Troll Project

Matt Cole

October 23, 2016

Introduction

Internet trolls are considered a menace in nearly all online communities as they strive to generate emotional responses and cause disagreements between users. Motivations behind these users are unclear, but some social scientists suspect that a strange phenomenon known as the "disinhibition effect" may be to blame. Masked behind the apparent anonymity provided by many forms, sites, and the internet in general, social reservations that facilitate normal, face-to-face conversations can disappear, resulting in sometimes wild and rude behavior. This 'troll behavior' can stretch from simply posting the same status repeatedly to annoy and clog streams of information to violent threats and many 'things' in between. The effect of these trolls on Twitter, in particular, has been linked to many popular users leaving the platform entirely. The damage trolls may cause moves further than individuals, with Twitter's troll problem considered to be partially responsible its recent loss of 35 billion in market capitalization according to some analysts (Hern, 2016). In this project, we focused on identifying political trolls on Twitter, an area considered to be particularly saturated with such users.

Data

Twitter data collection began by identifying and collecting the usernames, also known as screen names, of accounts that have taken part in the political diologue on Twitter. These usernames were collected from Twitter both through automated means in through the Twitter API and manually through the web app (https://www.twitter.com/) and was limited to English tweets. The initial data collection consisted of searching twitter, via the twitteR API wrapper for users with tweets containing one of the two major political party presidential candidate's twitter username within the tweet text (@hillaryclinton or @realdonaldtrump). This collection occured on October 1st, 2016 at 4:14:19 PM EST. Twitter API mechanics resulted in a mix of most recent tweets (up to the second), as well as 'hot' tweets that were somewhat recent and popular in terms of favorites and retweets. In total, 142 usernames were collected through this method. From October 7th through October 14th, because of a lack of trolls in the automatically sampled usernames, manual harvesting was conducted, consisting of browsing political figures tweets and responses, in an attempt to collect users with a higher proportion of trolls for modeling purposes. During this collection period, 132 additional usernames were added for a total of 274.

Once the list of 274 users who have participated in the twitter political discourse was created, each user's 10 most recent tweets, as well as relevant tweet data (favorites, location, retweets, etc.) and data associated with the account (friends, description, etc.), were compiled. Using only the text from each user's 10 most recent tweets, including retweets and replies, users were classified as a troll or non-troll using the following scheme:

Trolls were defined as users with at least one of the previous 10 tweets appearing to be both:

- inflammatory: intended to arouse angry or violent feelings
- extraneous: irrelevant or unrelated to the subject being dealt with

Ten tweets were chosen in an attempt to capture enough of each user's tweeting tendencies and classifying users as trolls with at least one inflammatory and extraneous tweet allows us to capture more subtle trolls. In total, all 274 users were classified as troll/non-troll after examining roughly 2,740 individual tweets.

Methods

Using each users previous 10 tweets, we were able to construct average tweet sentiment using the AFINN lexicon. This lexicon comprises of English words each with an integer valued sentiment score measured between minus five (negative) and plus five (positive). Stop words, or words that likely will not contributed to the sentiment understanding of the sentance such as: as, the, is, at, which, or on, were removed prior to text analysis.

Sentiment scores for each tweet were then calculated by summing together the values of each word, and averaged together. Additionaly, the NRC lexicon, which assigns scores to each of two sentiments (positive and negative) as well as eight emotions was used used to generate an ‘angry’ score comprising of anger, fear, and disgust subscores as well as a ‘happy’ score comprising of joy and trust emotion subscores. As emojis have become increasingly popular in today’s web-based communication, an emoji dictionary was scrapped from http://www.unicode.org and emoji usage as well as sentiments were determined and averaged across tweets. Tweet source (how tweets were sent) was collected and divided into three categories, browser-based tweets, mobile tweets, and other tweet systems which included automated (bots) and unknown tweet sources.

Exploratory data analysis revealed that no single factor showed good seperation, between trolls and non trolls (figure 1). Several covariates expected to have a strong relationship with troll status such as number of followers did not appear to show such in the 2 dimensional plots (figure 1). However, others such as AFFIN sentiment score did show a possible relationship, although weak at best (figure 1).

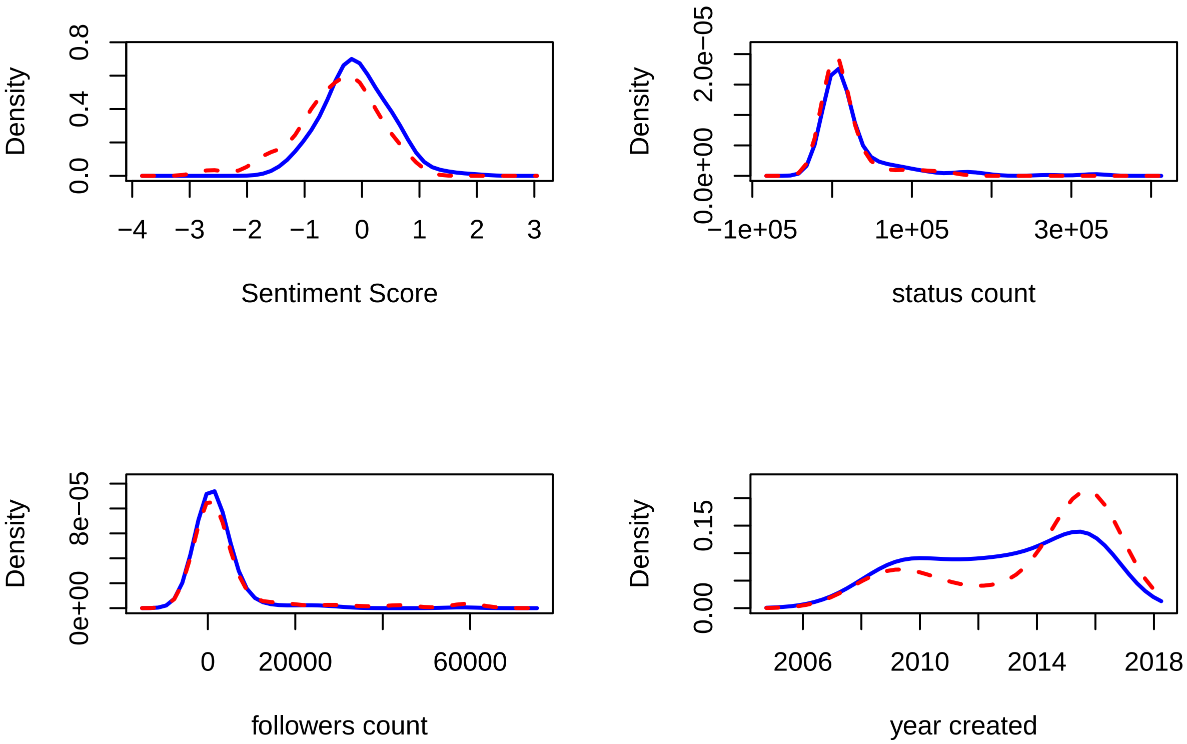

Figure 1 Smoothed density plots of common twitter metrics in both troll and non-troll groups. Trolls are denoted by the red line while non-trolls are denoted by the blue line. Distribution of AFFIN sentiment score and year of account creation appear to potentially be different in the two groups while there is no noticible difference in follower or status count.

We focused on three models to predict troll status: a simple logistic regression, a ‘full’ logistic regression model consisting of multiple covariates as well as a classification trees. In order to assess model fit, 20% of the data was randomly partitioned to a designated ‘testing’ data set comprising of 55 observations, of which 21 are trolls. The remaining 219 users (including 0 trolls) were used to construct all models and such models were tested by predicting the status of the testing set.

Classification and regression trees (CART) are a group of supervised machine learning methods which construct decision tree models utilizing observed data. Unlike other popular methods for classification or regression, CART methods do not produce global models, where a single predictive formula is employed to make decisions or predictions over all observations (James et al., 2013). Instead, CART methods break down the feature space containing the observations into small subspaces (called recursive partitioning) until the subspaces can be represented by relatively simple models. Benefits of CART methods include its high level of interpretability, automatic variable selection, and ease of visualization.

A classification tree was grown using the rpart package in R. Because of the unballanced dataset, we opted to weight the samples by the reciprical of the proportion of our data in each class (troll or non-troll). The classification tree's nodes were pruned back by ruducing the number of nodes to minimize the 10-fold cross-validated error rate (Figure 2).

Logistic regression models the probability of a user being a troll for each observation i (p_i) by treating the logit function, $log(\frac{p_i}{1-p_i})$ as a linear function of the covariates. In this study, a major limitation may be logistic regression's inability to capture non-linear trends associated with the covariates. However, this limitation can be partially overcome by the addition of splines or interaction terms. First, we examined a simple logistic regression only consisting of AFFIN sentiment, to better understand the analysis and found a prediction error of 0.31. In addition, we see that without controlling for any other covariates, for each 1 unit decrease in average AFINN sentiment score, expected odds of being a troll increased by a factor of 0.9225591. Another logistic model was fit, forward selection to minimize 10-fold cross validation. We ended with a model utilizing average AFINN sentiment score, the number of followers, number of lists the user is on, if the user is on mobile, if the user is using the 'default egg' photo, and the number of friends (people following) (table 1).

Results

For our classification tree, We found a cv prediction error rate of 0.193, sensitivity of 0.859, specificity of 0.78, and a AUC of 0.758 (figure 1, table 2). This showed a considerably higher degree of accuracy compared with the logistic regression models that were produced.

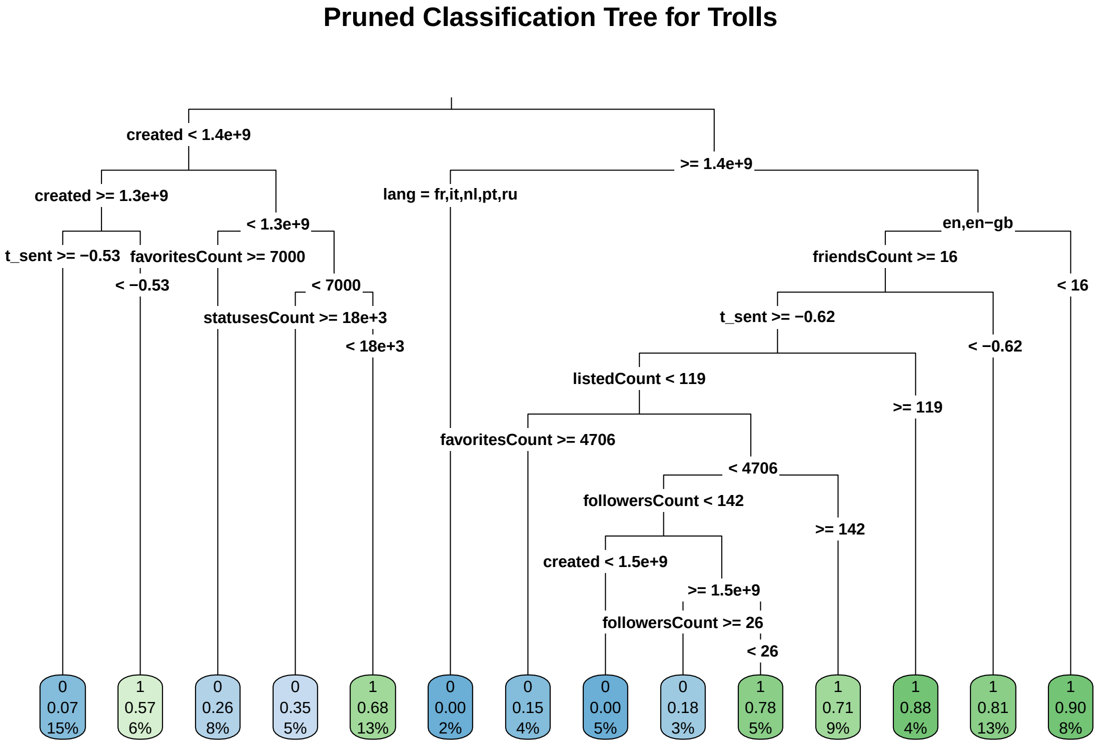

Figure 2 Pruned regression tree visualization. Nodes indicate, prediction (0 or 1 for troll status), mean troll value, and percentage of data contained in each node.

Table 1: Logistic regression coefficients

	Estimate	Std. Error	z value	Pr(>|z|)
Intercept	0.12767	0.26379	0.48398	0.62840
Average Tweet Sentiment (AFFIN)	-0.89837	0.22131	-4.05925	0.00005
Follwers	0.00011	0.00003	3.21587	0.00130
Listed Count	-0.00408	0.00174	-2.34981	0.01878
Mobile User (binary)	-0.05335	0.02907	-1.83505	0.06650
Default Image (binary)	0.69431	0.40901	1.69757	0.08959
Number of Friends	-0.00015	0.00008	-1.90624	0.05662

Table 1 Logistic regression coefficients.

```r
x1<-c(log1_err_rate, log1_sens, log1_spec, log1_auc)
x2<-c(log2_err_rate, log2_sens, log2_spec,log2_auc)
x3<-c(tree_err_rate, tree_sens, tree_spec, tree_auc)
x_all<-as.data.frame(rbind(x1,x2,x3))
rownames(x_all) <- c("Simple Logistic", "Full Logistic", "Regression Tree")
colnames(x_all) <- c("Error Rate", "Sensitivity", "Specificity", "AUC")
kable(x_all)
```

	Error Rate	Sensitivity	Specificity	AUC
Simple Logistic	0.4000000	0.6190476	0.5882353	0.6750700
Full Logistic	0.3818182	0.5714286	0.6470588	0.6974790
Regression Tree	0.1934307	0.8095238	0.5588235	0.7584034

Table 2 Comparison of all three models.

Discussion

results

It was expected that the regression tree would be able to outperform the logistic regression as the relationship between covariates and troll status seem opaque at best and would likely be best modeled with a non-parametric method. As we saw, the classification tree was in fact able to outperform the logistic regression models with respect to 10-fold cross-validation prediction error rate and recorded a sensitivity of 0.86 and a specificity of 0.78.

This study did shed light on a strange phenomena of different classes of trolls. From our anecdotal experience, there seemed to be three types or classes of twitter trolls.

- 'personal accounts' comprising of students, professionals, etc. whom occasionally say mean/emotionally charged things on twitter.
- bots: usually newly created, low tweet/follower/friend count which seem to spew politically charged propaganda.

- troll/fan pages: these accounts seem to be semi-autonomous as they respond to policial figures (@therealdonaldtrmp, @hillaryclinton) almost instantaneously while also being able to respond to certain accounts in a thought out manner. These accounts have many tweets, were created long ago and have many followers. In fact, Fortune, POLITICO and the New York Magazine all have run stories of these 'bot armies'.

limitations

The main limitation of this study was a lack of available data, particularly trolls. Although there were 274 observations there were only 92 trolls in our dataset, more would likely allow us to identify additional relationships and interactions between variables and troll status. In addition, there is the possibility for human error as a single person classified all tweets. This error could come from keystroke error or implicit political bias. Political bias could be difficult to identify and overcome where beliefs of what actually occured could alter whether a statement is either inflamitory or extraneous. A good example of this are statements such as "Bill Clinton is a rapist", which were classified as both inflamitory and extraneous, but to an individual whom believes that the statement is true, would likely not view it as extraneous as it is simply stating the truth.

Generalization

It is plausible that these models could generalize to other spaces within twitter or even outside of the social media site where trolling is a serious issue. However, the model relies heavily on non-text covariates such as friends, dates, and followers and applying this model to, an anonymous online form would likely be less effective, as this user data would be missing. the models here may not generalize to sites where the long posts are common as tweets are capped at 140 characters, however potentially averaging sentiments or bad words (ie. average sentiment per 100 characters or 10 words) may possibly be effective.

future work

In the future, it would be beneficial to further optimize tuning parameters associated with these models and inspect how other algorithms perform, particularly convolutional neural networks as well as support vector machines which may be able to identify aditional hard to see relationships in the data resulting in better prediction accuracy. In addition, future work should include identifying word n-grams which may allow us to gain additional insight beyond lexicons. n-grams are popular in computational linguistics as two words in sequence contain additional information about the meaning of the sentence compared to lexicon analysis of the same two words. Future would could focus on classifiny individual tweets as troll / non-troll tweets or even non-binary degrees of 'troll', for instance, classifying three types of trolls mentioned earlier in the discussion.

Thank you

Thanks to Finn Årup Nielsen manually labeled the words in the AFINN lexicon from 2009-2011, which was used in this analysis.

Thank you to http://unicode.org/emoji/charts/full-emoji-list.html for letting me borrow the emoji database without asking.

References:

Hern, Alex. "Did Trolls Cost Twitter $3.5bn and Its Sale?" The Guardian. Guardian News and Media, 18 Oct. 2016. Web. 21 Oct. 2016.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). An introduction to statistical learning (Vol. 6). New York: springer.

Martin, J. H., & Jurafsky, D. (2000). Speech and language processing. International Edition, 710.

Packages

Hadley Wickham and Romain Francois (2016). dplyr: A Grammar of Data Manipulation. R package version 0.5.0. https://CRAN.R-project.org/package=dplyr

Terry Therneau, Beth Atkinson and Brian Ripley (2015). rpart: Recursive Partitioning and Regression Trees. R package version 4.1-10. https://CRAN.R-project.org/package=rpart

Hadley Wickham, Jim Hester and Romain Francois (2016). readr: Read Tabular Data. R package version 1.0.0. https://CRAN.R-project.org/package=readr

Bowman, A. W. and Azzalini, A. (2014). R package 'sm': nonparametric smoothing methods (version 2.2-5.4) URL http://www.stats.gla.ac.uk/~adrian/sm, http://azzalini.stat.unipd.it/Book_sm

Stephen Milborrow (2016). rpart.plot: Plot 'rpart' Models: An Enhanced Version of 'plot.rpart'. R package version 2.1.0. https://CRAN.R-project.org/package=rpart.plot

Jeff Gentry (2015). twitteR: R Based Twitter Client. R package version 1.1.9. https://CRAN.R-project.org/package=twitteR