Detection and Classification of Twitter Trolls

Guanghao Qi

October 2016

1 Introduction

A troll is a person that posts controversial and provocative messages to upset people and draw attention from the public [5]. Among the social media, trolls reside predominantly in Twitter. According to a study on abusive social media mentions from football fans, 88% abuse occurs on Twitter [2]. The large number of trolls deteriorate the cyber-environment and expose people to constant threat of emotional damage. In 2012, Jessica Laney, a 16-year-old girl in Florida, committed suicide due to huge emotional pressure caused by trolls [3]. People are quitting Twitter to avoid cyber-abuse. For example, some celebrities closed their Twitter accounts after troll hits, others who never had a Twitter account, like Jennifer Lawrence, continue to stay away from it [4].

To combat the loss of core users and regain momentum in user growth, Twitter tripled the staff who are responsible for chasing down trolls [2]. However, this is costly and unefficient. This summer, Twitter finally banned one of its biggest trolls, Milo Yiannopoulos, after allowing him to send hateful posts for years. Users also took measures to protect themselves from trolls, publishing tips for identifying trolls and ways to deal with them, which are subjective to individual judgement and inconvenient to implement. Others created self-reporting based troll databases, which may be susceptible to selection bias.

For the well-being for both Twitter and its users, an automatic identification algorithm is needed to overcome the shortcoming of previous methods. Our approach uses tweets as unit of analysis and match them with a list of negative words. After detecting the troll tweets, we can naturally trace back to their sources and find the troll users. By applying our algorithm to data from changing time domain, we can get a dynamic list of trolls to keep people updated.

2 Methods

To detect trolls by identifying tweets that contain trolling content, we developed an algorithm based on negative word matching. We only target the troll tweets that explicitly contain negative words since they are linguistically complex.

2.1 Data collection

To collect data from Twitter, we registered a Twitter App and set up API access. Authentication and search were completed using R package twitteR version 1.1.9. In the search for tweets, we specified the key words, time domain, language and the number of tweets requested. In the search for user profiles, we provided a list of user names that we are interested in.

2.2 Troll detection and classification

As we mentioned above, the detection algorithm is based on calculating the frequency of negative words in the tweet. The raw material is a negative word list L put together from three different sources:

- A: list of swear words from NoSwearing [Link].
- B: list of insulting slang words and phrases from The Online Slang Dictionary [Link].
- C: list of bad words from CMU Professor Luis von Ahn's personal website [Link].

In addition, we also downloaded a list of 100 most common words, denoted by D, from Wikipedia [Link]. A involves multiple webpages and was downloaded manually. B, C and D are downloaded using the R web scraping package rvest version 0.3.2. They are listed in chronological order of collection. L is the unique collection of A, B and C after removing the words in D, i.e. $L = A \cup B \cup C - D$.

Besides the word frequencies in L, we are also interested in the word frequencies in individual lists A, B and C. A mainly contains very strong swear words. B contains some insulting words and many phrases. C contains words with negative sentiment but are not as insulting as A. To remove overlap and common words, we redefine the lists as

$$L_1 = A - D,\ L_2 = B - (A \cup C \cup D),\ L_3 = C - (A \cup D).$$

A and C are allowed to keep their overlapping words with B since they contain no phrases, which are harder match since have lots of variation. A is more insulting than C thus is given higher priority.

A two-step procedure is used to detect troll tweets:

1. For each tweet, count the number of words that appear in L. Denote the count by NWC (negative word count). Keep the tweets of which the NWC passes its 80% quantile.
2. Further classification for the remaining tweets by logistic regression using word counts in L_1, L_2 and L_3 (denoted by NWC_1, NWC_2 and NWC_3) as well as the sender's profile information: friend count (FrC), follower count (FoC) and total favorite count (FaC).

In step 1, the 80% quantile is a liberal threshold to narrow down the scope and still keeps almost all the trolls. And we believe that no more than 20% tweets are trolls even in a controversial topic. Since word counts are discrete, we pick the threshold that is closest to 80% quantile in practice, which may be off by a few percents.

Next, the senders of the identified trolling tweets are marked as trolls. Classification is purely based on the troll's follower count. A troll with many followers is more capable of spreading their posts thus has higher influence. A study shows that the 95% quantile of follower counts is 819 [1]. Therefore a troll with 819 followers or more is marked as a star troll, and otherwise a common troll.

3 Results

3.1 Data

We apply our algorithm to the tweets that comment on Brad Pitt and Angelina Jolie, who are a prominent couple and long-time focus of public attention. Their recent divorce sparked a heated discussion on the Internet and can be an area that people like to troll about. We searched for 450 tweets posted between 03/01/2005, the approximate date that Pitt divorced from ex-wife Jennifer Aniston and 10/08/2016, three weeks after the Pitt-Jolie divorce, by which there should be plenty of comments on Twitter. However, all tweets retrieved by the search function is on the most recent date 10/07/2016. For later method evaluation, we randomly sampled 100 from the 450 tweets and had a trained researcher manually label them as troll or non-troll.

Tweets are also needed to train our algorithm. The training data are the comments on the 100 most followed twitter accounts [Link]. For each of them, we searched for 90 tweets in the same time frame as the test data. Due to API limits, only tweets about the top 74 accounts are retrieved and all of them are on 10/07/2016. The actual number of tweets for each user varies since there may not be up to 90 accessible tweets. The number 90 is also based on the API rate limit 450 tweets/15min. In that case we can search for one topic every 3 minutes. A total of 5854 tweets were collected.

For all our collected tweets, we downloaded theirs senders' profile information, including FrC, FoC and FaC described in Section 2.2.

3.2 Troll detection and classification

Step 1 of the algorithm uses the L panel of Figure 1. See Supplementary Figure 1 for the words in L that appear in the training tweets. 1 is the integer closest to the 80% quantile. 1125 tweets satisfied $NWC \geq 1$. 1121 are left after removing the cases with missing data. To implement step 2, troll or non-troll labels of the tweets are required when fitting the logistic regression model. To reduce the burden of manual work, we had a trained researcher label only 200 tweets randomly sampled from the 1121 that passed the first step screening.

The model was fitted on the 200 labeled tweets using negative word counts $NWC_{1,2,3}$ and sender specific variables FrC, FoC and FaC (See Supplementary Figure 2 for their distributions). Since we do not know which among FrC, FoC and FaC are predictive, we first fitted a logistic model on each of them separately. The p values for testing the regression slopes are 0.104, 0.351, and 0.259. Only FrC is close to significance under the criterion p value ≤ 0.1.

Then we fitted a logistic prediction model on NWC_1, NWC_2, NWC_3 and FrC and used backward model selection that. It turns out that the final prediction model only involves NWC_2 and FrC. A potential explanation is that words counts in L_1 and L_3 are only useful in the first step screening but are not predictive in the second stage.

We applied the procedure to the Pitt-Jolie test data. 24 tweets passed the word frequency screening. Then we used the logistic model trained from the training data to predict the probability of being a troll tweet. Since 61 of the 200 tweets in the training set are trolls, we threshold the predicted probability of the test set at the 69.5-th quantile. It returns 8 tweets. The sensitivity is 0.27 and the specificity is 0.96.

See the supplementary material for the troll tweets. The senders' user names, follower count and category is listed in Table 1.

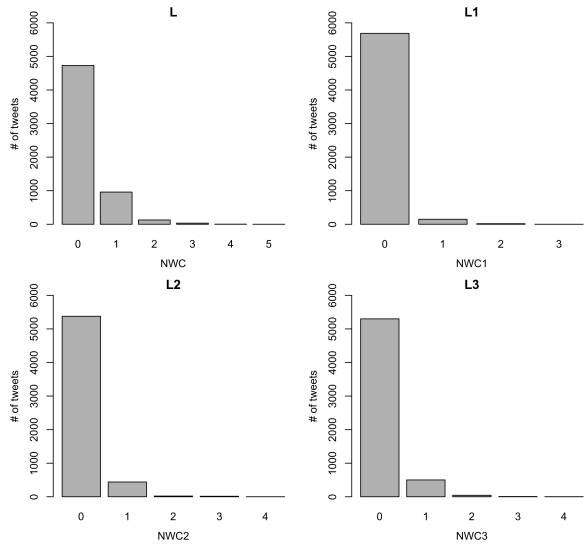

Figure 1: Distribution of NWC's. NWC, NWC_1, NWC_2 and NWC_3 for a tweet are the number of words in L, L_1, L_2 and L_3 that appear in the training tweet.

User name	Follower count	Star/Common
bazmi119	1467	Star
olzanskisrauhl	1221	Star
helensullivann	811	Common
AngelinaJolie49	9090	Star
zorinely	530	Common
libbyJones6	1022	Star
mango gab	1079	Star
georgeclony fan	593	Common

Table 1: Twitter trolls identified by our algorithm, their number of followers and category. A troll is marked as star if his/her follower count > 819 (95% quantile of follower counts), and as common otherwise.

4 Discussion

Our algorithm automatically detects troll tweets by using mainly negative words frequencies and also sender profile information as a supplement. As noted in section 3.2, ti gives a low sensitivity and a high specificity. This is due to both the sparcity of troll tweets and the conservativeness of our method. The prediction accuracy can be improved by enlarging the training data set, which requires more manual labeling.

References

[1] Jon Bruner. Tweets loud and quiet. O'Reilly, Dec. 2013. URL `https://www.oreilly.com/ideas/tweets-loud-and-quiet`.

[2] Jim Edwards. One statistic shows that twitter has a fundamental problem facebook solved years ago, Apr. 2015. URL `http://www.businessinsider.com/statistics-on-twitter-abuse-rape-death-threats-and-trolls-2015-4`.

[3] Steve Robson and Lydia Warren. 'can you kill yourself already?' the vile online messages from internet trolls 'that led girl, 16, to hang herself'. Daily Mail, Dec. 2012. URL `http://www.dailymail.co.uk/news/article-2246896/Jessica-Laney-16-committed-suicide-internet-trolls-taunted-told-kill-herself.html`.

[4] The Telegraph. Twitter trolls: The celebrities who've been driven off social media by abuse, 2014. URL `http://www.telegraph.co.uk/women/womens-life/11238018/Celebrity-Twitter-trolls-The-famous-people-whove-been-driven-off-social-media-by-abuse.html`.

[5] Wikipedia. Internet troll — wikipedia, the free encyclopedia, 2016. URL https://en.wikipedia.org/w/index.php?title=Internet_troll&oldid=745032976.