An Empirical Research: "Wikipedia Vandalism Detection using VandalSense 2.0"

Notebook for PAN at CLEF 2011

F. Gediz Aksit

Maastricht University
g.aksit@student.maastrichtuniversity.nl

Abstract. Wikipedia despite having a very small budget has been among the top ten most visited websites for over half a decade. Being this visible also generated the problem of ill intended people modifying Wikipedia in a destructive manner. VandalSense is an experimental tool programmed by F. Gediz Aksit to automatically identify vandalism on Wikipedia through the use of machine learning and text mining as well as the use of years of personal experience. VandalSense is not intended to replace traditional recent changes patrolling and instead it is intended to be a tool to compliment it.

1. Introduction

Wikipedia, the free encyclopedia has been under the constant attack of vandalism. Vandalism is generally quickly removed, "but in one particularly well-publicized incident, false information was introduced into the biography of American political figure John Seigenthaler and remained undetected for four months."[1]

Furthermore Wikipedia is no longer just an online encyclopedia that one can simply check the previous revision on vandalized pages. WikiReader[2], Wikibrowse[3] for OLPC (One Laptop Per Child), WikiMiner[4] are among the many offline uses of the encyclopedia with no access to the history page. Furthermore vandalism in offline editions may remain unchecked for years in these offline devices which in some cases are the only means of information where internet or even books are unavailable. This only elevates the importance of the elimination of Vandalism from Wikipedia.

2. Approach

My approach was based on the initial analysis of the problem itself. Wikipedia receives millions of edits every day and these edits can be classified initially as edits from logged in users (Accounts) and edits from anonymous users (IPs) which will be referenced respectively in the rest of this document. German (de) and Spanish (es) editions of Wikipedia were analyzed using the PAN 2011 corpus for both these language editions. The two language editions of Wikipedia will be referenced by their respective two letter shortcut for the rest of this document. The

distinction between the types of contributors is visible in the test and training corpus as well.

Edit behavior between account and IP type contributors differs considerably. By very nature accounts are intended to have a more established contribution history while IP edits are intended to be quick edits often months apart. In the light of this distinction it is possible to distinguish edits classified as vandalism on the training corpus based on their account types before the actual training phase.

As visible from the pie charts training corpus for both language editions have about 30% of the revisions that are vandalism. As also clearly visible of the pie charts vast majority of the vandalism comes from IP edits and only a minority comes from logged in users. De wiki training corpus has about 71% (270 out of 379) of the total IP edits that are vandalism. Same corpus also has about 4% (24 out of 596) of the total account edits that are vandalism. Es wiki training corpus has about 56% (292 out of 521) of the total IP edits that are vandalism. Same corpus also has about 3% (15 out of 462) of the total account edits that are vandalism. As a result as visible in the pie charts the throwaway IP accounts are more prone to vandalism than logged in users. Because edit behaviors of accounts and IPs are very different and because the small potential of training from so few edits by accounts edits by accounts were ignored completely.

Based on the statistics from the training corpus, de wiki is predicted to have about 3,400 out of 84,114 revisions by accounts while 19,800 out of 27,840 revisions by IPs are predicted to be vandalism. Es wiki on the other hand is predicted to have about 500 out of 15,577 revisions by accounts while 10,313 out of 18,417 revisions by IPs are to be vandalism.

3. Features

The creative nature of vandalism makes its identification more than tricky. Keywords such as vulgar words and keywords such as Nazi and Jew are common in vandalism edits however same words – even vulgar words – are welcome on their respective articles.

Extracting features had proven to be a challenge as each case of vandalism observed differs from each other significantly. A diff algorithm by M. Hertel[5] was used to compare revisions. The diff result is then stripped of punctuation and then it is stemmed using snowball libraries.[6] Wiki markup was not stripped as the markup itself is used differently by vandals where vandals typically use the wiki markup in a less familiar and sloppy way. This project intends to detect three types of vandalism.

- **Blanking:** This term is used to define edits that result in the mass removal of content. Edits of this nature typically removes one or more paragraphs. While there are legitimate reasons for such mass removal information such as the removal of copyrighted material, such edits are almost always conducted by accounts rather than IPs.
- **Gibberish:** This term is used to define edits that result in the mass inclusion of content. Edits of this nature typically includes one or more paragraphs. While

there may be legitimate reasons for such mass inclusion of information such as copy paste of material from freely licensed sources, such edits are almost always conducted by accounts (more so flagged bots) rather than IPs.

- **Sneaky:** This term is used to define edits that result in the addition of a small amount of content that is intended to change the meaning of a few sentences or add a shot well structured messages to avoid detection. Such an edit was used with Seigenthaler biography controversy mentioned in the introduction section. Sneaky vandalism revisions typically contain similar keywords which include but not limited to profanity.

4. Classification

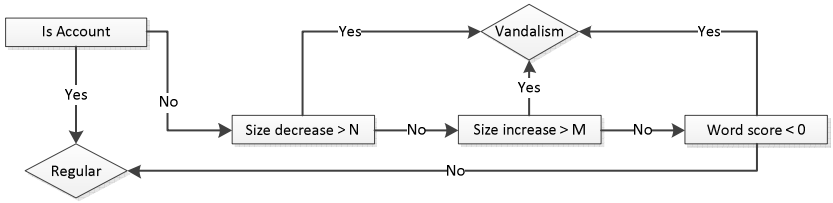

Figure 1: Decision tree structure

User type as well as the three features discussed above were observed and were used for classification purposes. A decision tree structure was implemented with an order that is intended to identify the more obvious kind of vandalism first without involving an unnecessary and time consuming word analysis reducing resource usage.

$$Entropy(S) = \sum -p(I)\log_2 P(I)$$

$$Gain\ (S, A) = Entropy(S) - \sum \left((|S_v|/|S|) * Entropy(S_v) \right)$$

Equation 1: Entropy and Entropy gain calculation

All of the three classifiers based on edit patterns require threshold values to operate. These values are determined through the use of statistical entropy and entropy gain which is also a key feature of the ID3 algorithm.[7] The tree was generated by hand due to the unstructured nature of the data as well as the minimal availability of meta data. Features are particularly hard to find as vandalism has far too many flavors with far too many different patterns. The same patterns can also be observed with regular edits.

Vandalism detection is achieved by initial count of the regular and vandalism revisions. These numbers are then compared to the revisions to threshold values for the latter three classifications. Byte values are looped while seeking a higher entropy gain. This strategy however has a flaw. Because the majority of the revisions are regular edits, entropy gain converges towards misidentifying vandalism revisions as regular edits. To circumvent this entropy gain is capped at a reasonable amount. The value of .6 was observed to be the more reasonable amount for entropy gain.

- **User Type:** User type classifier identifies if an edit is by an IP or an account. Account edits are flagged as regular edits for the reasons discussed in the approach section.
- **Deletion:** Deletion classifier calculates the deletion amount based on how many bytes of information was removed even if the information is replaced by some other content. The threshold value *N* is calculated for this classifier.
- **Addition:** Addition classifier conversely calculates the inclusion amount based on how many bytes of information was included even if the information replaces some other existing content. The threshold value *M* is calculated for this classifier.
- **Word Score:** This classifier calculates word score based on the frequency of the words appearance in regular and vandalism revisions. The effectiveness of word score classifier is limited as the training set is not a fair representation of the entire respective languages.

$$Word\ score = \left(Good\ score\ \frac{x}{1000}\right) - \left(Bad\ score\ \frac{1000-x}{1000}\right)$$

Equation 2: Word score calculation

Although the limited size of the training set makes it difficult to have a reasonable understanding of the language processed, vandalism identification through statistical frequency of words that appear in vandalism and regular edits was attempted. This produced almost random results. It was observed that words common in vandalism revisions can also be common in regular edits. For instance stop words are as expected common in both vandalism and regular edits. Instead of stripping stop words and spending time to manually or algorithmically identify words common on both types of edits, entropy gain was employed to weight the good and bad word scores for vandalism identification. A weight value was used to shift the weight towards good or bad edit score depending on the targeted entropy gain value. For word score, each words positive score (its frequency in good revisions) and bad score (its frequency in bad revisions) are individually calculated, weighted and then the weighted bad score is subtracted from the good score. If the concluding scores computation is negative in value, that revision is considered to be vandalism. This approach eliminates problems stemming from stop words as well.

5. Visualization

An ASP.Net web application was developed on top of the algorithms used to generate the submission to PAN. Intention behind this is to expand the project for human use to expand the training set based on human submission. The web application does not have a submission feature implemented yet. Visualization aspect of the project is essential as the intended use of the entire project is to assist recent changes patrol by weighting revisions instead of making edits to the wiki directly.

The web application end of the project is also intended to better analyze the inner workings of the project particularly that of the word score calculations. The web application is also capable of analyzing the live recent changes processing up to last

500 revisions (restriction due to Wikipedia's API limit). This web application can be accessed through http://eva.no-ip.biz/VandalSenseWeb/ and will be maintained as long as resources allow it. The three fields (upper diff, lower diff, word point) are actually the ID3 gain values. These fields exist to allow fine tuning. The change field displays the threshold values including the individual values for words before they get weighted.

A word cloud representation was implemented to show the frequency of words in the added revision. The colors represent if the word is positive (green), negative (red) or neutral/unknown (grey). The color itself isn't weighted. An interesting feature of the web application is the inclusion of Google Earth API.[8] The API allows the search of vandalism revisions which may help identify edits from unrelated IP ranges that are geographically nearby which could in return be used to identify more vandalism patterns expanding the training set.

6. Conclusions

The approach taken intended to avoid false positives as much as possible. Only a minority of the edits to Wikipedia is vandalism and false positives would only contribute to the problem. Both runs on both wikis (de and es) had a recall of 25% with a precision of about 60% for de wiki and 75% for es wiki. Based on the test set and training set statistics as discussed in the approach section es wiki receives a greater percentage of IP edits than of account edits in comparison to de wiki. The increase in accuracy of Spanish Wikipedia is probably due to training set of es wiki having a greater number of revisions for IP edits than of the training set of de wiki.

7. References

[1] http://www.usatoday.com/news/opinion/editorials/2005-11-29-wikipedia-edit_x.htm, [Online accessed: 23 June 2011]

[2] http://www.thewikireader.com/, [Online accessed: 23 June 2011]

[3] http://wiki.laptop.org/go/Wikibrowse, [Online accessed: 23 June 2011]

[4] http://meta.wikimedia.org/w/index.php?title=WikiMiner&oldid=503664, [Online accessed: 23 June 2011]

[5] http://www.mathertel.de/Diff/Default.aspx, [Online accessed: 23 June 2011]

[6] http://snowball.tartarus.org/, [Online accessed: 23 June 2011]

[7] http://www.cise.ufl.edu/~ddd/cap6635/Fall-97/Short-papers/2.htm, [Online accessed: 23 June 2011]

[8] http://code.google.com/apis/earth/, [Online accessed: 23 June 2011]