Anti-Vandalism Research: The Year in Review

Andrew G. West

Wikimania `11 – August 5, 2011
BIG IDEA: Survey recent anti-vandalism progress (2010+)

- On-Wikipedia developments
  - Algorithms generating vandalism probabilities
  - Tools/frameworks applying those scores
- Academic developments
  - Standardizing evaluation
  - Collaboration between techniques
  - Cross-language evaluation
- Future techniques and applications
  - Pending changes, smarter watchlists
  - Envisioning improved frameworks
Survey Approach

- 50++ practical tools and academic writings for anti-vandalism (see [1,2]).
- Non-exhaustive, focus on the representative, popular, and recent
- English Wikipedia only; Zero-delay detection

VANDALISM: An edit that is:
- Non-value adding
- Offensive
- Destructive in removal
On-Wikipedia
Anti-Vandalism Algorithms:

0. Regexp/std-static-rules (pre-2010)
1. Content-driven reputation
2. Language statistics
3. Metadata analysis
Algs: Static Rules

Snippet of scoring list used by ClueBot

- “suck” => -5
- “stupid” => -3
- “haha” => -5
- ... 
- [A-Z][^a-z]{30,0} => -10
- !{5,} => -10
- ... 
- “[[.*]]” => +1
- “[[Category:.*]]” => +3

- **en.wiki:** Cluebot
- **3.5 years; 1.6 mil. edits** via ≈105 rules
- **es.wiki:** AVBot

**Standard pre-2010**
- Still popular outside en.wiki
- Technically simple

**Manually written rule sets**
- Author intuition
- Regular expressions over obscenities; lang-patterns

**Weaknesses**
- Not language-portable
- Easily obfuscated
- Time-consuming
- Inexact weighting
• Core intuition: Content that survives is good content
  • Good content accrues reputation for its author
  • Use author reputation to judge new edits
• Implemented as WikiTrust [3] on multiple Wikipedia’s
• Weakness: New editors have null reputation (i.e., Sybil attack)
Core intuition:

- Vocabularies differ between vandalism and innocent edits
- An automatic way to discover the obscenity word lists

ClueBotNG [4] (CBNG)

- Current autonomous guardian on en.wiki
- 250k edits in 6 months

Weaknesses: Rare words, need labeled corpus
Algs: Metadata

• Core intuition: Ignore actual text changes, and...
  • Use associated metadata (quantities, lengths, etc.).
  • Predictive model via machine-learning.
• Implemented in STiki [5]
• Subsets extremely common in other systems
• Weaknesses: Needs corpus

EDITOR
• registered?, account-age, geographical location, edit quantity, revert history, block history, is bot?, quantity of warnings on talk page

ARTICLE
• age, popularity, length, size change, revert history

REVISION COMMENT
• length, section-edit?

TIMESTAMP
• time-of-day, day-of-week

Example metadata features
On-Wikipedia
Applying Scores:

1. Autonomous (i.e., bot) reversion
2. Prioritizing human patrollers
Scores: Bots

- Advantages
  - Quick, no human latency
  - Always on, never tired
- Yet, ultimately incomplete
  - Conservative false-positive tolerances (0.5% for CBNG)
  - Plenty of borderline cases
  - One-revert rule
  - Purists: “non-democratic”
- Discarded scores have meaning that should be further utilized
Scores: Huggle [6]

[WP:Huggle]
Scores: Huggle [6]

Nice as a GUI tool, but lacks efficiency:

- Very simple queuing logic
  - Anonymous users first
  - Sort by # of talk page warnings
  - White-lists for trusted users
- Poor workload distribution
  - Everyone looking at same copy
  - Reverted edits “disappear”
  - Innocent edits re-inspected
- No server-side component
- Windows only
Scores: STiki [7]

ACTIVE QUEUES:
- STiki “metadata”
- WikiTrust
- CBNG (overflow)

......
API to include more

- Edit queue semantics
  - Enqueue: A PRIORITY queue ordered by vandalism scores
  - Dequeue: (1) classified by GUI, or (2) newer edit on page
- Thus, “innocent” edits are not re-inspected
- Edit reservation system avoids simultaneous work
- Server-side queue storage and written in Java; performance notes
Academic Progress

Note: STiki (metadata) and WikiTrust (content-reputation) are practical implemented systems of academic origin. ClueBot (bad word) + Cluebot-NG (Bayesian language) → Velasco [8]
Corpus Standardization

- Pre-2010, approaches developed independently
  - Everyone had their own evaluation technique
  - Or the corpora produced were trivially small/biased
  - Non-comparable results and claims
- Enter the Potthast corpus [9]
  - ≈32,000 labeled revisions from en.wiki
  - Labeling was outsourced, with robustness
  - Now a standard in the field

<table>
<thead>
<tr>
<th>RID:</th>
<th>7121</th>
<th>9752</th>
<th>4839</th>
<th>9582</th>
</tr>
</thead>
<tbody>
<tr>
<td>LABEL - RID</td>
<td>SPAM - 7121</td>
<td>HAM - 9752</td>
<td>HAM - 4839</td>
<td>SPAM - 9582</td>
</tr>
</tbody>
</table>

Amazon Mechanical Turk

Wikipedia The Free Encyclopedia

Artificial Artificial Intelligence
2010 Vandalism Detection Competition [10]
- 9 entries tested over Potthast corpus [9]
- Spanned all features/techniques
- Winning approach was language one [8]

That event and Wikimania 2010 allowed the authors of the three major techniques to meet, propose a big “meta-classifier” [11]. Goals:
- Improve the end-game performance
- Isolate overlap between techniques to help understand which individual features and feature subsets are driving performance
To give an idea of scale:

The combination of the three methods results in **70+ data points/features being given to the machine-learning framework**

Problem space is quite well-covered!
Meta-Algorithm (2)

- Combined approach dominated with PAN-2011 winning technique
  - Unique capture
  - Current baseline!
- High precision suggests bot-operation success
- Vocabulary data helpful when present; but “rare words” hurt
- Online implementation

Text = Shallow props. -- Language = Vocabularies
2011 Vandalism Detection Competition [12]

Two rule changes relative to 2010:
1. Train/test corpora span three natural languages (German, English, Spanish)
2. The ability to leverage ex post facto evidence (after the edit was made)

Notebook papers not published until September:
• However, results have been revealed
• Fortunate to explain the most successful approach [13]
• Strategy: More metadata, less language-specific features
  • Create a portable model applicable for 197+ editions
• Evaluation results:
  • Consistent feature strength
  • Language features prove moderately helpful when included
  • Why is English out-performed?
• Motivating example: Wikipedia 1.0 Project
• Use “future” evidence after edit was committed to help score.
  • E.g.: Has the content persisted? What is the next comment?
• WikiTrust system a specialist at this task (helps WP1.0)
• Surprisingly minimal performance increase (next slide)

<table>
<thead>
<tr>
<th>Time</th>
<th>User</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jan. 1</td>
<td>Jimbo</td>
<td>“Initializing article”</td>
</tr>
<tr>
<td>Feb. 6</td>
<td>111.37.<em>.</em></td>
<td>(null)</td>
</tr>
<tr>
<td>Jun. 5</td>
<td>west.andrew</td>
<td>“RV vand. by 111.37.<em>.</em>”</td>
</tr>
<tr>
<td>Jun. 5</td>
<td>NewishUser</td>
<td>“Add recent events”</td>
</tr>
<tr>
<td>Aug. 4</td>
<td>120.831.<em>.</em></td>
<td>“I is super vandal!”</td>
</tr>
</tbody>
</table>

What version should one pick?
Why is there not a greater performance increase?

Possibly subjective nature of vandalism?
• “... Active Learning and Statistical Language models” [14]
  • Concentrate on tagging types of vandalism.
  • Could use to create type-specific models
• “... the Banning of a Vandal” [15]
  • Formal look at warning process, Huggle, AIV, etc.
Future of Anti-Vandalism
Future: Pend. Changes

- Basically like taking the STiki queue, and moving top edits under PC
- Reduce [[WP:PC]] workload
  - 1/3 of all PC reviewed edits were innocent
  - Avoid [[WP:Bite]]
  - No one has to maintain “protected pages” lists
Future: Watchlists

Future: MW Support

Vandalism clearinghouse
- Bot/GUI collaboration
- Explicit and formal “innocent”
- Akin to new page patrol

WMF support for software
- Provide host space for AV algorithms (reliability!)
Future: Acute Subsets

External link spam ➔

“Dangerous content” ↓

Revision history of "Test Page"

- (cur) (prev) 02:06, 14 January 2011 WikiUser (Talk | contribs) (38 bytes) (Add details) (undo)
- (cur) (prev) 02:01, 14 January 2011 Andrew (Talk | contribs) (26 bytes) (Revert vandalism)
- (cur) (prev) 00:00, 14 January 2011 SuperVandal (Talk | contribs) (comment removed) [deleted]
- (cur) (prev) 23:59, 13 January 2011 76.99.208.144 (Talk) (26 bytes) (Minor grammatical fix) (undo)
- (cur) (prev) 23:59, 13 January 2011 Andrew (Talk | contribs) (24 bytes) (Creating initial content)


### Table 4

Kullback-Leibler divergence (i.e., information-gain) ranking for English features. Ex post facto signals are indicated by “(F)” (but ranking is independent, so a zero-delay list would have the same ordering). Foreign language features are not included for brevity.
Table 6. Area-under-curve (AUC) measurements for feature sets over training data. This is done for precision-recall (PR) and receiver-operating characteristic (ROC) curves. Feature sets include a control classifier (random, RND), zero-delay (ZD), and including ex post facto data (ALL).

<table>
<thead>
<tr>
<th>METRIC</th>
<th>GERMAN</th>
<th></th>
<th></th>
<th>ENGLISH</th>
<th></th>
<th></th>
<th>SPANISH</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>RND</td>
<td>ZD</td>
<td>ALL</td>
<td>RND</td>
<td>ZD</td>
<td>ALL</td>
<td>RND</td>
<td>ZD</td>
</tr>
<tr>
<td>PR-AUC</td>
<td>0.302</td>
<td>0.878</td>
<td>0.930</td>
<td>0.074</td>
<td>0.773</td>
<td>0.801</td>
<td>0.310</td>
<td>0.868</td>
</tr>
<tr>
<td>ROC-AUC</td>
<td>0.500</td>
<td>0.958</td>
<td>0.981</td>
<td>0.500</td>
<td>0.963</td>
<td>0.968</td>
<td>0.500</td>
<td>0.946</td>
</tr>
</tbody>
</table>

Table 7. Measuring the impact of language-specific features (Tab. 3). Feature sets are evaluated with (W) and without (WO) the inclusion of language-specific signals. Otherwise, acronyms are as defined as in Tab. 6. PR-AUC is the singular metric used in this comparison.

<table>
<thead>
<tr>
<th>LANG</th>
<th>ZD-WO</th>
<th>ZD-W</th>
<th>DIFF%</th>
<th>ALL-WO</th>
<th>ALL-W</th>
<th>DIFF%</th>
</tr>
</thead>
<tbody>
<tr>
<td>(PR-AUC) DE</td>
<td>0.881</td>
<td>0.878</td>
<td>-0.34%</td>
<td>0.930</td>
<td>0.930</td>
<td>±0.00%</td>
</tr>
<tr>
<td>(PR-AUC) EN</td>
<td>0.737</td>
<td>0.773</td>
<td>+4.89%</td>
<td>0.776</td>
<td>0.801</td>
<td>+3.22%</td>
</tr>
<tr>
<td>(PR-AUC) ES</td>
<td>0.805</td>
<td>0.868</td>
<td>+7.83%</td>
<td>0.988</td>
<td>0.986</td>
<td>-0.20%</td>
</tr>
</tbody>
</table>