MS&E 231 HW3 Part II

Andrew (Foster) Docherty, Michael Spencer, Jorge Nam Song

I. Objective

Our objective in Part II of HW3 was to develop a model with Vowpal Wabbit that best predicts whether a tweet was sent by Donald J. Trump himself or by one of his staffers. Our definition of the "best" model in this case was the model which maximized test set AUC. Our methodology below details the process by which we developed a series of models and selected one for use in this competition.

II. Methodology

A. Splitting the data

In order to split the data, we wrote an R script to take a random 80/20 split of the provided trump_data.tsv file. That is, 20% of the data was randomly assigned to the test set while 80% of the data was randomly assigned to the training set, without regard for the time of the tweets. We did not consider the time of the tweets when splitting the data, thus our splitting may be less than optimal given there could potentially be some relationship between the content of tweets and the time in which they were tweeted. In this sense, our model would benefit from "seeing" the future before it has happened.

B. Feature engineering

Our strategy for developing a predictive model for classifying whether tweets from @realDonaldTrump were written by Donald J. Trump himself or one of his staffers was to first develop a set of features that we believed to be useful for a predictive model. The data provided included three columns: one for the "Trump"/"Staff" label, one for the date and time of day, and one with the raw text of the tweet. From this data we developed three Vowpal Wabbit namespaces, or groups of features to pipe into our training process.

The first namespace was "clock," translating the date and time of day data into a format that was useful. We hypothesized that the time of day would have some predictive power in our models, however as a circular variable (the 23rd hour jumps to the 0th hour) we needed a non-numeric way to encode this data. We tested both hourly indicator variables and 4-factor "part-of-day" indicators, taking values "night," "morning," "afternoon," or "night" depending on which 6-hour block the hour fell in, and found that the hourly indicator variables achieved a higher test AUC all other things equal. This finding is reflected in our final model.

The next namespace was "stats," a mix of summary and descriptive statistics of the tweet that we felt added valuable information to categorizing and interpreting different types of tweets. Included in these are a range of features that make sense for analyzing any given tweet, such as whether or not it is a retweet and the presence of media ("https:"), to features we felt were particularly important in analyzing Trump's tweets, such as the number of capital letters, number of hashtags, and number of @' used. While the first few features are fairly self-explanatory, we chose the last few based on Trump's typically boisterous and aggressive presence on Twitter. By analyzing the raw data and by simply following the account on Twitter, it appeared as though Donald Trump's tweets contained more capital letters, hashtags, and @'s on average, which led us to believe such stats would be good indicators of who actually sent the tweet. Prior to submitting, we also decided to include the relative length of the tweet (i.e. whether it was over or under 75 characters), and found that this significantly improved our AUC by almost .01.

The last namespace was "text," a modified version of the tweet body. Our belief ex-ante was that the text body of the tweet itself would both contain some of the richest information used to separate Trump from his staff's writing styles, as well as serve an opportunity for Vowpal Wabbit's modeling implementations to shine. The challenge with the text body was that it was not structured in a way that was conducive to modeling. There were numerous punctuations, variations in capitalizations, retweets, and other objects that would lead to trouble with processing and interpreting the text. Thus, we took a number of steps to reshape this data into an agreeable format. Rather than hit every detail, we will highlight some of the significant changes in this discussion. The first is the use of "dummy" text variables, for example using "timedummy" for any piece of text we believed to be a time of day or "mentiondummy" for any mention in the words. This step standardized components of the tweet language that held no new variation in their original format beyond indicating these "dummy" attributes. After this, we cleaned up the text by taking steps such as removing punctuation, removing extraneous spaces, and making all letters lowercase for words like "MAKE!" and "make" to have the same meaning. Note that this did not affect our counting of hashtags, @'s, or capital letters in the "stats" namespace. We did not pursue a stemming approach, both for time constraints and more importantly our belief that such variation could contribute to stylistic differences between Trump and his staff. As a last step, we decided to exclude retweets in this namespace since they were written by other people and may influence our interpretation of say Trump's style. With this last piece of the feature set complete, we moved on to exploring different modeling choices.

C. Model selection

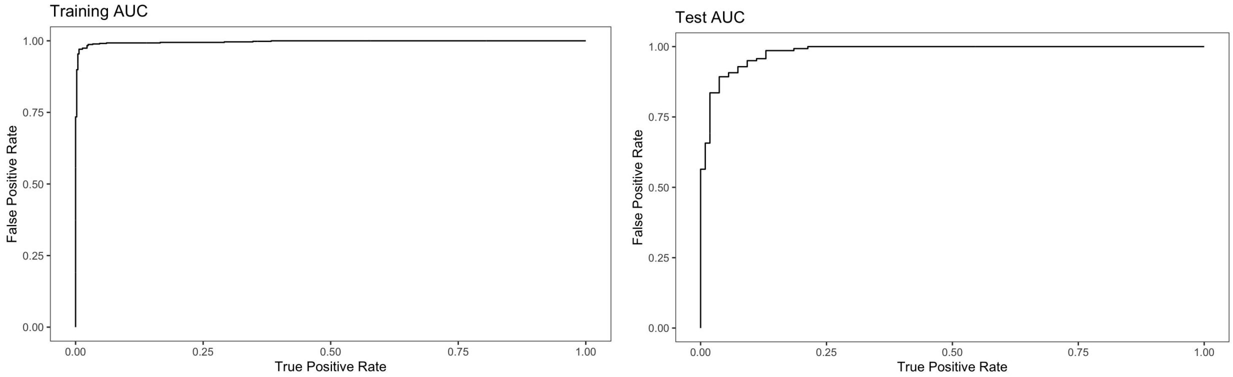
We explored a variety of modeling choices for selecting our final model. Breaking these considerations down, we first explored the "clock" namespace choice as mentioned in the section above to see which way of encoding tweet time was better in terms of test AUC. Upon testing each, we determined that the hour encoding led to a higher test AUC, likely due to the fact that it is a more granular feature and thus allowed for more flexibility in the model. By analyzing the variable importance, we see exactly that. The morning hours of 9, 10, and 11AM appeared important in determining whether the tweet was indeed sent by Trump, whereas the evening hours of 7, 8, and 9 PM were more important in determining if a staffer sent the tweet. Next, we compared using hinge loss versus logistic loss for our model, and settled on logistic loss as it performed better via test AUC. We also explored Vowpal Wabbit's "--ngram" option, which automatically creates strings of *n* words long to be used as a proxy for writing style. Given the work we did upfront to encode the text body, this option provided a clear improvement to our test AUC, and we settled on the 2-gram given empirical results. The inclusion of the text namespace allowed us to see that tweets with "trump" unironically tended to be tweeted by Trump himself. Likewise, tweets with hashtags such as "trump2016" and "imwithyou" tended to come from staffers. To check for potentially strong interaction effects in the "stats" namespace, we investigated quadratic and cubic interactions between the "stats" features as well as quadratic and cubic interactions between the "stats" features and the hourly indicators ("clock" namespace). We found there to indeed be interaction effects, with the cubic interactions within the "stats" namespace and the quadratic interactions between the "stats" and "clock" namespaces to both improve test AUC.

Our exploration of models then went further beyond the features and more to the modeling and search strategies. While Vowpal Wabbit's default modeling and search strategy was a solid choice, we also wanted to compare to other models that perform well on high-dimensional data. After exploring options such as bagging and boosting, we found strong merit to neural networks and decided to explore them further. Using the options discussed above, we varied the number of hidden units for a few of the settings explored above and found that 2 hidden units consistently performed best. Further, we explored the sensitivity of the test AUC to various L1 and L2 regularization values for neural net with 2 hidden units, and found that our AUC could be slightly improved with L1 = 0.0001 and L2 = 0.001. However, this regularized neural network was still suboptimal relative to the default Vowpal Wabbit implementation, and as such we decided to not pursue neural networks for our final model.

III. Analysis of final model

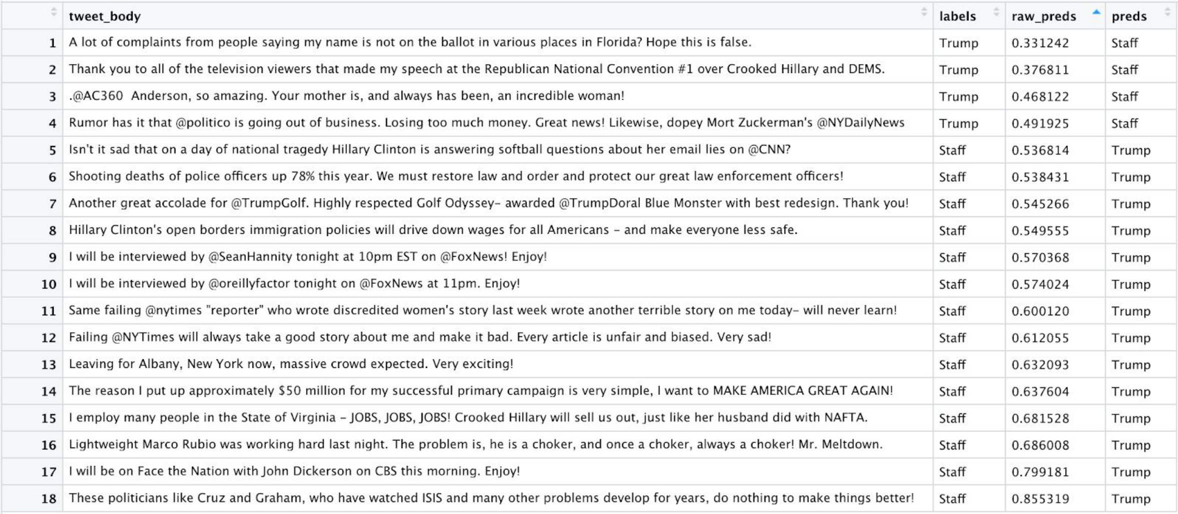
Our feature engineering and model selection investigations led us to our final choice for competition submission. This model was a standard Vowpal Wabbit implementation with a logistic loss function, cubic interactions within the "stats" namespace, quadratic interactions between the "clock" and "stats" namespaces, and 2-gram encoding of the tweet's text body. We estimated our test AUC for this model to be approximately 0.982, which we believed to be a solid benchmark for classifying "Trump" versus "Staff" tweets.

Figure 1 - Training and Test AUC of our Final Model



There were a number of reasons we felt confident about this model beyond just test AUC, however. For one, we found that the test accuracy of approximately .927 was also the best we had achieved. Further, looking at the tweets that were inaccurately predicted in Figure 2, it seems as though the model was able to cover several writing styles that were common to either Trump versus his staff, but unable to piece apart tweets such as the short, "I will be interviewed...Enjoy" style emails that both Trump and his staffers use often. We did however restrain ourselves from attempting to hard-code certain names or phrases in the tweets that were misclassified, as this after-the-fact feature engineering becomes an exercise of "memorizing" the test data. We took the stance that while certain characters and statistics played a role in the model and could be measured, the presence or interaction of words like "Hillary Clinton" would be contingent on counterfactual scenarios that we can't anticipate and thus should not be used as features (e.g. what if Trump's staff sends a tweet with "Hillary Clinton" three times after we bias the term to favor Trump?).

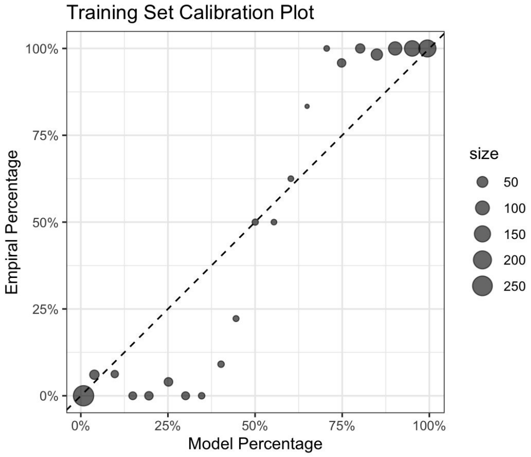
Figure 2 - Misclassified Tweets with Predicted Probabilities

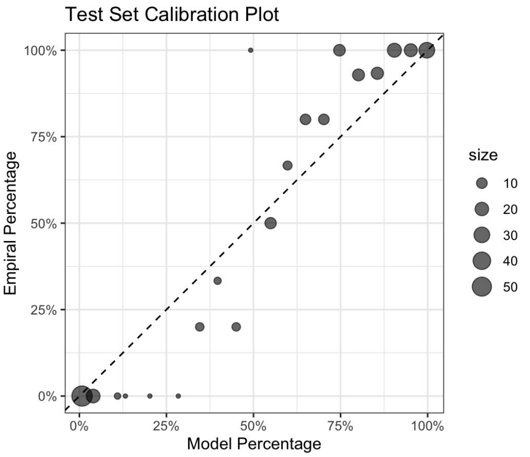


	tweet_body	labels	raw_preds	preds
1	A lot of complaints from people saying my name is not on the ballot in various places in Florida? Hope this is false.	Trump	0.331242	Staff
2	Thank you to all of the television viewers that made my speech at the Republican National Convention #1 over Crooked Hillary and DEMS.	Trump	0.376811	Staff
3	.@AC360 Anderson, so amazing. Your mother is, and always has been, an incredible woman!	Trump	0.468122	Staff
4	Rumor has it that @politico is going out of business. Losing too much money. Great news! Likewise, dopey Mort Zuckerman's @NYDailyNews	Trump	0.491925	Staff
5	Isn't it sad that on a day of national tragedy Hillary Clinton is answering softball questions about her email lies on @CNN?	Staff	0.536814	Trump
6	Shooting deaths of police officers up 78% this year. We must restore law and order and protect our great law enforcement officers!	Staff	0.538431	Trump
7	Another great accolade for @TrumpGolf. Highly respected Golf Odyssey– awarded @TrumpDoral Blue Monster with best redesign. Thank you!	Staff	0.545266	Trump
8	Hillary Clinton's open borders immigration policies will drive down wages for all Americans – and make everyone less safe.	Staff	0.549555	Trump
9	I will be interviewed by @SeanHannity tonight at 10pm EST on @FoxNews! Enjoy!	Staff	0.570368	Trump
10	I will be interviewed by @oreillyfactor tonight on @FoxNews at 11pm. Enjoy!	Staff	0.574024	Trump
11	Same failing @nytimes "reporter" who wrote discredited women's story last week wrote another terrible story on me today– will never learn!	Staff	0.600120	Trump
12	Failing @NYTimes will always take a good story about me and make it bad. Every article is unfair and biased. Very sad!	Staff	0.612055	Trump
13	Leaving for Albany, New York now, massive crowd expected. Very exciting!	Staff	0.632093	Trump
14	The reason I put up approximately $50 million for my successful primary campaign is very simple, I want to MAKE AMERICA GREAT AGAIN!	Staff	0.637604	Trump
15	I employ many people in the State of Virginia – JOBS, JOBS, JOBS! Crooked Hillary will sell us out, just like her husband did with NAFTA.	Staff	0.681528	Trump
16	Lightweight Marco Rubio was working hard last night. The problem is, he is a choker, and once a choker, always a choker! Mr. Meltdown.	Staff	0.686008	Trump
17	I will be on Face the Nation with John Dickerson on CBS this morning. Enjoy!	Staff	0.799181	Trump
18	These politicians like Cruz and Graham, who have watched ISIS and many other problems develop for years, do nothing to make things better!	Staff	0.855319	Trump

From our final model, we identified the following variables as being the most important in predicting Trump: the tweet was a retweet (RelScore: 100%), the time was 11 AM (RelScore: 55.84%), and the time was 10 AM (RelScore: 34.97%). The relationship between retweets and Trump makes sense, but its strength is shocking. It's plausible that Trump is the only one retweeting things as he scrolls through Twitter, whereas staffers were more focused on broadcasting information and original content. Nevertheless, the fact that a retweet always indicated Trump tweeted it is surprising. In addition, the relative strength of the hour variables is not surprising, but there are not very intuitive answers. Perhaps Trump's scheduling allows him to be on his phone in the late morning hours while his staffers are setting up or handling logistics. Similarly, we were able to find that the presence of a link (RelScore: -54.41%), the presence of an ampersand (RelScore: -37.41%), and the lack of a retweet (RelScore -37.21%) were important in predicting whether a Staffer tweeted.

Using the calibration plots below, we can see that our model is still not perfect. A good model would be aligned directly with the diagonal, however when our model is not extremely sure of its prediction, it either overestimates the likelihood a staffer tweeted or underestimates the likelihood Trump himself tweeted. Future steps would include trying to remedy this and either generate more confident predictions or explore an alternative model to logistic regression.





IV. Limitations

There are a few limitations to our approach in selecting a final model that we would like to acknowledge. First, as with most analyses, we want to note that we did not explore the full breadth of modeling options, let alone those available in Vowpal Wabbit. We solely focused on the modeling choices and tuning parameters that we hypothesized would serve us well in this investigation. If one were interested in further research into classifying Trump tweets, we would recommend exploring other modeling options. Another caveat to our approach relates to our feature engineering: we did not explore any NLP or other sentiment analysis techniques to try to extract further second-order information out of the text body. As with the modeling choices, we would recommend experimenting with these modern tools in follow-up analyses. In the search for the final model, we also want to note that we did not employ any automated approaches such as grid-search for parameters to maximize test AUC. Our iterative exploration of possible combinations represents only a subset of the potential paths, and final models, that could have been explored. Finally, we want to acknowledge the limitations of calibration plots and variable importance measures. While these are excellent tools for reason about the predictions a model is making, it is impossible to completely describe the effect of an individual feature or modeling change in such a high-dimensional and high-covariance setup.