2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)

Troll Vulnerability in Online Social Networks

Paraskevas Tsantarliotis
Department of Computer Science & Engineering
University of Ioannina, Greece
Email: ptsantar@cs.uoi.gr

Evaggelia Pitoura
Department of Computer Science & Engineering
University of Ioannina, Greece
Email: pitoura@cs.uoi.gr

Panayiotis Tsaparas
Department of Computer Science & Engineering
University of Ioannina, Greece
Email: tsap@cs.uoi.gr

***Abstract*—Trolling describes a range of antisocial online behaviors that aim at disrupting the normal operation of online social networks and media. Combating trolling is an important problem in the online world. Existing approaches rely on human-based or automatic mechanisms for identifying trolls and troll posts. In this paper we take a novel approach to the trolling problem: our goal is to identify the targets of the trolls, so as to prevent trolling before it happens. We thus define the troll vulnerability prediction problem, where given a post we aim at predicting whether it is vulnerable to trolling. Towards this end, we define a novel troll vulnerability metric of how likely a post is to be attacked by trolls, and we construct models for predicting troll-vulnerable posts, using features from the content and the history of the post. Our experiments with real data from Reddit demonstrate that our approach is successful in recalling a large fraction of the troll-vulnerable posts.**

1. Introduction

Online social media and networks have emerged as the principal forum for the public discourse. However, this open global forum is threatened by users that actively try to undermine its operation. Such users engage in discussions without the intention of constructively contributing to the dialogue, but rather to disrupt it, and they are commonly referred to as *trolls*.

Due to the severity of the problem, there is significant effort in identifying and combating trolls. Related research along these lines includes detecting vandalism [1] and vandals [2] in Wikipedia, bad behavior in multi-player online games [3], and trolling comments in social new sites [4], [5]. A recent study analyzes users who were banned from three large online discussion communities to identify the characteristics of their behavior and how this behavior changes through time [6]. Another line of research in troll detection assumes the availability of a signed social graph among users where signs indicate positive and negative relationships among users. Then, troll detection is modeled as a ranking problem in this graph [7], [8].

In this paper, we take a different approach to the trolling problem. Instead of detecting trolls, we focus on identifying the possible targets of trolls. Given a post, we ask whether it is likely to attract trolls in the future, that is, how *vulnerable* the post is to trolls. To quantify the vulnerability of a post, we define the *Troll Vulnerability Rank* (*TVRank*) metric, based on the amount of trolling activity that followed that post. We then introduce the *Troll Vulnerability prediction problem*, where the goal is to predict which posts will acquire high *TVRank* value. Using historical data, we train models for the prediction task and apply them to new posts.

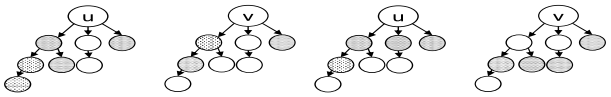
Our approach has a major advantage compared to traditional troll detection mechanisms in that it is *pro-active*: rather than detecting and removing trolls after they occur, we try to anticipate the troll activity and take preventive actions to eliminate it before it appears. Furthermore, modeling troll vulnerability offers valuable insights into what makes a post susceptible to trolling behavior. Finally, the *TVRank* value is a useful metric in itself, because it offers a way to measure the severity of the troll activity with respect to a post.

2. Model of Troll Vulnerability

We assume that trolling occurs within an online user-engagement ecosystem, such as a social network, a micro-blogging system, or a discussion forum. Users contribute content in the form of posts, and they interact with each other, creating discussions. We model interactions between posts as a directed graph $G = (V, E)$, where nodes $u \in V$ correspond to posts and there is an edge (u, v), from post u to post v, if v is a reply to u. For example, in Twitter, nodes may correspond to tweets and there is an edge from a tweet (node) u to all tweets (if any) that this tweet refers to. Similarly, in Facebook, nodes may correspond to comments on user posts.

In this paper, we will use Reddit, a popular online discussion forum, as our running example. In this case, the conversation graph of the posts defines a tree. The root of the tree corresponds to the initial post (message) that generated the discussion. Each node of the tree, other than the root, has a unique parent, and there is a directed edge from the parent-comment node to the child-comment node, indicating that the child comment is a reply to the parent comment. A comment may have multiple replies (children), but each comment replies to a single previous comment (the parent). The tree structure in posts is common to many social media.

IEEE/ACM ASONAM 2016, August 18-21, 2016, San Francisco, CA, USA
978-1-5090-2846-7/16/$31.00 ©2016 IEEE



(a) Trolling Volume (b) Proximity

Figure 1: Examples of the properties. Shaded nodes correspond to trollings and non-shaded nodes to non-trollings.

We note that our metrics are applicable to general graph structures as well.

For the following, we use the term *troll* to refer to a user that acts disruptively, and *trolling* to refer to a post with disruptive content. Our goal is to define a metric that quantifies the vulnerability of a post to trolling attacks. We first describe some intuitive properties that such a metric must satisfy.

First, clearly, posts that attract a large number of trollings must have high vulnerability.

Property 1 (Trolling Volume)**.** *The vulnerability rank of a post should increase with the number of its descendants that are trollings.*

Figure 1a shows an example of a discussion tree, where the shaded nodes are trollings. We consider node u to be more vulnerable than node v, since u has more trolling descendants than v.

Second, the proximity of trolling descendants should also be accounted for in the definition of troll vulnerability.

Property 2 (Proximity)**.** *The vulnerability rank of a post should increase with its proximity to trollings.*

For example, in Figure 1b, nodes u and v have the same number of trolling descendants. However, we consider node u to be more vulnerable, because node u is closer to its trolling descendants than node v.

To capture trolling volume and proximity, we use Random Walks with Restarts (RWR) for the definition of the troll vulnerability. Intuitively, we relate the vulnerability of a node u with the probability that a random walk starting from u will visit a trolling. The RWR takes place in the subtree rooted at u, where at each transition there is a chance α that the random walk restarts at u. For each descendant v of u, let $p_u(v)$ be the probability that the random walk is at node v after an infinite number of transitions. RWRs have been widely used to define the strength of the relationship between two nodes in a graph and are the building blocks of many metrics such as topic-sensitive PageRank [9].

We now define the troll vulnerability rank of a node as follows.

Definition 1 (Troll Vulnerability Rank)**.** *The Troll Vulnerability Rank (TVRank) of a post u is defined as:*

$$TVRank(u) = \sum_{\substack{v \text{ is a trolling} \\ v \neq u}} \frac{p_u(v)}{1 - p_u(u)},$$

The *TVRank*(u) value is the probability that the RWR visits a trolling, given that it is visiting a descendant of u. The higher the *TVRank* value of a post, the more vulnerable the post is. Our definition naturally incorporates the desired properties. In order for the *TVRank* to be high, a node must have a large fraction of its descendants to be trollings. Furthermore, due to the restart, distant trolling descendants have a smaller effect on the *TVRank*(u) than closer ones.

In addition to having a high *TVRank* value, for a post to be characterized as vulnerable, we ask that it also satisfies the following property.

Property 3 (Popularity)**.** *To be consider vulnerable, a node must have a large enough number of descendants (not necessarily trollings).*

The popularity property requires for a post to generate enough traffic in order to be of interest to moderators. For example, consider a node v that has just one descendant in total. Even if this descendant is a trolling response, there is no additional interaction and no further responses, so this is clearly a failed attempt at trolling.

Definition 2 (Post Vulnerability)**.** *A post u is considered vulnerable to trolls if it has at least K, $K > 0$, descendants and $TVRank(u) \geq \theta$, $0 \leq \theta \leq 1$, where K and θ are parameters that control the sensitivity of post vulnerability.*

The θ value determines the intensity of trolling activity that a post needs to generate for the post to be considered vulnerable. When moderation needs to be strict (for instance, to avoid insults in a social media where kids participate), a lower θ value allows prompt notification for potential trolling behavior. The threshold value K determines the minimum number of responses that a post needs to generate for the post to be considered important enough to be characterized as vulnerable.

3. Troll Vulnerability Prediction

In this section, we present preliminary results for the troll vulnerability prediction task.

Dataset. Our dataset contains posts from the Reddit social network website. We retrieved 20 submissions from each of 18 subreddits based on their popularity, resulting in 555,332 comments.

Although, identifying trollings is a problem orthogonal to our approach, to evaluate the performance of the troll vulnerability prediction task, we need to first detect trollings in our dataset. We focus on the anti-social part of trolls, i.e., we detect comments that contain offensive content. To this end, we modified a publicly available classifier produced in a Kaggle competition for detecting offensive content [1]. We detected 9,541 trollings in our dataset, which amounts to 1.7% of the dataset with accuracy more than 80%.

Prediction Task. Our goal is for a given a post to predict whether the post will be vulnerable to trolls or not. We treat the problem as a two-class classification problem, with the positive class corresponding to the vulnerable posts, and

1. http://goo.gl/UL2VuE

θ	0.15				0.20				0.25				0.30				0.35			
K	A	P	R	AUC	A	P	R	AUC	A	P	R	AUC	A	P	R	AUC	A	P	R	AUC
2	0.81	0.05	0.67	0.81	0.81	0.04	0.67	0.81	0.82	0.03	0.67	0.82	0.83	0.03	0.68	0.82	0.84	0.02	0.67	0.82
3	0.81	0.04	0.68	0.81	0.82	0.03	0.67	0.82	0.83	0.02	0.68	0.83	0.85	0.02	0.69	0.84	0.86	0.01	0.70	0.85
5	0.81	0.02	0.69	0.83	0.83	0.02	0.70	0.84	0.86	0.01	0.73	0.87	0.88	0.01	0.76	0.89	0.90	0.01	0.79	0.90
8	0.82	0.01	0.72	0.86	0.84	0.01	0.74	0.88	0.86	0.01	0.78	0.90	0.89	0.01	0.80	0.91	0.90	0.01	0.80	0.92

TABLE 1: Performance of the model with different values of K and θ. *A*, *P*, *R*, *AUC* stand for accuracy, precision, recall, and area under the ROC curve respectively.

the negative class to the non-vulnerable posts and build a classification model. For defining the positive class (i.e., the set of vulnerable posts), we use Definition 2. We design features that capture various aspects of the post and its past. Note that we only consider ancestors of the post, since we want to decide on its vulnerability, before the post receives any replies (i.e., acquires any descendants).

We group features in four categories, namely, content, author, history and participants. Content features include features related to the text of the post. Author features try to capture the behavior of the author of the post in the social setting. History-related features are extracted from the conversation tree of the post. Finally, the features related to the participants in a discussion contain information about the authors of the previous comments. For the classification, we use a Logistic Regression classifier with a 5-fold cross validation.

Considering each feature category independently, the classifier using the participant features performs the best, followed by the author features, while content features proved to be the weakest of the four groups. This indicates that features related to the users that post the comments carry a stronger signal. Combining features improves the prediction, with the classifier using features from all four groups being the best.

Troll Vulnerability Results. We report results of the classification model that uses all features from all four groups. Table 1 shows the accuracy (A), precision (P), recall (R) and area under the ROC curve (AUC) values of our classifier for different values of K and θ. Accuracy is very high in all cases, but this is basically due to the fact that the classes are highly unbalanced, with a small positive class and a dominant negative class. Precision is low, again due to the imbalance of the dataset. Recall and AUC are the most interesting measures, since we want to make sure that we avoid false-positives. Our results show that we are able to retrieve most of the vulnerable posts.

Larger values of K and θ increase the selectivity in the troll-vulnerability definition, resulting in fewer comments considered as vulnerable. The recall and AUC of the classifier improve when the classes of vulnerable comments become more selective. A reasonable tradeoff between selectivity and performance is achieved for $\theta = 0.3$ and $K = 2$, which results in 3,853 comments being characterized as vulnerable (which amounts to about 2.5 trollings per vulnerable comment, on average), while attaining high values for all performance metrics.

4. Conclusions

Understanding and detecting trolling behavior in social networks has attracted considerable attention. In this paper, we took a different approach shifting the focus from the trolls to their victims. We introduced the novel concept of troll vulnerability to characterize how susceptible a post is to trolls, and we proposed a metric based on random walks for measuring it. We then introduced the troll vulnerability prediction problem: given a post, predict whether this post will attract trolls in the future. Our initial results using the Reddit dataset are promising, suggesting that a proactive treatment of trolls is feasible.

Acknowledgments

This work has been supported by the Marie Curie Reintegration Grant project titled JMUGCS which has received research funding from the European Union.

References

[1] 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*, 2011.

[2] S. Kumar, F. Spezzano, and V. S. Subrahmanian, "VEWS: A wikipedia vandal early warning system," in *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2015.

[3] J. Blackburn and H. Kwak, "Stfu noob!: predicting crowdsourced decisions on toxic behavior in online games," in *Proceedings of the 23rd International Conference on World Wide Web (WWW)*, 2014.

[4] E. Cambria, P. Chandra, A. Sharma, and A. Hussain, "Do not feel the trolls," in *Proceedings of the 3rd International Workshop on Social Data on the Web (ISWC)*, 2010.

[5] S. O. Sood, E. F. Churchill, and J. Antin, "Automatic identification of personal insults on social news sites," *Journal of the American Society for Information Science and Technology*, vol. 63, no. 2, pp. 270–285, 2012.

[6] J. Cheng, C. Danescu-Niculescu-Mizil, and J. Leskovec, "Antisocial behavior in online discussion communities," in *Proceedings of ICWSM*, 2015.

[7] S. Kumar, F. Spezzano, and V. Subrahmanian, "Accurately detecting trolls in slashdot zoo via decluttering," in *Advances in Social Networks Analysis and Mining (ASONAM)*. IEEE, 2014.

[8] Z. Wu, C. C. Aggarwal, and J. Sun, "The troll-trust model for ranking in signed networks," in *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining (WSDM)*, 2016.

[9] T. H. Haveliwala, "Topic-sensitive pagerank," in *Proceedings of the 11th International World Wide Web Conference (WWW)*, 2002.