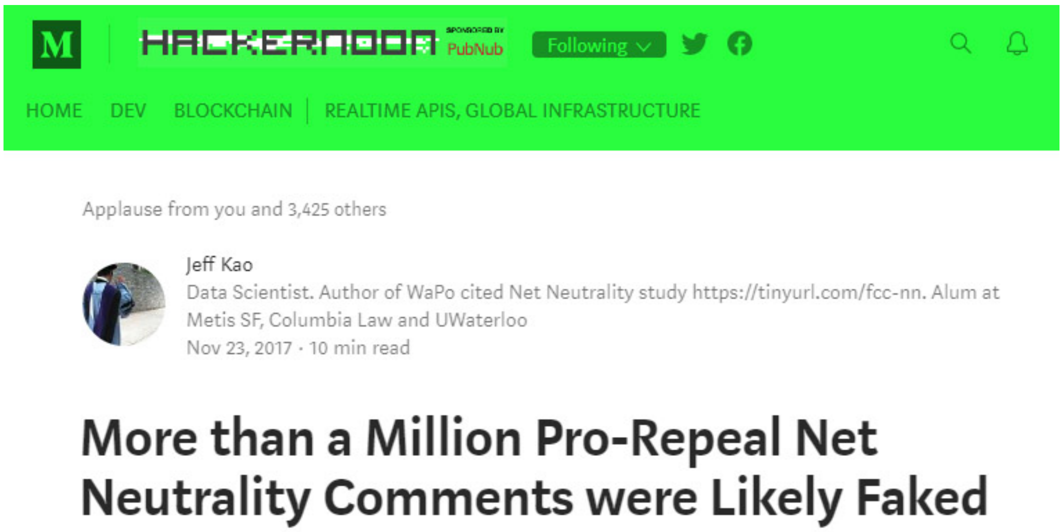
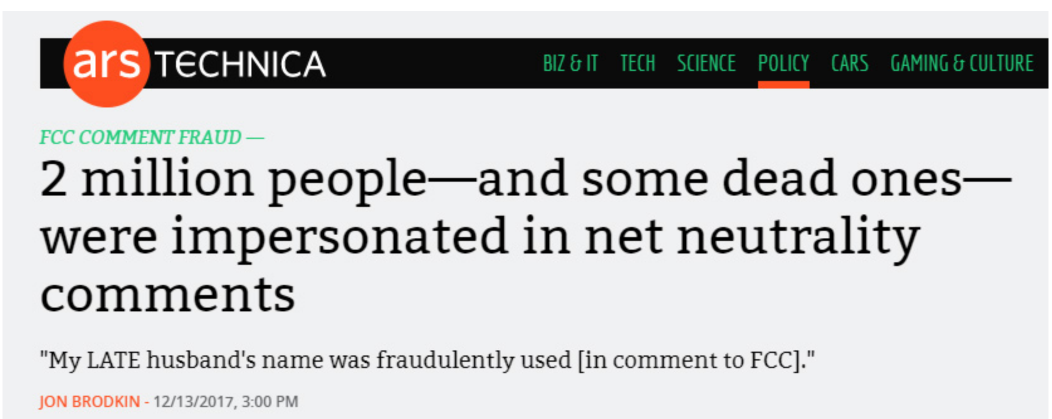
Analysis of FCC Net Neutrality Submissions

In the winter of 2017 there were numerous articles about quantity of fake comments submitted regarding the repeal of Net Neutrality laws by the FCC.



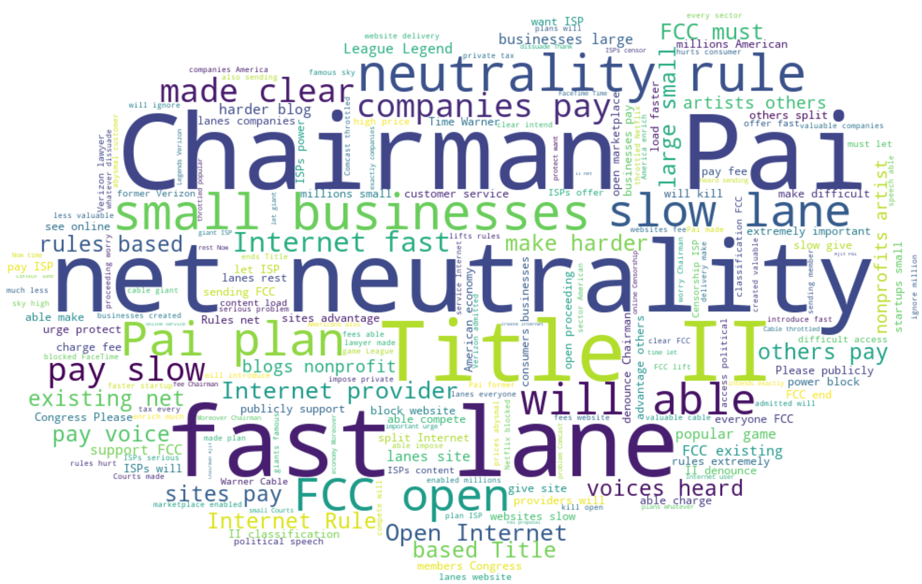
A [blog post published by Jeff Kao] caught my attention and I followed up with him on his analysis of the text. He provided me with the unedited 22 million filings available. I analyzed a sample of 3 million of them to see what I could find to develop my own features based around the text of faked comments.



Exploratory Data Analysis

The sample of 3 million FCC Net Neutrality comments were taken from the 22 million submitted. After importing the data from SQL into a csv file, I cleaned and investigated the data to extract the email, zip code and state details.

My first step was to take a look at the text of the comments:

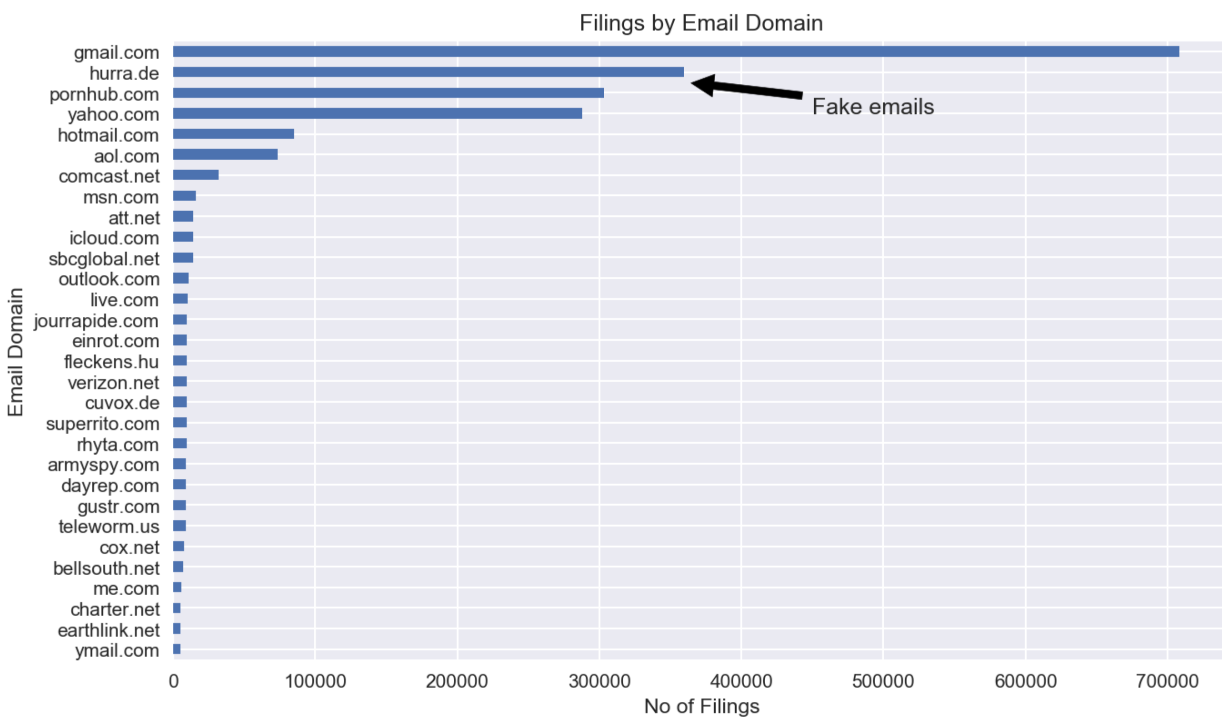


The first thing that caught my eye was the preponderance of identical word pairings. These are probably an indicator of faked or spammed comments if they repeat that frequently.

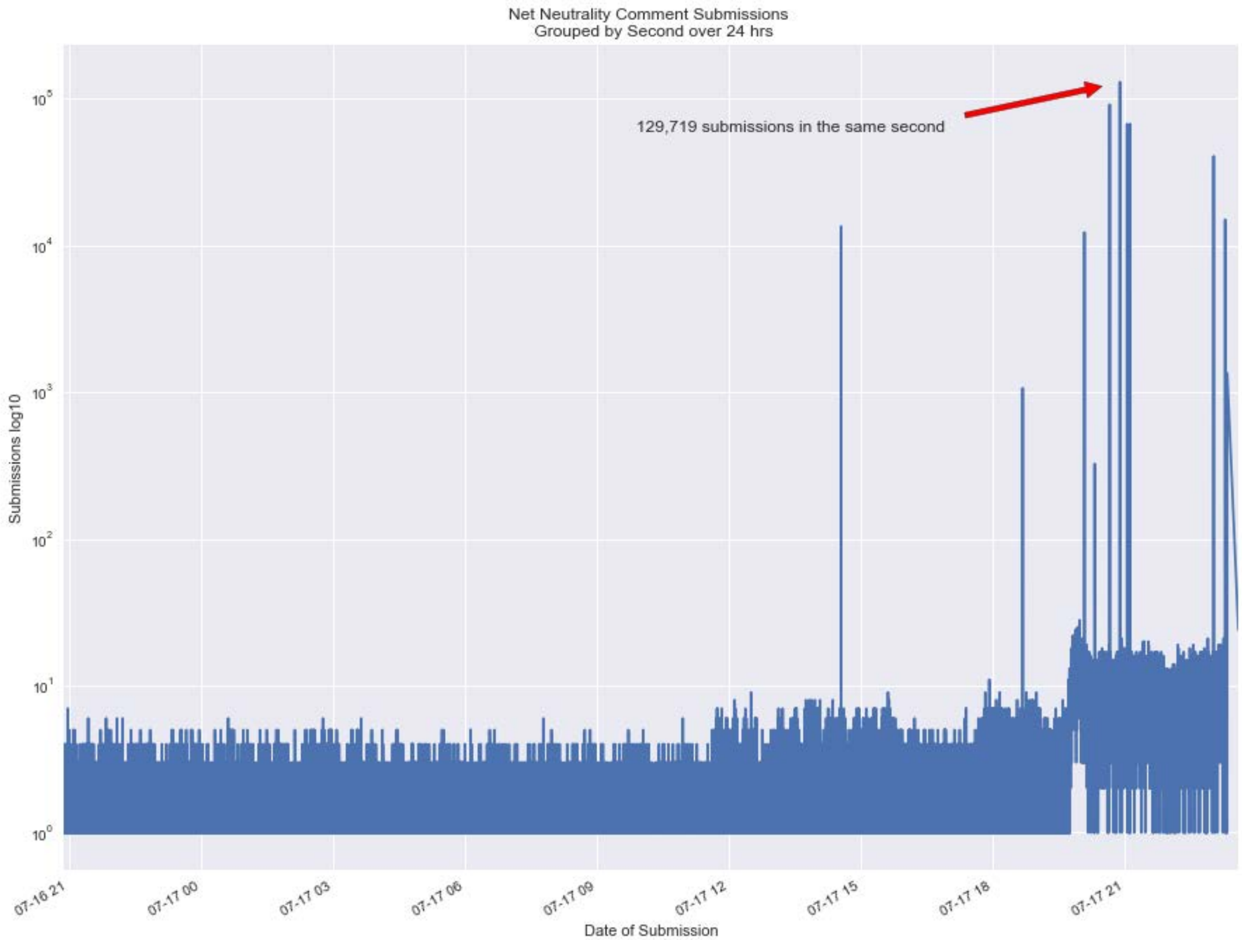
Identifying Fake or Not Comments

There were many clues as to which submissions were faked, for example:

- An IP address was provided instead of a zip code
- The city in the address was situated in the wrong state (Chicago wrongly situated in the state of Virginia)
- In a column 'internationaladdressentity' a valid US zip code was listed free form text field along with street address and state. However the city was omitted, something native residents would not do. In addition to a superfluous addition of "United States of America was typed but not needed.
- There were submissions also had either blank contact emails or the email address was written in all caps, where nothing else was. There was even a set of duplicate submissions filed by johndoe@gmail.com.
- It has already been reported in the press that hundreds (thousands) of these appeared to submitted during the same exact second. Something a spam bot would do as it generates the fake submissions.
- Fake email domains were used, and some clearly so in that their domain does not exist such as ‘hurra.de’. As well as several hundred thousand Russian employees of pornhub.com.



Here is a diagram of submissions filed in the same second over a 24 hour period starting July 17th 9pm. At one point the peak exceeded 129,000 in the same second (July 17th 8:53:08 pm).



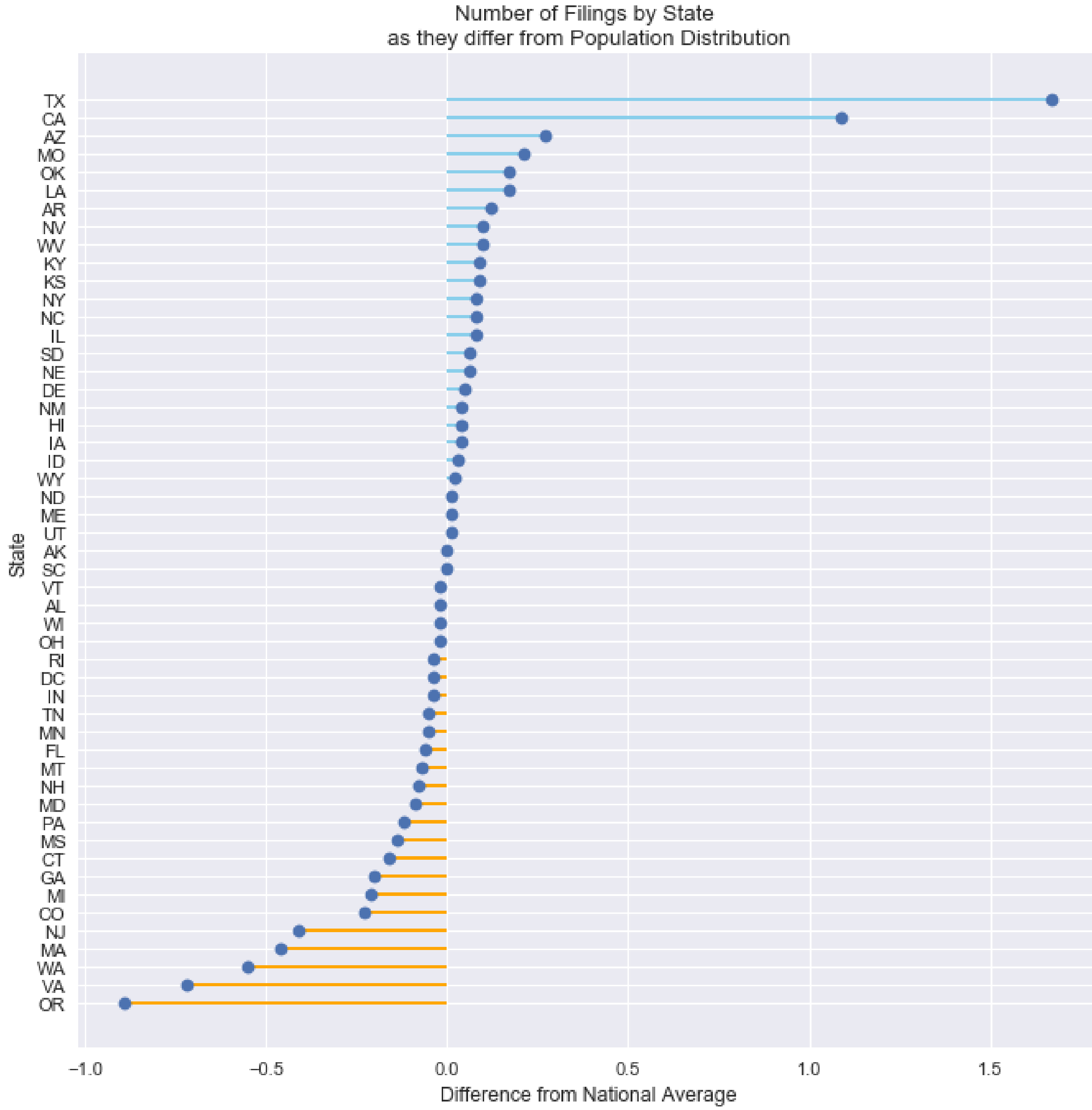
Time of Submission	Number of Submissions
2017-07-17 20:53:08	129286
2017-07-17 20:39:21	90679
2017-07-17 21:07:07	67025
2017-07-17 21:02:51	66852
2017-07-17 23:01:02	40426
2017-07-17 23:16:41	14947
2017-07-17 14:33:14	13433
2017-07-17 20:04:55	12237
2017-07-17 23:19:39	1345
2017-07-17 18:40:41	1064
2017-07-17 20:18:43	325
2017-07-17 20:53:49	52
2017-07-17 19:58:15	28
2017-07-17 19:55:19	25
2017-07-17 19:56:51	24

Were Fake Filings Statistically Significant?

I used 2016 census estimates by state to compare with the filings to see the variation. The proportion of filings from each state should be similar to the proportion of the US population in that state (and the District of Columbia).

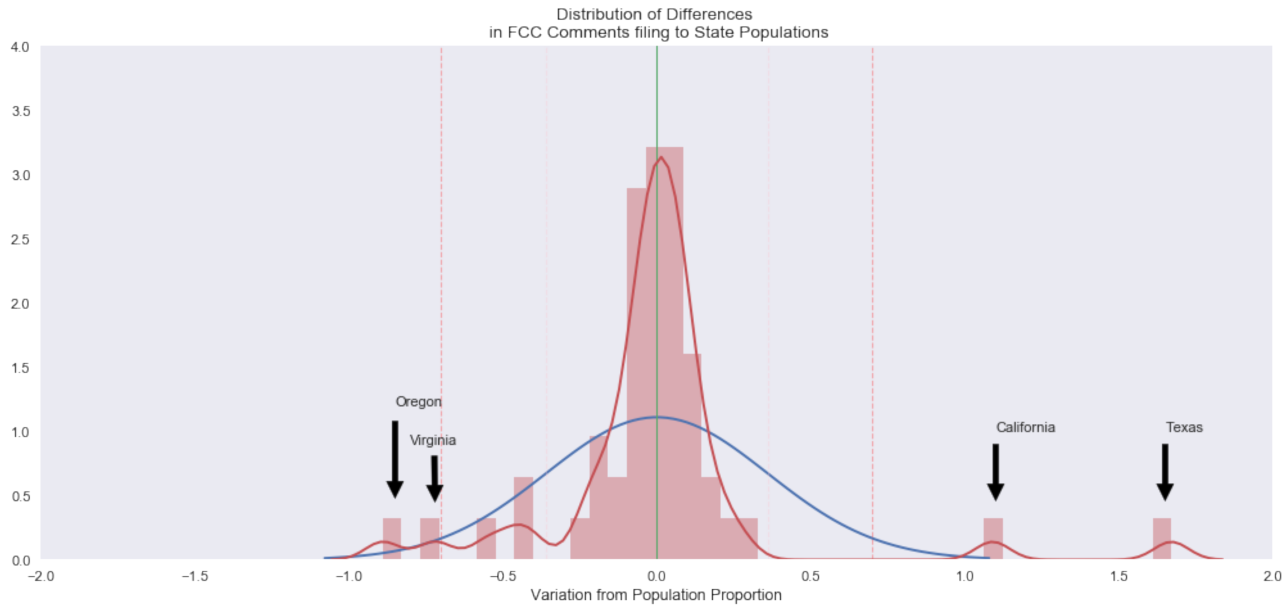
State	Population Estimate 2016	% of US Population	Number of Submissions	% of Submissions	Difference from Population %
Alabama	4863300	1.51	4441	0.23	**-0.02**
Alaska	741894	0.23	29970	1.53	**0.00**
Arizona	6931071	2.14	15605	0.80	**0.27**
Arkansas	2988248	0.92	36573	1.87	**0.12**
California	39250017	12.15	216119	11.06	**1.09**
Colorado	5540545	1.71	37914	1.94	**-0.23**
Connecticut	3576452	1.11	24834	1.27	**-0.16**

Visualizing the differences by state; the variations are largest in Texas, California, Virginia and Oregon.



The comments should be distributed by states in proportion to the population. My assumption is that the difference from the state average has a mean of 0 and a standard deviation of 1.

These 4 aforementioned values from Texas, California, Virginia and Oregon are well outside the 95% area of the Normal curve as indicated by the horizontal dark red lines either side of the center (2 standard deviations).



Labeling the Features for Supervised Learning

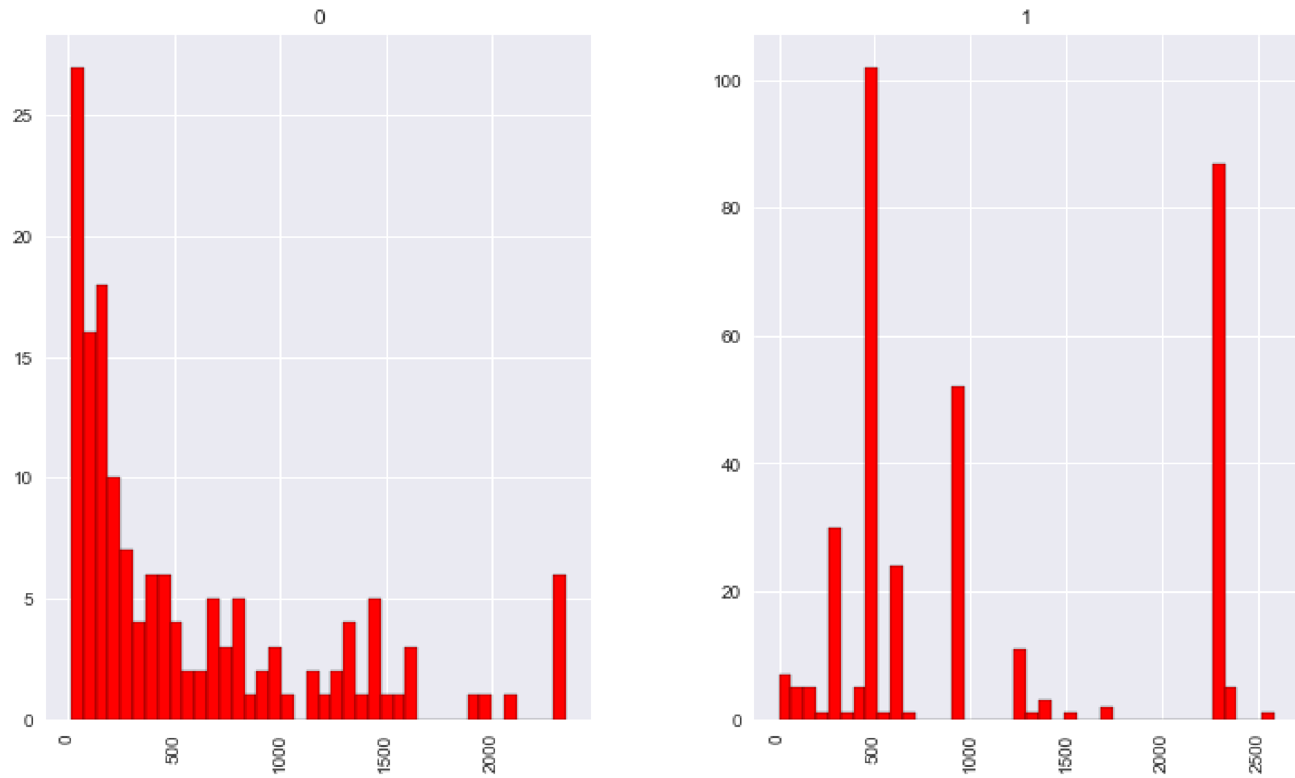
I sampled 500 rows from the dataset labeled them as Fake or Not (1 or 0). Even though the repetition of some of the fake ones is a give away there were other clues, such as missing address details that a local would not make or an email address from pornhub, hurra.de or even an email in all caps with no capitalization in the comments, address etc. As mentioned earlier some of clues of a fake submission were: missing value entries in a field rather than blank values, indicating an automated spam bot submitting the form.

Which I simplified to:

Sample of Fake or Real Text	Fake	Submissions Per Second	Text Length
Net neutrality should NOT be eradicated for th...	0	9	205
The unprecedented regulatory power the Obama A...	1	3	638
The FCC's Open Internet Rules (net neutrality ...	1	4	2311
The FCC's Open Internet Rules (net neutrality ...	1	5	2332
The FCC's Open Internet Rules (net neutrality ...	1	129286	228

Within my sample I also added text comment length and wanted to see if that or submissions per second were also good features. As can be seen on in this histogram, the Fake comments, labeled as '1', have high values for length with the majority of them at 500 and 2300. Whereas the Real comments, labeled as '0', have a text length 500 or less.

Comparison of Comment Text Length Real(0) or Fake(1)



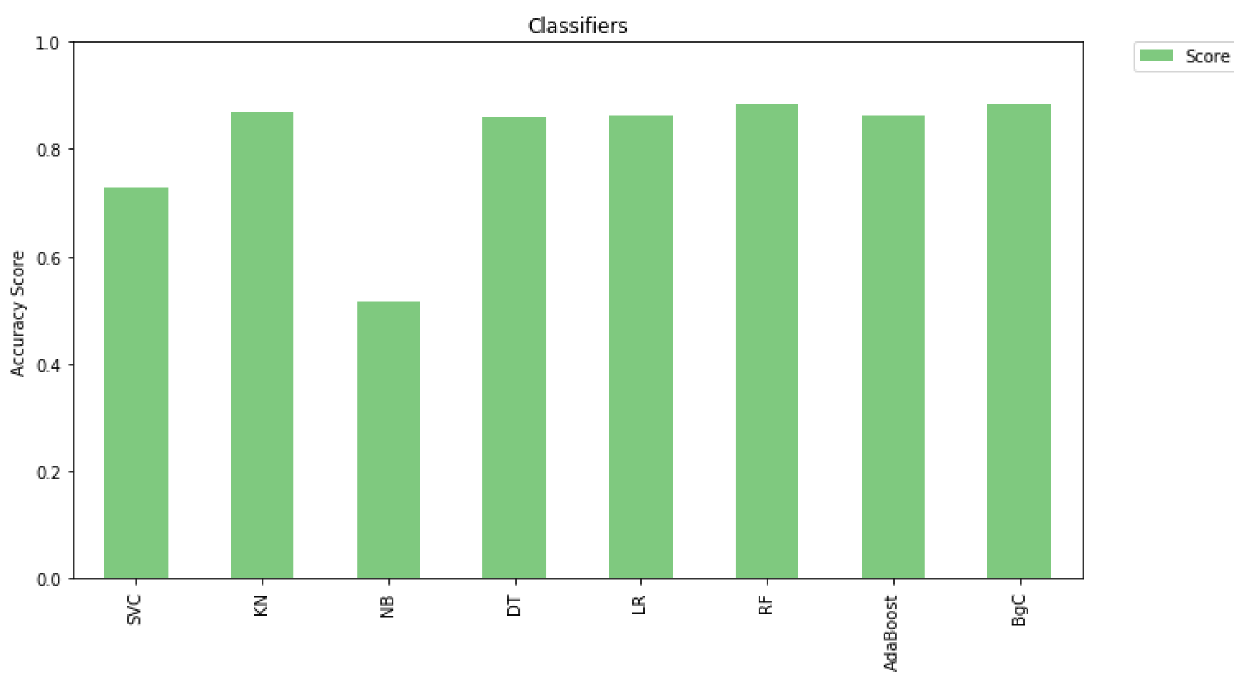
I group of 8 classifiers to iterate through, with a training size of 60% of the 500 labeled data, and generated the accuracy scores. I have 2 additional features which I want to try, Length of Text and Submission Frequency.

- SVC
- K Nearest Neighbors
- Multinomial Naive Bayes
- Decision Tree
- Logistic Regression
- Random Forest
- Adaptive Boost
- Bagging

Initially just on the TFIDF created from the bag of words output this accuracy score:

Classifier	Score
SVC	0.728643
KN	0.869347
NB	0.517588
DT	0.859296
LR	0.864322
RF	0.884422
AdaBoost	0.864322
BgC	0.884422

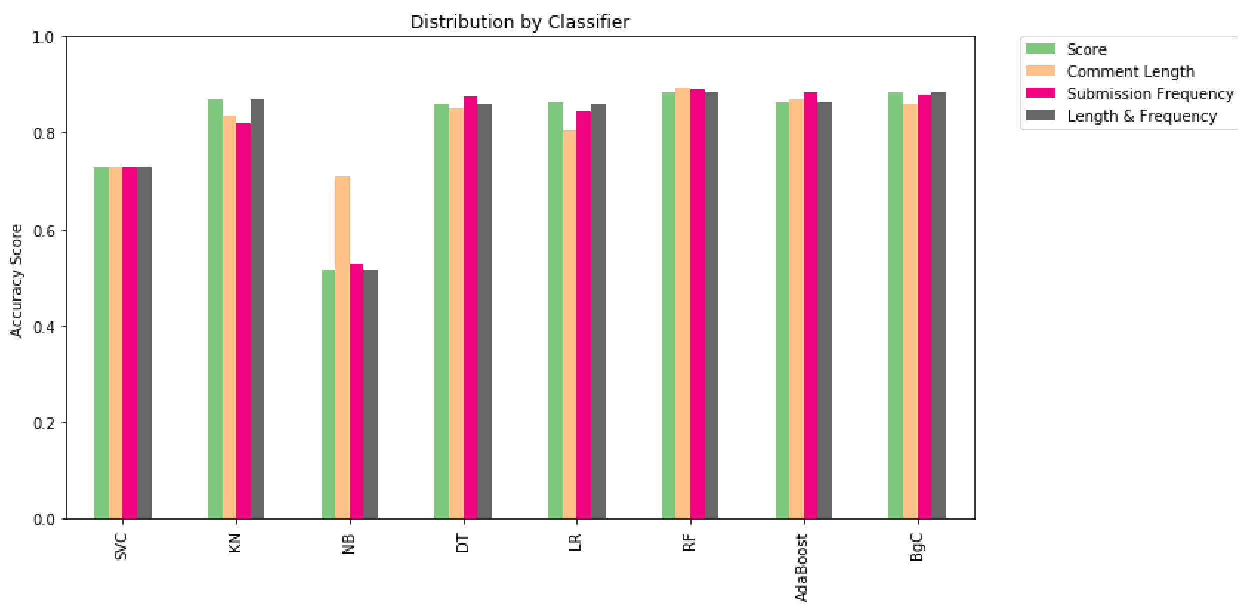
Plotted out here is a visual comparison:



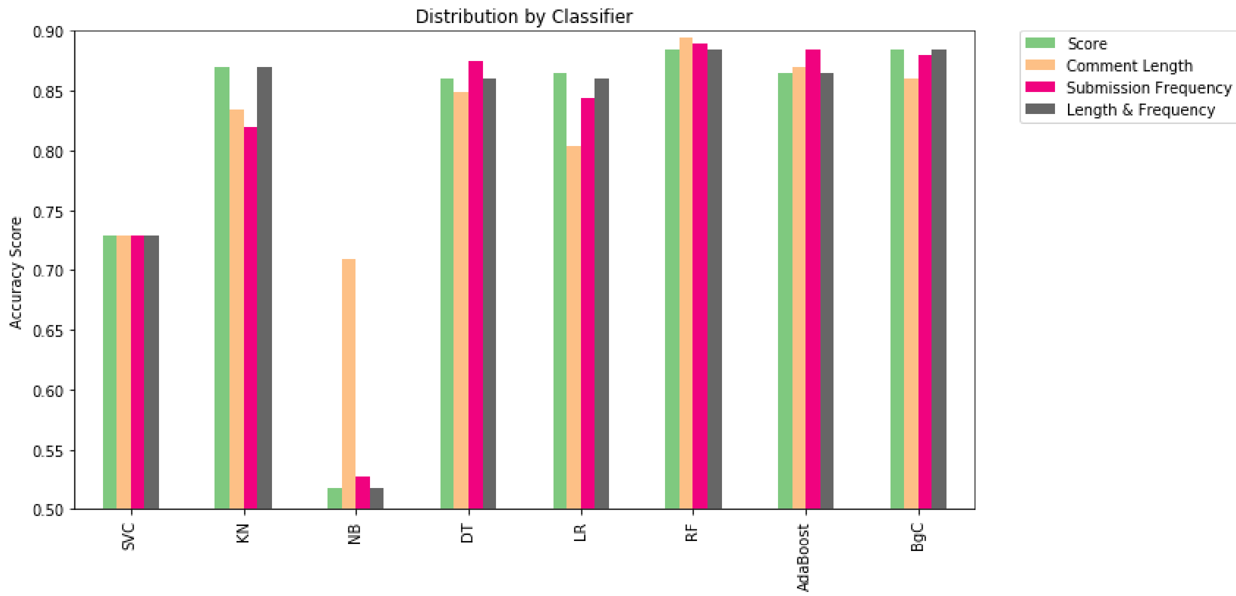
Iterating through the TFIDF with the other two features, the Random Forest Classifier gives the best score with the variations and the best result with just the TFIDF and Comment Length.

	Score	Comment Length	Submission Frequency	Length & Frequency
SVC	0.728643	0.728643	0.728643	0.728643
KN	0.869347	0.834171	0.819095	0.869347
NB	0.517588	0.708543	0.527638	0.517588
DT	0.859296	0.849246	0.874372	0.859296
LR	0.864322	0.804020	0.844221	0.859296
RF	0.884422	0.894472	0.889447	0.884422
AdaBoost	0.864322	0.869347	0.884422	0.864322
BgC	0.884422	0.859296	0.879397	0.884422

Plotted out here is a visual comparison:



For a detail comparison here is the same chart with a narrower range for the display:



The highest group of accuracy scores is coming from the Random Forest classifier. With the TFIDF, the addition of either Comment Length or Submission Frequency gives slight improvement (0.89). The addition of both features to the TCDIF yields little improvement however (0.88).

Classification Report: RFC	precision	recall	f1-score	support
0 (Real)	0.75	0.93	0.83	54
1 (Fake)	0.97	0.88	0.92	145
avg / total	0.91	0.89	0.90	199

Acknowledgements

Data: Jeff Kao
https://hackernoon.com/more-than-a-million-pro-repeal-net-neutrality-comments-were-likely-faked-e9f0e3ed36a6

Word Cloud: Nikhil Kumar Singh
https://github.com/nikhilkumarsingh/wordcloud-example/blob/7a77e97c4da135b67ad924be96269d6bb68a0fe6/mywc.py

Chorogrid Plot: lavinben88
https://plot.ly/~lavinben88/116/chorogrid-tutorial-part-2-chorogri/

Classifier Iterator: Evgeny Volkov
SMS spam detection with various classifiers