all versions i of a document, we propose an algorithm that proceeds as follows. For the initial version, we let $C_1 = [[(w_1^1, 1), (w_2^1, 1), \ldots, (w_{m_1}^1, 1)]]$. For $1 \leq i < n$, the algorithm computes C_{i+1} from $C_i = [c_0^i, c_1^i, \ldots, c_k^i]$ and v_{i+1}. To compute the live chunk c_0^{i+1}, we match contiguous portions of text in v_{i+1} with contiguous text in any of the chunks in C_i; the matching words in v_{i+1} are labeled with the version index that labels them in C_i, and represent words that were introduced prior to version v_{i+1}. Any words of v_{i+1} that cannot be thus matched are considered new in v_{i+1}, and are labeled with version index $i + 1$. The dead chunks $c_1^{i+1}, \ldots, c_l^{i+1}$ of C_{i+1} are then obtained as the portions of the chunks in C_i that were not matched by any text in v_{i+1}. We allow the same text in C_i to be matched multiple timed in v_{i+1}: if a contributor copies multiple times text present in v_i or in prior versions in order to obtain v_{i+1}, the replicated text should not be counted as new in v_{i+1}. Considering replicated text as new would open the door to a *duplication attack,* whereby an attacker duplicates text in a revision r_i, and then removes the original text in a revision $r_k : v_{k-1} \rightsquigarrow v_k$ with $k > i$. From version v_k onwards, the text would be attributed to the attacker rather than to the original author.

Matching v_{i+1} with C_i is a matter of finding matches between text strings, and several algorithms have been presented in the literature to accomplish this in an efficient manner (see, e.g., [8, 17, 14, 1]). We experimented extensively, and the algorithm that gave the best combination of efficiency and accuracy was a variation of a standard greedy algorithm. In standard greedy algorithms, such as [14, 1], longest matches are determined first. In our algorithm, we define a notion of match *quality,* and we determine first matches of highest quality. To define match quality, we let m_{i+1} be the length of v_{i+1}, and we let m' be the length of the chunk of C_i where the match is found (all length and indices are measured in number of words). Let l be the length of the match, and assume that the match begins at word $k' \leq m'$ in the chunk, and at word $k_{i+1} \leq m_{i+1}$ in v_{i+1}. We define match quality as follows:

- If the match occurs between v_{i+1} and the live chunk, then the quality is:
$$\frac{l}{\min(m_{i+1}, m')} - 0.3 \cdot \left|\frac{k'}{m'} - \frac{k_{i+1}}{m_{i+1}}\right| .$$
- If the match occurs between v_{i+1} and a dead chunk, then the quality is 0 if $l < 4$, and is $l/\min(m_{i+1}, m') - 0.4$ otherwise.

Thus, the quality of a match is the higher, the longer the match is. If the match is with the live chunk, a match has higher quality if the text appears in the same relative position in v_{i+1} and in v_i. Matches with dead chunks have somewhat lower quality than matches with the live chunk: this corresponds to the fact that, if some text can be traced both to the previous version (the live chunk), and to some text that was previously deleted, the most likely match is with the text of the previous version. Moreover, matches with dead chunks have to be at least of length 4: this avoids misclassifying common words in new text as reintroductions of previously-deleted text. The coefficients in the above definition of quality have been determined experimentally, comparing human judgements of authorship to the

algorithmically computed ones for many pages of the Italian Wikipedia. The text survival algorithm we developed is efficient: the main bottleneck, when computing text authorship, is not the running time of the algorithm, but rather, the time required to retrieve all versions of a page from the MySQL database in which Wikipedia pages are stored.[5]

3.2 Computing Edit Distances

For two versions v, v', the edit distance $d(v, v')$ is a measure of how many word insertions, deletions, replacements, and displacements are required to change v into v' [17]. In order to compute the edit distance between two versions v and v', we use the same greedy algorithm for text matching that we used for text survival, except that each portion of text in v (resp. v') can be matched *at most once* with a portion of text in v' (resp. v). Thus, text duplication is captured as an edit. The output of the greedy matching is modified, as is standard in measurements of edit distance, so that it outputs a list L of elements that describe how v' can be obtained from v:

- $I(j, k)$: k words are inserted at position j;
- $D(j, k)$: k words are deleted at position j;
- $M(j, h, k)$: k words are moved from position j in v to position h in v'.

We compute the total amount I_{tot} of inserted text by summing, for each $I(j, k) \in L$, the length k; similarly, we obtain the total amount D_{tot} of deleted text by summing, for each $D(j, k) \in L$, the length k. We take into account insertions and deletions via the formula $\max(I_{tot}, D_{tot}) - \frac{1}{2}\min(I_{tot}, D_{tot})$: thus, every word that is inserted or removed contributes 1 to the distance, and every word that is replaced contributes $\frac{1}{2}$. The motivation for this treatment of replacements is as follows. Suppose that author a_i edits v_{i-1} adding a new paragraph consisting of k words, obtaining v_i, and suppose that author a_{i+1} then rewrites completely the paragraph (keeping it of equal length), obtaining v_{i+1}. If we accounted for insertion and deletions via $I_{tot} + D_{tot}$, we would have $d(v_{i+1}, v_{i-1}) = k$ and $d(v_{i+1}, v_i) = 2k$: consequently, according to (1) and Rule 2, a_i's reputation would suffer. With our formula, we have instead $d(v_{i+1}, v_{i-1}) = k$ and $d(v_{i+1}, v_i) = k/2$, so that a_i's reputation increases. This is the second mechanism we have for ensuring that authors whose contributions are rewritten, rather than rolled back, still receive credit for their effort.

We account for text moves between versions v and v' as follows. Let l, l' be the lengths (in words) of v and v', respectively. Each time a block of text of length k_1 exchanges position with a block of text of length k_2, we count this as $k_1 \cdot k_2 / \max(l, l')$. Thus, a word that moves across k other words contributes $k/\max(l, l')$ to the distance: the contribution approaches 1 as the word is moved across the whole document. The total contribution M_{tot} of all moves can be computed by adding $k \cdot k'$, for all pairs of moves $M(j, h, k) \in L$ and $M(j', h', k') \in L$ such that $j < j'$ and $h > h'$ (this ensures that every crossing is counted once). We finally define:

$$d(r, r') = \max(I_{tot}, D_{tot}) + M_{tot} - \tfrac{1}{2}\min(I_{tot}, D_{tot}).$$

[5]The measurement was done on a PC with AMD Athlon 64 3000+ CPU, two hard drives configured in RAID 1 (mirroring), and 1 GB of memory.

Due to the nature of the greedy algorithms used for text matching, and of the definitions above, our edit distance is not guaranteed to satisfy the triangular inequality. However, we found experimentally that the proposed edit distance, on Wikipedia pages, satisfies the triangular inequality within approximately one unit (one word) for well over 99% of triples of versions of the same page.

4. EVALUATION METRICS

We now develop quantitative measures of the ability of our content-driven reputation to predict the quality of future revisions. For a revision $r_i : v_{i-1} \rightsquigarrow v_i$ in a sequence $v_0, v_1, \ldots, v_n$ of versions, let $\rho_t(r_i) = txt(i, i)$ be the new text introduced at r_i, and $\rho_e(r_i) = d(v_{i-1}, v_i)$ be the amount of editing involved in r_i. We define edit and text longevity as follows:

- The *edit longevity* $\alpha_e(r_i) \in [-1, 1]$ of r_i is the average of $ELong(i, j)$ for $i < j \leq \min(i + 3, n)$.
- The *text longevity* $\alpha_t(r_i) \in [0, 1]$ of r_i is the solution to the following equation:

$$\sum_{j=i}^{n} txt(i, j) = txt(i, i) \cdot \sum_{j=1}^{n} (\alpha_t(r_i))^{j-i} . \qquad (2)$$

Thus, $\alpha_t(r_i)$ is the coefficient of exponential decay of the text introduced by r_i: on average, after k revisions, only a fraction $(\alpha_t(r_i))^k$ of the introduced text survives. As all quantities in (2) except $\alpha_t(r_i)$ are known, we can solve for $\alpha_t(r_i)$ using standard numerical methods. We also indicate by $rep(r_i)$ the reputation of the author a_i of r_i at the time r_i was performed. We note that $rep(r_i)$ is computed from the history of the Wikipedia before r_i, while $\alpha_e(r_i)$ and $\alpha_t(r_i)$ depend only on events after r_i. Moreover, $\alpha_e(r_i)$ and $\alpha_t(r_i)$ can be computed independently of reputations.

Let R be the set of all revisions in the Wikipedia (of all articles). We view revisions as a probabilistic process, with R as the set of outcomes. We associate with each revision a probability mass (a weight) proportional to the number of words it affects. This ensures that the metrics are not affected if revisions by the same author are combined or split into multiple ones. Since we keep only the last among consecutive revisions by the same user, a "revision" is a rather arbitrary unit of measurement, while a "revision amount" provides a better metric. Thus, when studying edit longevity, we associate with each $r \in R$ a probability mass proportional to $\rho_e(r)$, giving rise to the probability measure $\Pr_e$. Similarly, when studying text longevity, we associate with each $r \in R$ a probability mass proportional to $\rho_t(r)$, giving rise to the probability measure $\Pr_t$.

In order to develop figures of merit for our reputation, we define the following terminology (used already in the introduction in informal fashion):

- We say that the edit performed in r is *short-lived* if $\alpha_e(r) \leq -0.8$.
- We say that the new text added in r is *short-lived* if $\alpha_t(r) \leq 0.2$, indicating that at most 20% of it, on average, survives from one version to the next.
- We say that a revision r is *low-reputation* if $\log(1 + rep(r)) \leq \log(1 + c_{maxrep})/5$, indicating that the reputation, after logarithmic scaling, falls in the lowest 20% of the range.

Correspondingly, we define three random variables $S_e, S_t, L : R \mapsto \{0, 1\}$ as follows, for all $r \in R$:

- $S_e(r) = 1$ if $\alpha_e(r) \leq -0.8$, and $S_e(r) = 0$ otherwise.
- $S_t(r) = 1$ if $\alpha_e(r) \leq 0.2$, and $S_t(r) = 0$ otherwise.
- $L(r) = 1$ if $\log(1 + rep(r)) \leq \log(1 + c_{maxrep})/5$, and $L(r) = 0$ otherwise.

For short-lived text, the *precision* is $prec_t = \Pr_t(S_t = 1 \mid L = 1)$, and the *recall* is $rec_t = \Pr_t(L = 1 \mid S_t = 1)$. Similarly, for short-lived edits, we define the precision is $prec_e = \Pr_e(S_e = 1 \mid L = 1)$, and the recall is $rec_e = \Pr_e(L = 1 \mid S_e = 1)$. These quantities can be computed as usual; for instance,

$$\Pr_e(S_e = 1 \mid L = 1) = \frac{\sum_{r \in R} S_e(r) \cdot L(r) \cdot \rho_e(r)}{\sum_{r \in R} L(r) \cdot \rho_e(r)}.$$

We also define:

$$boost_e = \frac{\Pr_e(S_e = 1 \mid L = 1)}{\Pr_e(S_e = 1)} = \frac{\Pr_e(S_e = 1, L = 1)}{\Pr_e(S_e = 1) \cdot \Pr_e(L = 1)}$$

$$boost_t = \frac{\Pr_t(S_t = 1 \mid L = 1)}{\Pr_t(S_t = 1)} = \frac{\Pr_t(S_t = 1, L = 1)}{\Pr_t(S_t = 1) \cdot \Pr_t(L = 1)}$$

Intuitively, $boost_e$ indicates how much more likely than average it is that edits produced by low-reputation authors are short-lived. The quantity $boost_t$ has a similar meaning. Our last indicator of quality are the *coefficients of constraint*

$$\kappa_e = I_e(S_e, L)/H_e(L) \qquad \kappa_t = I_t(S_t, L)/H_t(L),$$

where I_e is the *mutual information* of S_e and L, computed with respect to $\Pr_e$, and H_e is the entropy of L, computed with respect to $\Pr_e$ [2]; similarly for $I_t(S_t, L)$ and $H_t(L)$. The quantity κ_e is the fraction of the entropy of the edit longevity which can be explained by the reputation of the author; this is an information-theoretic measure of correlation. The quantity κ_t has an analogous meaning.

To assign a value to the coefficients c_{scale}, c_{slack}, c_{punish}, c_{text}, c_{len}, and c_{maxrep}, we implemented a search procedure, whose goal was to find values for the parameters that maximized a given objective function. We applied the search procedure to the Italian Wikipedia, reserving the French Wikipedia for validation, once the coefficients were determined. We experimented with $I_e(S_e, L)$ and $prec_e \cdot rec_e$ as objective functions, and they gave very similar results.

5. EXPERIMENTAL RESULTS

To evaluate our content-driven reputation, we considered two Wikipedias:

- The Italian Wikipedia, consisting of 154,621 articles and 714,280 *filtered* revisions; we used a snapshot dated December 11, 2005.
- The French Wikipedia, consisting of 536,930 articles and 4,837,243 *filtered* revisions; we used a snapshot dated October 14, 2006.

Of both wikipedias, we studied only "NS_MAIN" pages, which correspond to ordinary articles (other pages are used as comment pages, or have other specialized purposes). Moreover, to allow the accurate computation of edit longevity, we used only revisions that occurred before October 31, 2005 for the Italian Wikipedia, and before July 31, 2006 for the French one. Our algorithms for computing content-driven reputation depend on the value of six parameters, as mentioned earlier. We determined values for these parameters by searching the parameter space to optimize the coefficient of constraint, using the Italian Wikipedia as a training set; the values we determined are:

$$c_{scale} = 13.08 \qquad c_{punish} = 19.09 \qquad c_{len} = 0.60$$
$$c_{slack} = 2.20 \qquad c_{text} = 0.60 \qquad c_{maxrep} = 22026$$

We then analyzed the Italian and French Wikipedias using the above values. The results are summarized in Table 1. The results are better for the larger French Wikipedia; in particular, the reputation's ability to predict short-lived edits is better on the French than on the Italian Wikipedias. We are not sure whether this depends on different dynamics in the two Wikipedias, or whether it is due to the greater age (and size) of the French Wikipedia; we plan to study this in further work. We see that edits performed by low-reputation authors are four times as likely as the average to be short-lived.

To investigate how many of the edits had a short life due to bad quality, we asked a group of 7 volunteers to rate revisions performed to the Italian Wikipedia. We selected the revisions to be ranked so that they contained representatives of all 4 combinations of high/low reputation author, and high/low longevity. We asked the volunteers to rate the revisions with +1 (good), 0 (neutral), and −1 (bad); in total, 680 revisions were ranked. The results, summarized in Table 2, are striking. Of the short-lived edits performed by low-reputation users, fully 66% were judged bad. On the other hand, less than 19% of the short-lived edits performed by high-reputation users were judged bad. We analyzed in detail the relationship between user reputation, and the percentage of short-lived text and edits that users considered bad. Using these results, we computed the approximate recall factors on the Italian Wikipedia of content-driven reputation for *bad* edits, as judged by users, rather than short-lived ones:

- The recall for short-lived edits that are judged to be bad is over 49%.
- The recall for short-lived text that is judged to be bad is over 79%.

These results clearly indicate that our content-driven reputation is a very effective tool for spotting, at the moment they are introduced, bad contributions that will later be undone. There is some margin of error in this data, as our basis for evaluation is a small number of manually-rated revisions, and human judgement on the same revisions often contained discrepancies.

The fact that so few of the short-lived edits performed by high-reputation authors were judged to be of bad quality points to the fact that edits can be undone for reasons unrelated to quality. Many Wikipedia articles deal with current events; edits to those articles are undone regularly, even though they may be of good quality. Our algorithms do not treat in any special way current-events pages. Other Wikipedia edits are administrative in nature, flagging pages that need work or formatting; when these flags are removed, we classify it as text deletion. Furthermore, our algorithms do not track text across articles, so that when text is moved

	Precision		Recall		Boost		Coeff. of constr.	
	Edit $prec_e$	Text $prec_t$	Edit rec_e	Text rec_t	Edit $boost_e$	Text $boost_t$	Edit κ_e	Text κ_t
Excluding anonymous authors:								
French Wikipedia	23.92	5.85	32.24	37.80	4.21	4.51	7.33	6.29
Italian Wikipedia	14.15	3.94	19.39	38.69	4.03	5.83	3.35	7.17
Including anonymous authors:								
French Wikipedia	48.94	19.01	82.86	90.42	2.35	2.97	25.29	23.00
Italian Wikipedia	30.57	7.64	71.29	84.09	3.43	3.58	19.83	17.49

Table 1: Summary of the performance of content-driven reputation over the Italian and French Wikipedias. All data are expressed as percentages.

Reputation	Judged bad	Judged good
Short-lived edits:		
Low [0.0–0.2]	66 %	19 %
Normal [0.2–1.0]	16 %	68 %
Short-lived text:		
Low [0.0–0.2]	74 %	13 %
Normal [0.2–1.0]	14 %	85 %

Table 2: User ranking of short-lived edits and text, as a function of author reputation, for the Italian Wikipedia. In square brackets, we give the interval where the normalized value $\log(1+r)/\log(1+c_{maxrep})$ of a reputation r falls. The precentages do not add to 100%, because users could also rank changes as "neutral".

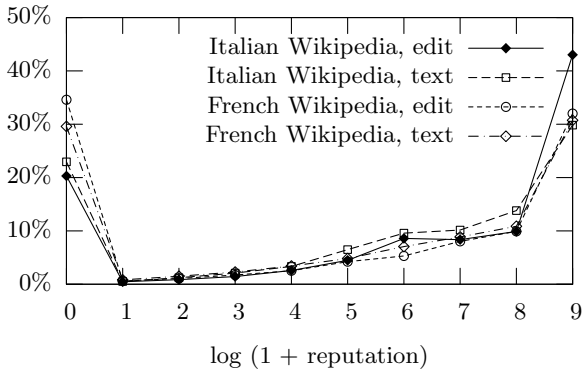

Figure 1: Percentage of text and edit contributed to the Italian and French Wikipedias, according to author reputation. The data includes anonymous authors.

from one article to another, they classify it as deleted from the source article.

From Table 1, we note that the precision is low, by search standards. Our problem, however, is a prediction problem, not a retrieval problem, and thus it is intrinsically different. The group of authors with low reputation includes many authors who are good contributors, but who are new to the Wikipedia, so that they have not had time yet to build up their reputation.

Figure 1 provides a breakdown of the amount of edits and text additions performed, according to the reputation of the author.

5.1 Comparison with Edit-Count Reputation

We compared the performance of our content-driven reputation to another basic form of reputation: *edit count*. It is commonly believed that, as Wikipedia authors gain experience (through revision comments, talk pages, and reading articles on Wikipedia standards), the quality of their submissions goes up. Hence, it is reasonable to take edit count, that is, the number of edits performed, as a form of reputation. We compare the performance of edit count, and of content-driven reputation, in Table 3. The comparison does not include anonymous authors, as we do not have a meaningful notion of edit-count for them. As we can see, according to our metrics, content-driven reputation performs slightly better than edit-count reputation on both the Italian and French Wikipedias.

We believe that one reason edit-count based reputation performs well in our measurements is that authors, after performing edits that are often criticized and reverted, commonly either give up their identity in favor of a "fresh" one, thus zeroing their edit-count reputation (thus "punishing" themselves), or stop contributing to the Wikipedia. However, we believe that the good performance of edit count is an artifact, due to the fact that edit count is applied to an already-existing history of contributions. Were it announced that edit count is the chosen notion of reputation, authors would most likely modify their behavior in a way that both rendered edit count useless, and damaged the Wikipedia. For instance, it is likely that, were edit count the measure of reputation, authors would adopt strategies (and automated robots) for performing very many unneeded edits to the Wikipedia, causing instability and damage. In other words, edit count as reputation measure has very little prescriptive value. In contrast, we believe our content-driven reputation, by prizing long-lasting edits and content, would encourage constructive behavior on the part of the authors.

5.2 Text Age and Author Reputation as Trust Criteria

The age of text in the Wikipedia is often considered as an indicator of text trustworthiness, the idea being that text that has been part of an article for a longer time has been vetted by more contributors, and thus, it is more likely to be correct [3]. We were interested in testing the hypothesis that author reputation, in addition to text age, can be a useful indicator of trustworthiness, especially for

	Precision		Recall		Boost		Coeff. of constr.	
	Edit $prec_e$	Text $prec_t$	Edit rec_e	Text rec_t	Edit $boost_e$	Text $boost_t$	Edit κ_e	Text κ_t
Italian Wikipedia:								
Content-driven reputation	14.15	3.94	19.39	38.69	4.03	5.83	3.35	7.17
Edit count as reputation	11.50	3.32	19.09	39.52	3.27	4.91	2.53	6.35
French Wikipedia:								
Content-driven reputation	23.92	5.85	32.24	37.80	4.21	4.51	7.33	6.29
Edit count as reputation	21.62	5.63	28.30	37.92	3.81	4.34	5.61	6.08

Table 3: Summary of the performance of content-driven reputation over the Italian and French Wikipedias. All data are expressed as percentages. Anonymous authors are not included in the comparison.

text that has just been added to a page, and thus that has not yet been vetted by other contributors. Let *fresh text* be the text that has just been inserted in a Wikipedia article. We considered all text that is fresh in all the Italian Wikipedia, and we measured that 3.87 % of this fresh text is deleted in the next revision. In other words, Pr(deleted | fresh) = 0.0387. We then repeated the measurement for text that is both fresh, and is due to a low-reputation author: 6.36 % of it was deleted in the next revision, or Pr(deleted | fresh and low-reputation) = 0.0636. This indicates that author reputation is a useful factor in predicting the survival probability of fresh text, if not directly its trustworthiness. Indeed, as remarked above, since text can be deleted for a number of reasons aside from bad quality, author reputation is most likely a better indicator of trustworthiness than these figures indicate.

5.3 Conclusions

After comparing edit and text longevity values with user quality ratings for revisions, we believe that the largest residual source of error in our content-driven reputation lies in the fact that our text analysis does not include specific knowledge of the Wikipedia markup language and Wikipedia conventions. We plan to make the text analysis more precise in future work.

Acknowledgements. We thank Marco Faella for helping organize the manual rating of revisions to the Italian Wikipedia, and we thank the anonymous reviewers for their insightful and helpful feedback.

6. REFERENCES

[1] R. Burns and D. Long. A linear time, constant space differencing algorithm. In *Performance, Computing, and Communication Conference (IPCCC)*, pages 429–436. IEEE International, 1997.

[2] T. Cover and J. Thomas. *Elements of Information Theory*. J. Wiley & Sons, 1991.

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

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

[5] C. Dellarocas. The digitization of word-of-mouth: Promises and challenges of online reputation systems. *Management Science*, October 2003.

[6] J. Golbeck. *Computing and Applying Trust in Web-Based Social Networks*. PhD thesis, University of Maryland, 2005.

[7] R. Guha, R. Kumar, P. Raghavan, and A. Tomkins. Propagation of trust and distrust. In *Proc. of the 13th Intl. Conf. on World Wide Web*, pages 403–412. ACM Press, 2004.

[8] J. Hunt and M. McIlroy. An algorithm for differential file comparison. Computer Science Technical Report 41, Bell Laboratories, 1975.

[9] S. Kamvar, M. Schlosser, and H. Garcia-Molina. The eigentrust algorithm for reputation management in p2p networks. In *Proc. of the 12th Intl. Conf. on World Wide Web*, pages 640–651. ACM Press, 2003.

[10] J. Kleinberg. Authoritative sources in a hyperlinked environment. *J. ACM*, 46(5):604–632, 1999.

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

[12] D. Lucking-Reiley, D. Bryan, N. Prasad, and D. Reeves. Pennies from Ebay: The determinants of price in online auctions. Working paper, Vanderbilt University, 1999.

[13] D. McGuinness, H. Zeng, P. da 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.

[14] E. Myers. An o(ND) difference algorithm and its variations. *Algorithmica*, 1(2):251–266, 1986.

[15] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. Technical report, Stanford Digital Library Technologies Project, 1998.

[16] P. Resnick, R. Zeckhauser, E. Friedman, and K. Kiwabara. Reputation systems. *Comm. ACM*, 43(12):45–48, 2000.

[17] W. Tichy. The string-to-string correction problem with block move. *ACM Transactions on Computer Systems*, 2(4), 1984.

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

[19] H. Zeng, M. Alhoussaini, L. Ding, R. Fikes, and D. McGuinness. Computing trust from revision history. In *Intl. Conf. on Privacy, Security and Trust*, 2006.