7-2010

Spatio-Temporal Analysis of Wikipedia Metadata and the STiki Anti-Vandalism Tool

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

Sampath Kannan
University of Pennsylvania, kannan@cis.upenn.edu

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

Follow this and additional works at: http://repository.upenn.edu/cis_papers

Part of the Computer Sciences Commons

Recommended Citation

6th International Symposium on Wikis and Open Collaboration (WikiSym ’10), Gdańsk, Poland, July 7-9, 2010.

This paper is posted at ScholarlyCommons. http://repository.upenn.edu/cis_papers/454
For more information, please contact repository@pobox.upenn.edu.
Spatio-Temporal Analysis of Wikipedia Metadata and the STiki Anti-Vandalism Tool

Abstract
The bulk of Wikipedia anti-vandalism tools require natural language processing over the article or diff text. However, our prior work demonstrated the feasibility of using spatio-temporal properties to locate malicious edits. STiki is a real-time, on-Wikipedia tool leveraging this technique.

The associated poster reviews STiki’s methodology and performance. We find competing anti-vandalism tools inhibit maximal performance. However, the tool proves particularly adept at mitigating long-term embedded vandalism. Further, its robust and language-independent nature make it well-suited for use in less-patrolled Wiki installations.

Keywords
Wikipedia, collaborative applications, information security, intelligent routing, spatio-temporal processing

Disciplines
Computer Sciences | Physical Sciences and Mathematics

Comments
6th International Symposium on Wikis and Open Collaboration (WikiSym ’10), Gdańsk, Poland, July 7-9, 2010.

This other is available at ScholarlyCommons: http://repository.upenn.edu/cis_papers/454
Spatio-Temporal Analysis of Wikipedia Metadata and the STiki Anti-Vandalism Tool* †

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

Sampath Kannan
University of Pennsylvania
Philadelphia, PA, USA
kannan@cis.upenn.edu

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

ABSTRACT

The bulk of Wikipedia anti-vandalism tools require natural language processing over the article or diff text. However, our prior work demonstrated the feasibility of using spatio-temporal properties to locate malicious edits. STiki is a real-time, on-Wikipedia tool leveraging this technique.

The associated poster reviews STiki's methodology and performance. We find competing anti-vandalism tools inhibit maximal performance. However, the tool proves particularly adept at mitigating long-term embedded vandalism. Further, its robust and language-independent nature make it well-suited for use in less-patrolled wiki installations.

Categories and Subject Descriptors

H.5.3 [Group and Organization Interfaces]: collaborative computing, computer-supported cooperative work;
K.6.5 [Management of Computing and Information Systems]: Security and Protection

General Terms

Design, Management, Human Factors, Security

1. SPATIO-TEMPORAL DETECTION

We informally define Wikipedia vandalism to be any revision that is non-value adding, offensive, or destructive in its removal of content. Detecting vandalism is difficult; it has many varied and subtle forms.

To this end, our prior research [3] investigated the spatio-temporal properties of metadata as a means of vandalism detection. The metadata of an edit includes: the (1) timestamp of the edit, (2) article being edited, (3) user-name or IP of the editor, and (4) the revision comment. Meanwhile, temporal properties are a function of the time at which an event occurs and spatial properties are appropriate wherever a distance or membership function can be defined.

Our prior work [3] identifies ten spatio-temporal properties (see Tab. 1) that are effective in locating malicious edits. Simple features include the edit time-of-day, revision comment length, etc. Aggregate features combine time-decayed behavioral observations (feedback) to create reputations [2] for single entities and spatial groupings thereof.

2. THE STIKI TOOL

STiki [1] exploits the aforementioned logic. It consists of:

- Server-Side Engine: Listens on IRC for a Wikipedia edit, fetches metadata, and compiles the feature-set. Machine learning assigns a real-value vandalism score, which is the priority for insertion into the edit queue.
- Client-Side GUI: Pops the edit queue, presenting likely vandalism to users, via colored edit diffs (see Fig. 1).

An edit is also de-queued if a newer one is made on the same article. A STiki workflow diagram is given in Fig. 2. Both the GUI executable and source are available at [1].

3. STIKI PERFORMANCE

STiki has been used to revert over 2k instances of vandalism, yet the hit-rate (the percentage of time vandalism is displayed) has failed to meet off-line expectations [3]. Consider that the median active duration (time in queue) of the 10k most poorly scoring edits is around 3 minutes: The many autonomous anti-vandalism tools/bots prevent STiki from displaying much of the vandalism it finds. While STiki's hit-rate is ≈10%, analysis has shown it would be 50%+ (to a reasonable depth) if competing tools were not present.

Thus, STiki and its language-independence may be well suited for less-patrolled settings (e.g., foreign language editions of Wikipedia or corporate Wiki's). Even so, STiki has proven capable of finding embedded vandalism on English Wikipedia – that which escapes initial detection. The median age of an edit reverted by STiki is approximately 4.25 hours, nearly 200× that of conventional reversions.

4. EXTENSION & FUTURE WORK

To remedy the modest hit-rate, extension of the spatio-temporal feature-set is planned. With the inclusion of lightweight natural-language features, STiki could also evolve into a general-purpose anti-vandalism tool. The STiki framework will provide a convenient test-bed for these new features and other future vandalism mitigation strategies.
References


Figure 1: STiki GUI displaying a revision exhibiting vandalism (nonsense).

Figure 2: Simplified STiki workflow diagram.

Table 1: STiki features [3].

<table>
<thead>
<tr>
<th>#</th>
<th>FEATURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Edit time-of-day</td>
</tr>
<tr>
<td>2</td>
<td>Edit day-of-week</td>
</tr>
<tr>
<td>3</td>
<td>Time-since (TS) editor registration (first-edit)</td>
</tr>
<tr>
<td>4</td>
<td>TS article last edited</td>
</tr>
<tr>
<td>5</td>
<td>TS editor last vandalized</td>
</tr>
<tr>
<td>6</td>
<td>Rev. comment length</td>
</tr>
<tr>
<td>7</td>
<td>Article reputation</td>
</tr>
<tr>
<td>8</td>
<td>Categorical reputation (grouping over articles)</td>
</tr>
<tr>
<td>9</td>
<td>Editor reputation</td>
</tr>
<tr>
<td>10</td>
<td>Geographical reputation (grouping over editors)</td>
</tr>
</tbody>
</table>
Spatio-Temporal Analysis of Wikipedia Metadata and the STiki Anti-Vandalism Tool

Andrew G. West • Sampath Kannan • Insup Lee

Department of Computer and Information Science -- University of Pennsylvania -- Philadelphia, PA, USA

BIG IDEA

Spatio-temporal properties of edit metadata (editor, article, timestamp, and revision comment) can be leveraged to detect Wikipedia vandalism comparably to NLP based methods:

- Simple features (i.e., time-of-day), in addition to historical reputations for editors, articles, and spatial groupings thereof are used.
- Such features have language-independence, efficiency, and robustness not found in traditional detection mechanisms (i.e., NLP).
- STiki [1], is a real-time, on-Wikipedia tool utilizing the technique, already shown feasible off-line in our prior work [3].

EDIT LABELING: ROLLBACK

Need to label edits exhibiting vandalism (ex-post facto) to:
1. Show features effective (and eventually to train over them)
2. Form basis of historical reputations (vandalism = misbehavior)

ROLLBACK

An administrative form of [undo]:
- Revisions undone are Offending Edits (OEs), likely vandalism
- Autonomously parse-able
- Vandalism defined case-by-case

AGGREGATE FEATURES (REPUTATIONS)

IDEA: Use entity-specific reputation; augment with spatial reputations [2], which will have more historical data.

<table>
<thead>
<tr>
<th>#</th>
<th>FEATURE</th>
</tr>
</thead>
<tbody>
<tr>
<td>7</td>
<td>Article reputation</td>
</tr>
<tr>
<td>8</td>
<td>Category reputation</td>
</tr>
<tr>
<td>9</td>
<td>Editor reputation</td>
</tr>
<tr>
<td>10</td>
<td>Country reputation</td>
</tr>
</tbody>
</table>

STiki PERFORMANCE & FUTURE

Performance metric: hit-rate (% of displayed edits that are vandalism):
- Offline-analysis [3] shows hit-rate should be 50%+
- In fact, >10% due to competing tools/bots (often autonomous)

Successes and alternative uses:
- STiki has reverted over 5000 instances of vandalism on en-wiki
- Combats embedded vandalism well. Median age of vandalism reverted by STiki is 4.25 hours, nearly 200× of conventional reverts.
- May be best suited for less-patrolled Wikis (e.g. foreign lang. eds.)

Future improvements:
- Include lightweight NLP features (a general-purpose tool)
- Alternative detection (link spam? more ST-features?)

REFERENCES & ACKNOWLEDGEMENTS


This research was supported in part by ONR MURI N00014-07-1-0907. POC: Insup Lee, lee@cis.upenn.edu