Snooping Wikipedia Vandals with MapReduce

Michele Spina[1], Dario Rossi[1,2], Mauro Sozio[2], Silviu Maniu[2], Bogdan Cautis[2]
[1]LINCS, Paris, France – first.last@enst.fr
[2]Telecom ParisTech, Paris, France – first.last@telecom-paristech.fr

***Abstract*—In this paper, we present and validate an algorithm able to accurately identify anomalous behaviors on online and collaborative social networks, based on their interaction with other fellows. We focus on Wikipedia, where accurate ground truth for the classification of vandals can be reliably gathered by manual inspection of the page edit history. We develop a distributed crawler and classifier tasks, both implemented in MapReduce, with whom we are able to explore a very large dataset, consisting of over 5 millions articles collaboratively edited by 14 millions authors, resulting in over 8 billion pairwise interactions. We represent Wikipedia as a signed network, where positive arcs imply constructive interaction between editors. We then isolate a set of high reputation editors (i.e., nodes having many positive incoming links) and classify the remaining ones based on their interactions with high reputation editors. We demonstrate our approach not only to be practically relevant (due to the size of our dataset), but also feasible (as it requires few MapReduce iteration) and accurate (over 95% true positive rate). At the same time, we are able to classify only about half of the dataset editors (recall of 50%) for which we outline some solution under study.**

I. Introduction

Wikipedia is one of the most actively used online social networks, where content is collaborative edited and improved by a very large set of participants. While other popular social networks allow users to either share personal information (e.g., Facebook) or at least present information in a personalized way (e.g., Twitter), Wikipedia enforces a rigorous editorial process in order to ensure information to be as accurate and neutral as possible. Due to the openness of the process, Wikipedia is exposed to vandalism, defined in [1] as any addition, removal or change of content in a deliberate attempt to compromise its integrity. Some examples of vandalism include for instance: spam to promote external sites, addition of nonsense/injuries/provocative text or images, unjustified removal of legitimate text, deliberate addition of false information.

A number of tools such as automated bots (e.g.,Cluebot), filters (e.g.,abusefilter), and editing assistants (e.g., Huggle and Twinkle), assist in locating acts of vandalism. Yet, despite benefits of simplicity and precision, techniques based on pattern matching have very low recall, are difficult to maintain and tune, and are inherently limited across language barriers. As a result, research has focused on automated and statistical classification, that are generally based either on textual data or article metadata (see Sec. V). Fewer work instead exist that leverage, as we do in this work, much more extended information (e.g., interactions of a specific editor on the *full set* of his articles) than those generally scrutinized in each specific case under exam (i.e., metadata about the specific interaction).

In this paper, we propose and validate an algorithm for the classification of Wikipedia vandals based on their mutual interaction on the whole article corpus: due to the sheer size of Wikipedia, both our crawler and classifier are implemented on MapReduce. While we foresee that such tool can be used for the online detection of Wikipedia vandals, for the time being we assess the classification performance on a very large but static snapshot of the english Wikipedia website. Our MapReduce crawler is interesting per se, as it gathers a total of $261{\cdot}10^6$ revisions of $5{\cdot}10^6$ articles, resulting in $8{\cdot}10^9$ pairwise interactions between over $14{\cdot}10^6$ editors – several orders of magnitude larger than most Wikipedia vandal detection study.

By associating negative and positive weights to different types of interaction (e.g., insert, delete, revert, etc.), we build a signed graph among Wikipedia editors: reputation of nodes is then computed as sum of positive and negative arcs in the signed network. Our MapReduce classifier finds a subset of graph with nodes having highest reputation, that constitutes a set of reliable editors: we then infer vandal behavior when authors have consistently negative interactions with the reliable editor set. Overall, we classify about half of all editors, with 95% accuracy (resulting from a careful manual validation of random instances of our results) and discuss how iterative approach can be used to extend the classification to the remaining half (that we aim at pursuing as future research work).

II. Workflow

Our methodology is as follows: we (i) develop a distributed Wikipedia crawler in MapReduce, that we use to (ii) build a signed network of editor interactions, from which we (iii) extract a set of reliable editors with an iterative MapReduce filtering, upon which we (iv) classify the remaining editors as vandals depending their interaction with neighboring editors.

A. Crawler

The English Wikipedia contains around 27 million pages, each having an average of 19 revisions [2]. Obviously, this amount of data cannot be feasibly processed on a single machine, sequentially: as such, we develop a distributed crawler in MapReduce, which consists of 2 jobs. This choice has not only the advantage of distributing the crawling load among a set of 32 machines in our cluster, but can also exploit the

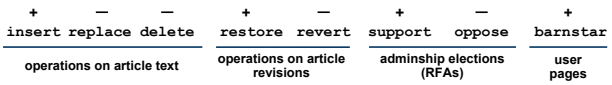

Fig. 1: The raw interaction vector

MapReduce framework for partial preprocessing of the raw data[1] gathered during crawling.

The first job in the MapReduce chain is the extraction of text interactions. Starting from an input consisting of a list of articles, each mapper is tasked with extracting the entire revision history of an article, via API calls. Then, interactions between contributor pairs are computed for each revision. The mapper then presents two inputs to the reducers: (i) a set of contributor pairs and their interaction on a revision, and (ii) a list of contributors detected by the mapper. One reducer is then tasked with the aggregation of the interactions for each user pair, while the second generates a list of unique contributors to be used in the next job of the chain.

The second job in the chain takes as input the contributor list, and then parses the HTTP sources of election and contributor profile. As in the case of the text interactions, the reducer aggregates, via summation, the interactions of unique contributor pairs.

B. Signed network

Interactions among wikipedia editors can be of two flavors: (i) community interactions or (ii) interactions on article content. For instance, community interactions can be retrieved from Request for Adminship elections (RFAs), where users can participate as candidates or as voters (votes can be positive or negative). Another, more infrequent type of interaction is represented by the exchange of barnstars, which are prizes that users can give to users with a significant level and quality of contributions (so they are always positive interactions).

As for (ii), user interaction either create a revision of an article by editing (adding, removing) text or by reverting the text to a previous version: interactions can be quantified assigning ownership at word level by analyzing text differences between two consecutive revisions on an article. As depicted in Fig. 1, we aggregate all interactions between a pair of editors in a single vector, where text *insertion (ins)* are interpreted as constructive (positive unity weight) while *replacement (rep)* and *deletion (del)* of text are seen as destructive (negative unity weight). Similarly, *restores (res)* of a revision are interpreted as positive interactions, while *reverts (rev)* of a revision are negative ones.

The kendall's $k = \frac{ins-(del+rep)}{ins+(del+rep)}$ coefficient is used to assess the sign of a set of textual interactions (specifically, $k \in [-1, 1]$ and positive interaction requires $k \geq 0.5$). Based on preliminary results (see Sec. III), we consider editor-pairs having at least 2 interactions, and we further require the total number of words implied in the interactions to be at least $\|del + rep + ins\|_1 \geq 10$). Then, scores are aggregated in a single sign by a majority voting, considering kendall k, restore/revert and community interaction of as individual bulletins. More details on this procedure are available in [17].

It could be argued that different actions could be given different weights: e.g., proportionally to the number of ins/del/rep words; or giving more weight to rev/res than to ins/del/rep, etc. However setting weights is not an easy task: as such, we opt for simplicity and defer a sensitivity analysis for future work.

C. Reliable editor set

Our vandal detection methodology relies on finding, within the full signed Wikipedia network W, a set of reliable editors R that can assist the judgment of the remaining editors. More formally, defining the reputation $rep(x)$ of an editor x as the sum of signs of all edges directed to x, the problem can be phrased as finding a subgraph $R \subset W$ whose nodes have the highest mutual average reputation.

This problem can be stated as finding the densest subgraph, where the density of a graph $G(V, E)$ is measured $\frac{2|E|}{|V|(|V|-1)}$. While the problem is NP-hard, [6] proposes a greedy approximation that is guaranteed to converge, for any $\epsilon > 0$, in $O(log_{1+\epsilon} n)$ passes yielding an approximation factor of $2(1 + \epsilon)$.

Shortly, starting from the Wikipedia graph $G_0 = W$, at each iteration i the algorithm computes the average reputation $E[rep(G_i)]$, and it removes from G_i all nodes whose degree is less than $(1 + \epsilon)E[rep(G_i)]$. The subgraph with the highest average reputation, among all the subgraph obtained at each iteration, is the reliable editor set (note that this is not necessarily the last step, since removing nodes also removes positive edges, so that reputation is not monotonous in i). Since the signed network of Wikipedia has 14M nodes, by using $\epsilon = 1$ the algorithm will have at maximum 24 MapReduce iterations (as discussed in Sec. VI, sensitivity to ϵ is part of our future work).

The algorithm in [6] can be parallelized. We implement it by running 5 MapReduce jobs with the following tasks: (i) compute reputation of editors in $rep(G_i)$ and the number of editors in S, (ii) compute average reputation $E[rep(G_i)]$, (iii) flag editors whose reputation is below $(1 + \epsilon)E[rep(G_i)]$ (iv) delete links whose source is flagged and (v) delete links whose destination is flagged.

To speed up computation in step (ii) we use a combiner equal to the single reducer node. In step (iii) instead no reducer is used, but we still exploit the mapping phase to parallelize computation across nodes. Then, mapper in step (iv) just takes as input the output of step (iii), while mapper in (v) just needs swaps sources and destination of output of step (iii). Pseudocode and further implementation details, that we omit here for lack of space, are available in [23] for the interested reader.

D. Classification

Classification task is simply explained with the help of Fig. 2. Denote the full signed network as W, and the reliable

[1]For each interaction among two editors on any article, the raw dataset contains the users, article and revision ID, the number of words inserted/deleted/replaced/retained, and restore/revert operations.

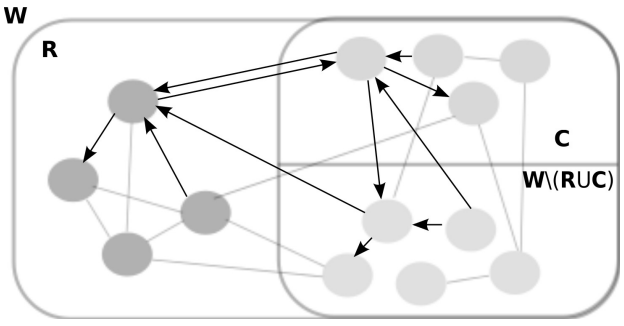

Fig. 2: Vandals detection methodology synoptic

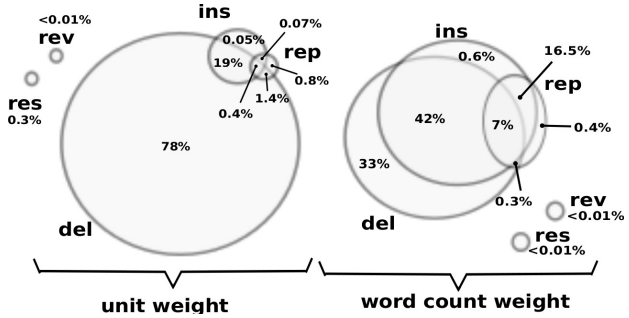

Fig. 3: Characterization of Wikipedia interactions, at aggregate level, with equal weights (left) or weights proportional to the number of words (right)

editor set found in the previous set as $R \subset W$. By definition, editors in R are not considered to be vandals. Editors in the residual set $W \backslash R$ can be then further divided into two set. Notably, a set C of editors whom some reliable editor in R had interacted with (and that can be classified) and a residual set $W \backslash (R \cup C)$ of editors that has not direct interaction with editors in R (and that cannot be classified based on R). As Fig. 2 shows, editors in $W \backslash (R \cup C)$ potentially interacted with editors in R (e.g., a potential vandal inserting, or deleting or reverting a reliable editor text), while the reverse is not true (i.e., the reliable editor has not had direct interaction with the potential vandal). In simple terms, we miss a direct reliable viewpoint of $W \backslash (R \cup C)$ (see Sec. VI for a potential solution). Classification is then merely done by majority voting (with threshold 0.5): i.e., the reputation of an editor in C is gathered by summing up the incoming edges to that editor coming from editors in R: an editor is labeled as "vandal" when negative edges outweighs the positive ones.

Two remarks are worth pointing out. First, the sign of an arc already compactly summarizes possibly several interaction between a pair of editors: hence, votes could be weighted on the ground of the number of interactions (e.g., judgment of a reliable editor with several interactions could be weighted more than a reliable editor with a single interaction). Again, in this phase of our work, we resort to equal weight for simplicity. As before, we defer sensitivity study of majority vote threshold to future work.

TABLE I: Edit length statistics, in words, for different interaction types (Int) and levels of granularity (G).

G	Int	mean	median	90-th	99.9-th
Aggr.	rep	38	3	40	4,411
	del	25	2	17	3,937
	ins	159	12	168	14,256
Single	rep	31	3	39	2,697
	del	45	3	54	4,287
	ins	56	10	79	3,488
Pair	rep	33	3	36	3,460
	del	23	2	17	3,289
	ins	89	10	101	6,890

III. Dataset

The MapReduce crawler allows us to explore a significant portion of the English Wikipedia. Overall, our dataset consists of $5{\cdot}10^6$ articles for a total of $261{\cdot}10^6$ revisions, resulting in $8{\cdot}10^9$ pairwise interactions between over $14{\cdot}10^6$ editors. As we will substantiate in Sec. V, this size is several orders of magnitude bigger than what is usually considered for vandal detection: hence, we believe that making the data set publicy available is an important contribution to the scientific community.

A. Characterization of interaction type

First, we gauge the relative popularity of Wikipedia interactions. While in the present study we are not considering weights while building the signed network, this would be a useful indication in order to equalize weights among different categories (e.g., give rev/res a higher weight than ins/del/rep). We note that, possibly multiple changes are effectuated in a single edit (e.g., some test is possibly deleted, other is inserted and other replaced), which is represented as overlap in Fig. 3. Statistics are computed either in terms of the raw number of interactions (left), or by weighting each interaction on the respective length in words (right). As clearly emerges from Fig. 3, while the majority of interactions are deletion, we have that the amount of deleted text is much smaller than the amount of the inserted one.

Importantly, while revert and restore interactions can be very informative concerning the likely presence of vandalism (e.g., a legitimate author can restore a version previous that vandalism occur; or a vandal can get rid of legitimate article improvements), their occurrence is too low in practice (well below 1%) to be useful without weighting.

B. Characterization of interaction length

It is possible to analyze Wikipedia interactions at different levels of granularity: namely, from the more coarse to the finer-grained, (i) aggregate of all editors, (ii) individual editors, (iii) editor-pair. In the first case, statistics will be biased by the most active editors. The second viewpoint is instead unbiased view respect to very active editors, but possibly over-represent sporadic editors. Finally, the latter viewpoint takes into consideration dynamics of interactions between editors, so that frequently interacting pairs of editors are weighted as much as more infrequent contacts. We argue a preliminary assessment

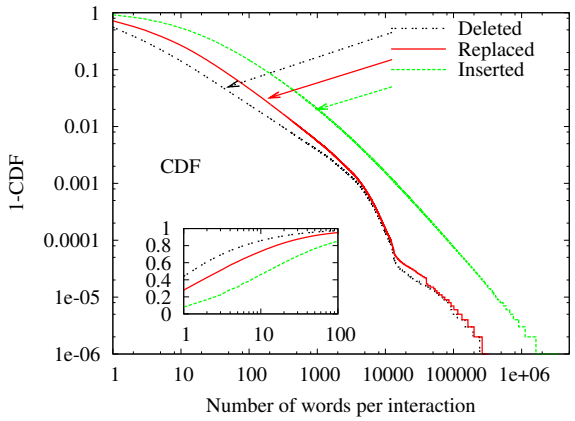

Fig. 4: Length of ins/del/rep, aggregate statistics

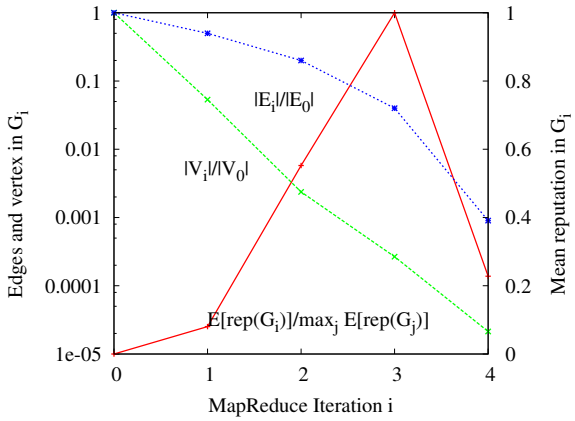

Fig. 5: Convergence of $(2 + 2\epsilon)$-approximation of densest subgraph

of all viewpoints to be instrumental for the definition of a signed network – were the wealth of interaction information is quantized in a single sign.

Tab. I reports mean, median, 90-th and 99.9-th percentile of edit length, in words, for different interaction types (replacement, deletion, insert) and levels of granularity (aggregate, individual, pairwise). Additionally, distribution of edit length at aggregate level are reported in Fig. 4. On the one hand, it can be seen that the bulk of edits is relaively short: only 10% of all edits insert more than 168 words; moreover, this length is exacerbated by few editors, since 90% of all editors (editor-pairs) insert less than 80 (101) words per edit.

On the other hand, it can be seen that an exiguous (0.1%) number of edits exhibit abnormal edit length – deleting, replacing and inserting possibly several *thousand* words per edit (exceeding 10^6 inserted words as shown in Fig. 4). While these abnormally long edits are very likely due to vandals due to vandals, we argue that almost any vandal detection mechanism based on edit length, can be easily worked around by splitting long edits in multiple shorter ones – which would likely create additional load on Wikipedia databases, rendering the detection technique useless if not harmful. Also, notice that longest delete/replace are shorter than insertion – which is due to the fact that while the maximum number of inserted word depends on the attacker resources (i.e., time and bandwidth), the number of deleted words is upper-bounded by the article length.

Overall, we get the 10-words minimum threshold (summing up all interactions over an editor pair) in building a signed network a reasonable tradeoff between (i) the size of the resulting dataset, as less than 1% of editors are filtered out and the (ii) relevance of the data, as we additionally avoid more complex per-interaction thresholds.

IV. Experimental Results

From the crawled dataset (Sec. III), we gather a signed network (Sec. II-B) consisting of 13.8 million editors and 99.3 million edges. By applying the densest subgraph approximation (Sec. II-C) on the signed network, we gather a reliable editor set consisting of 3932 nodes, that we use to classify the rest of editors (Sec. II-D).

Recall that, starting from the whole Wikipedia graph W at iteration $i = 0$, $W = G_0 = (E_0, V_0)$, at each iteration we remove from the graph nodes having a reputation $1 + \epsilon$ smaller than the average. Fig. 5 reports (logscale, left y-axis) the number of edges $|E_i|/|E_0|$, normalized to that of the initial graph, and the normalized number of nodes $|V_i|/|V_0|$ at the i-th iteration (where it can be seen that the number of nodes exponentially reduces at each iteration).

The picture also reports (linscale, right y-axis) the average reputation $E[rep(G_i)]/max_j E[rep(G_j)]$ of the nodes in G_i: reputation has a peak at the 3rd iteration (hence, our reliable set $R = G_3$), after which the algorithm converges at the 4th iteration. We first describe properties of the resulting graph R, compared to the whole set W and to the residual editors to be classified $W \backslash R$ in Tab. II. Notice that our classification algorithm can be applied only to the subset $C \subset W \backslash R$, consisting of 6.4M editors that have direct interactions with some of the 3932 editors in R. Specifically, our criterion is to say that members of $C^- = \{c \in C : rep(c) < 0\}$ are vandals while $C^+ = C \backslash C^-$ are legitimate editors.

We validate classification accuracy by manually constructing the ground truth. Manual validation is performed by browsing to the Wikipedia page showing the revision history, that is of generally straightforward interpretation, as it contains visually readable difference across versions, and is occasionally annotated with other useful informations (such as blocked account or IPs). For each R, C^+ and C^- set, we perform a stratified sampling of the whole population according to their reputation and manually validate a total of 300 sample articles. For each graph, we stratify populations in 10 groups according to the reputation, and sample 10 editors per reputation stratum.

Results of the validation are reported in Tab. III, showing very high true positive and true negative rates in the range 93%-95% (with furthermore tight upper and lower bound of the confidence interval computed according to the Wilson

TABLE II: Characteristics of R and W

	W	R	$W \setminus R$
$\|V\|$ $(\cdot 10^6)$	13.8	0.004	13.8
$\|E\|$ $(\cdot 10^6)$	99.3	4.0	52.3
$\|E^+\|$ $(\cdot 10^6)$	87.6	3.9	46.6
$\|E^-\|$ $(\cdot 10^6)$	11.7	0.07	5.8
$E[rep(G)]$	5.50	995.94	2.95
$density$	$5.0 \cdot 10^{-7}$	0.26	$2.7 \cdot 10^{-7}$

	$\|E^+(\cdot)\|$ $(\cdot 10^6)$		$\|E^-(\cdot)\|$ $(\cdot 10^6)$	
	R	$W \setminus R$	R	$W \setminus R$
R	3.9	11.4	0.07	5.1
$W \setminus R$	25.7	46.6	0.7	5.8

	$\|E^+(\cdot)\|/\|E(\cdot)\|$		$\|E(\cdot)\|/\|E(W)\|$	
	R	$W \setminus R$	R	$W \setminus R$
R	93.3%	68.3%	4.1%	16.6%
$W \setminus R$	97.3%	88.9%	52.7%	52.7%

TABLE III: Classification accuracy for three different sets

	R	C^+	C^-
True Positive (TP)	-	-	95%
True Negative (TN)	95%	93%	-
False Positive (FP)	-	-	5%
False Negative (FN)	5%	7%	-

score). On the other hand, recall, we have that recall of our method is $(|R| + |C|)/|W| = 49.3\%$. Hence, despite we are able to correctly classify on the order of several million editors –much larger size than what is done in the literature– still an equal number of editors remain unclassified, which we discuss in Sec. VI.

V. Related Work

Due to the success of online collaborative and social networks in general, and of Wikipedia in particular, there have been an increased interest in their study over the last few years. At high level, we have either (i) measurement studies addressing a multitude of social networks, (ii) studies modeling OSN as signed networks or (iii) mechanism for vandal detection.

Online social network measurement. For what concerns measurements, due to the multitude (and varying popularity) of social networks, a large literature almost covers their full spectrum, with work closely following the timeline and hype of new platforms – in loosely reverse chronological order, Google+[16], Gowalla[5], an undisclosed Chinese social network[29], Twitter[24], [22], Renren[12], [28], Facebook[26], Wikipedia[25] and Flickr[18]. While previous work on Wikipedia also addressed [25] a passive study of the workload it generates, our active crawling is instrumental in characterizing the type of interaction among editors for the definition of a signed network.

Signed networks. Internet applications have been modeled as signed networks (in which nodes representing users, resources, etc., establish *negative* or *positive* links with other nodes) since early 2000 as, e.g., for reputation of P2P networks [13] or spam [9]. More recently, local notion of trust in a network have been used for social networks [14] and Wikipedia [7]. Along these line, several proposal try to measure the worthiness of contributors to Wikipedia. In [3], a measure of trustworthiness of text are derived based on editor interactions, while [11] exploits interaction to build a reputation systems. With this regard, closer work to ours deals with edge sign *prediction*, having an existing signed network as input. Especially, [15] use a logistic regression model for link prediction, based on a feature vector consisting of the types of directed triads (i.e., relationship involving a groups of three nodes) a link is involved in. Building on our preliminary work [17] (that was however based on a much smaller scale of 563 articles for 910K total revisions and a total of 198K unique contributors), we propose to infer an implicit signed network directly from user interactions.

Wikipedia vandals. Closer in spirit to our work, recent effort focused on automated statistical classification of Wikipedia vandals, based either on textual data or article metadata. For instance, arguing that vandalism often involves the use of unexpected words to draw attention, [8] exploit the fitness (or unfit) of a new edit when compared with language models built from previous versions (though the method is applied to anecdotal dataset consisting of just 2 articles with about 8,000 revisions each). Others have focused on mining the text-style, offering that deep syntactic patterns based on probabilistic context free grammar (PCFG) discriminate vandalism more effectively than shallow lexico-syntactic n-grams [10].

However, text-mining is a relatively cumbersome, so that features associated to metadata may be preferable due to their lightweight. On this line, STiki [27] observes that metadata of malicious edit exhibits peculiar spatial (e.g., revision comment length) and temporal properties (e.g., time-of-day) unlike those associated with innocent edits (applied to a fairly larger dataset 298 million edits). Authors in [21] focus on systematic definition of features (such as compressibility, ratio of Uppercase characters, length of the longest word, frequency of vulgar terms, size of the edit compared to the previous version, anonymity of the editor, etc.) reporting 83% precision and 77% recall (but is however based on an exiguous set of 940 human-assessed edits from which 301 edits are classified as vandalism).

Research effort has also produced a labeled corpus PAN-WVC-10 [21] (by crowdourcing ground-truth using Amazon's Mechanical Turk) consisting of 32,452 edits on 28,468 Wikipedia articles, with 2,391 vandals (smaller than ours, but with exhaustive ground truth). On the PAN-WVC-10 corpus, which has become fairly popular since, more recent work [20], [19] reported an accuracy of 96% (thus comparable to ours). Yet, as previously pointed out, vandals could easily "game" some features (e.g., breaking long suspicious edits in many smaller ones) that would not go undetected with our approach (i.e., since breaking an edit will result in many negative interactions), confirming its interest.

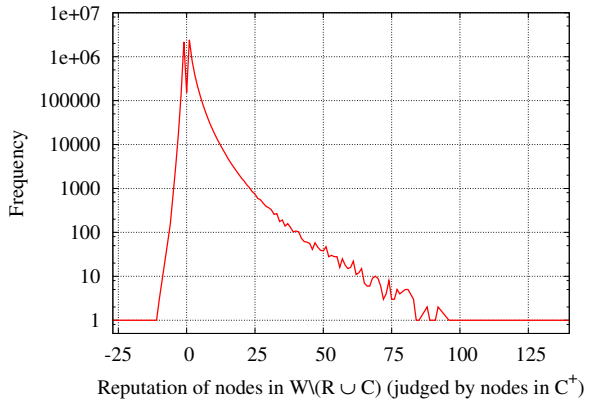

Fig. 6: Reputation of nodes in $W \setminus (R \cup C)$

VI. Discussion

We present the first extremely large scale study of Wikipedia vandal detection. Though preliminar, this work already achieves quite some contributions, ranging from developing a distributed MapReduce crawler, to offering the dataset to the community, to the accurate classification of Wikipedia vandals. Despite classification results are already fairly accurate, a trivial, but necessary, extension of this work concerns a sensitivity analysis of the detection thresholds and settings of the numerous parameters involved at several stages.

More interestingly, a limit of the current classification method is that it relies on direct link between high-reputation and low-reputation nodes, which are not always available. At the same time, we could iterate the process by letting editors in C^+ judge editors in $W \setminus (R \cup C)$. As shown by the reputation of the residual nodes in Fig. 6, there is a consistent fraction of nodes having null or negative reputation (about 2.4M), that could be classified according to majority voting of nodes in C^+ (and recursively apply the methodology). As can be gathered from Fig. 6, this can be expected to significantly increase the recall (an additional 2.4M/13.8M or 17% of editors could be classified as vandals in the first recusion), though further experiments would be needed to assess the accuracy degradation in the recursion (where a sensitivity analysis may be thus more relevant).

Finally, more recent joint work [4] of authors of [19], [21], [3] combine trust-based mechanism with metadata and text features. Specifically, [4] achieve 75% precision at 80% recall, or 99% precision at 30% recall. We believe iterative application of our method can further improve recall well beyond 50% while keeping accuracy close to 95%, sitting at interesting point in the tradeoff. While we believe our approach to be especially fit for very large scale dataset, our future work will also investigate whether direct comparison on the PAN-WVC-10 corpus is possible (notice that the editor graph we crawled has a different time span, so that it may be difficult to calibrate our signed network for a fair comparison on that dataset).

References

[1] http://en.wikipedia.org/wiki/Wikipedia:Vandalism.

[2] http://en.wikipedia.org/wiki/Wikipedia:Statistics.

[3] B. T. Adler, K. Chatterjee, L. de Alfaro, M. Faella, I. Pye, and V. Raman. Assigning trust to Wikipedia content. In *WikiSym*, 2008.

[4] B. T. Adler, L. De Alfaro, S. M. Mola-Velasco, P. Rosso, and A. G. West. Wikipedia vandalism detection: Combining natural language, metadata, and reputation features. In *Computational Linguistics and Intelligent Text Processing*, pages 277–288. Springer, 2011.

[5] M. Allamanis, S. Scellato, and C. Mascolo. Evolution of a location-based online social network: analysis and models. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2012.

[6] B. Bahmani, R. Kumar, and S. Vassilvitskii. Densest subgraph in streaming and mapreduce. *VLDB*, 5(5):454–465, 2012.

[7] U. Brandes, P. Kenis, J. Lerner, and D. van Raaij. Network analysis of collaboration structure in Wikipedia. In *WWW*, 2009.

[8] S.-C. Chin, W. N. Street, P. Srinivasan, and D. Eichmann. Detecting wikipedia vandalism with active learning and statistical language models. In *4th ACM Workshop on Information Credibility*, pages 3–10, 2010.

[9] Z. Gyöngi, H. G. Molina, and J. Pedersen. Combating web spam with TrustRank. In *VLDB*, 2004.

[10] M. Harpalani, M. Hart, S. Singh, R. Johnson, and Y. Choi. Language of vandalism: Improving wikipedia vandalism detection via stylometric analysis. In *Association for Computational Linguistics*, volume 2, pages 83–88, 2011.

[11] S. Javanmardi, C. Lopes, and P. Baldi. Modeling user reputation in wikis. *Stat. Anal. Data Min.*, 2010.

[12] J. Jiang, C. Wilson, X. Wang, P. Huang, W. Sha, Y. Dai, and B. Y. Zhao. Understanding latent interactions in online social networks. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2010.

[13] S. D. Kamvar, M. T. Schlosser, and H. G. Molina. The eigentrust algorithm for reputation management in p2p netwokrs. In *WWW*, 2003.

[14] J. Kunegis, A. Lommatzsch, and C. Bauckhage. The Slashdot Zoo: mining a social network with negative edges. In *WWW*, 2009.

[15] J. Leskovec, D. Huttenlocher, and J. Kleinberg. Predicting positive and negative links in online social networks. In *WWW*, 2010.

[16] G. Magno, G. Comarela, D. Saez-Trumper, M. Cha, and V. Almeida. New kid on the block: Exploring the google+ social graph. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2012.

[17] S. Maniu, B. Cautis, and T. Abdessalem. Building a signed network from interactions in wikipedia. In *ACM Databases and Social Networks*, 2011.

[18] A. Mislove, M. Marcon, K. P. Gummadi, P. Druschel, and B. Bhattacharjee. Measurement and analysis of online social networks. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2007.

[19] S. M. Mola-Velasco. Wikipedia vandalism detection. In *WWW*, pages 391–396. ACM, 2011.

[20] M. Potthast and T. Holfeld. Overview of the 2nd international competition on wikipedia vandalism detection. In *CLEF*, 2011.

[21] M. Potthast, B. Stein, and R. Gerling. Automatic vandalism detection in wikipedia. In *Advances in Information Retrieval*, pages 663–668. 2008.

[22] T. Rodrigues, F. Benevenuto, M. Cha, K. Gummadi, and V. Almeida. On word-of-mouth based discovery of the web. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2011.

[23] M. Spina. Finding Wikipedia vandals with a reputation based algorithm in MapReduce. In *MSc Thesis, Telecom ParisTech,*.

[24] K. Thomas, C. Grier, D. Song, and V. Paxson. Suspended accounts in retrospect: an analysis of twitter spam. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2011.

[25] G. Urdaneta, G. Pierre, and M. Van Steen. Wikipedia workload analysis for decentralized hosting. *Computer Networks*, 53(11), 2009.

[26] B. Viswanath, A. Mislove, M. Cha, and K. P. Gummadi. On the evolution of user interaction in facebook. In *2nd ACM workshop on Online Social Networks*, 2009.

[27] A. G. West, S. Kannan, and I. Lee. Detecting wikipedia vandalism via spatio-temporal analysis of revision metadata? In *3rd European Workshop on System Security*, pages 22–28. ACM, 2010.

[28] Z. Yang, C. Wilson, X. Wang, T. Gao, B. Y. Zhao, and Y. Dai. Uncovering social network sybils in the wild. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2011.

[29] X. Zhao, A. Sala, C. Wilson, X. Wang, S. Gaito, H. Zheng, and B. Y. Zhao. Multi-scale dynamics in a massive online social network. In *ACM SIGCOMM Internet Measurement Conference (IMC)*, 2012.