The Role of Probabilistic Grammar in Wikipedia Vandalism Detection

Natalia Timakova

timakova@berkeley.edu

Abstract

Vandalism detection is a long-standing problem for Wikipedia. The Objective Revision Evaluation Service (ORES) team at Wikimedia Foundation has been working on the solution for several years. Right now, they reached 92.42% accuracy (ROC-AUC) for the classifier that sorts all new edits into damaging and not damaging categories, and 96.72% AUC score for the classifier that sorts edits into ones made in good faith and bad faith. In addition to the existing 81 features, I included six probabilistic context free grammar features, with no improvement for the ROC AUC score. As my analysis suggests, grammar features rank from 12th to 72th in the feature importance rating in both Damaging and Goodfaith models, which means they can play an important role in the classification process, but need more training data to become truly useful.

1. Introduction

The inspiration for this paper came from my internship at Wikimedia Foundation in the summer 2017. I was working with ORES team trying to improve the classifier that is currently used to detect vandalism in the first few minutes after it happens. Vandalism, as Wikipedia defines it, is “the act of editing the project in a malicious manner that is intentionally disruptive” and includes “the addition, removal, or other modification of the text or other material that is either humorous, nonsensical, a hoax, or that is an offensive, humiliating, or otherwise degrading nature” [7].

The variety of vandalism acts is stunning and presents an interesting classification problem. Some of the latest examples that attracted media attention were when White House Press Secretary Sean Spicer's Wikipedia page was vandalized and his picture replaced with that of Muhammad Saeed al-Sahhaf, Iraqi Foreign and Information Minister under Saddam Hussein; Dana J. Botne's page description was edited to read that he was “the newest sock puppet for the Trump Administration”, and Paul Ryan's picture was added to a list of invertebrates, with the edit summary stating that he was added due to his lack of a spine.

Other vandalism examples are less humorous and hard to label as made in good-faith or bad-faith even manually. However, Wikipedia’s definition of vandalism requires detecting not only the fact of damage but also a malicious intent. This made ORES team build two classifiers instead of one: one that distinguishes a damaging edit from non-damaging one and another one that determines bad-faith.

As of 2017, ORES used Gradient Boosting (GB) classifier and Random Forest (RF) classifier, complemented with meta-data features like whether a user is anonym, as well as language features like a list of obscene lexemes or difference between the number of dictionary words before and after the edit. Both Damaging and Goodfaith models ran with 10-fold cross-validation, and GB almost always won the voting.

In this work, I included six probabilistic context free grammar (PCFG) features into both ORES' models to see if these features are capable of improving the state-of-the-art classification accuracy.

2. Previous work

Since 2016, when J. Tramullas et.al. published their brief literature review on Wikipedia vandalism, only a few interesting and relevant papers on this topic came out. One of them [8] discusses some spatio-temporal characteristics of the edits that are most likely to be made vandals. The author, in particularly, discovered that vandalism is more likely on weekdays than on weekends, and is equally of characteristic for all seasons except Summer. “While most edits are made between 14 and 17 hours, the ratio of vandalism to all edits peaks much earlier at around 9 hours with two more peaks occurring at 13 hours and at 19 hours,” Johannes Kiesel et.al. report. According to the authors, this pattern suggests “that vandalism is connected to labor (working hours)” and may be a tactic to cope with stress or boredom. It is very likely, that most of the vandals during these peak hours are pupils and students.

Another recent paper that addresses Wikipedia vandalism problem via studying rather an editor, than an edit, is published by Shuhan Yuan et.al [1]. The authors developed “a multi-source long-short term memory network (M-LSTM) to model user behaviors by using a variety of user edit aspects as inputs, including the history of edit reversion information, edit page titles and categories.” The assumption here, obviously, is that a vandal account stays active long enough to distinguish its behavior from benign accounts. S. Yean's best accuracy was 93.93% (F1) which is state-of-the-art at that moment.

However, this result might be misleading. The researchers do not specify whether they use only registered accounts or anonymous ones as well. Given their experiment design, they must be including some anonymous accounts, which are represented as IPs. There might be one, two and more users under one IP editing Wikipedia at different points in time, and not all of them are vandals. Therefore, labeling a certain IP as being a vandal may be not productive. On the other hand, if they used only registered accounts for their analysis, it cannot be very practical too, because a significant number of vandals edit Wikipedia from anonymous accounts.

Another recent and interesting work (Diyi Yang et.al. [9]) discusses various edit intentions including vandalism. A taxonomy of edit intentions in Wikipedia revisions the authors developed (Table 1) is worth keeping in mind when generating features for anti-vandal classifier as they constitute most of the data we have and represent the background from which the vandalism is supposed to be distinguished.

Interestingly, the Cronbach alpha (α) among the 13 editors that were employed in this classification work was the highest when they were to identify vandalism and counter vandalism. Some other less ambiguous and most prevalent edit intentions are copy editing, elaboration, fact update and Wikificaiton. Those, apparently, will constitute the competing background I'll need to distinguish vandalism from and will be responsible for the major part of false positives.

Label	Description	α	% of all revisions
Clarification	Specify or explain an existing fact or meaning by example or discussion without adding new information	0.394	0.7
Copy Editing	Rephrase; improve grammar, spelling, tone, or punctuation	0.800	11.8
Disambiguation	Relink from a disambiguation page to a specific page	0.401	0.3
Elaboration	Extend/add substantive new content; insert a fact or new meaningful assertion	0.733	12.0
Fact Update	Update numbers, dates, scores, episodes, status, etc. based on newly available information	0.744	5.5
Point of View	Rewrite using encyclopedic, neutral tone; remove bias; apply due weight	0.629	0.3
Process	Start/continue a wiki process workflow such as tagging an article with cleanup, merge or deletion notices	0.786	4.4
Refactoring	Restructure the article; move and rewrite content, without changing the meaning of it	0.737	1.9
Simplification	Reduce the complexity or breadth of discussion; may remove information	0.528	1.6
Verification	Add/modify references/citations; remove unverified text	0.797	5.4
Wikification	Format text to meet style guidelines, e.g. add links or remove them where necessary	0.664	33.1
Vandalism	Deliberately attempt to damage the article	0.894	2.5
Counter Vandalism	Revert or otherwise remove vandalism	0.879	1.9
Corpus Size			4,977

Table 1: A taxonomy of edit intentions in Wikipedia revisions, Cronbach's α agreement and the distributions of edit intention among the corpus (the percentages do not sum up to 100% because one revision could belong to multiple categories). [9]

Most of the other related papers were published around 2010-2011 and inspired by the International Competitions on Wikipedia Vandalism Detection. Two of them report a successful use of features based on PCFG. In particularly, Harpalani [5] quotes Raghavan's [11] PCFG model, and then Bhosale [10] draws the same features from Harpalani and Raghavan, using the model described below:

(1) Using generic PCFG parser, tree-bank each training document in the target (Wikipedia vandalism) corpus;

(2) Using only vandalism examples in your tree-banked target corpus, train a new PSFG parser C_{vandal}. Similarly, train $C_{regular}$ on a non-vandalism examples of the corpus.

(3) For each test document in the corpus, compare the probability of the edit determined by C_{vandal} and $C_{regular}$, where the parser with the higher score determines the class of the edit.

As the Harpalani's test on naturally imbalanced data shows, a few PCFG features can add 1.3 pp to the ROC AUC of 91.6%. These features were: the difference in the maximum log-likelihood score, the difference in the mean log-likelihood score, the difference in the standard deviation of the mean log-likelihood score and the difference in the sum of the log-likelihood scores. As their feature analysis demonstrates, the difference in the maximum and the mean PCFG scores were the most important among the probabilistic grammar features.

In the Bhosale work, PCFG-features increased ROC AUC from 0.893 bag-of-words baseline to 0.903.

As the previous work shows, language features can make a difference in the 90-th percentile, but to get to this percentile, one needs to employ meta-features. Following Thomas Adler et.al [4], “language features only provide an additional 6% of performance over the combined efforts of language-independent features.” Harpalani et.al. [5] also found that features like the total number of author contributions and how long the author has been registered responsible for the biggest relative gain in accuracy score.

Most of the researchers used 2010 PAN Wikipedia vandalism corpus Potthast et al. (2010), which is a non-random sample labeled by mechanical turkers, not Wikipedians. I am using Wikipedia’s own manually tagged corpora resulted from their campaign they held in 2015 among their users and volunteers. And this is the major difference in my approach comparing to the previous research.

3. Method

In my research, I define an edit as the unit of analysis, as it is in the current revision scoring model at Wikimedia. For the English version of the model, ORES team uses the corpus of 20,000 tagged Wikipedia edits, informally called diffs.

Following the ORES framework, I classify edits as damaging/not damaging and made in good-faith or bad-faith. An edit classified both as damaging and bad-faith is considered vandalism, according to Wikipedia’s definition of the term.

The existing ORES classifier does not use PCFG features, so evaluating these features in the current Gradient Boosting model is beneficial for Wikipedia.

I added six new PCFG features: delta of min, max and mean sentence syntax scores calculated on two treebanks, each pretrained on vandalism and featured (i.e., high-quality) articles, respectively.

As Step 1, I took two PCFG treebanks, or two sets of rules of context free grammar (CFG), trained by ORES team on the examples of Wikipedia vandalism (7000 articles) and Wikipedia featured articles (5000 articles, that are supposed to be free of vandalism). As Step 2, using SpaCy library, I parsed every diff (i.e., article text before and after the edit). For Step 3, I applied Kasami algorithm (which I upgraded from the old one built by ORES) to calculate the probability of each sentence-tree in a diff (that is, in the “before” and “after” revisions) to belong to the vandal and to the valid treebanks. As a result, with respect to vandal and valid CFGs, every diff yielded four sets of probabilities: two representing a diff before the edit (*revision.parent.syntax_score_if_valid, revision.parent.syntax_score_if_vandal*) and another two representing the diff after the edit (*revision.syntax_score_if_valid, revision.syntax_score_if_vandal*). Finally, I calculated the deltas in min, mean and max scores of the sets, subtracting the diff before the edit from the diff after, separately for vandal and valid probabilities. Here is the example calculation for a diff containing just one sentence:

Diff before the edit:

```
>>> sents = ['A vehicle wireless charging standard by Qualcomm.']
>>> sentences2syntax_scores_if_valid(sents)
[-0.3010299956639812]
sentences2syntax_scores_if_vandal(sents)
[-0.3010299956639812]
```

Diff after the vandal edit:

```
>>> sents = ['A vehicle wireless is boss well i
love line isloses  anent more']
>>> sentences2syntax_scores_if_valid(sents)
[-1.3766937125395522]
sentences2syntax_scores_if_vandal(sents)
[-1.2524500016406703]
```

Features:
-1.37-(-0.3-) = **1.07 (valid)** and -1.25-(-0.3) = **0.95 (vandal)**

The number **[-0.3010299956639812]** suggests that Kasami failed to attribute this texts either to vandal or to valid treebank. However, the feature extractor will use this number nonetheless. To compare the resulted features to the output of a non-vandal, valid example, I parsed another short diff:

```
>>> sents = ['"Mark Walter" is a founder and the chief executive officer of Guggenheim Partners, a privately held global financial services with more than $170 billion in assets under management and headquarters in Chicago and New York.']
>>> sentences2syntax_scores_if_valid(sents)
[-0.4980951372336739]
sentences2syntax_scores_if_vandal(sents)
[-0.45486469443971417]
```

```
>>> sents = ['"Mark Walter" is a founder and the chief executive officer of Guggenheim Partners, a privately held global financial services firm with more than $170 billion in assets under management and headquarters in Chicago and New York.']
>>> sentences2syntax_scores_if_valid(sents)
[-0.49129978752437414]
sentences2syntax_scores_if_vandal(sents)
[-0.44956004965434404]
```

Features: **0.0068 (valid); 0.0053 (vandal)**

These numbers are significantly smaller, but it can only represent the smaller difference in the texts before and after the edit (only one word inserted).

4. Results

As a result, I discovered that my new features did not improve the score of the current ORES classifier. Below, I am comparing the output of the baseline model and the one containing the six PCFG features.

Best performing classifier	**Parameters**	**ROC AUC (%)**

GradientBoosting	learning_rate=0.1,
	max_depth=7,
	n_estimators=700,
	max_features="log2"
		96.72

GradientBoosting	learning_rate=0.5,
	max_depth=7,
	n_estimators=500,
	max_features="log2"
		96.58

Table 2: Baseline performance for the Goodfaith model and the new model performance

Best performing classifier	**Parameters**	**ROC AUC (%)**

GradientBoosting	learning_rate=0.01,
	max_depth=5,
	n_estimators=700,
	max_features=log2
		92.42

GradientBoosting	learning_rate=0.01,
	max_depth=5,
	n_estimators=700,
	max_features="log2"
		92.5

Table 3: Baseline performance for the Damaging model and the new model performance

Both models had input of 19,442 observations; the data is very imbalanced, with pop-rate true/false as 0.97/0.03 in the Goodfaith model, and 0.03/0.97 in the Damaging.

metric	Goodfaith	Damaging
recall false	0.502	0.962
recall true	0.981	0.564
precision false	0.472	0.984
precision true	0.983	0.342

Table 4: Recall and precision for two classes of both models

As we can see from the Table 4, even with the new syntax features, both models still poorly predict minority class: bad-faith edits and damaging edits. These together are vandalism and our actual target.

5. Discussion: feature importance

In order to see whether the new features played an important role in the new model, I analyzed the feature importance for both models (both using GradientBoosting). The feature importance factors are readily available for any scikit-learn tree-based estimator and are indicative of the information gain obtained by selecting a specific feature as a node-splitting attribute.

Out of 87 features, the six PCFG deltas occupy places from 12 to 79 with the scores from 49.79 to 0.56 (on the 100-point scale).

For both (Goodfaith and Damaging) models, my feature importance analysis indicates that some of the new features, particularly **mean_syntax_score_if_[valid|vandal]_delta** are quite informative. Yet, these features yielded little to no score gains in my experiment. The explanation lies within the treebanks' sizes.

Since the diffs' syntax scores are defined with regard to a corpus/treebank, the bias of the estimator depends on the size of the treebank. For instance, for a sentence **s** with a syntax parse tree **T(s), syntax_score_if_vandal(s)** equals **log $P(T(s)|T_{vandal})$**. Unfortunately, the treebanks pre-trained by ORES are relatively small, likely leading to biased sentence probability estimates. As a result, the features, while deemed important, may be misleading to the classifier.

N	Feature name	Score
1	Log of seconds since user registration + 1	100.0
2	# markups per token in the text before the edit	30.43
3	whether user is anonymous	29.83
…		
15	**mean_syntax_score_if_vandal_delta**	**18.86**
…		
17	**mean_syntax_score_if_valid_delta**	**18.31**
…		
62	**min_syntax_score_if_vandal_delta**	**2.80**
…		
64	**min_syntax_score_if_valid_delta**	**2.66**
…		
74	**max_syntax_score_if_vandal_delta**	**0.85**
…		
76	**max_syntax_score_if_valid_delta**	**0.56**
…		
85	whether user is a curator	0.01
86	whether user has advanced rights	0.0007
87	whether page is a draftspace	0.00007

Table 5: Feature importance for Goodfaith model, GradientBoosting

Another observation is that **min_syntax_score_if_[valid|vandal]_delta** turns out to be not quite as sensitive to grammatically invalid edits as I expected. This indicates that the prior valid treebank already has plenty of grammatically invalid or incomplete sentences - e.g., section titles, image captions, unfiltered metadata, etc. - suggesting the need for more extensive pre-filtering of sentences of interest.

N	Feature name	Score
1	Log of seconds since user registration + 1	100.0
2	Log of text length before the edit + 1	71.35
3	# of uppercase words per word in the text before the edit	69.65
…		
12	**mean_syntax_score_if_vandal_delta**	**49.79**
…		
16	**mean_syntax_score_if_valid_delta**	**47.13**
…		
58	**diff.min_syntax_score_if_vandal_delta**	**3.38**
…		
64	**min_syntax_score_if_valid_delta**	**2.59**
…		
70	**max_syntax_score_if_valid_delta**	**1.57**
…		
72	**max_syntax_score_if_vandal_delta**	**1.47**
…		
85	whether user is curator	0.0057
86	whether page is draftspace	0.0018
87	whether user has advanced rights	0.0008

Table 6: Feature importance for Damaging model, GradientBoosting

6. Future work

In this work, I tried to reproduce the results of Harplani et al.[5] using a better corpus and a higher baseline. Although I did not reach a comparable score gain, I also started with a higher baseline.

There is still hope that PCFG features do have some potential and can help earn a better ROC AUC score for the current ORES models. For this to happen, more training data of a higher quality needed to build the treebanks.

References

1. Shuhan Yuan, Panpan Zheng, Xintao Wu, and Yang Xiang. 2017. Wikipedia Vandal Early Detection: from User Behavior to User Embedding, *arXiv:1706.00887v1.*

2. Heindorf, S., Potthast, M., Stein, B., Engels, G.: Vandalism detection in wikidata. *In: CIKM (2016)*

3. Kumar, S., Spezzano, F., Subrahmanian, V.: Vews: A wikipedia vandal early warning system. *In: KDD (2015)*

4. B. Thomas Adler, Luca de Alfaro, Santiago M. Mola-Velasco, Paolo Rosso, Andrew G. West. Wikipedia Vandalism Detection: Combining Natural Language, Metadata, and Reputation Features, 2-2011, *University of Pennsylvania Scholarly Commons*

5. Manoj Harpalani, Michael Hart, Sandesh Singh, Rob Johnson, and Yejin Choi. Language of Vandalism: Improving Wikipedia Vandalism Detection via Stylometric Analysis, *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics:shortpapers, pages 83–88,* Portland, Oregon, June 19-24, 2011.

6. Stefan Heindorf, Martin Potthast, Benno Stein, Gregor Engels. Towards Vandalism Detection in Knowledge Bases: Corpus Construction and Analysis. *Proceedings of*

the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2015

7. https://en.wikipedia.org/wiki/Vandalism_on_Wikipedia

8. Johannes Kiesel, Martin Potthast, Matthias Hagen and Benno Stein. Spatio-temporal Analysis of Reverted Wikipedia Edits. *Association for the Advancement of Artificial Intelligence,* 2017

9. Diyi Yang, Aaron Halfaker, Robert Kraut, Eduard Hovy. Identifying Semantic Edit Intentions from Revisions in Wikipedia *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 1990–2000*, Copenhagen, Denmark, September 7–11, 2017. ACL

10. Shruti Bhosale, Heath Vinicombe, and Raymond J. Mooney. Detecting Promotional Content in Wikipedia. *In Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing (EMNLP 2013)*, pp. 1851--1857, Seattle, WA, October 2013.

11. Sindhu Raghavan, Adriana Kovashka, and Raymond Mooney. 2010. Authorship attribution using probabilistic context-free grammars. *In Proceedings of the ACL,* pages 38–42, Uppsala, Sweden, July. Association for Computational Linguistics.

Appendix: Qualitative analysis

Seeking to better understand the problem, I studied 100 diffs from the training sets (50 in Russian and 50 in English) labeled as damaging and good-faith or damaging and bad-faith in proportion 25:25. A few takeaways from this study regarding the potential features are:

1). Although anonyms are still prevalent among vandals, their proportion to all vandals in the English Wikipedia is, probably, 2/3. Quite often, vandals use registered accounts rather than reveal their IP addresses, especially in the English Wikipedia. So whether the user is anonymous not always gives a robust signal: not all vandals are anons and not all anons are vandals. In the English subset I reviewed, 18 damaging bad-faith edits were committed under an IP and 10 - under a registered account. (However, for Russian Wikipedia, this proportion is different: 20 IPs vs. 1 registered.) For the Enwiki, this means the anon feature needs to be complemented with the number of previous edits made by this IP account in some continual period.

2). One of the typical vandal behavioral patterns is making several edits to different places in one or two articles in a span of 5-10 minutes. A good-faith editor would take more time to figure out an edit, and also would try to make them all in one commit. If it's several commits, then a good-faith editor must be improving the same - his own - edit. A bad-faith editor would rather take several random shots to make the reversion of all his malicious edits harder. The feature derived from this observation might be a number of previous edits by the same editor to the same article minus a number of previous edits by the same editor to the same article in the same place.

3). Sometimes, vandals lie in the comment to their edit that it was a "minor" edit (there is a tag for that) or it was some sort of a necessary correction. Hence, the comment is not a reliable feature.

4). One popular vandal use of Wikipedia is a "shameless plug" when a user inserts some content irrelevant to the article but most likely related to a business the vandal promotes. I've seen at least once when such a content was copy-pasted into three different articles being irrelevant to all of them. In theory, PSFG features described above may handle it.

To read about this qualitative analysis in more detail refer to this Wikimedia page.