**(1) Kaggle team name:**

想了想还是南ことり俺の嫁吧

**(2) Score on the public leaderboard:**

0.85000

**(3) The preprocessing techniques used:**

Here I used Naive Bayers as model, to generate the feature for classification. I use the text content of each twitter to exact features. For each twitter, split them into words and for each word, hash them into a certain place of a list. Finally take the list as features of input training data.

**(4) What you learned while exploring the data:**

The twitter text itself is a good feature to classify Trump himself and his staff as you can easily find the difference between these texts. Whether it’s a retweet is also a good feature but unfortunately I found that in the test data all the tweets are not retweet so I have to throw this feature out. For other features, like the favorite count or retweet count, I tried to classify them but the result turned out not good. The time of sending may be a feature to improve the correctness but I cannot find a good way to evaluate this feature.

**(5) How you selected your model**

Just by trying, at first I use a SVM to classify these features but it turns to work well on training data but really bad on validation data. Compared to it, kernel SVM is a little bit better, but the validation error is still around 30%. At first I use the favorite count, id, and retweet count and so on as input feature, the result is also bad. However while I extract features for SVM I was thinking about using the features for SVM to train a Naive Bayers model, it turns out a good way and successfully reduces the error rate. Then, to reduce the error rate, I choose to use a k-fold method.

**(6) What features you extracted**

The text content is the only feature. Hash each twitter into a list as the features for training. The size of hashing list is decided by a random number. By trying lots of times, choose the one with the least average validation error rate as the final submission.

Besides, there are several features to help classify in the content. For example, tweet with a URL link is more likely from an iphone while tweet with “Crooked Hillary Clinton” is more likely from android. Beside these, I also find some emotion-related words, like “angry” or “crazy”, are more likely from android, so taking these factors into consideration could largely improve the accuracy rate.