NLP Assignment 3 Report

Einat Shusterman

I. Experiment Details

A. Overview

The aim of the assignment was to distinguish tweets written by Donald Trump from tweets written by his assistants. I analyzed the data and used different text processing and classification algorithms to find the best model for the job.

B. Data Exploration

I wanted to analyze Trump tweets semantic nature vs his assistants tweets semantic nature. for that I created for each tweet in the training data Tweet Embedding vector which was the average vector of all the words vectors inside the tweet. Then I used K-mean clustering to identify different clusters in the space, Also I used T-sne algorithm for dimensionality reduction. Fig 1 shows the results.

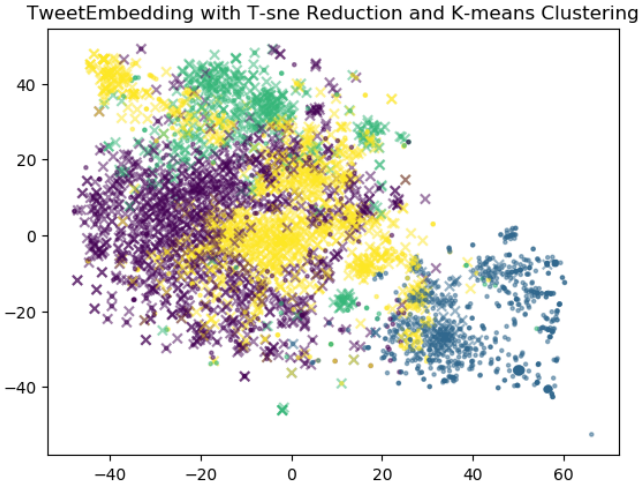

Fig. 1. Trump Tweets are shown with 'x' shape and Trump assistants tweets with 'o' shape

In Table I is shown the label percent on each cluster, some clusters like Purple and Yellow captured well Trump label and cluster Blue capture half of Trump assistants label, I found that very interesting since it reasonable that Trump and his assistants will express the same semantics (since they talk on the same events/opinions) but from this analyze we can see that half of the a assistants semantics is different than Trump's.

	Purple	Blue	Yellow	Green
Trump Percent	40	4	40	14
Not Trump Percent	19	54	17	9

TABLE I

Clustering Result - Label Percent for each Cluster

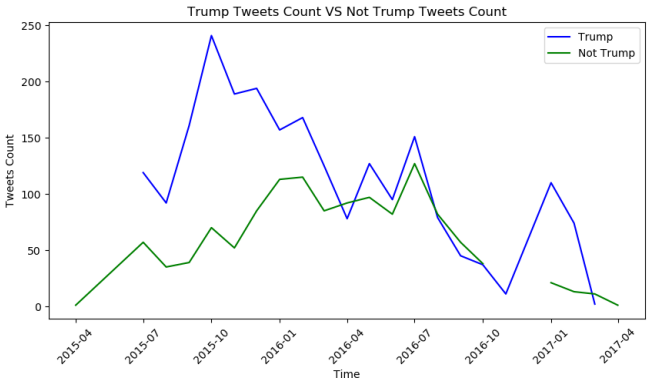

Fig. 2. Tweets distribution over time

furthermore, I wanted to see how the Tweets distributed over time, on Fig 2 it's clearly to see that during the 2016 United States presidential election Trump reduced his activity on Twitter, which can verify the claim that Trump was kept away from his Twitter account during the campaign. After Trump win the election, on 8/11/2016 we can see increase in Trump tweets count. Also from Fig 3 we can see Trump sentiments on his tweets over time, especially before, during and after his campaign. We can infer why Trump was kept away from his Twitter account (his positive tweets reduced and his negative tweets increased). Also on Fig 4 we can see Trump assistants sentiments from tweets which for the most part stays positive

which make sense from marketing point of view.

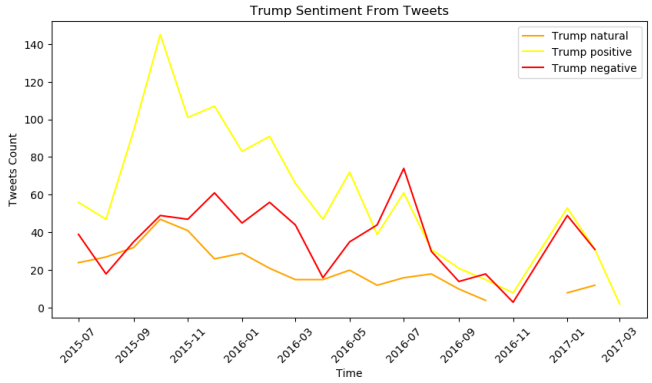

Fig. 3. Trump Tweets Sentiment Distribution Over Time

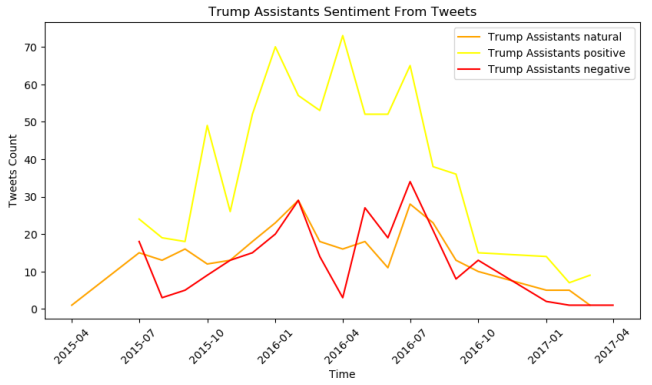

Fig. 4. Trump Assistants Tweets Sentiment Distribution Over Time

C. Classification Models

1) SVM
2) Logistic Regression
3) FeedForward Neural Netwrok
4) Random Forest

D. Data preprocess

I Tagged tweet as written by Trump only if tweet device was Android and tweet wasn't a re-tweet (to remove noise from real Trump tweets) and tweet user was - 'realDonaldTrump'. I sorted the data by tweet time and remove tweets with Nan as time. The tweet text was preprocessed differently for each text feature, I will explain each preprocess technique on next subsection.

E. Features Used

- **TF-IDF** : Since I wanted the model to learn the 'Donald Trump language model I had to use features that present the away he express. I used the TfidfVectorizer from sklearn and preprocess the tweet text with lower the tokens, removed punctuation and removed stop words. I took the most 9000 best n-grams, for unigram, bigram and trigram.
- **Sentiments** : I tried to capture the author sentiments from the tweet text, so I used sentiment analyzer from nltk. from preprocess point of view I removed web links and lower the text (I kept stop words and punctuation because its important features to extract sentiment)
- **K-means Clustering On Word Embedding** : I tried to capture the semantic from the tweet text, since Donald Trump and his assistants tweet on similar topics The word Embedding alone didn't helped much for classification so I used K-means clustering with 4 clusters. I used the google news pretrained words Embedding with vector size 300 (I kept only the words vectors from the trained Data) and each Tweet Embedding vector was an average of all the words Embedding in the tweet text. The preprocess stage was the same as on TF-IDF.
- **Tweet Weekday** :I tried to capture Donald Trump habits and schedule, therefore the days of the week were he is more active.
- **Tweet Hour** : I used that feature from the same reasons mentioned above.
- **Tweet Time As Relative To Election Day** : I tried to distinguish between tweets before the United States election (8-11-2016) and after. I thought that the text tweets will be much differ before the election and after.

II. Algorithms Comparison

I split the train data to train set and test set with TimeSeriesSplit from sklearn because I wanted the evaluation scores will be as close to reality as it can be. Since I learn on past data and predict on present/future data.

A. SVM

Since SVM algorithem such 'rbf' and 'linear' didn't worked well with high dimension features Data, I reduced the features dimension with SVD algorithm to 400 dimensions. for 'sigmoid' I didn't

reduce the features dimension. I run experiment on different kernel types. Table II shows the results.

	Linear	**sigmoid**	**RBF**
Auc	**0.83**	0.74	0.8
Precision	**0.86**	0.76	0.86
Recall	**0.85**	0.76	0.83
F1-score	**0.85**	0.76	0.82

TABLE II

SVM KERNEL ALGORITHMS EXPERIMENTED

B. Logistic Regression

Logistic regression handle well with features dimension so no dimentionality reduction needed. I used 'liblinear' solver since I read that it the most suits solver for small Datasets. The results were: Auc: 0.82 , Precision: 0.87 , Recall: 0.85 , F1-Score: 0.84.

C. Random Forest

The results were: Auc: 0.83 , Precision: 0.87 , Recall: 0.85 , F1-Score: 0.84.

D. FeedForward Neural Netwrok

I used FNN with cross-Entropy loss function and 2 hidden layers, each with 200 nodes, learning rate of 0.001 and batch size of 256. The results were: Auc: 0.8 , Precision: 0.81 , Recall: 0.81 , F1-Score: 0.8 .

III. CONCLUSIONS

from the given result, I decided to use the SVC linear since it had the best results.