



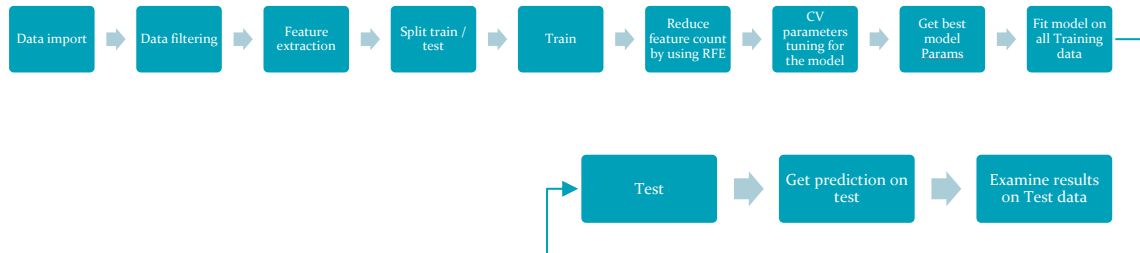
## NLP Task 2

DOCUMENT CLASSIFICATION, CLUSTERING AND TOPIC MODELS

## Task 1- Classification: Who Controls this Account

In this task we will attempt to cluster unlabeled tweets posted by Mr trump (or by his entourage), and extract some insights about the tweets and it originator

The process taken in this part:



input data

Set of ~3500 tweets given between April 2015 to April 2017, given in the following format:

UID	User	Content	date	source
Tweet key id	Poster user name	The tweet content	Post date and time	The device posted the tweet
Normal number	String	String	Date-time	string

### data filtering

Prior to the classification task, we filtered the given data by the following rules:

UID	User	Content	date	source
Na	Leave only user: "realDonaldTrump"	Na	Na	Leave only: "Iphone" or "android"

After this step data rows are \*enter amount\*

### Tweet cleaning

In order to prepare data to enter to classifier, the tweet content gone through the process of string cleaning:

1. Remove of stop words
2. Word stemming

### Feature extraction

Since data is given mainly in string and date-time format, I will attempt to extract numerical attributes prior to the classification process

Output attribute	Input	Description	Output format	Number of attributes	examples
Len	Tweet content	Length of string	Int	1	144,84,39
capital_letters	Tweet content	Count the number of capital letters in tweet	Int	1	2,3,20
capital_percent	Tweet content	percentage of capital letters to total chars in tweets	float	1	0.2,0.53

links	Tweet content	Count the number of links in tweet	Int	1	1,2,3
Part of day	date-time	One hot of the hour divided into parts of day: morning, evening...	Boolean	4	"night","morning",
Weekend	date-time	One hot of the day of week divided into week	Boolean	2	"weekend" "daywork"
num_hashtags	Tweet content	Extract he number of hashtags in the tweet	Int	1	1,2,3
num_mentions	Tweet content	Extract he number of user mentioning in the tweet	Int	1	1,2,3
TF-IDF	Tweet content	Processing the string through TF-IDF process	Matrix of Float	Number of words	0.3,0.5..
Hour	date-time	Given the hour of the tweet	Int	1	19,20,04

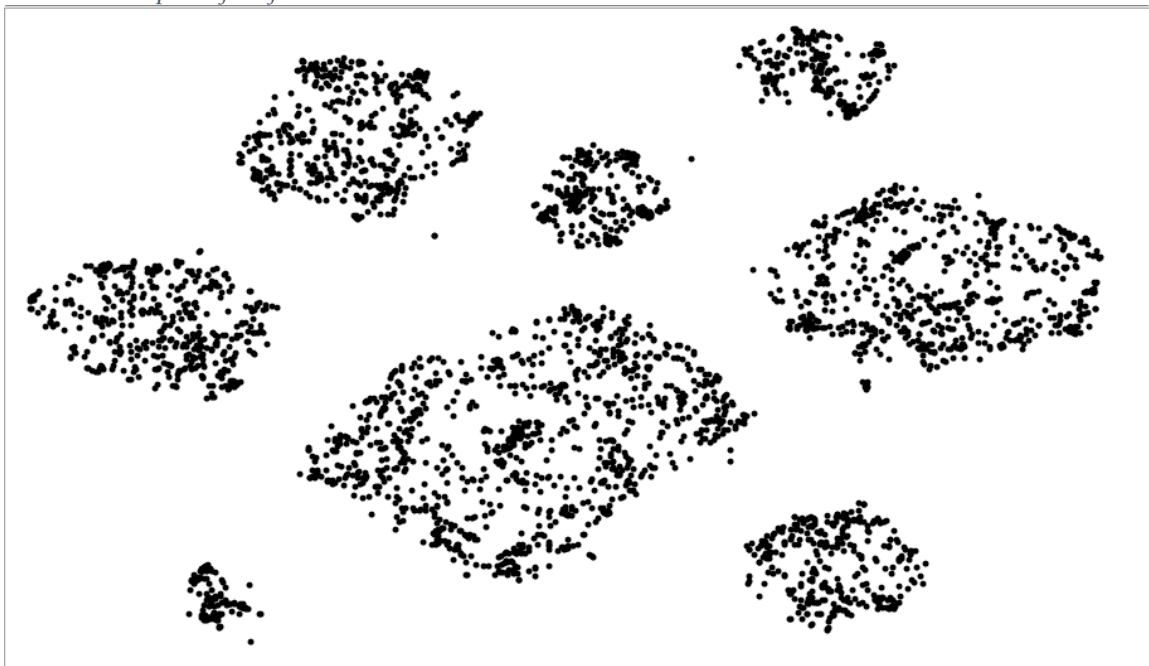
### Data presentation

We have presented the data in a manner of matrix of values, given by the pre-processing described above, in total 1037 total features where only 19 where manually extracted and rest are given by TF-IDF

### General data presentation:

In this part I took all of the featured extracted and plotted on TSNE scatter plot

Table 1 TNSE plot of all features extracted



Features extracted do carry meaning and with the right model we can hopefully provide good classification for detecting “iPhone” or “android” device,

### Cross validation parameter optimization:

In the process of finding the best parameters to fit the models, first the data set was split into 80% train set and 20% test set,

The train test was used for configuration optimization

While test was used for assessing the model performance

the following configurations was used:

	Features to use by RFE	Kernel	Gamma	C (CSV)	C (LR)
SVC	[5, 10, 20, 40, 70]	rbf	[1e-3, 1e-4]	[1, 10, 100, 1000] [1, 10, 100, 1000]	[0.001, 0.01, 0.1, 1, 10, 100, 1000]
SVC		linear			
Logistic regression		NA			

I’ve used Cross Validation of 5 folds

### Features selection:

I’ve inputted a mass number of 1031 feature into the RFE process in order to select the top features to use for the classification process, reviewing the top features in each section:

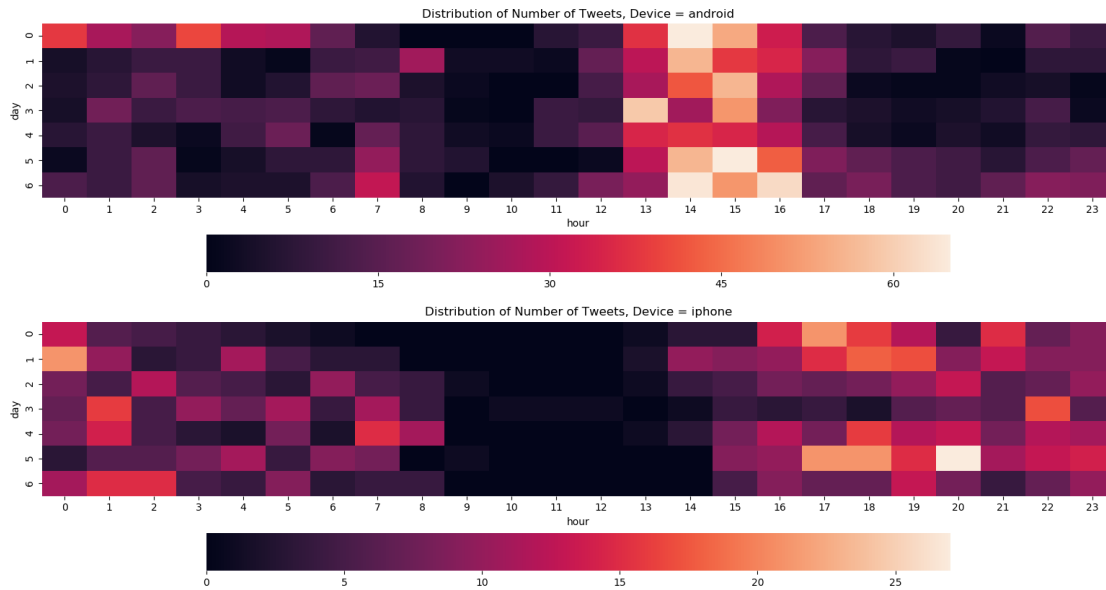
Table 2 top features extracted by RFE

5	10	20	40	70
links	links	links	links	capital_percent
num_hashtags	num_hashtags	length	length	links
num_mentions	num_mentions	num_hashtags	num_hashtags	length
l_realdonaldtrump	l_amp	num_mentions	num_mentions	num_hashtags
l_rt	l_donald	l_amp	morning	num_mentions
	l_donaldtrump	l_copi	afternoon	morning
	l_join	l_donald	l_amp	afternoon
	l_realdonaldtrump	l_donaldtrump	l_big	l_americafirst
	l_rt	l_everybodi	l_care	l_amp
	l_thank	l_fail	l_copi	l_appreci
		l_fox	l_donald	l_best
		l_greatli	l_donaldtrump	l_big
		l_join	l_dopey	l_bush

		<a href="#">l_pme</a>	<a href="#">l_ericbol</a>	<a href="#">l_care</a>
		<a href="#">l_realdonaldtrump</a>	<a href="#">l_everybodi</a>	<a href="#">l_congratul</a>

Note that the list is ordered, meaning in features =5 *links* was more useful then *num\_hashtags*  
What's clearly visible, is the fact that manually extracted features where much more useful than  
The TD-IDF features, this is also was noticeable in the part 2 of this assignment where using only  
TF-IDF features was not as sacksful as classification in this segment  
To view visually the contribution of the *time* for example, the plot of time of the tweets can be seen  
in the table bellow

*Table 3 time of tweet heat map, by count*



Its possible to see that if we will use this information correctly we can use it to our advantage in  
classifying the tweets

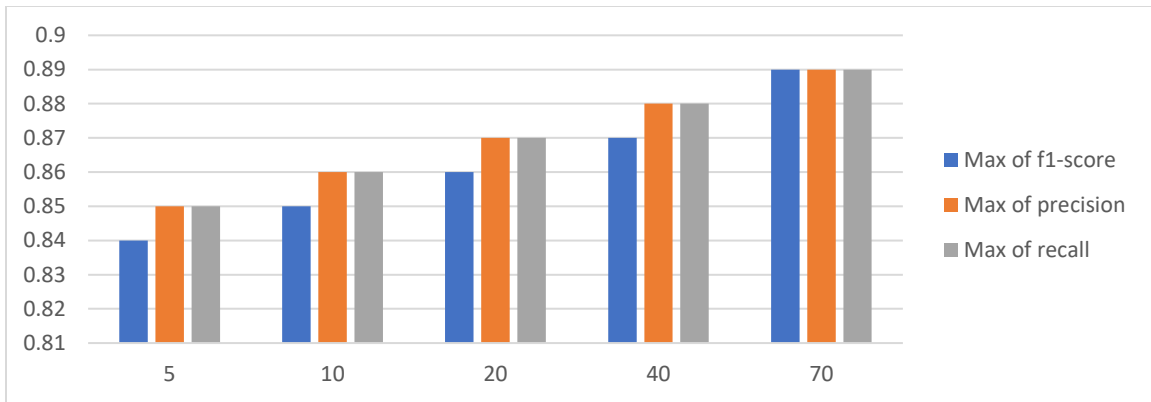
### **Model tuning and result:**

I've used the Grid - search function to preform optimization on the correct set,  
the model was used to extract best

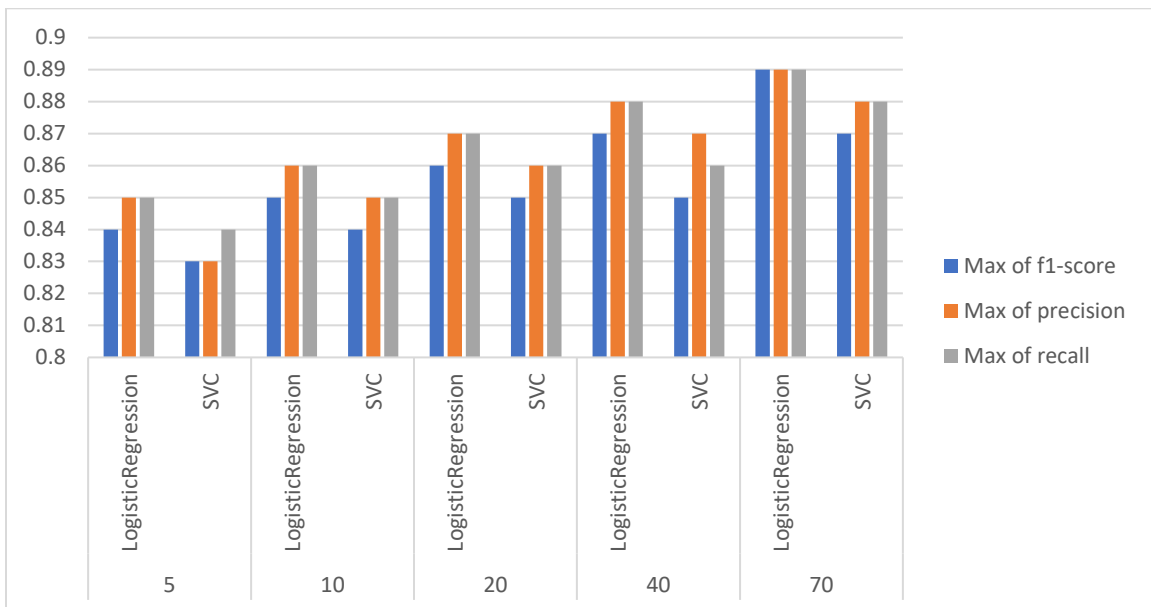
### **Features effect on model**

To examine the effect of number of features on both model we look at the precision / recall / fi-  
score in different set of features

*Table 4 max score, for different feature selection*



One can notice, that feature selection has effect on the classification performance, while its not big, it does contribute rise of 8% to all methods of comparison, if we plot it between the 2 models, we will notice that SVM suffers more from lack of features:

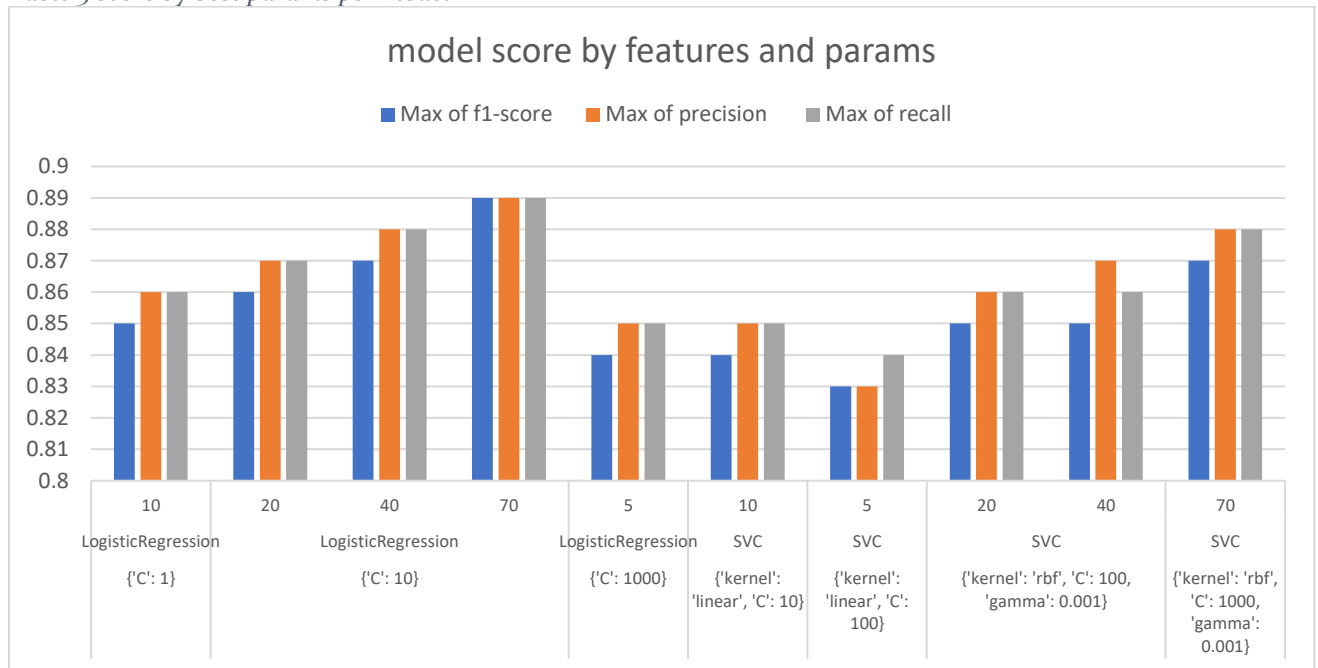


From this part we can conclude that bigger set of features (until 70) are better performing than lower number of features and both models are using the additional features for classification

### Results analysis

To answer the question what is the right model and the right prams we look at the following plot

Table 5 score by best params per model



This plot summarizes all the runs preforms and plot the TEST set on the best fitted model, what insights we can see in the plot

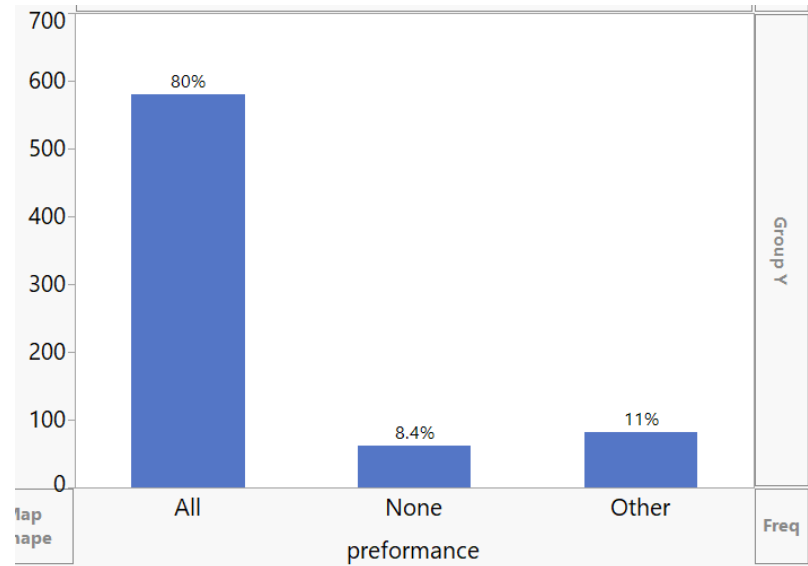
1. The best model and parameters is Logistic regression with 70 features with C of 10
2. If we will want a more stable model, we can use the LG model with 10 features with C=1, this will give us better regularization results
3. As expected LG with very high value C was preforming good on trainset (was chosen as best parameter) but poorly on the test set – very important to test on separated set
4. As seen before SVC effected by number of features used
5. SVC with kernel linear performed poorly in compare to RBF
6. RBF best preformed with many features with small value of GAMMA, this means that the algorithm wasn't in a need to break the data into small clusters, but in general big clusters in hyperspace was found – contributing to the stability and repeatability of the model, the parameter C was high (100,1000) meaning very intolerance to the “misclassification” in the train stage.
7. Why F1 score is lower? Well F1 is given to certain class, when the data is skewed, and there is one class with small amount of instances, it gets same weight as a big class, in our test data we had ~500 android and ~200 iPhone, every mistake in “iPhone” carry more weight than other mistakes

### *Difference between the models:*

The test set was 20% which is 723 instances, when comparing the 10 different configurations from the plot above, we see the following information

That all 10 models managed to correctly classify 80% of the data, and None managed to classify 8.4% of the data Leaving the changes between the models to a small number of 11%

This means that all algorithm performs relatively well, and the differences are from the 11 % of the data





## Task 2- Topic Modeling and Clustering

In this part we deal with topic modeling algorithm and clustering in order to get insights on 2 sets of data, the first, as part one; trump tweets, the second is comments given from the American public to the FCC concerning net neutrality

### Data filtering

Trump tweet: as part 1

FCC comments: as dataset was big, 1% of the data was loaded

### Text cleaning

Trump tweet: as part 1

FCC comments: stemming, and stop-words removal

### Feature extraction

For both data sets TF-IDF was used to depict the data in a vector of numerical values

In this sections no manual feature extraction was used since I wanted to compare same algorithm on a “generic” text and not twitter specific

### Results:

#### FCC NET NEUTRALITY COMMENTS

##### **Number of topics:**

The first step in processing the data, LDA run on the TF-IDF dataset using big amount of topics (1000) and in order to inspect the main topics in the dataset, I run clustering with various parameters and algorithms; Kmeans, DBscan and TSNE for visual inspections:

as can be seen from figure 1, the topics are loosely connected and cannot be easily clustered,

All besides 2 very clear clusters (straight lines) which K-means and DB scan were not able to cluster, but TSNE shows them in a clear manner, visual inspection of these topics shows that they are very similar and one is on a very positive nature (words like: good, knowledge, protect) and one on negative nature (words including: oppose, protect, and rules)

this steps focus our next steps on a smaller number of topics ranging from 2 to 20

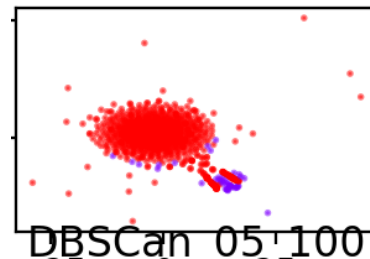


Figure 1

This demonstrates clearly the “long tail” phenomena, while there are many possible topics to produce by the given algorithm – only few are the most common ones

### Number of topics wordcloud:

In the process of unraveling the data, I run LDA and topic modeling on the given data, the results can be seen in figure 2 where bigger words are the more prominent words in each topic:

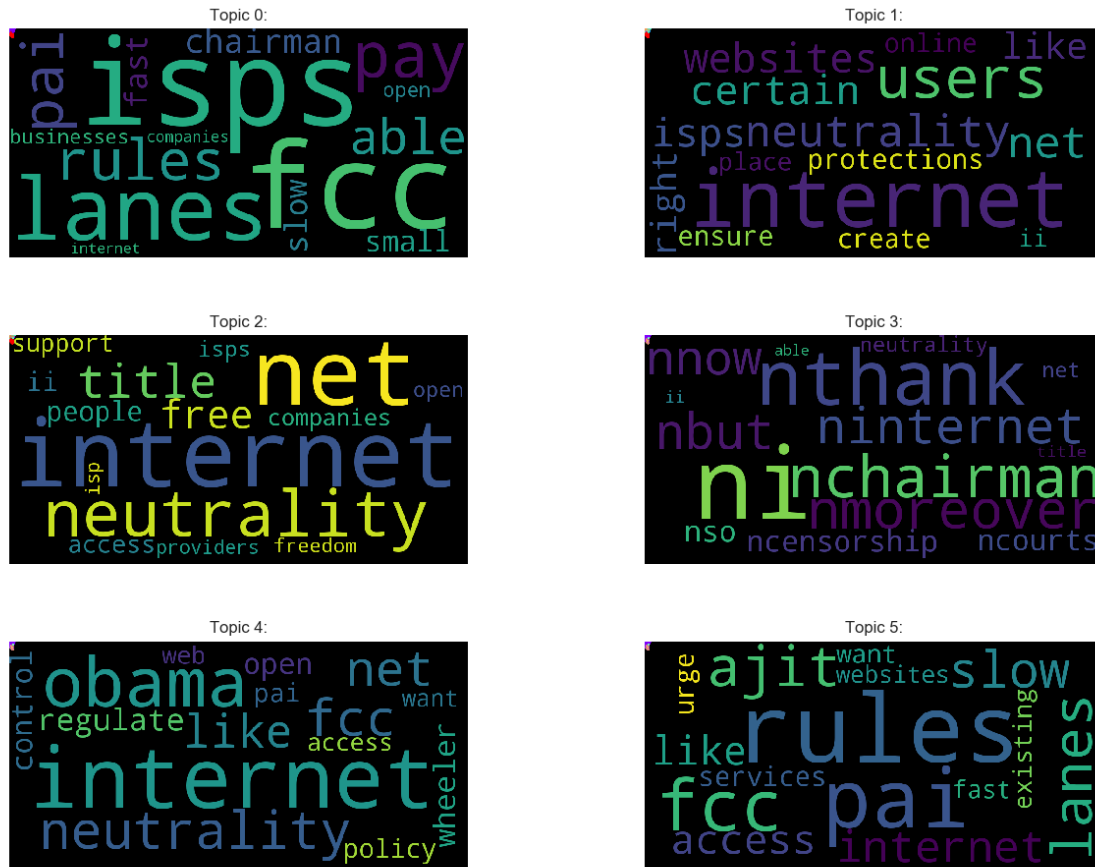


Figure 2 topics top words

Observing the data, we can see for example that topic 2 is positive about maintaining the net neutrality, other topics are forward regulation and governance policing this issue (i.e. topic 4), this gives somewhat of a confidence that topics are also correlate with the user stand in this topic

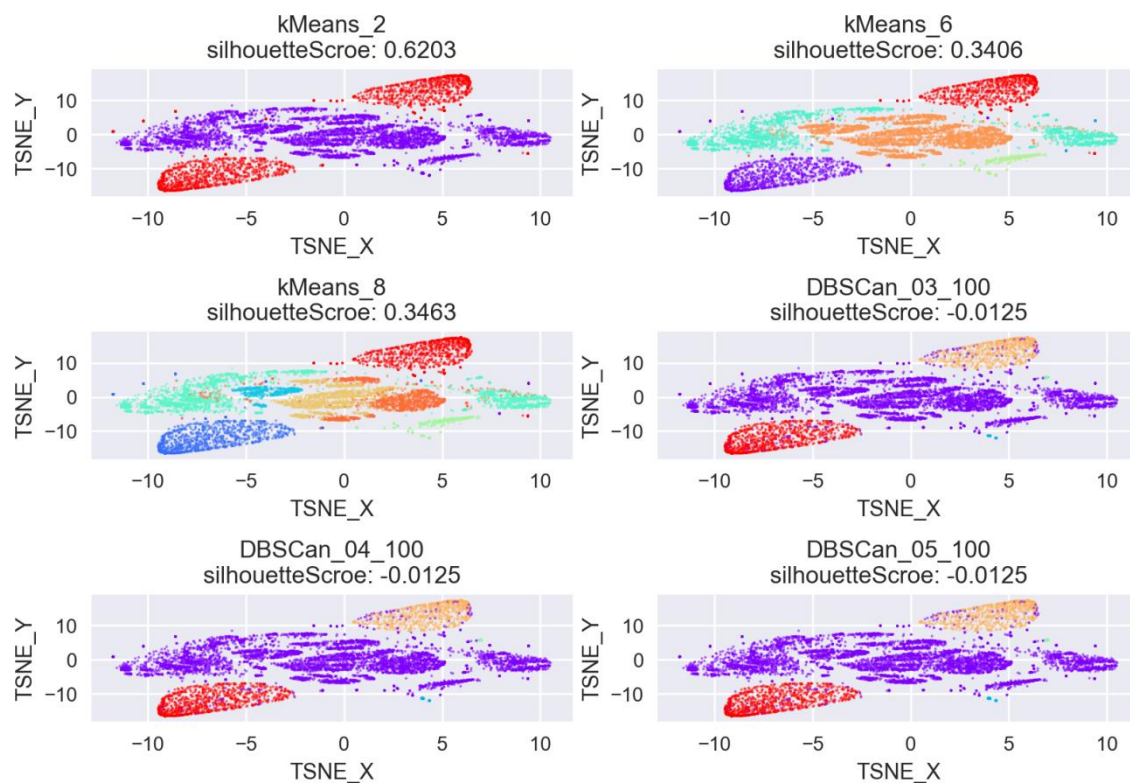
### Sentences clustering:

In this part we will examine the clustering of the input sentences by their topics affiliations, reminder: a sentence input eventually will be presented by number of percentage of his affiliation to certain topics:

“I support the net freedom” → 20% topic A, 50% topic B, 10% topic C .....

We will put this vector per sentence into a clustering algorithm and examine its outcome

First step we will use the following configurations:  
 Number of words for TFIDF:1000  
 Number of topics: 6



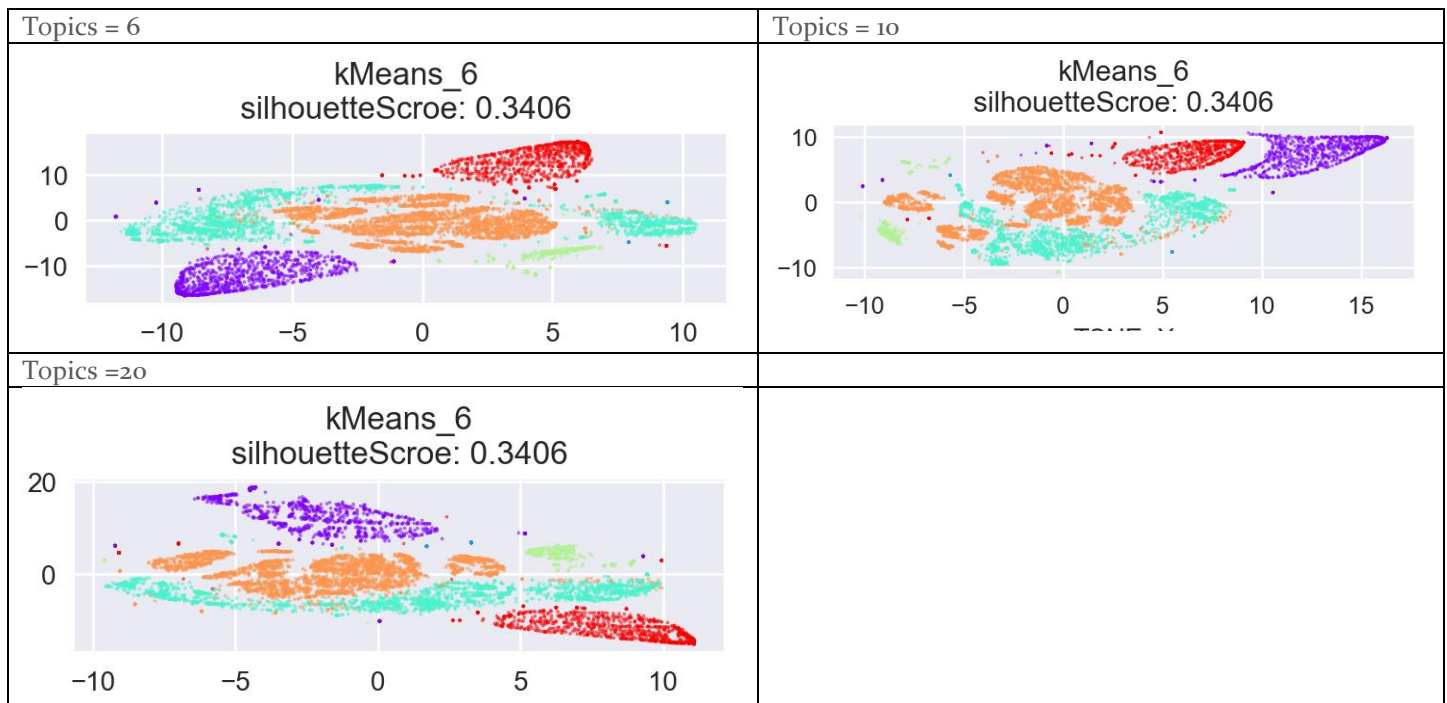
*Figure 3 sentences clustering*

In figure 3 one can see the outcome of 4 algorithms, all sentences plotted in 2d manner after dimensionality reduction using TSNE, the colors are the clustering given by Kmeans + DBscan and the silhouette score of each clustering, it is noticeable that TSNE present in a visual manner the sentences clusters, which in some cases are cluster within a cluster (the center shape) where there is one overhaul connecting point between the sentences, but even within this cluster, there is a finer clustering to sub-clustering, second, Kmeans outperforms the DBscan clustering, this we understand also by the silhouette score (DB scan is negative) and also visually seeing that the total number of clusters found is not sufficient, third we can observed the silhouette score of all the clustering algorithm, judging by the silhouette score, Kmeans with 6 clusters give the “lowest” (better) score with the minimal number of clusters- there for we will suggest using K=6 to maximize the algorithm performance

**Number of topics in combination of clusters found:**

To give judgment on the relations between number of clusters and number of topics (both should be given in kMeans and LDA)

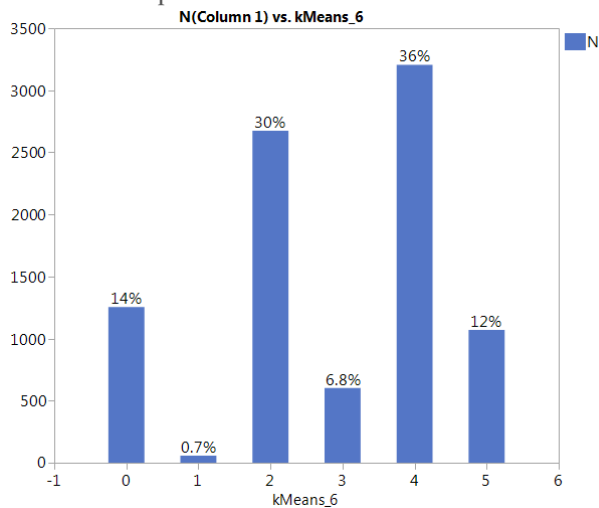
Each drawing is a plot of the TSNE out from a given TF-IDF matrix, produced by different number of topics,



Form the drawing above, we see that with different set of topics [6,10,20] still the actual clusters founds is between 4-6 – this remain visually clear and also in all cases, K=6 had the best silhouette score

In addition in we plot the binning of clusters by count

The top 3 clusters are 80% of the data, showed that even we can lower the number of clusters and number of topics and still be able to cluster most of the data.



### **Bot generated comments**

To examine the questions of BOT generated comments we can use a finer clustering and examine each clusters example of sentences,

Quick overview of cluster no5, present the following sentences:

Although they are NOT the same text, and signed by “different” people, this cluster is a prime example of a very suspicious behavior, in the cluster example bellow, they are FORWARD the net neutrality

Column 1	0
26	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
27	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
28	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
29	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
30	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
31	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
32	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
33	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
34	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
35	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
36	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
37	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
38	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
39	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
40	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
41	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
43	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
44	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
45	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
46	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
299	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
385	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
462	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
463	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
493	In all honesty, there is no way that this would benefit the citizens of this country. Please conside...
505	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
738	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
891	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
897	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
903	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
907	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
927	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
929	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
938	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...
939	The FCC Open Internet Rules (net neutrality rules) are extremely important to me. I urge you to ...

Figure 4 cluster 5 sentences example

## TRUMP TWEETS

### *Number of topics:*

Like FCC we will first try to cluster the topics, Running the application with 500 topics, plotting in onto 2D and cluster shows no evidence clustering, this can also be seen by the high silhouette score, lowering the topics number to 100 yielded the same noisy results.

So we will continue to test the right number of topics in other manners

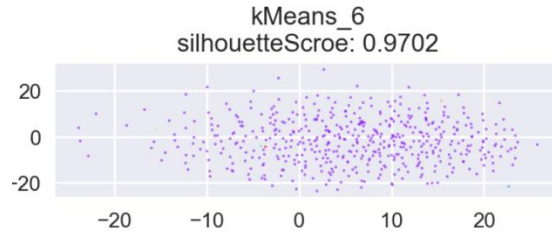
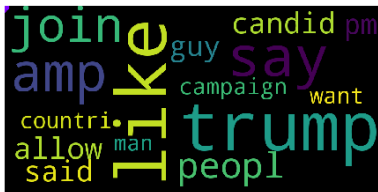


Figure 5 topic clustering

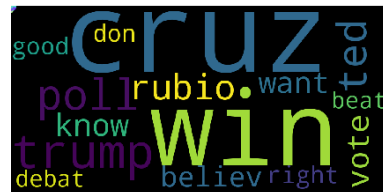
### *Topics analysis:*

Reviewing the topics visually:

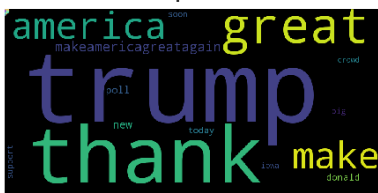
Topic 0:



Topic 1:



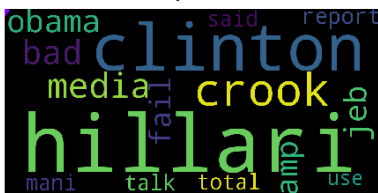
Topic 2:



Topic 3:



Topic 4:



Topic 5:

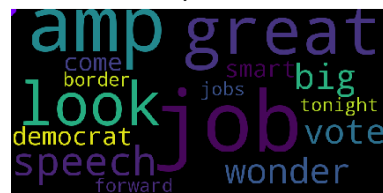
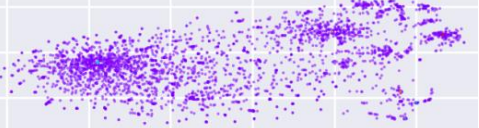
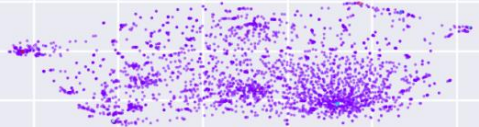



Figure 6 topics top words

Looking at the topics, we can see for example topic 1: centered around the elections within the republic party (words like: cruz, rubio..) topic 2 is about trump and making America great (again?!) Topic 4 is about Hilton and Obama... this makes a lot of sense, and give confidence that topics found reflect the actual topics trumps discuss about.

### *Sentences clustering:*

As seen at the first part, this data is much more diverse than FCC and we might plotting the tweets clustering, in different configuration proves that is true:

TSNE plot	Number of topics	Number of top words
	50	1000
	10	1000
	6	1000

We can see that data is much more diverse and hard to cluster, at least by TSNE visualization  
After running few numbers of topics, I've decided to go on with 6 topics in the topic model process

NOTE: this TSNE is very much different from the TSNE we had in part1 where trumps tweets made clear clusters, after investigation, we see that TF-IDF carry little influence on the data presentation, and the more prominent information is from the manually extracted features extracted in part 1, in this part I did not extracted additional manual features to align with the FCC data which did not carry time information or links / hashtags

### *Clustering the sentences:*

Clustering the sentences given by their affiliation to topics, showed that DBscan wasn't successful in finding any cluster, and K-means presented uneven clusters and



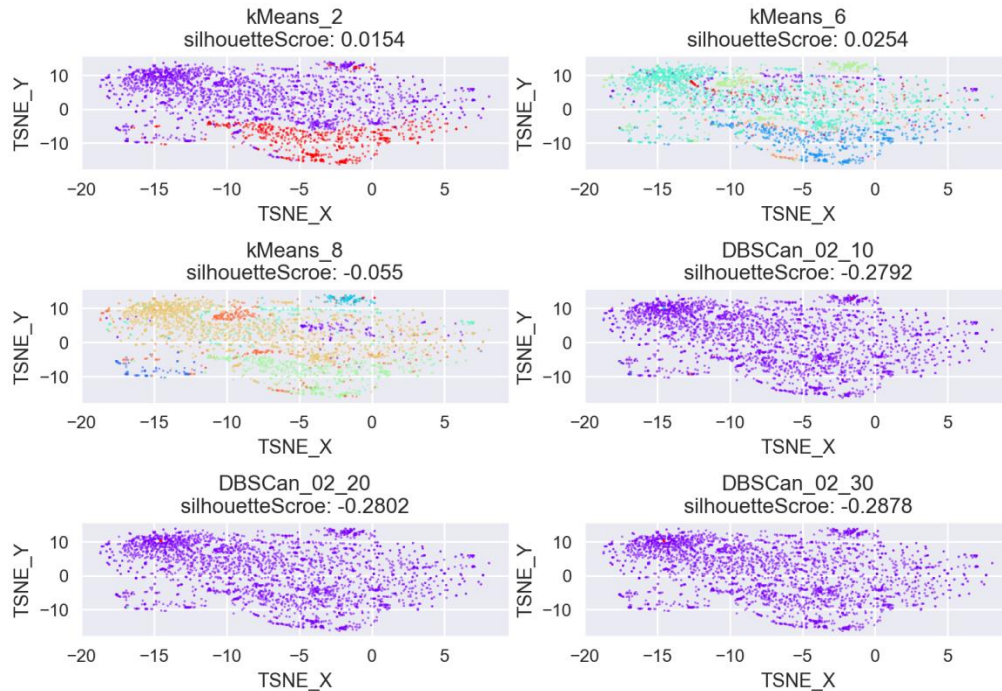


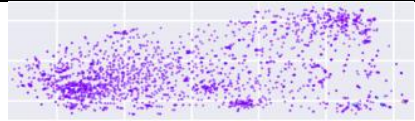

Figure 7 TSNE of topic model 6

We do have somewhat of contradiction here, where TSNE seems spares but silhouette score is quite low - in that case, I will inspect cluster manually, and try to assess its compactness:

Cluster	Example sentences
Cluster 4	wow, poll number announc gone roof! neg poll fake news, like cnn big poll announc morn face nation
Cluster 5	Not clear affiliation
Cluster 3	make america great make america safe great again thank america - great

And it is seems that clusters of the sentence do have somewhat of a theme, making me trust the silhouette score

Since, topic model on all “RealTrump” data wasn’t successful, I will test the same method on the data, devied into 2, device=iphone and device=android

Android	Iphone
	

It is can be seen that “android” (presumably trump) is more coherent and talks about the same topics, while “iPhone” (presumably his staff) are more diverse.

