Ensuring the Integrity of Wikipedia: A Data Science Approach

Francesca Spezzano

Computer Science Department
Boise State University
francescaspezzano@boisestate.edu

Abstract. In this paper, we present our research on the problem of ensuring the integrity of Wikipedia, the world's biggest free encyclopedia. As anyone can edit Wikipedia, many malicious users take advantage of this situation to make edits that compromise pages' content quality. Specifically, we present DePP, the state-of-the-art tool that detects article pages to protect with an accuracy of 93% and we introduce our research on identifying spam users. We show that we are able to classify spammers from benign users with 80.8% of accuracy and 0.88 mean average precision.

1 Introduction

Wikipedia is the world's biggest free encyclopedia read by many users every day. Thanks to the mechanism by which anyone can edit, its content grows and is kept constantly updated. However, malicious users can take advantage of this open editing mechanism to seriously compromise the quality of Wikipedia articles.

The main form of content damaging is vandalism, defined by Wikipedia itself as "any addition, removal, or change of content, in a deliberate attempt to compromise the integrity of Wikipedia"[1]. Other forms of damaging edits are page spamming [2] and dissemination of false information, e.g. through hoax articles [11].

In this paper, we discuss two research efforts which have the common goal of ensuring the content integrity of Wikipedia.

First, we introduce DePP, the state-of-the-art tool detecting article pages to protect [12]. Page protection is a mechanism used by Wikipedia to place restrictions on the type of users that can make edits to prevent vandalism, libel, or edit wars [3]. Our DePP system achieves an accuracy of 93% and significantly improves over baselines.

Second, we present our work on spam users identification [8]. We formulate the problem as a binary classification task and propose a set of features based on user editing behavior to separate spam users from benign ones. Our results show that we reach 80.8% classification accuracy and 0.88 mean average precision and beat ORES, the most recent tool developed by Wikimedia to assign damaging scores to edits.

2 Related Work

Detecting damaging edits. Plenty of work has been done on detecting damaging edits, particularly vandalism (see [7] for a survey). Currently, ClueBot NG [4] and STiki [5] are the state-of-the-art tools used by Wikipedia to detect vandalism. ClueBot NG is a bot based on an artificial neural network which scores edits and reverts the worst-scoring edits. STiki is an intelligent routing tool which suggests potential vandalism to humans for definitive classification. It works by scoring edits on the basis of metadata and reverts, and computing a reputation score for each user.

Recently, Wikimedia Foundation launched a new machine learning-based service, called Objective Revision Evaluation Service (ORES) [6] which measures the level of general damage each edit causes. More specifically, given an edit, ORES provides three probabilities predicting (i) whether or not it causes damage, (ii) if it was saved in good-faith, and (iii) if the edit will eventually be reverted. These scores are available through the ORES public API [1].

Regarding spam edits detection specifically, previous work concentrated on the problem of predicting whether a link contained in an edit is spam or not, whereas, in this paper, we predict whether a user is a spammer or not by considering her edit history. [14] created the first Wikipedia link-spam corpus, identified Wikipedia's link spam vulnerabilities, and proposed mitigation strategies based on explicit edit approval, refinement of account privileges, and detecting potential spam edits through a machine learning framework. The latter strategy, described by the same authors in [13], relies on features based on (i) article metadata and link/URL properties, (ii) HTML landing site analysis, and (iii) third-party services used to discern spam landing sites. This tool was implemented as part of STiki and has been used on Wikipedia since 2011. Nowadays, this STiki component is inactive due to a monetary cost for third-party services.

An Early Warning System for Vandals. In our previous work [10], we addressed the problem of vandalism in Wikipedia from a different perspective. We studied for the *first* time the problem of early prediction of vandal users. The proposed system, called VEWS (Vandal Early Warning System) [2], leverages differences in the editing behavior of vandals vs. benign users and detect vandals with an accuracy of over 85% and outperforms both ClueBot NG and STiki. Moreover, as an early warning system, VEWS detects, on average, vandals 2.39 edits before ClueBot NG. The combination of VEWS and Cluebot NG results in a fully automated system that does not leverage any human input (e.g. edit reversion) and further increases the performances.

Page protection. When a page article is heavily vandalized, administrators may decide to protect the page by restricting its access. To the best of our knowledge, little research has been done on the topic of page protection in Wikipedia.

[1] `http://ores.wikimedia.org`

[2] Dataset and code are available at `http://www.cs.umd.edu/~vs/vews/`

Hill and Shaw [9] studied the impact of page protection on user patterns of editing. They also created a dataset (they admit it may not be complete) of protected pages to perform their analysis. There are not currently bots on Wikipedia that can search for pages that may need to be protected. WikiMedia does have a script [3] available in which administrative users can protect a set of pages all at once. However, this program requires that the user supply the pages, or the category of pages to be protected and is only intended for protecting a large group of pages at once. There are some bots on Wikipedia that can help with some of the wiki-work that goes along with protecting or removing page protection. This includes adding or removing a template to a page that is marked as protected or no longer marked as protected. These bots can automatically update templates if a page protection has expired.

3 Detecting Pages to Protect

The first problem we address consists in deciding whether or not a page should be protected by Wikipedia administrators. Page protection consists in placing restrictions on the type of users that can edit a Wikipedia page. Common motivations that an administrative user may have in protecting a page include (i) consistent vandalism or libel from one or more users, and (ii) avoiding edit wars [3]. There are different levels of page protection for which different levels of users can make edits (or, in general, perform actions on the page): fully protected pages can be edited (or moved) only by administrators, semi-protected pages can be edited only by autoconfirmed users, while move protection does not allow pages to be moved to a new title, except by an administrator. Page protections can also be set for different amounts of time, including 24 or 36 hours, or indefinitely.

Currently, English Wikipedia contains over five million pages. Only a small percentage of those pages are currently protected, less than 0.2 percent. However, around 17 pages become protected every day (according to the number of protected pages from May 6 through Aug 6, 2016). This ratio shows how it is difficult for administrative users to monitor over all Wikipedia pages to determine if any need to be protected. Users can request pages to be protected or unprotected but an administrative user would have to analyze the page to determine if it should be protected, what level of protection to give, and for how long the protection should last, if not indefinitely. All this work is currently *manually* done by administrators.

To overcome this problem, we propose `DePP`, the *first* automated tool to detect pages to protect in Wikipedia [12]. `DePP` is a machine learning-based tool that works with two novel set of features based on (i) users page revision behavior and (ii) page categories. More specifically, the first group of features includes the following six base features:

E1 *Total average time between revisions*;

[3] `https://www.mediawiki.org/wiki/Manual:Pywikibot/protect.py`

E2 *Total number of users making 5 or more revisions*;
E3 *Total average number of revisions per user*;
E4 *Total number of revisions by non-registered users*;
E5 *Total number of revisions made from mobile device*;
E6 *Total average size of revisions.*

In addition to the above base features, we also include an additional set of features taking into account the page editing pattern over the time. We define these features by leveraging the features E1-E6 as follows. For each page, we consider the edits made in the latest 10 weeks and we split this time interval into time frames of two weeks (last two weeks, second last two weeks, etc.). Then, we compute features E1 to E6 within each time frame. The idea of these features is to monitor features E1-E6 over time to see if some anomaly starts to happen at some point. For instance, if a page is new we may observe a lot of edits of larger size in a short time after the page is created as users are building the content of the page. Later when the content is stable, we may observe fewer edits of smaller size representing small changes in the page. On the other hand, if the content of the page was stable and suddenly we observe a lot of edits from many users, it may indicate the page topic became controversial and the page may need protection.

The second group of features use information about page categories and includes:

NC *Number of categories a page is marked under*;
PC *Probability of protecting the page given its categories*: given all the pages in the training set T and a page category c, we compute the probability $\texttt{pr}(c)$ that pages in category c are protected as the percentage of pages in T having category c that are protected. Then, given a page p having categories $c_1, \ldots, c_n$, we compute this feature as the probability that the page is in at least one category whose pages have a high probability to be protected as $PC(p) = 1 - \prod_{i=1}^{n}(1 - \texttt{pr}(c_i))$.

In addition to the above two features, we define another group of features that shows how much features E1-E6 vary for a page p w.r.t. the average of these values among all the pages in the same categories as p. Specifically, given the set of pages in the training set T, we computed the set C of the top-100 most frequent categories. Additionally, for each category $c \in C$, we averaged the features E1-E6 among all the pages (denoted by T_c) having category c in the training set. Then, for each page p we computed 600 features (6 times 100), one for each feature Ei ($1 \leq i \leq 6$) and for each category $c \in C$ as follows:

$$C(Ei, c) = \begin{cases} |Ei(p) - \texttt{avg}_{p' \in T_c}(Ei(p'))| & \text{if } p \text{ is in category } c \\ 0 & \text{otherwise} \end{cases}$$

where $Ei(p)$ is the value of the feature Ei for the page p. The aim of this group of features is to understand if a page is anomalous w.r.t. other pages in the same category.

	Accuracy
B1	55.964%
B2	65.617%
B3	73.361%
B1+B2+B3	78.089%
DePP	**93.237%**

Table 1. DePP accuracy results and comparison with baselines. Everything is computed with random forest.

All the features that we propose are language independent as they do not consider page content. As a consequence, DePP is general and able to work on any version of Wikipedia.

To test our DePP system we built a balanced dataset [4] containing all edit protected articles until to Apr. 7, 2016 (6,799 pages) and an almost equal number of randomly selected unprotected pages (6,824), for a total of 13.6K article pages, and up to the last 500 most recent revisions for each selected page. For protected pages, we only gathered the revisions up until the most recent protection. If there was more than one recent protection, we gathered the revision information between the two protections. This allowed us to focus on the revisions leading up to the most recent page protection. Revision information that we collected included the user who made the revision, the timestamp of the revision, the size of the revision, the categories of the page, and any comments, tags or flags associated with the revision.

The DePP accuracy in the prediction task on 10-fold cross validation is reported for random forest (the best performing algorithm as compared to Logistic Regression, SVM, and K-Nearest Neighbor) in Table 1. As we can see, DePP is able to classify pages to protect from pages that do not need protection with an accuracy of 93.237%. As no automated tool detecting which page to protect exists in Wikipedia, we defined some baselines to compare our results. One of the main reasons for protecting a page on Wikipedia is to stop edit wars, vandalism or libel from happening, or continuing to happen on a page. Thus, we used the following baselines: [B1] *Number of revisions tagged as "Possible libel or vandalism"*; [B2] *Number of revisions that Cluebot NG or STiki reverted as possible vandalism*; [B3] *Number of edit wars between two users in the page.*

As we can see in Table 1, DePP significantly beats each individual baseline and the combination of all the three.

4 Spam Users Identification

Another problem that compromises the content quality of Wikipedia articles is spamming. Wikipedia, like most forms of online social media, receives continuous spamming attempts every day. Since all non-protected pages are open for editing by any type of user, inevitably happens that malicious users have the opportunity to post spam messages into any open page. These messages remain

[4] Dataset available at `http://bit.ly/wiki_depp`

on the page until they are discovered and removed by another user. Specifically, Wikipedia recognizes three main types of spam, namely "advertisements masquerading as articles, external link spamming, and adding references with the aim of promoting the author or the work being referenced"[2].

Currently, no specific tool is available on Wikipedia to identify neither spam edits or spam users. Tools like Cluebot NG and STiki are tailored toward vandalism detection, while ORES is designed to detect damaging edits in general. As in the case of page protection, the majority of the work to protect Wikipedia from spammers is done *manually* by Wikipedia users (patrollers, watchlisters, and readers) who monitor recent changes in the encyclopedia and, eventually, report suspicious spam users to administrators for definitive account blocking.

To fight spammers on Wikipedia, we study the problem of identifying spam users from benign ones [8]. Our work is closer in spirit to [10] as the aim is to classify users by using their editing behavior instead of classifying a single edit as vandalism [4,5], spam [13] or generally damaging [6].

We propose a machine learning-based framework using a set of features which are based on research that has been done regarding typical behaviors exhibited by spammers: similarity in edit size and links used in revisions, similar time-sensitive behavior in edits, social involvement of a user in the community through contribution to Wikipedias talk page system, and chosen username. We did not consider any feature related to edit content so that our system would be language independent and capable of working for all Wikipedia versions. Moreover, we do not rely on third-party services, so there is no overhead cost as in [13].

The list of features we considered in our system are as follows:

User edit sizes based features : average size of edits, standard deviation of edit sizes, and variance significance (previous feature normalized by user average edit size).

Edit timing behavior based features : average and standard deviation of time between edits.

Links in edits based features : Unique link rating (the ratio of unique links posted by a user to the total number of links posted by the user) and link ratio in edits (number of edits that a user makes which contain links).

Talk page edit ratio : this is the ratio of talk pages edited by a user that correspond with the main article pages that a user edits.

Username based features : Zafarani and Liu [15] showed that aspects of users' usernames themselves contain information that is useful in detecting malicious users. Thus, in addition to the features based on users' edit behaviors, we also considered four additional username related features: number of digits in a username, ratio of digits in a username, number of leading digits in a username, and unique character ratio in a username.

To test our framework, we built a new dataset [5] containing 4.2K (half spammer and half benign) users and 75.6K edits as follows. We collected all Wikipedia

[5] Dataset available at `http://bit.ly/wiki_spammers`

	Our Features	ORES	Our Features + ORES
Accuracy	**80.8%**	69.7%	**82.1%**
MAP	**0.88**	0.695	**0.886**

Table 2. Spammers identification accuracy and Mean Average Precision (MAP) results in comparison with ORES. Everything is computed with XGBoost.

users (up to Nov. 17, 2016) who were blocked for spamming from two lists maintained on Wikipedia: "Wikipedians who are indefinitely blocked for spamming" [6] "Wikipedians who are indefinitely blocked for link spamming" [7]. The first list contains all spam users blocked before Mar 12, 2009, while the second one includes all link-spammers after Mar 12, 2009 to today. We gathered a total of 2,087 spam users (we only included users who did at least one edit) between the two lists considered.

In order to create a balanced dataset of spam/benign users, we randomly select a sample of benign Wikipedia users of roughly the same size as the spammer user set (2,119 users). To ensure these were genuine users, we cross-checked their usernames against the entire list of blocked users provided by Wikipedia [8]. This list contains all users in Wikipedia who have been blocked for any reason, spammers included. For each user in our dataset, we collected up to their 500 most recent edits. For each edit we gathered the following information: edit content, time-stamp, whether or not the edit is done on a Talk page, and the damaging score provided by ORES.

We run 10-fold cross validation on several machine learning algorithms, namely SVM, Logistic Regression, K-Nearest Neighbor, Random Forest, and XGBoost, to test the performances of our features. Experimental results are shown in Table 2 for the best performing algorithm (XGBoost). Here we can see that our system is able to classify spammers from benign users with 80.8% of accuracy and it is a valuable tool in suggesting potential spammers to Wikipedia administrators for further investigation as proved by a mean average precision of 0.88.

We compared our tool with ORES only, as the tool proposed in [13] is no longer used and Cluebot NG and STiki are designed specifically for vandalism and not spam. To compare our system with ORES, we considered the edit damaging score. More specifically, given a user and all her edits, we computed both the average and maximum damaging score provided by ORES and used these as features for classification. Results on 10-fold cross validation with XGBoost (the best performing classifier) are reported in Table 2, as well. As we can see, ORES performances are poor for the task of spammer detection (69.7% of accuracy

[6] `http://en.wikipedia.org/wiki/Category:Wikipedians_who_are_indefinitely_blocked_for_spamming`

[7] `http://en.wikipedia.org/wiki/Category:Wikipedians_who_are_indefinitely_blocked_for_link-spamming`

[8] `http://en.wikipedia.org/wiki/Special:BlockList`

and a mean average precision of 0.695). However, combining our features with ORES further increases the accuracy to 82.1%.

5 Conclusions

In this paper, we addressed the problem of ensuring the integrity of Wikipedia pages and presented our research on detecting pages to protect and identifying spam users. Our experimental results show that we are able to classify (i) article pages to protect with an accuracy of 93% and (ii) spammers from benign users with 80.8% of accuracy and 0.88 mean average precision.

Both the methods proposed do not look at edit content and, as a consequence, they are generally applicable to all versions of Wikipedia.

References

1. `http://en.wikipedia.org/wiki/Wikipedia:Vandalism`.
2. `http://en.wikipedia.org/wiki/Wikipedia:Spam`.
3. `http://en.wikipedia.org/wiki/Wikipedia:Editwarring`.
4. `http://en.wikipedia.org/wiki/User:ClueBot_NG`.
5. `http://en.wikipedia.org/wiki/Wikipedia:STiki`.
6. `http://meta.wikimedia.org/wiki/Objective_Revision_Evaluation_Service`.
7. B. Thomas Adler, Luca de Alfaro, Santiago Moisés Mola-Velasco, Paolo Rosso, and Andrew G. West. Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *Computational linguistics and intelligent text processing*, pages 277–288, 2011.
8. Thomas Green and Francesca Spezzano. Spam users identification in wikipedia via editing behavior. In *International AAAI Conference Web and Social Media*, pages 532–535, 2017.
9. Benjamin Mako Hill and Aaron D. Shaw. Page protection: another missing dimension of wikipedia research. In *International Symposium on Open Collaboration*, pages 15:1–15:4, 2015.
10. Srijan Kumar, Francesca Spezzano, and VS Subrahmanian. Vews: A wikipedia vandal early warning system. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 607–616, 2015.
11. Srijan Kumar, Robert West, and Jure Leskovec. Disinformation on the web: Impact, characteristics, and detection of wikipedia hoaxes. In *International World Wide Web Conference*, pages 591–602, 2016.
12. Kelsey Suyehira and Francesca Spezzano. Depp: A system for detecting pages to protect in wikipedia. In *International Conference on Information and Knowledge Management*, pages 2081–2084, 2016.
13. Andrew G West, Avantika Agrawal, Phillip Baker, Brittney Exline, and Insup Lee. Autonomous link spam detection in purely collaborative environments. In *International Symposium on Wikis and Open Collaboration*, pages 91–100, 2011.
14. Andrew G. West, Jian Chang, Krishna Venkatasubramanian, Oleg Sokolsky, and Insup Lee. Link spamming wikipedia for profit. In *Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, pages 152–161, 2011.
15. Reza Zafarani and Huan Liu. 10 bits of surprise: Detecting malicious users with minimum information. In *International Conference on Information and Knowledge Management*, pages 423–431, 2015.