भारतीय प्रौद्योगिकी संस्थान दिल्ली
Indian Institute of Technology Delhi

# Mid Term Report of MTD702 Project 2

## Computational approaches for Early Detection of Fake News


Supervisor:

Prof. Dr. Niladri Chatterjee

Submitted by:

Abhishek Bansal

2019MAS7058

Department of Mathematics,

Indian Institute of Technology Delhi

# Contents

# ABSTRACT

The demand for fake news detectors is increasing due to the threat posed by fake news sources to public sentiments and democracy with the rapid growth of electronic and social media. News agencies often use fake news to create public opinion in order to achieve their goal, political, commercial etc. In this project, we intend to develop a Machine learning based system that could help us to detect fake news. Our intended domain of application is election related data gathered from Kaggle. We'll study unigram, bigram and trigram word frequency distribution and analyze whether they have significant distribution in detection of fake news articles.


**Keywords:** Unigram, Bigram, Trigram, Word Frequency.

# 1. Introduction

In the modern era, it is a common observation that people used to spent most of their time online on social media platforms. People tend to incline more towards social media rather than traditional news platforms. With the onset of past decade, it has been seen that our generation has become increasingly dependent on social media and they do not actually cross verify the facts and figures of what is happening in the world. We have really limited our domains to electronic media and gadgets. This thing was not seen 3-4 decades earlier because the electronic media were not that much advanced in those days. Following are the two primary reasons for this change in behavior:

**(i)** It is very cheap to consume news on social media
compared with traditional news media, such as newspapers; and

**(ii)** It is very easy share, comment, and discuss the news with others on social media.

These reasons are existing as an essential constituent or characteristic in the nature of these social media platforms. Although social media has reduced our efforts to much extent in many fields but its advancement have proven to much more disadvantageous in many other fields too. One of the prominent disadvantages

they bring with them is the propagation of Fake News. Actually, the problem of spread of fake news is not a new thing, it was present earlier also but the onset of electronic media fueled this problem to much larger extent.

Presently, spreading of any kind of news becomes easy with the arrival of the electronic gadgets and the underlying social media. The traditional mode of news is not as much effective in communication as that of social media today. These social media platforms in today's time easily influence the mass and they contain the information that seems to be visually and audibly appealing but they take the people far away from the truth. So, Fake news have become the biggest danger to democracy and right to freedom of expression. The influenced people have been bound to act in a manner in which they would not, had they been aware of the truth. For example, during U.S. 2016 presidential elections, the most popular fake news was spread on Facebook than the most popular authentic mainstream. Fake news makes the opinion of the people biased on various sensitive issues. During events like elections which can affect the future of whole nation, spreading of fake is an intolerable thing as every information should be transparent and genuine in order for people to make wise decisions. In this article, we'll try to study various computational models in order to detect fake news articles from a set of given documents.

# 2. What is a fake news?

Literally saying, there is no universally accepted definition of the term called "fake news". Although existing studies usually connect fake news to the notions like *false news*, *deceptive news*, *satire news*, *misinformation*, *disinformation*, *rumor*, etc. and based on these notions, there are two key features that help us to understand the difference between their corresponding definition of fake news: (I) **authenticity** (containing any false statement or not which can be verified), (II) **intent** (aiming to mislead or entertain the public). Table given below will help us better to distinguish between the aforementioned notions based on these two features.

| Notion | Authenticity | Intent |
|---|---|---|
| **False news** | Non-factual | Undefined |
| **Deceptive news** | Non-factual | Mislead |
| **Satire news** | Non-unified* | Entertain |
| **Misinformation** | Non-factual | Undefined |
| **Disinformation** | Non-factual | Mislead |
| **Rumor** | Undefined | Undefined |

**Table 2.1.** Distinguishing various notions of fake news

In this article, we'll define the narrow definition of fake news as follows:

**Definition 2.1**: (FAKE NEWS) Fake news is a source of news that is intentionally false and verifiably false. [Allcott and Gentzkow 2017[1]; Shu et al. 2017[2]]

This definition will serve our purpose to much larger extent. This narrow definition emphasizes on both news authenticity and intent. My main focus for this article will be only on determining the authenticity of a News Article since I am working on Computation Approaches for Early Detection and the models involving intent along with authenticity are a bit involved and thus, will increase the complexity of the model.

*where compositions that imitate somebody's style in a humorous way differ from satires in their use of non-factual information to spread humor [Tandoc Jr et al. 2018][3]

# 3. Literature Review

Various commonly used techniques already existing for fake news detection are:

**1) Fact Checking (Knowledge – Based Dectection)**

Fact-checking is the process of verifying information in realistic and non-realistic sources of text in order to determine its veracity and correctness. It can be conducted before **(ante hoc)** or after **(post hoc)** the text is published. **Internal fact-checking** is the one done-in house by the publisher, while in **external fact-checking**, the process of analyzing the text is done by the third-party. It can be done manually or automatically by using machines.

**2) Style – Based Detection**

In the same way as knowledge-based fake news detection, style-based fake news detection also deals with the veracity of the news content. Nevertheless, knowledge-based methods mainly evaluate the factuality of the given news, while style-based methods can detect intent behind news as well, i.e., is there an intention to mislead the public or not? Style-based methods aims to capture the style of writing of news content.

**3) Stance-Based Detection**

Stance-based approaches take into account users' viewpoints from relevant post contents to conclude the veracity of original news articles. It is the method of determining from a post whether the user stands in favor of, is neutral, or is against some target entity, idea, or event [4]. **Explicit stances** are direct expressions of emotion or opinion, like the “thumbs up" and “thumbs down" reactions expressed in Facebook. **Implicit stances** are often automatically extracted from social media posts.

**4) Propagation-Based Detection**

Propagation-based method for fake news detection gives reasoning about the interrelations of relevant social media posts to predict credibility of news. The basic assumption is that the credibility of a news event is highly related to the credibilities of relevant social media posts. Both homogeneous (single type of entities, such as post or event [5]) and heterogeneous credibility networks (different types of entities, such as posts, sub-events, and events [6; 7].) can be built for propagation process.

# 4. Aim of our Experiment

We'll firstly extract the feature set for each document in the given corpus of the true news as well as fake news articles. We'll preprocess those features and try to find unigram, bigram and trigram word frequencies, word clouds, cumulative term frequencies, cumulative document frequencies for each of these features which will help us in finding cumulative Term frequency Score and cumulative Document frequency Score for each document containing those feature sets based on which we can conclude whether a news article is true or fake.

# 5. Working of the Algorithm

The following sequence of steps has been used in our approach:

1) **Gathering Training Data Set for True and Fake News Articles:**

We've taken the two datasets from Kaggle: one of them consisting of true news articles and the other consisting fake news articles. The first one is named as True.csv and it contains 100 True News articles out of which we'll use 80 for training purposes. While the second one is named as Fake.csv and it contains 100 Fake News Articles out of which we'll use 80 for training purposes.

2) **Some part of training set is removed to be used for prediction later:**

Some part of training set, i.e., 20 documents from both True.csv and Fake.csv is removed for true and fake news articles for testing purposes. We have merged these 40 documents in a single file and named it Testing.csv. The removed training set is fed into the prediction algorithm to classify the document as True or Fake.

3) **Use the Training Data Set to teach the Algorithm:**

In this phase every True and Fake News Article are read word by word and common patterns are extracted from them. We have used Term Frequency and Document Frequency approach for unigram, bigram and trigram Model in our analysis (the details in subsection 6.3. involving experiments). The useful information extracted from the True and Fake News Articles is stored in dictionary in python code which will be used by the prediction algorithm later in order to make predictions for each of testing article based on the features it contains.

**4) Predict the removed Training Data Set as True or Fake:**

In this phase we fed the removed training Data Set into our prediction algorithm and it predicts the documents as True or Fake.

**5) Calculate Accuracy of the results:**

We check the accuracy of our results and see if the documents are correctly predicted as True or Fake.

**6) Varying the size of testing data:**

Finally, we vary the size of the testing data to compare and see whether which model performs better at what size of testing data.

# 6. Experiment and Observations

## 6.1. Dataset

We have used two .csv files as dataset for our experiment. They are named as True.csv and Fake.csv. We have downloaded this dataset[1] from Kaggle.com. These datasets are very large but we have taken only 100 documents each from True.csv and Fake.csv. Although these datasets are labelled, we haven't used these labels anywhere in our prediction algorithms. So, our algorithm is based on Unsupervised Machine Learning Model. The dataset chosen by us is mainly based on U.S. Presidential Elections. We have taken 20 out of 100 news articles for testing purposes each from True.csv and Fake.csv and merged these 40 articles in one file and named it as Testing.csv.

| File name | Label | Source | Quantity | Documents taken for testing purpose |
|---|---|---|---|---|
| True.csv | TRUE | Kaggle.com | 100 | 20 |
| Fake.csv | FAKE | Kaggle.com | 100 | 20 |

**Table 6.1.** Statistics of true and fake news article

We are thus left with 80 documents each for True.csv and Fake.csv which will be used by us for training purposes and forms the basis for our prediction algorithm.

1. https://www.kaggle.com/clmentbisaillon/fake-and-real-news-dataset

# 6.2. Experiment apparatus

**Feature Extraction and preprocessing:** We use an extraction model to extract train triples from 18 fake news, which is used to train FML, and extract train triples from 18 true news, which is used to train FML. Simultaneously, we use an extraction model to extract test triple sets from 7 fake news and 7 true news, which means translating each news item into a triple set with a fake or true label. We use OpenIE to perform triple extraction. However, OpenIE does not perform very well on triple extraction from news articles so we use four methods to improve the quality of the entities and relations in the triples extracted:

- We have used NLTK's word_tokenize to convert any text into tokens. For exam ple, the text: "This is a sample sentence, showing off the stop words filteration. " will be changed to the list: [' `` ', 'This', 'is', 'a', 'sample', 'sentence', ',', 'showin g', 'off', 'the', 'stop', 'words', 'filteration', '.', " ‘‘ "].
- We have used NLTK's stopwords to remove the most commonly used words in documents. For example, the list of tokens: [' `` ', 'This', 'is', 'a', 'sample', 'sentence', ',', 'showing', 'off', 'the', 'stop', 'words', 'filteration', '.', """] will be changed to ['``', 'This', 'sample', 'sentence', ',', 'showing', 'stop', 'words', 'filteration', '.', " ‘‘ "].
  The stop words available in python NLTK are:
  {'yourselves', 'this', 'by', 'up', 'of', 'again', 'it', 'during', 'when ', 'than', 'hers', 'for', 'mustn', 'needn', 'been', 'what', 'having', ' if', 'themselves', 'theirs', 'with', 'below', 'all', 'until', 'my', 'is n', 'which', 'about', 'were', 'won', "shouldn't", 'as', 'be', 'do', 'be cause', 'yours', 'didn', 'aren', 'o', 'are', "couldn't", "you're", 'so' , 'most', 'here', 'hasn', "don't", "you'll", 'before', 'a', "aren't", ' its', 'his', 'ma', 's', 'd', 'off', 'don', 'there', 'those', 'does', 'y ourself', 'in', 'out', "should've", 'couldn', 'who', "won't", "she's", 'the', 'shouldn', "shan't", 'further', 'your', "you've", 'y', 'some', ' now', 'had', 'not', 'herself', 'each', 'itself', 'other', 'i', 'while', 'them', 'on', 'such', 'him', "weren't", "hasn't", "hadn't", 'same', 'he r', 'against', 'did', 've', "wasn't", 'or', 'myself', 'above', 'from', 'll', "you'd", "that'll", 'then', 'ours', 'but', 'very', 'whom', 'over' , 'wasn', 'she', 'doing', 'just', 'being', 'these', 'will', "haven't", 'can', 'am', 'have', 'an', 'm', 'weren', 'you', 'himself', 'me', 'is', 'how', 'ourselves', 'shan', 'haven', 'their', 'hadn', 'through', 'shoul d', "wouldn't", "isn't", 'few', 'down', 'more', 'own', 'any', 'doesn', 'ain', 'and', 'was', 'after', 'only', 't', 'they', "doesn't", 'that', ' at', 'has', 'nor', 'why', 'once', 'both', "mightn't", 'wouldn', 'betwee n', 'where', "it's", 'into', 'he', 'under', 're', "didn't", 'we', 'our' , "needn't", 'too', 'no', 'mightn', "mustn't", 'to'}
- Additionally, we have removed any word of length < 3. This will help us to avoid many punctuation marks (for example: comma, quotations, parenthesis, colon,

semicolon, etc.), Non-alphanumeric characters (for example: exclamation mark (!), at (@), question mark (?), dollar sign ($), etc.), etc. which occurs in documents commonly. Also, it helps us to avoid many single numbers and abbreviations. For example, “John O. Brennan” will be changed to “John Brennan”, etc.

- We have used NLTK’s WordNetLemmatizer to change any verbs to their present tense form. For example: “gone” will be changed to “go”, “handled” will be changed to “handle”, etc.
- We have used NLTK’s SnowballStemmar to convert data to lower case, to remove apostrophe ‘s from possessives, to singularize the plurals, to modify the last syllable of any word. For example, “Trump’s” will be changed to “trump”, “mules” will be changed to “mule”, “denied” will be changed to “deni”, etc.
- We have used NLTK’s ngrms in order to work with Bigrams and Trigrams.

**Article Summarization:** We have taken only first 300 words from each article to summarize our news articles. This is done in order to prevent an increase in the runtime. Otherwise, it took too long for a program to run.

## 6.3. Using Training Data Set to characterize every word:

We have used three models for our task namely:

1. Unigram Word Frequency Model
2. Bigram Word Frequency Model
3. Trigram Word Frequency Model

**FOR EVERY DOCUMENT IN TRUE.CSV FILE**

- Read every word in the document.
- Tokenize the whole document.
- Remove stop words (extremely common words used like ‘the’, ‘an’ etc).
- Lemmatize and stem the remaining words obtained after removing stop words.
- Took only first 300 words for summarization.
- Calculate word frequency and sort them in decreasing order of frequency for each of the three Models.

- Generate a common result for the entire corpus True.csv.
  (Arrange words by decreasing order of frequency, i.e., the word which has appeared the maximum times in the entire True.csv at the top and the word which has appeared the minimum time in the entire True.csv at the bottom).
- Calculate CTF, CDF, Net CTF, Net CDF for each word (details in later part of this subsection 6.3).

**FOR EVERY DOCUMENT IN FAKE.CSV**

- Read every word in the document.
- Tokenize the whole document.
- Remove stop words (extremely common words used like ‘the’, ‘an’ etc).
- Lemmatize and stem the remaining words obtained after removing stop words.
- Took only first 300 words for summarization.
- Calculate word frequency and sort them in decreasing order of frequency for each of the three Models.
- Generate a common result for the entire corpus Fakee.csv.
  (Arrange words by decreasing order of frequency, i.e., the word which has appeared the maximum times in the entire Fake.csv at the top and the word which has appeared the minimum time in the entire Fake.csv at the bottom).
- Calculate CTF, CDF, Net CTF, Net CDF for each word (details in later part of this subsection 6.3).

I have created 3 folders namely, Unigram, Bigram, Trigram and put both True.csv and Fake.csv Data Set in each of them. Also, I have put Testing.csv in ach of three folders, and finally ran the Python program using Jupyter IDE which calculated the words on decreasing order of frequency count and the results are shown in the upcoming parts of this subsection 6.3.

## 6.3.1. Unigram Word Frequency Model

The following results were obtained for True Documents*:

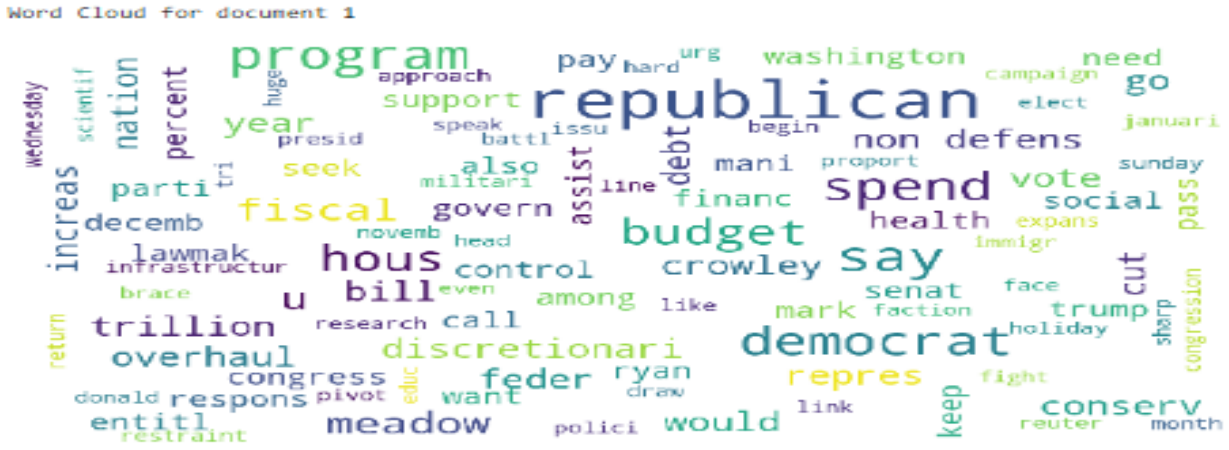
```
16632 words were found in the corpus
The unique identified words with their corresponding word frequencies for top 50 words from true news articles are given below:
        say   (330)
        trump   (273)
        u.s.   (188)
        republican   (160)
        democrat   (138)
        hous   (138)
        would   (123)
        presid   (118)
        state   (115)
        bill   (112)
        year   (110)
        reuter   (98)
        senat   (96)
        court   (79)
        also   (75)
        elect   (68)
        donald   (68)
        washington   (64)
        administr   (60)
        feder   (59)
        percent   (57)
        vote   (56)
        govern   (56)
        repres   (56)
        make   (56)
        congress   (55)
        rule   (54)
        white   (52)
        includ   (51)
        offic   (51)
        immigr   (50)
        investig   (50)
        judg   (50)
        could   (50)
        mueller   (48)
        program   (48)
        depart   (47)
        legisl   (47)
        wednesday   (43)
        statement   (43)
        lawmak   (43)
        unit   (43)
        busi   (42)
        offici   (42)
        polit   (41)
        campaign   (40)
        week   (40)
        thursday   (40)
        countri   (39)
        report   (39)
```

*The complete results are not shown here.

Here it means that the word ‘say’ appeared 330 times in all True documents.

6.3. Using Training Data Set to characterize every word

Word Clouds for first 10 True News articles:

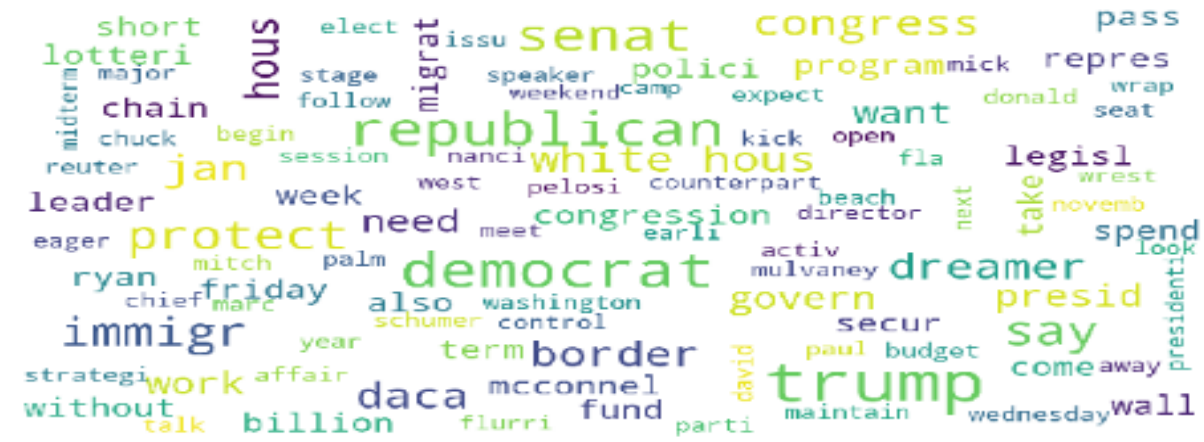


6.3. Using Training Data Set to characterize every word

Word Cloud for document 4

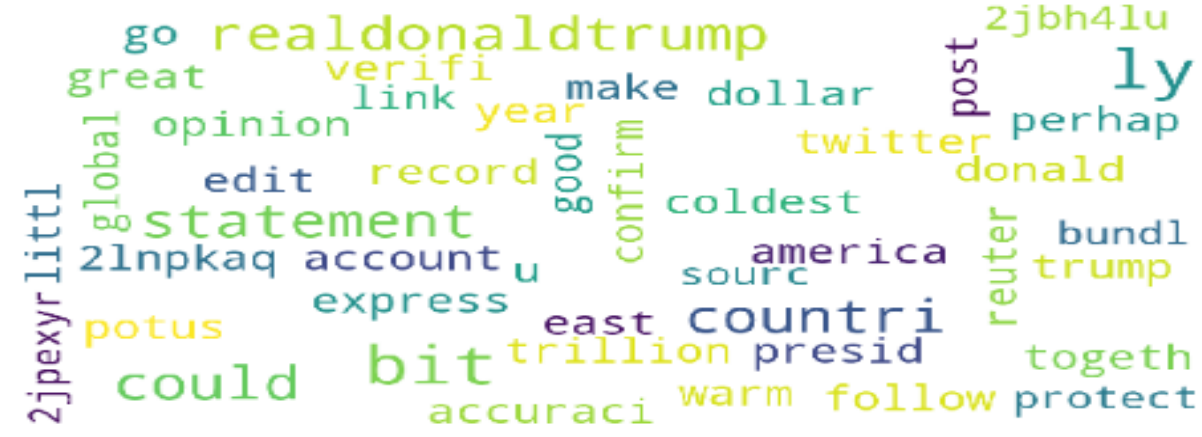

Word Cloud for document 5


Word Cloud for document 6



6.3. Using Training Data Set to characterize every word





6.3. Using Training Data Set to characterize every word

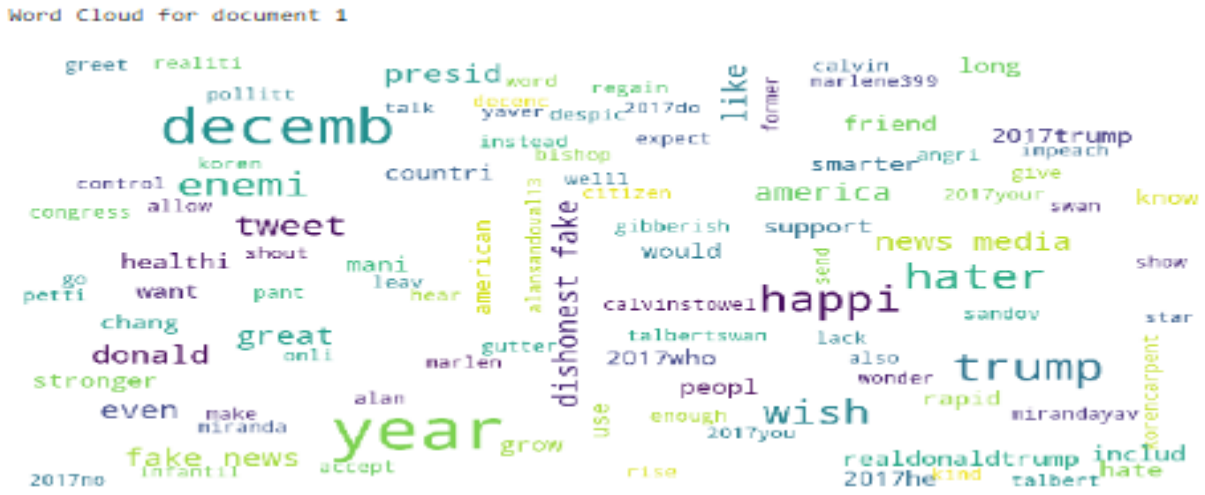
Similarly, the following results were obtained for Fake Documents:

```
16846 words were found in the corpus
The unique identified words with their corresponding word frequencies for top 50 words from fake news articles are given below:
        trump   (461)
        presid   (128)
        donald   (125)
        decemb   (120)
        say   (115)
        novemb   (111)
        time   (109)
        imag   (106)
        moor   (89)
        year   (89)
        like   (89)
        peopl   (77)
        senat   (77)
        make   (75)
        tweet   (74)
        even   (70)
        know   (65)
        white   (65)
        republican   (61)
        would   (60)
        call   (60)
        take   (60)
        news   (59)
        hous   (58)
        flynn   (58)
        former   (56)
        american   (54)
        think   (51)
        go   (51)
        alabama   (49)
        also   (49)
        state   (47)
        nation   (46)
        campaign   (44)
        well   (43)
        twitter   (43)
        come   (42)
        support   (40)
        report   (39)
        mueller   (39)
        offic   (38)
        person   (38)
        right   (38)
        democrat   (38)
        russian   (38)
        accus   (37)
        back   (37)
        first   (37)
        investig   (37)
        https   (37)
```

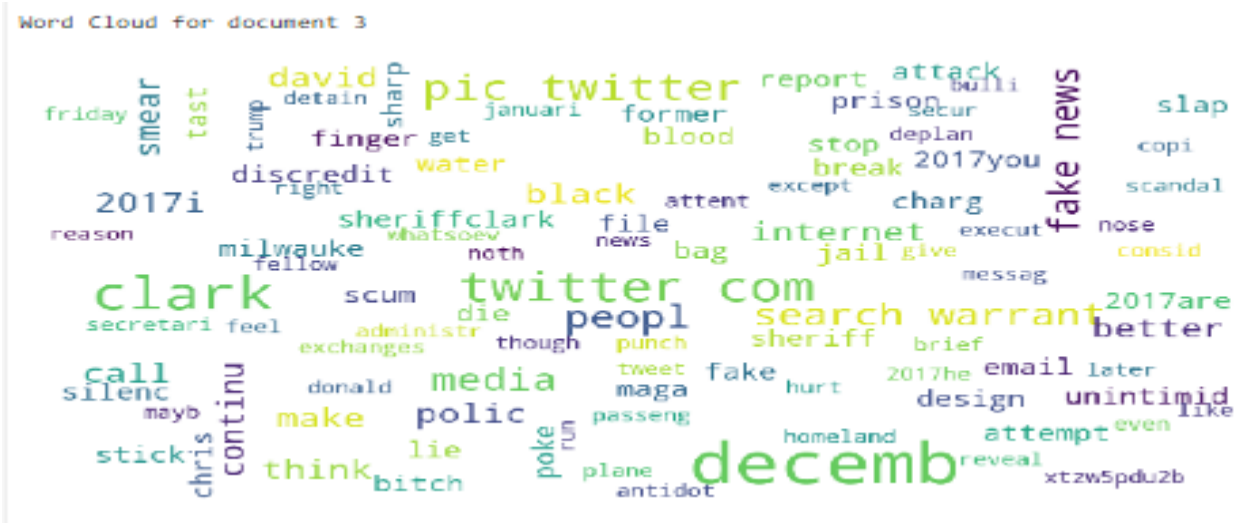
A snapshot of the results is attached here. Here the words are sorted by decreasing order of frequency in the Fake.csv.

6.3. Using Training Data Set to characterize every word

Word Clouds for first 10 Fake News articles:

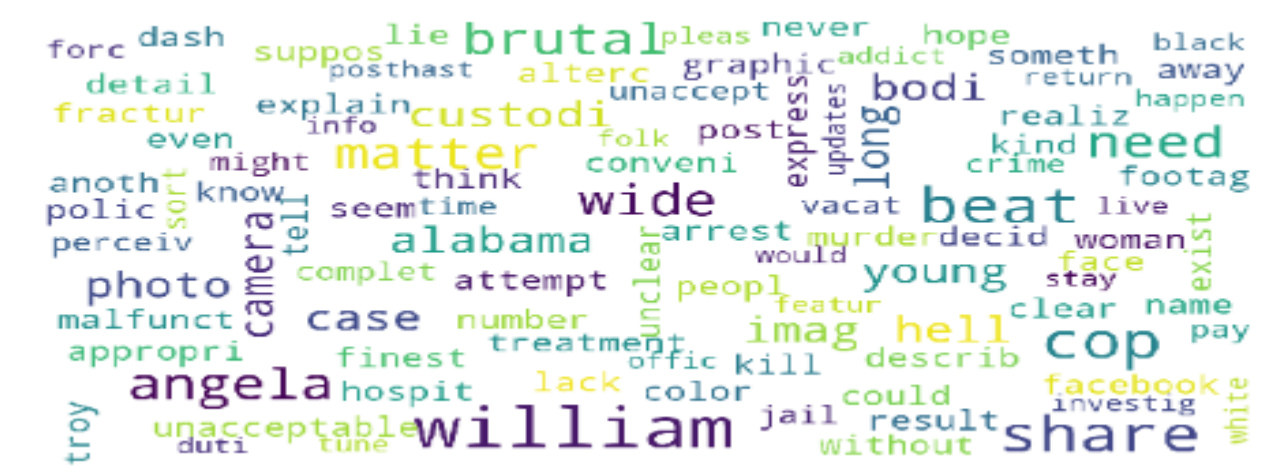






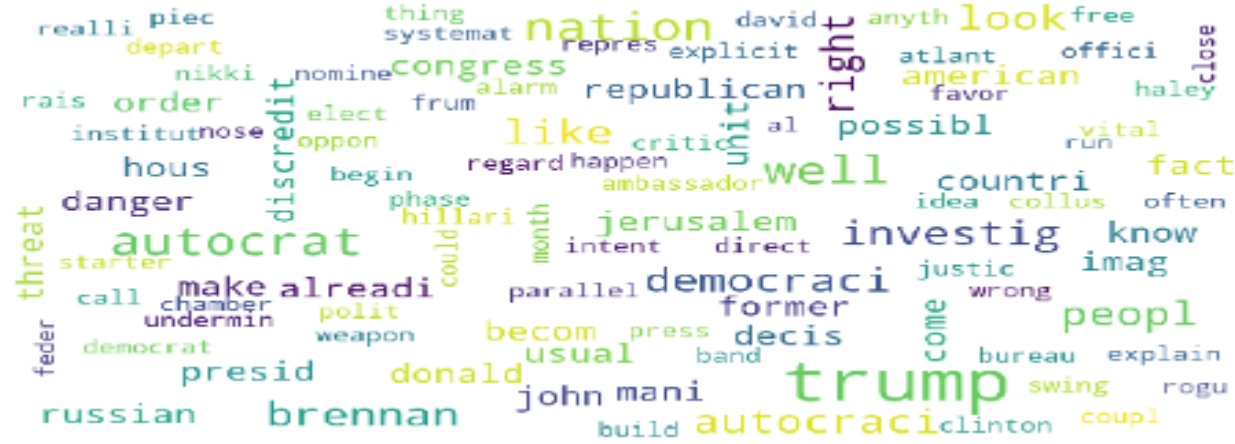
6.3. Using Training Data Set to characterize every word



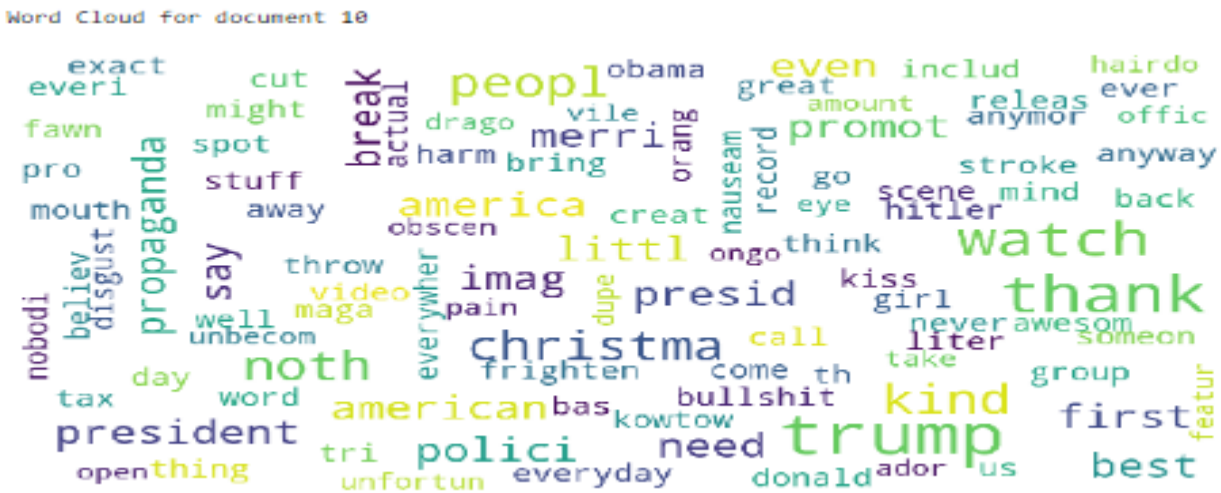


6.3. Using Training Data Set to characterize every word





6.3. Using Training Data Set to characterize every word




**Problem with directly calculating the frequency:**
Here the problem was that: Words like 'administr' (in True Documents) which had occurred in lesser number of documents (20 out of 80) compared to the words like ‘make’ (in True Documents) which had occurred in a greater number of documents (34 out of 80), but ‘administr’ had a high frequency in those documents in which it occurred. This gives us an impression that ‘administr’ must be occurring in a greater number of True documents than ‘make’. However, this is not the case at all.

One of the other problems that was found out was in the input set, **some documents were big and some were very small so documents** that were big were actually dictating the result.

In order to solve the above problem, we normalized the word count. The **Term Frequency** was used:

$$\boldsymbol{Term\ Frequency}['\boldsymbol{w}'] = \frac{No.\ of\ times\ a\ word\ 'w'\ appeared\ in\ a\ Document}{Total\ number\ of\ words\ in\ the\ Document}$$

If all the words in the document are same then term frequency of the common word will be 1.

- For the Entire True.csv Data Set the following tests were ran (Similarly for Fake.csv Data Set as well):

For every Document $D_j \in$ Corpus/ Dataset $\{D_1, D_2, D_3.....D_{80}\}$

Remove the Stop words, lemmatize and stem the remaining words and for every remaining word 'w' in $D_j$

If word 'w' is new, never appeared in any other document
Calculate the term frequency for that word

If word already appeared previously in some previous document
Term frequency = Term frequency of 'w' for $D_j$ + the older term frequency for that word

Display the result in decreasing order of Cumulative Term Frequency. We will call it **CTF**. Thus:

**$(n_1/N_1) + (n_2/N_2) + (n_3/N_3) + ..... + (n_{80}/N_{80})$ = Cumulative sum of Term Frequency for word 'w'**

$n_1$ = no. of times word 'w' appeared in document 1
$N_1$= total no of words in Document 1 excluding stop words

$n_2$ = no. of times word 'w' appeared in document 2
$N_2$ = total no of words in Document 2 excluding stop words

$n_3$ = no. of times word 'w' appeared in document 3
$N_3$ = total no. of words in Document 3 excluding stop words
.

.

.

$n_{80}$ = no. of times word 'w' appeared in document 80
$N_{80}$ = total no. of words in Document 80 excluding stop words
For every word ‘w’
Calculate the Cumulative Term Frequency for word ‘w’
Display the result in decreasing order of Cumulative Term Frequency.

A similar approach was used to calculate Cumulative Document frequency. We will call it **CDF**.

**Document Frequency** for a word 'w' is defined as no. of documents a word appeared in. Since the no. of documents could keep on varying, we tried to normalize it.

$$\boldsymbol{CDF}['\boldsymbol{w}'] = \frac{No.of\ documents\ a\ word\ 'w'\ appeared\ in}{Total\ number\ of\ documents\ in\ the\ corpus/Data\ Set}$$

Basically, it is a measure of the percentage of documents a word appeared in.

- For the Entire True.csv Data Set the following tests were ran (Similarly for Fake.csv Data Set as well):

For every Document $D_j \in$ Corpus/ Dataset $\{D_1, D_2, D_3.....D_{80}\}$

Remove the Stop words, lemmatize and stem the remaining words and for every remaining word ‘w’ in $D_j$

Calculate the Cumulative Document Frequency for word ‘w’

Display the result in decreasing order of Cumulative Document Frequency. Thus:

n/N = Cumulative Document Frequency for word 'w'

n = No. of Documents a word ‘w’ appeared in
N = Total no. of documents in the Set

The following results were obtained for True Documents showing CTF and CDF in decreasing order of frequency. (Only Top Results are shown here)

| Word | Cumulative Term Frequency * | Cumulative Document Frequency |
|---|---|---|
| say | 1.476195 | 0.800000 |
| trump | 1.273139 | 0.825000 |
| u.s. | 1.111425 | 0.900000 |
| hous | 0.748159 | 0.550000 |
| republican | 0.714917 | 0.637500 |
| reuter | 0.687426 | 1.000000 |
| bill | 0.635938 | 0.425000 |
| democrat | 0.630889 | 0.512500 |
| presid | 0.605505 | 0.787500 |
| would | 0.568186 | 0.662500 |
| state | 0.544942 | 0.562500 |
| year | 0.506914 | 0.625000 |
| senat | 0.479564 | 0.450000 |
| washington | 0.430551 | 0.625000 |
| donald | 0.409277 | 0.812500 |
| also | 0.351765 | 0.537500 |
| court | 0.350390 | 0.262500 |
| repres | 0.342504 | 0.462500 |
| statement | 0.326994 | 0.312500 |
| make | 0.324974 | 0.425000 |
| govern | 0.323218 | 0.412500 |
| elect | 0.313555 | 0.337500 |
| congress | 0.302804 | 0.375000 |
| thursday | 0.291247 | 0.312500 |
| pass | 0.278806 | 0.312500 |
| vote | 0.274533 | 0.350000 |
| white | 0.262247 | 0.287500 |
| wednesday | 0.256777 | 0.362500 |
| legisl | 0.253352 | 0.350000 |
| rule | 0.249969 | 0.287500 |
| feder | 0.247081 | 0.412500 |
| could | 0.240657 | 0.387500 |
| administr | 0.230843 | 0.250000 |
| week | 0.230335 | 0.350000 |
| offici | 0.229452 | 0.325000 |

Results showing in decreasing order of CTF for True Documents

Here the word ‘administr’ has dropped down much below the word ‘make’ as per CTF which is good thing.

| Word | Cumulative Term Frequency | Cumulative Document Frequency * |
|---|---|---|
| reuter | 0.687426 | 1.000000 |
| u.s. | 1.111425 | 0.900000 |
| trump | 1.273139 | 0.825000 |
| donald | 0.409277 | 0.812500 |
| say | 1.476195 | 0.800000 |
| presid | 0.605505 | 0.787500 |
| would | 0.568186 | 0.662500 |
| republican | 0.714917 | 0.637500 |
| washington | 0.430551 | 0.625000 |
| year | 0.506914 | 0.625000 |
| state | 0.544942 | 0.562500 |
| hous | 0.748159 | 0.550000 |
| also | 0.351765 | 0.537500 |
| democrat | 0.630889 | 0.512500 |
| repres | 0.342504 | 0.462500 |
| includ | 0.211379 | 0.462500 |
| senat | 0.479564 | 0.450000 |
| bill | 0.635938 | 0.425000 |
| make | 0.324974 | 0.425000 |
| govern | 0.323218 | 0.412500 |
| feder | 0.247081 | 0.412500 |
| could | 0.240657 | 0.387500 |
| congress | 0.302804 | 0.375000 |
| wednesday | 0.256777 | 0.362500 |
| take | 0.191827 | 0.350000 |
| vote | 0.274533 | 0.350000 |
| week | 0.230335 | 0.350000 |
| legisl | 0.253352 | 0.350000 |
| elect | 0.313555 | 0.337500 |
| last | 0.178794 | 0.337500 |
| unit | 0.185355 | 0.325000 |
| like | 0.198368 | 0.325000 |
| offici | 0.229452 | 0.325000 |
| work | 0.147890 | 0.325000 |
| pass | 0.278806 | 0.312500 |

Results showing in decreasing order of CDF for True Documents

6.3. Using Training Data Set to characterize every word

The following results were obtained for Fake Documents showing CTF and CDF (Only Top Results are shown here)

| Word | Cumulative Term Frequency * | Cumulative Document Frequency |
|---|---|---|
| trump | 2.265490 | 0.900000 |
| say | 0.598736 | 0.662500 |
| donald | 0.587645 | 0.787500 |
| presid | 0.559557 | 0.662500 |
| imag | 0.555667 | 0.812500 |
| decemb | 0.500716 | 0.400000 |
| time | 0.485250 | 0.550000 |
| moor | 0.463776 | 0.200000 |
| novemb | 0.435927 | 0.325000 |
| like | 0.435660 | 0.600000 |
| senat | 0.389444 | 0.337500 |
| year | 0.384692 | 0.387500 |
| peopl | 0.378434 | 0.512500 |
| make | 0.356020 | 0.587500 |
| know | 0.338443 | 0.525000 |
| even | 0.336314 | 0.512500 |
| tweet | 0.329934 | 0.412500 |
| white | 0.324627 | 0.375000 |
| republican | 0.290857 | 0.350000 |
| flynn | 0.290592 | 0.137500 |
| call | 0.288349 | 0.412500 |
| take | 0.283722 | 0.500000 |
| hous | 0.281258 | 0.375000 |
| would | 0.276348 | 0.387500 |
| alabama | 0.264569 | 0.212500 |
| former | 0.260646 | 0.450000 |
| american | 0.258426 | 0.375000 |
| think | 0.253380 | 0.437500 |
| news | 0.252476 | 0.387500 |
| go | 0.250449 | 0.450000 |
| state | 0.244946 | 0.350000 |
| nation | 0.232740 | 0.337500 |
| also | 0.231933 | 0.450000 |
| campaign | 0.216696 | 0.362500 |
| well | 0.213416 | 0.400000 |

Results showing in decreasing order of CTF for Fake Documents

6.3. Using Training Data Set to characterize every word

| Word | Cumulative Term Frequency | Cumulative Document Frequency * |
|---|---|---|
| trump | 2.265490 | 0.900000 |
| imag | 0.555667 | 0.812500 |
| donald | 0.587645 | 0.787500 |
| presid | 0.559557 | 0.662500 |
| say | 0.598736 | 0.662500 |
| like | 0.435660 | 0.600000 |
| make | 0.356020 | 0.587500 |
| time | 0.485250 | 0.550000 |
| know | 0.338443 | 0.525000 |
| peopl | 0.378434 | 0.512500 |
| even | 0.336314 | 0.512500 |
| take | 0.283722 | 0.500000 |
| also | 0.231933 | 0.450000 |
| former | 0.260646 | 0.450000 |
| go | 0.250449 | 0.450000 |
| think | 0.253380 | 0.437500 |
| come | 0.208699 | 0.425000 |
| tweet | 0.329934 | 0.412500 |
| call | 0.288349 | 0.412500 |
| decemb | 0.500716 | 0.400000 |
| well | 0.213416 | 0.400000 |
| year | 0.384692 | 0.387500 |
| would | 0.276348 | 0.387500 |
| news | 0.252476 | 0.387500 |
| american | 0.258426 | 0.375000 |
| show | 0.159882 | 0.375000 |
| hous | 0.281258 | 0.375000 |
| white | 0.324627 | 0.375000 |
| campaign | 0.216696 | 0.362500 |
| offic | 0.196632 | 0.362500 |
| state | 0.244946 | 0.350000 |
| republican | 0.290857 | 0.350000 |
| mani | 0.161629 | 0.337500 |
| nation | 0.232740 | 0.337500 |
| senat | 0.389444 | 0.337500 |

Results showing in decreasing order of CDF for Fake Documents

The CTF and CDF were calculated for 16632 words in True.csv Data set.

The CTF and CDF were calculated for 16846 words in Fake.csv Data set.

This above exercise gave us a measure that which words are the most important for True and Fake Documents.

However, we noticed there are words which appear in both True and Fake Documents. For such words we are not sure whether they are ‘fake’ in nature or not.

In order to solve the above problem, we decided to give a **Net CTF** and **Net CDF** score for every word encountered so far.

The following convention was used: **All the True Documents were considered as positive and Fake Documents were considered as negative.**

TRUE = + ve
FAKE = - ve

The following formula was used to calculate NET CTF for every word 'w':

**Net CTF = (CTF for word 'w' in True.csv Data Set) - (CTF for word 'w' in Fake.csv Data Set)**

The following formula was used to calculate NET CDF for every word 'w':

**Net CDF = (CDF for word 'w' in True.csv Data Set) - (CDF for word 'w' in Fake.csv Data Set)**

The Net CTF and Net CDF were calculated and were stored for every word.

6.3. Using Training Data Set to characterize every word

The following results were obtained.

| Word | Net Cumulative Term Frequency * | Net Cumulative Document Frequency |
|---|---|---|
| u.s. | 1.063802 | 0.800000 |
| say | 0.877459 | 0.137500 |
| reuter | 0.687426 | 1.000000 |
| bill | 0.512724 | 0.237500 |
| hous | 0.466901 | 0.175000 |
| democrat | 0.439338 | 0.262500 |
| republican | 0.424061 | 0.287500 |
| washington | 0.392709 | 0.525000 |
| state | 0.299997 | 0.212500 |
| would | 0.291838 | 0.275000 |
| govern | 0.291102 | 0.350000 |
| thursday | 0.273740 | 0.275000 |
| court | 0.270871 | 0.137500 |
| repres | 0.268364 | 0.350000 |
| wednesday | 0.243602 | 0.325000 |
| feder | 0.223816 | 0.362500 |
| legisl | 0.221240 | 0.287500 |
| statement | 0.218235 | 0.125000 |
| rule | 0.215861 | 0.225000 |
| fund | 0.211084 | 0.187500 |
| lawmak | 0.202838 | 0.200000 |
| pass | 0.199181 | 0.175000 |
| congress | 0.196442 | 0.187500 |
| percent | 0.193161 | 0.175000 |
| program | 0.185693 | 0.162500 |
| approv | 0.184841 | 0.237500 |
| week | 0.175030 | 0.237500 |
| immigr | 0.174968 | 0.162500 |
| friday | 0.167930 | 0.187500 |
| expect | 0.165702 | 0.200000 |
| januari | 0.162591 | 0.212500 |
| depart | 0.162542 | 0.225000 |
| administr | 0.156228 | 0.112500 |
| vote | 0.155265 | 0.187500 |
| billion | 0.151724 | 0.150000 |

Results showing in decreasing order of Net CTF

6.3. Using Training Data Set to characterize every word

**Positive means the word has True Characteristics**

| Word | Net Cumulative Term Frequency | Net Cumulative Document Frequency * |
|---|---|---|
| reuter | 0.687426 | 1.000000 |
| u.s. | 1.063802 | 0.800000 |
| washington | 0.392709 | 0.525000 |
| feder | 0.223816 | 0.362500 |
| repres | 0.268364 | 0.350000 |
| govern | 0.291102 | 0.350000 |
| wednesday | 0.243602 | 0.325000 |
| republican | 0.424061 | 0.287500 |
| legisl | 0.221240 | 0.287500 |
| thursday | 0.273740 | 0.275000 |
| would | 0.291838 | 0.275000 |
| democrat | 0.439338 | 0.262500 |
| includ | 0.114092 | 0.250000 |
| approv | 0.184841 | 0.237500 |
| bill | 0.512724 | 0.237500 |
| year | 0.122222 | 0.237500 |
| week | 0.175030 | 0.237500 |
| rule | 0.215861 | 0.225000 |
| depart | 0.162542 | 0.225000 |
| state | 0.299997 | 0.212500 |
| januari | 0.162591 | 0.212500 |
| congression | 0.087913 | 0.212500 |
| victori | 0.119364 | 0.212500 |
| expect | 0.165702 | 0.200000 |
| overhaul | 0.108736 | 0.200000 |
| lawmak | 0.202838 | 0.200000 |
| friday | 0.167930 | 0.187500 |
| congress | 0.196442 | 0.187500 |
| recent | 0.081075 | 0.187500 |
| fund | 0.211084 | 0.187500 |
| vote | 0.155265 | 0.187500 |
| hous | 0.466901 | 0.175000 |
| pass | 0.199181 | 0.175000 |
| percent | 0.193161 | 0.175000 |
| dec. | 0.113767 | 0.175000 |

Results showing in decreasing order of Net CDF

**Positive means the word has True Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency * | Net Cumulative Document Frequency |
|---|---|---|
| trump | -0.992351 | -0.075000 |
| imag | -0.541730 | -0.787500 |
| decemb | -0.484049 | -0.350000 |
| novemb | -0.386256 | -0.187500 |
| moor | -0.334391 | -0.150000 |
| time | -0.307307 | -0.275000 |
| flynn | -0.286527 | -0.125000 |
| tweet | -0.278108 | -0.362500 |
| like | -0.237292 | -0.275000 |
| peopl | -0.225557 | -0.225000 |
| know | -0.223715 | -0.325000 |
| even | -0.179892 | -0.200000 |
| donald | -0.178369 | 0.025000 |
| alabama | -0.167411 | -0.125000 |
| cours | -0.161564 | -0.262500 |
| well | -0.160033 | -0.262500 |
| https | -0.151069 | -0.250000 |
| american | -0.148281 | -0.150000 |
| accus | -0.136844 | -0.162500 |
| think | -0.136428 | -0.250000 |
| come | -0.136121 | -0.237500 |
| person | -0.136067 | -0.137500 |
| video | -0.135406 | -0.212500 |
| call | -0.133367 | -0.175000 |
| child | -0.129882 | -0.150000 |
| america | -0.129869 | -0.225000 |
| back | -0.124828 | -0.112500 |
| go | -0.122940 | -0.225000 |
| world | -0.122067 | -0.175000 |
| russian | -0.119461 | -0.062500 |
| obama | -0.117285 | -0.062500 |
| word | -0.117225 | -0.237500 |
| women | -0.116758 | -0.112500 |
| thing | -0.115529 | -0.225000 |
| noth | -0.114289 | -0.162500 |

Results showing in increasing order of Net CTF

**Negative means the word has Fake Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency | Net Cumulative Document Frequency * |
|---|---|---|
| imag | -0.541730 | -0.787500 |
| tweet | -0.278108 | -0.362500 |
| decemb | -0.484049 | -0.350000 |
| know | -0.223715 | -0.325000 |
| time | -0.307307 | -0.275000 |
| like | -0.237292 | -0.275000 |
| well | -0.160033 | -0.262500 |
| cours | -0.161564 | -0.262500 |
| think | -0.136428 | -0.250000 |
| https | -0.151069 | -0.250000 |
| word | -0.117225 | -0.237500 |
| come | -0.136121 | -0.237500 |
| thing | -0.115529 | -0.225000 |
| go | -0.122940 | -0.225000 |
| seem | -0.097603 | -0.225000 |
| america | -0.129869 | -0.225000 |
| someth | -0.113559 | -0.225000 |
| peopl | -0.225557 | -0.225000 |
| fact | -0.092884 | -0.212500 |
| video | -0.135406 | -0.212500 |
| featur | -0.085255 | -0.212500 |
| realiti | -0.079977 | -0.200000 |
| even | -0.179892 | -0.200000 |
| novemb | -0.386256 | -0.187500 |
| news | -0.082668 | -0.187500 |
| star | -0.072073 | -0.187500 |
| realdonaldtrump | 0.053183 | -0.175000 |
| alway | -0.077836 | -0.175000 |
| call | -0.133367 | -0.175000 |
| world | -0.122067 | -0.175000 |
| actual | -0.100377 | -0.175000 |
| make | -0.031046 | -0.162500 |
| former | -0.064326 | -0.162500 |
| right | -0.104018 | -0.162500 |
| accus | -0.136844 | -0.162500 |

Results showing in increasing order of Net CDF

**Negative means the word has Fake Characteristics**

## 6.3.2. Bigram Word Frequency Model

The following results were obtained for True Documents*:

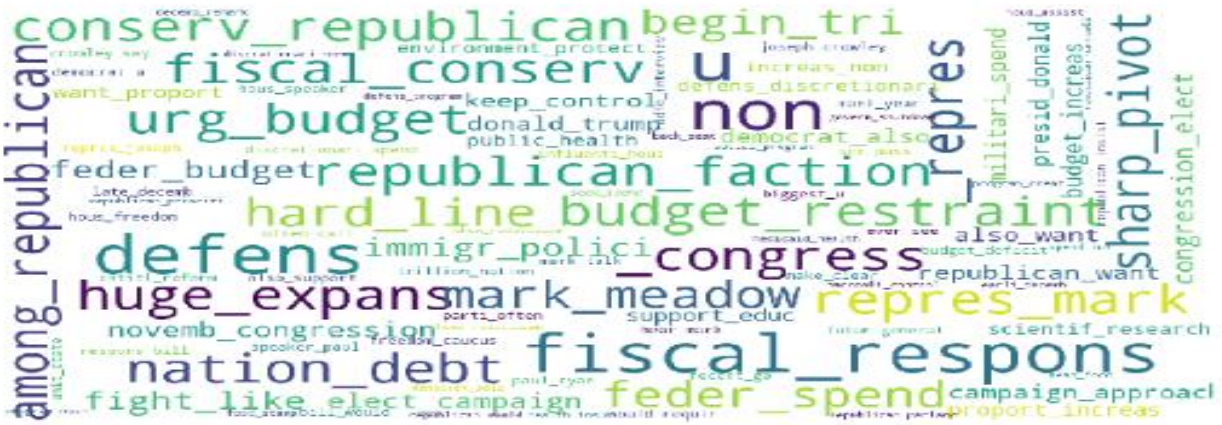
```
7564 bigrams were found in the corpus
The unique identified bigrams with their corresponding word frequencies for top 50 bigrams from true news articles are given be
low:
        ('donald', 'trump')   (65)
        ('presid', 'donald')   (60)
        ('unit', 'state')   (27)
        ('white', 'hous')   (25)
        ('take', 'offic')   (18)
        ('u.s.', 'presid')   (18)
        ('u.s.', 'hous')   (14)
        ('trump', 'administr')   (13)
        ('special', 'counsel')   (13)
        ('last', 'week')   (12)
        ('next', 'year')   (12)
        ('hillari', 'clinton')   (11)
        ('nation', 'secur')   (11)
        ('attorney', 'general')   (11)
        ('trump', 'say')   (11)
        ('feder', 'judg')   (11)
        ('justic', 'depart')   (11)
        ('offici', 'say')   (10)
        ('suprem', 'court')   (10)
        ('congression', 'elect')   (9)
        ('last', 'year')   (9)
        ('legisl', 'victori')   (8)
        ('opinion', 'express')   (8)
        ('counsel', 'robert')   (8)
        ('robert', 'mueller')   (8)
        ('twitter', 'account')   (8)
        ('u.s.', 'suprem')   (8)
        ('district', 'judg')   (8)
        ('follow', 'statement')   (8)
        ('sourc', 'link')   (8)
        ('u.s.', 'district')   (8)
        ('republican', 'presid')   (8)
        ('verifi', 'twitter')   (8)
        ('puerto', 'rico')   (8)
        ('u.s.', 'senat')   (7)
        ('govern', 'fund')   (7)
        ('u.s.', 'elect')   (7)
        ('would', 'like')   (7)
        ('first', 'major')   (7)
        ('also', 'say')   (7)
        ('barack', 'obama')   (7)
        ('hous', 'republican')   (7)
        ('health', 'insur')   (7)
        ('feder', 'bureau')   (7)
        ('former', 'presid')   (7)
        ('republican', 'lawmak')   (6)
        ('appeal', 'court')   (6)
        ('paul', 'ryan')   (6)
        ('presid', 'barack')   (6)
        ('republican', 'leader')   (6)
```

*The complete results are not shown here.

6.3. Using Training Data Set to characterize every word

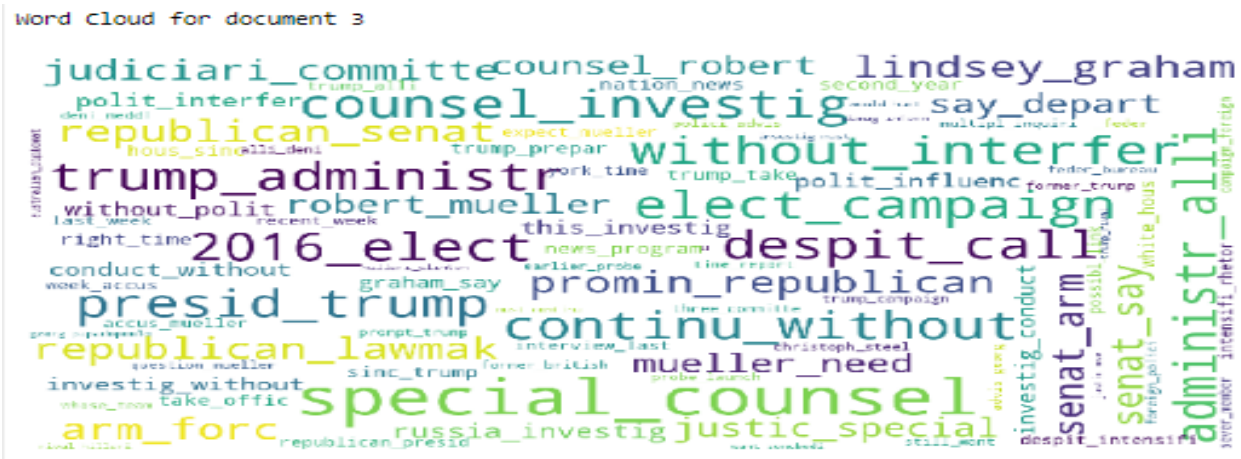
Bigram Word Clouds for first 10 True News articles:

Word Cloud for document 1



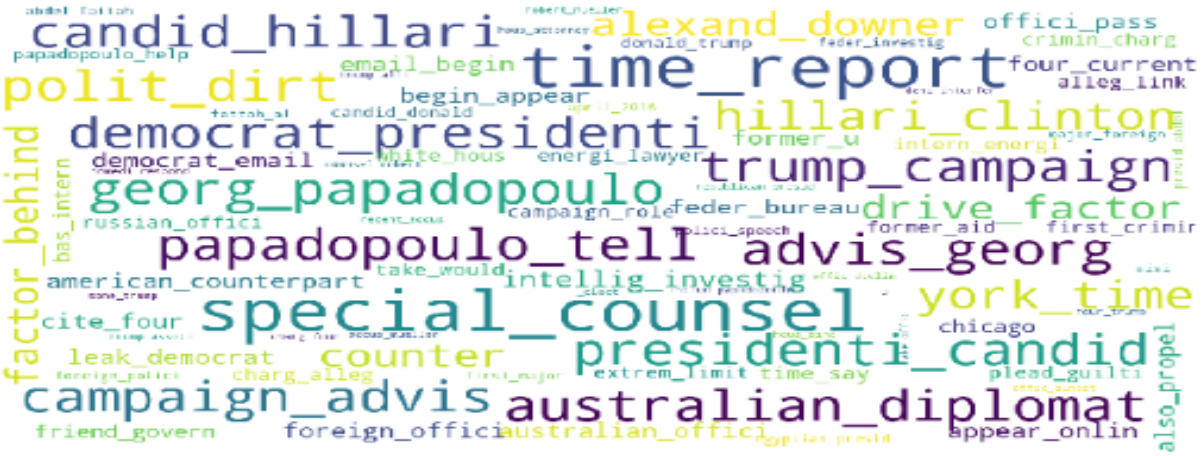

Word Cloud for document 2


Word Cloud for document 3

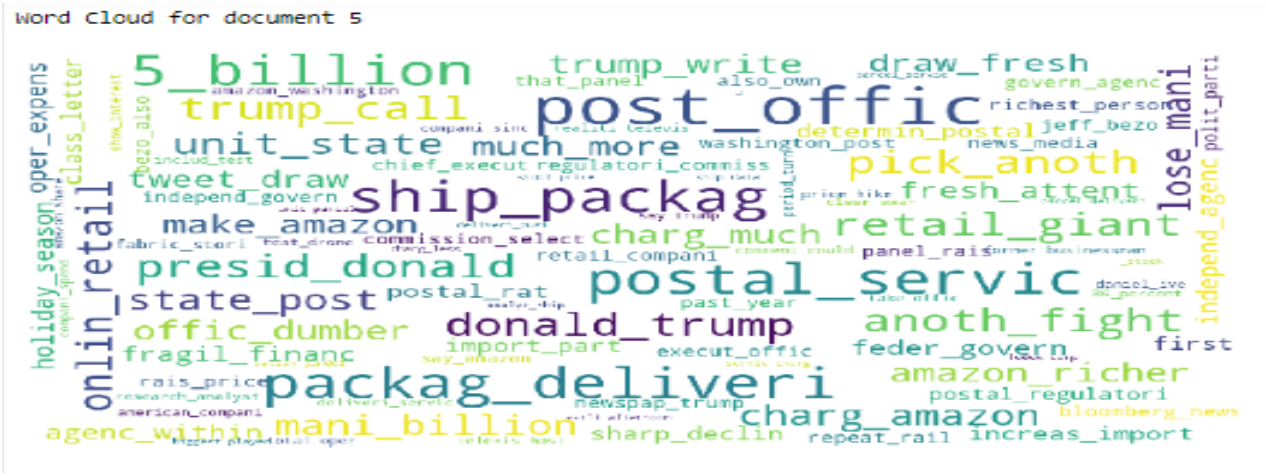


6.3. Using Training Data Set to characterize every word

Word Cloud for document 4




Word Cloud for document 5



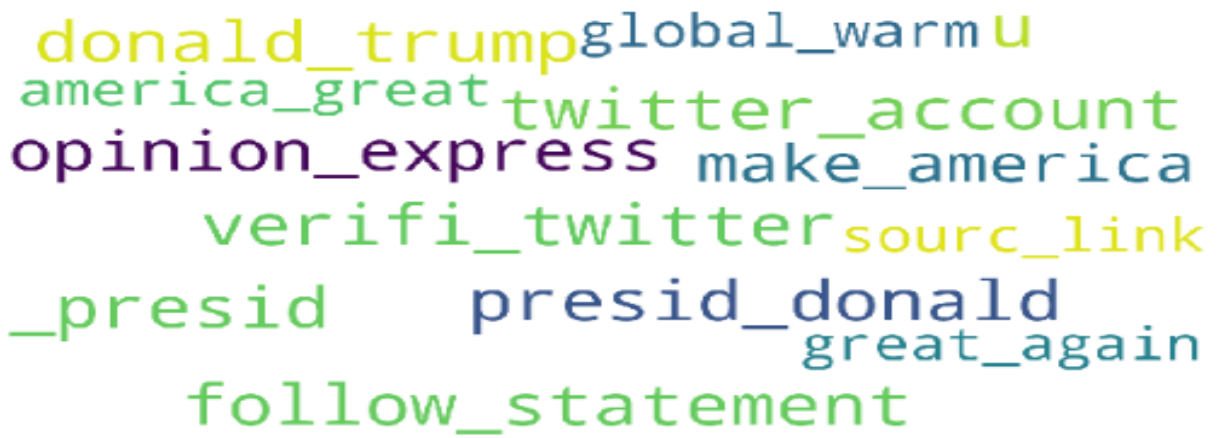

Word Cloud for document 6

6.3. Using Training Data Set to characterize every word

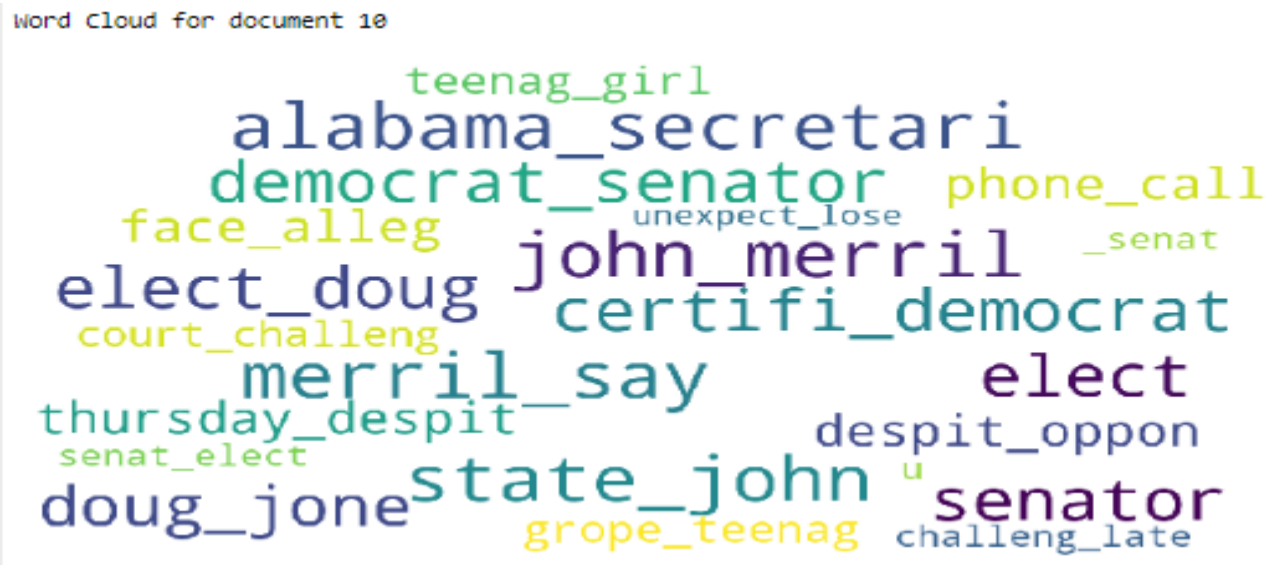
Word Cloud for document 7


Word Cloud for document 8


Word Cloud for document 9



6.3. Using Training Data Set to characterize every word

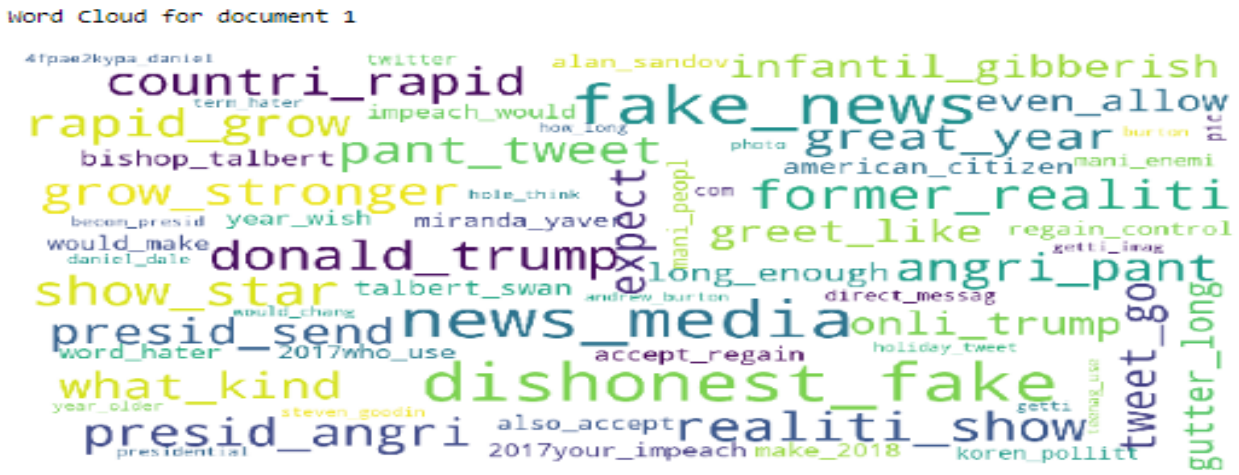


Similarly, the following results were obtained for Fake Documents:

```
5642 bigrams were found in the corpus
The unique identified bigrams with their corresponding word frequencies for top 50 bigrams from fake news articles are given be
low:
        ('donald', 'trump')   (61)
        ('white', 'hous')   (22)
        ('robert', 'mueller')   (15)
        ('unit', 'state')   (15)
        ('realiti', 'show')   (14)
        ('show', 'star')   (14)
        ('former', 'realiti')   (14)
        ('nation', 'secur')   (13)
        ('featur', 'imag')   (13)
        ('hillari', 'clinton')   (11)
        ('screen', 'captur')   (11)
        ('fake', 'news')   (11)
        ('special', 'counsel')   (11)
        ('child', 'molest')   (10)
        ('look', 'like')   (10)
        ('presid', 'trump')   (9)
        ('secur', 'advis')   (9)
        ('counsel', 'robert')   (9)
        ('doug', 'jone')   (8)
        ('chip', 'somodevilla/getti')   (8)
        ('somodevilla/getti', 'imag')   (8)
        ('senat', 'candid')   (8)
        ('michael', 'flynn')   (8)
        ('attorney', 'general')   (8)
        ('trump', 'say')   (8)
        ('former', 'nation')   (7)
        ('mani', 'peopl')   (7)
        ('york', 'time')   (7)
        ('even', 'though')   (7)
        ('sexual', 'harass')   (7)
        ('teenag', 'girl')   (7)
        ('trump', 'tweet')   (7)
        ('barack', 'obama')   (7)
        ('republican', 'parti')   (7)
        ('sexual', 'predat')   (7)
        ('last', 'year')   (6)
        ('plead', 'guilti')   (6)
        ('time', 'magazin')   (6)
        ('sexual', 'assault')   (6)
        ('trump', 'support')   (6)
        ('suprem', 'court')   (6)
        ('oval', 'offic')   (6)
        ('wong/getti', 'imag')   (6)
        ('former', 'presid')   (6)
        ('alex', 'wong/getti')   (6)
        ('alabama', 'senat')   (5)
        ('angerer/getti', 'imag')   (5)
        ('make', 'sure')   (5)
        ('advis', 'michael')   (5)
        ('drew', 'angerer/getti')   (5)
```

6.3. Using Training Data Set to characterize every word

Bigram Word Clouds for first 10 Fake News articles:



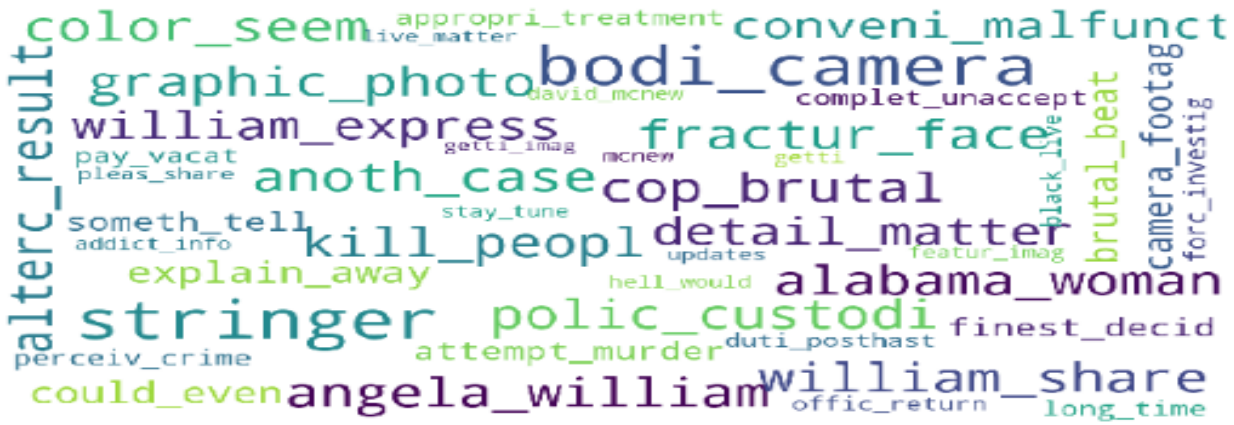


6.3. Using Training Data Set to characterize every word

Word Cloud for document 4

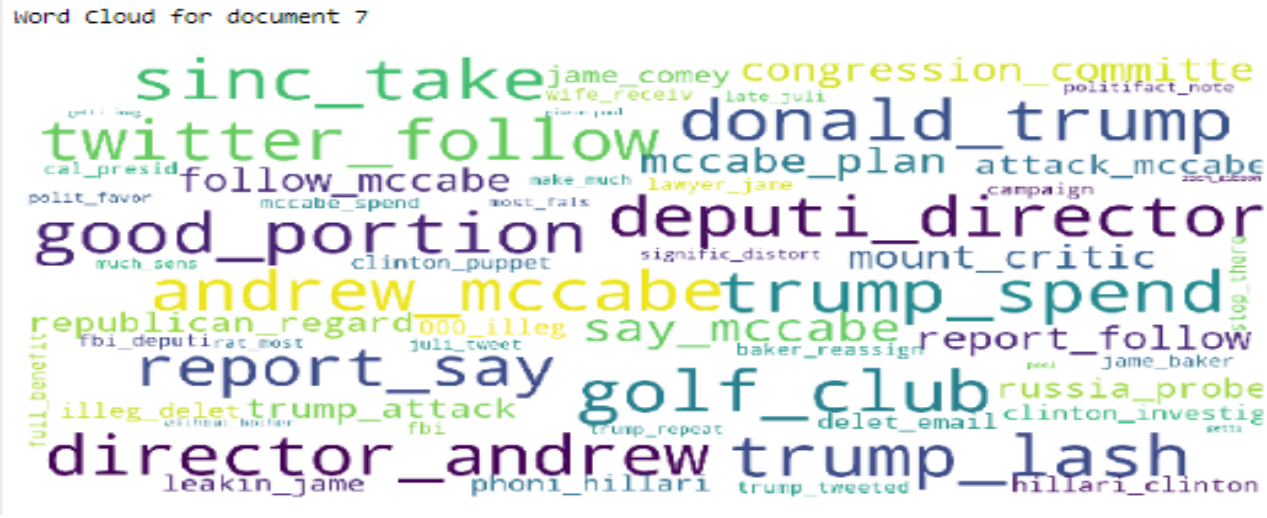

Word Cloud for document 5


Word Cloud for document 6



6.3. Using Training Data Set to characterize every word

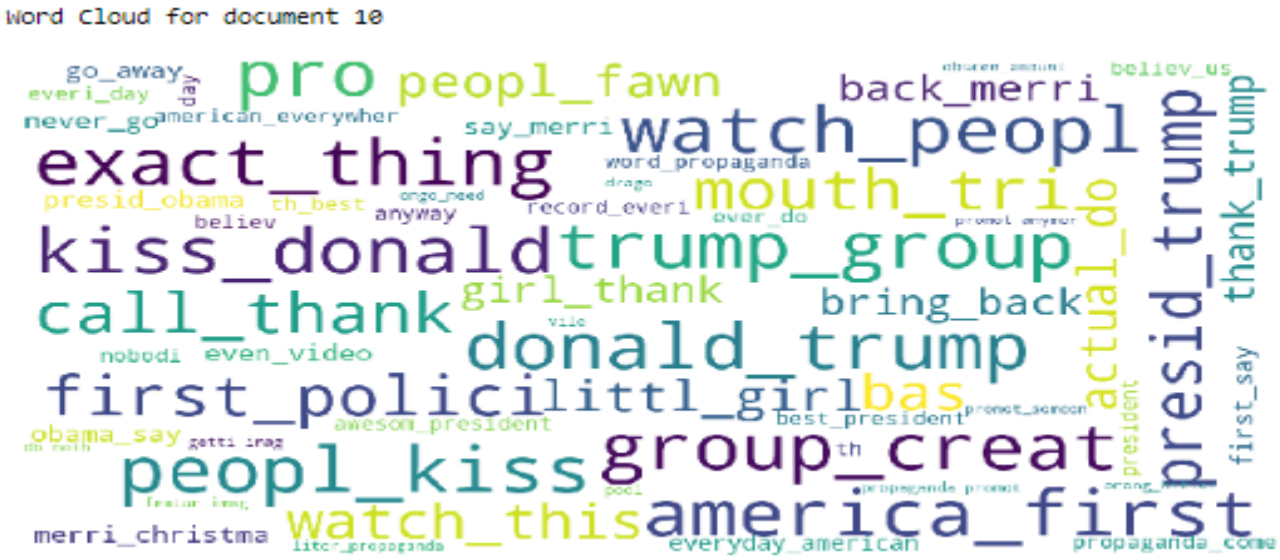
Word Cloud for document 7




Word Cloud for document 8


Word Cloud for document 9

6.3. Using Training Data Set to characterize every word



The following results were obtained for True Documents showing CTF and CDF in decreasing order of frequency. (Only Top Results are shown here)

| Word | Cumulative Term Frequency * | Cumulative Document Frequency |
|---|---|---|
| ('donald', 'trump') | 1.097698 | 0.812500 |
| ('presid', 'donald') | 1.013399 | 0.750000 |
| ('u.s.', 'presid') | 0.597634 | 0.225000 |
| ('sourc', 'link') | 0.355721 | 0.100000 |
| ('opinion', 'express') | 0.355721 | 0.100000 |
| ('verifi', 'twitter') | 0.355721 | 0.100000 |
| ('twitter', 'account') | 0.355721 | 0.100000 |
| ('follow', 'statement') | 0.355721 | 0.100000 |
| ('white', 'hous') | 0.350816 | 0.312500 |
| ('unit', 'state') | 0.301944 | 0.337500 |
| ('u.s.', 'hous') | 0.248957 | 0.162500 |
| ('puerto', 'rico') | 0.216320 | 0.100000 |
| ('u.s.', 'virgin') | 0.184678 | 0.062500 |
| ('virgin', 'island') | 0.184678 | 0.062500 |
| ('trump', 'say') | 0.174508 | 0.137500 |
| ('take', 'offic') | 0.168255 | 0.212500 |
| ('make', 'america') | 0.153827 | 0.050000 |
| ('america', 'great') | 0.153827 | 0.050000 |
| ('great', 'again') | 0.148925 | 0.037500 |
| ('special', 'counsel') | 0.147720 | 0.100000 |
| ('last', 'week') | 0.139153 | 0.150000 |
| ('u.s.', 'senat') | 0.132706 | 0.087500 |
| ('offici', 'say') | 0.131768 | 0.125000 |
| ('attorney', 'general') | 0.131513 | 0.112500 |
| ('nation', 'secur') | 0.130599 | 0.100000 |
| ('next', 'year') | 0.127685 | 0.150000 |
| ('natur', 'disast') | 0.126057 | 0.037500 |
| ('govern', 'open') | 0.124603 | 0.037500 |
| ('recoveri', 'effort') | 0.121429 | 0.025000 |
| ('doug', 'jone') | 0.119044 | 0.075000 |
| ('u.s.', 'state') | 0.118056 | 0.062500 |
| ('govern', 'fund') | 0.115912 | 0.087500 |
| ('feder', 'judg') | 0.109280 | 0.125000 |
| ('trump', 'administr') | 0.109013 | 0.162500 |
| ('justic', 'depart') | 0.108888 | 0.137500 |

Results showing in decreasing order of CTF for True Documents

6.3. Using Training Data Set to characterize every word

| Word | Cumulative Term Frequency | Cumulative Document Frequency * |
|---|---|---|
| ('donald', 'trump') | 1.097698 | 0.812500 |
| ('presid', 'donald') | 1.013399 | 0.750000 |
| ('unit', 'state') | 0.301944 | 0.337500 |
| ('white', 'hous') | 0.350816 | 0.312500 |
| ('u.s.', 'presid') | 0.597634 | 0.225000 |
| ('take', 'offic') | 0.168255 | 0.212500 |
| ('trump', 'administr') | 0.109013 | 0.162500 |
| ('u.s.', 'hous') | 0.248957 | 0.162500 |
| ('last', 'week') | 0.139153 | 0.150000 |
| ('next', 'year') | 0.127685 | 0.150000 |
| ('justic', 'depart') | 0.108888 | 0.137500 |
| ('trump', 'say') | 0.174508 | 0.137500 |
| ('hillari', 'clinton') | 0.106390 | 0.137500 |
| ('feder', 'judg') | 0.109280 | 0.125000 |
| ('offici', 'say') | 0.131768 | 0.125000 |
| ('suprem', 'court') | 0.090735 | 0.125000 |
| ('congression', 'elect') | 0.060574 | 0.112500 |
| ('attorney', 'general') | 0.131513 | 0.112500 |
| ('last', 'year') | 0.077001 | 0.112500 |
| ('puerto', 'rico') | 0.216320 | 0.100000 |
| ('republican', 'presid') | 0.069341 | 0.100000 |
| ('robert', 'mueller') | 0.088920 | 0.100000 |
| ('special', 'counsel') | 0.147720 | 0.100000 |
| ('counsel', 'robert') | 0.088920 | 0.100000 |
| ('legisl', 'victori') | 0.077102 | 0.100000 |
| ('sourc', 'link') | 0.355721 | 0.100000 |
| ('opinion', 'express') | 0.355721 | 0.100000 |
| ('verifi', 'twitter') | 0.355721 | 0.100000 |
| ('twitter', 'account') | 0.355721 | 0.100000 |
| ('follow', 'statement') | 0.355721 | 0.100000 |
| ('u.s.', 'suprem') | 0.073427 | 0.100000 |
| ('u.s.', 'district') | 0.080259 | 0.100000 |
| ('district', 'judg') | 0.080259 | 0.100000 |
| ('nation', 'secur') | 0.130599 | 0.100000 |
| ('hous', 'republican') | 0.088407 | 0.087500 |

Results showing in decreasing order of CDF for True Documents

6.3. Using Training Data Set to characterize every word

The following results were obtained for Fake Documents showing CTF and CDF (Only Top Results are shown here)

| Word | Cumulative Term Frequency * | Cumulative Document Frequency |
|---|---|---|
| ('donald', 'trump') | 0.915427 | 0.750000 |
| ('white', 'hous') | 0.327479 | 0.275000 |
| ('unit', 'state') | 0.220270 | 0.175000 |
| ('robert', 'mueller') | 0.209990 | 0.187500 |
| ('realiti', 'show') | 0.207299 | 0.175000 |
| ('show', 'star') | 0.207299 | 0.175000 |
| ('former', 'realiti') | 0.207299 | 0.175000 |
| ('featur', 'imag') | 0.200449 | 0.162500 |
| ('nation', 'secur') | 0.187988 | 0.137500 |
| ('screen', 'captur') | 0.185067 | 0.137500 |
| ('hillari', 'clinton') | 0.165637 | 0.137500 |
| ('fake', 'news') | 0.163889 | 0.112500 |
| ('special', 'counsel') | 0.151018 | 0.125000 |
| ('secur', 'advis') | 0.136081 | 0.112500 |
| ('child', 'molest') | 0.134705 | 0.100000 |
| ('look', 'like') | 0.131498 | 0.112500 |
| ('trump', 'say') | 0.130681 | 0.100000 |
| ('senat', 'candid') | 0.128855 | 0.100000 |
| ('counsel', 'robert') | 0.124916 | 0.100000 |
| ('doug', 'jone') | 0.122135 | 0.100000 |
| ('presid', 'trump') | 0.121102 | 0.112500 |
| ('chip', 'somodevilla/getti') | 0.120717 | 0.100000 |
| ('somodevilla/getti', 'imag') | 0.120717 | 0.100000 |
| ('michael', 'flynn') | 0.120368 | 0.100000 |
| ('attorney', 'general') | 0.117741 | 0.087500 |
| ('barack', 'obama') | 0.110332 | 0.087500 |
| ('sexual', 'harass') | 0.109452 | 0.087500 |
| ('teenag', 'girl') | 0.108147 | 0.087500 |
| ('republican', 'parti') | 0.107540 | 0.087500 |
| ('former', 'nation') | 0.103025 | 0.075000 |
| ('trump', 'tweet') | 0.100692 | 0.087500 |
| ('sexual', 'predat') | 0.098516 | 0.062500 |
| ('york', 'time') | 0.097985 | 0.087500 |
| ('oval', 'offic') | 0.095673 | 0.075000 |
| ('suprem', 'court') | 0.093008 | 0.075000 |

Results showing in decreasing order of CTF for Fake Documents

6.3. Using Training Data Set to characterize every word

| Word | Cumulative Term Frequency | Cumulative Document Frequency * |
|---|---|---|
| ('donald', 'trump') | 0.915427 | 0.750000 |
| ('white', 'hous') | 0.327479 | 0.275000 |
| ('robert', 'mueller') | 0.209990 | 0.187500 |
| ('realiti', 'show') | 0.207299 | 0.175000 |
| ('show', 'star') | 0.207299 | 0.175000 |
| ('former', 'realiti') | 0.207299 | 0.175000 |
| ('unit', 'state') | 0.220270 | 0.175000 |
| ('featur', 'imag') | 0.200449 | 0.162500 |
| ('hillari', 'clinton') | 0.165637 | 0.137500 |
| ('screen', 'captur') | 0.185067 | 0.137500 |
| ('nation', 'secur') | 0.187988 | 0.137500 |
| ('special', 'counsel') | 0.151018 | 0.125000 |
| ('fake', 'news') | 0.163889 | 0.112500 |
| ('look', 'like') | 0.131498 | 0.112500 |
| ('presid', 'trump') | 0.121102 | 0.112500 |
| ('secur', 'advis') | 0.136081 | 0.112500 |
| ('counsel', 'robert') | 0.124916 | 0.100000 |
| ('trump', 'say') | 0.130681 | 0.100000 |
| ('chip', 'somodevilla/getti') | 0.120717 | 0.100000 |
| ('somodevilla/getti', 'imag') | 0.120717 | 0.100000 |
| ('doug', 'jone') | 0.122135 | 0.100000 |
| ('senat', 'candid') | 0.128855 | 0.100000 |
| ('child', 'molest') | 0.134705 | 0.100000 |
| ('michael', 'flynn') | 0.120368 | 0.100000 |
| ('even', 'though') | 0.090462 | 0.087500 |
| ('barack', 'obama') | 0.110332 | 0.087500 |
| ('york', 'time') | 0.097985 | 0.087500 |
| ('republican', 'parti') | 0.107540 | 0.087500 |
| ('teenag', 'girl') | 0.108147 | 0.087500 |
| ('attorney', 'general') | 0.117741 | 0.087500 |
| ('sexual', 'harass') | 0.109452 | 0.087500 |
| ('trump', 'tweet') | 0.100692 | 0.087500 |
| ('mani', 'peopl') | 0.081824 | 0.075000 |
| ('plead', 'guilti') | 0.092121 | 0.075000 |
| ('former', 'presid') | 0.078686 | 0.075000 |

Results showing in decreasing order of CDF for Fake Documents

The following results were obtained for Net CTF and Net CDF for each bigram.

| Word | Net Cumulative Term Frequency * | Net Cumulative Document Frequency |
|---|---|---|
| ('presid', 'donald') | 0.957153 | 0.687500 |
| ('u.s.', 'presid') | 0.587733 | 0.212500 |
| ('sourc', 'link') | 0.355721 | 0.100000 |
| ('opinion', 'express') | 0.355721 | 0.100000 |
| ('verifi', 'twitter') | 0.355721 | 0.100000 |
| ('follow', 'statement') | 0.344485 | 0.087500 |
| ('twitter', 'account') | 0.331018 | 0.075000 |
| ('u.s.', 'hous') | 0.248957 | 0.162500 |
| ('puerto', 'rico') | 0.216320 | 0.100000 |
| ('u.s.', 'virgin') | 0.184678 | 0.062500 |
| ('virgin', 'island') | 0.184678 | 0.062500 |
| ('donald', 'trump') | 0.182271 | 0.062500 |
| ('take', 'offic') | 0.151306 | 0.200000 |
| ('make', 'america') | 0.136878 | 0.037500 |
| ('america', 'great') | 0.136878 | 0.037500 |
| ('great', 'again') | 0.131976 | 0.025000 |
| ('offici', 'say') | 0.131768 | 0.125000 |
| ('natur', 'disast') | 0.126057 | 0.037500 |
| ('last', 'week') | 0.125264 | 0.137500 |
| ('recoveri', 'effort') | 0.121429 | 0.025000 |
| ('u.s.', 'senat') | 0.120047 | 0.075000 |
| ('u.s.', 'state') | 0.118056 | 0.062500 |
| ('govern', 'fund') | 0.115912 | 0.087500 |
| ('feder', 'judg') | 0.109280 | 0.125000 |
| ('trump', 'administr') | 0.109013 | 0.162500 |
| ('govern', 'open') | 0.108730 | 0.025000 |
| ('econom', 'advis') | 0.107175 | 0.037500 |
| ('spend', 'bill') | 0.102950 | 0.075000 |
| ('u.s.', 'elect') | 0.097299 | 0.087500 |
| ('justic', 'depart') | 0.093736 | 0.125000 |
| ('john', 'merril') | 0.092817 | 0.037500 |
| ('state', 'john') | 0.092817 | 0.037500 |
| ('fulli', 'digest') | 0.090909 | 0.012500 |
| ('legisl', 'expect') | 0.090909 | 0.012500 |
| ('even', 'begin') | 0.090909 | 0.012500 |

Results showing in decreasing order of Net CTF

**Positive means the word has True Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency | Net Cumulative Document Frequency * |
|---|---|---|
| ('presid', 'donald') | 0.957153 | 0.687500 |
| ('u.s.', 'presid') | 0.587733 | 0.212500 |
| ('take', 'offic') | 0.151306 | 0.200000 |
| ('unit', 'state') | 0.081674 | 0.162500 |
| ('trump', 'administr') | 0.109013 | 0.162500 |
| ('u.s.', 'hous') | 0.248957 | 0.162500 |
| ('last', 'week') | 0.125264 | 0.137500 |
| ('feder', 'judg') | 0.109280 | 0.125000 |
| ('offici', 'say') | 0.131768 | 0.125000 |
| ('justic', 'depart') | 0.093736 | 0.125000 |
| ('congression', 'elect') | 0.060574 | 0.112500 |
| ('puerto', 'rico') | 0.216320 | 0.100000 |
| ('republican', 'presid') | 0.069341 | 0.100000 |
| ('legisl', 'victori') | 0.077102 | 0.100000 |
| ('sourc', 'link') | 0.355721 | 0.100000 |
| ('opinion', 'express') | 0.355721 | 0.100000 |
| ('verifi', 'twitter') | 0.355721 | 0.100000 |
| ('u.s.', 'suprem') | 0.073427 | 0.100000 |
| ('next', 'year') | 0.065674 | 0.100000 |
| ('follow', 'statement') | 0.344485 | 0.087500 |
| ('u.s.', 'district') | 0.064386 | 0.087500 |
| ('district', 'judg') | 0.064386 | 0.087500 |
| ('hous', 'republican') | 0.088407 | 0.087500 |
| ('u.s.', 'elect') | 0.097299 | 0.087500 |
| ('govern', 'fund') | 0.115912 | 0.087500 |
| ('twitter', 'account') | 0.331018 | 0.075000 |
| ('republican', 'leader') | 0.067440 | 0.075000 |
| ('appeal', 'court') | 0.058488 | 0.075000 |
| ('feder', 'bureau') | 0.067446 | 0.075000 |
| ('republican', 'lawmak') | 0.064405 | 0.075000 |
| ('first', 'major') | 0.055980 | 0.075000 |
| ('fiscal', 'year') | 0.051422 | 0.075000 |
| ('u.s.', 'senat') | 0.120047 | 0.075000 |
| ('victori', 'sinc') | 0.047807 | 0.075000 |
| ('spend', 'bill') | 0.102950 | 0.075000 |

Results showing in decreasing order of Net CDF

**Positive means the word has True Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency * | Net Cumulative Document Frequency |
|---|---|---|
| ('realiti', 'show') | -0.207299 | -0.175000 |
| ('show', 'star') | -0.207299 | -0.175000 |
| ('former', 'realiti') | -0.207299 | -0.175000 |
| ('featur', 'imag') | -0.200449 | -0.162500 |
| ('screen', 'captur') | -0.185067 | -0.137500 |
| ('child', 'molest') | -0.134705 | -0.100000 |
| ('senat', 'candid') | -0.128855 | -0.100000 |
| ('secur', 'advis') | -0.126990 | -0.100000 |
| ('robert', 'mueller') | -0.121071 | -0.087500 |
| ('chip', 'somodevilla/getti') | -0.120717 | -0.100000 |
| ('somodevilla/getti', 'imag') | -0.120717 | -0.100000 |
| ('michael', 'flynn') | -0.111278 | -0.087500 |
| ('trump', 'tweet') | -0.100692 | -0.087500 |
| ('sexual', 'predat') | -0.098516 | -0.062500 |
| ('sexual', 'harass') | -0.094959 | -0.075000 |
| ('former', 'nation') | -0.093934 | -0.062500 |
| ('alex', 'wong/getti') | -0.083556 | -0.075000 |
| ('wong/getti', 'imag') | -0.083556 | -0.075000 |
| ('time', 'magazin') | -0.082182 | -0.050000 |
| ('mani', 'peopl') | -0.081824 | -0.075000 |
| ('drew', 'angerer/getti') | -0.081479 | -0.062500 |
| ('angerer/getti', 'imag') | -0.081479 | -0.062500 |
| ('presid', 'trump') | -0.080858 | -0.062500 |
| ('plead', 'guilti') | -0.079926 | -0.062500 |
| ('alabama', 'suprem') | -0.079119 | -0.062500 |
| ('alabama', 'senat') | -0.078536 | -0.062500 |
| ('trump', 'support') | -0.076862 | -0.075000 |
| ('golf', 'cours') | -0.075457 | -0.050000 |
| ('oval', 'offic') | -0.075265 | -0.062500 |
| ('saturday', 'night') | -0.073072 | -0.062500 |
| ('white', 'supremacist') | -0.071680 | -0.062500 |
| ('republican', 'parti') | -0.071326 | -0.025000 |
| ('transit', 'team') | -0.071218 | -0.050000 |
| ('nativ', 'american') | -0.070228 | -0.025000 |
| ('orang', 'overlord') | -0.069848 | -0.050000 |

Results showing in increasing order of Net CTF

**Negative means the word has Fake Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency | Net Cumulative Document Frequency * |
|---|---|---|
| ('realiti', 'show') | -0.207299 | -0.175000 |
| ('show', 'star') | -0.207299 | -0.175000 |
| ('former', 'realiti') | -0.207299 | -0.175000 |
| ('featur', 'imag') | -0.200449 | -0.162500 |
| ('screen', 'captur') | -0.185067 | -0.137500 |
| ('look', 'like') | -0.048165 | -0.100000 |
| ('secur', 'advis') | -0.126990 | -0.100000 |
| ('chip', 'somodevilla/getti') | -0.120717 | -0.100000 |
| ('somodevilla/getti', 'imag') | -0.120717 | -0.100000 |
| ('senat', 'candid') | -0.128855 | -0.100000 |
| ('child', 'molest') | -0.134705 | -0.100000 |
| ('michael', 'flynn') | -0.111278 | -0.087500 |
| ('robert', 'mueller') | -0.121071 | -0.087500 |
| ('trump', 'tweet') | -0.100692 | -0.087500 |
| ('fake', 'news') | -0.062583 | -0.075000 |
| ('sexual', 'harass') | -0.094959 | -0.075000 |
| ('mani', 'peopl') | -0.081824 | -0.075000 |
| ('alex', 'wong/getti') | -0.083556 | -0.075000 |
| ('wong/getti', 'imag') | -0.083556 | -0.075000 |
| ('trump', 'support') | -0.076862 | -0.075000 |
| ('presid', 'trump') | -0.080858 | -0.062500 |
| ('plead', 'guilti') | -0.079926 | -0.062500 |
| ('oval', 'offic') | -0.075265 | -0.062500 |
| ('former', 'nation') | -0.093934 | -0.062500 |
| ('white', 'supremacist') | -0.071680 | -0.062500 |
| ('saturday', 'night') | -0.073072 | -0.062500 |
| ('alabama', 'suprem') | -0.079119 | -0.062500 |
| ('sexual', 'predat') | -0.098516 | -0.062500 |
| ('drew', 'angerer/getti') | -0.081479 | -0.062500 |
| ('angerer/getti', 'imag') | -0.081479 | -0.062500 |
| ('alabama', 'senat') | -0.078536 | -0.062500 |
| ('advis', 'michael') | -0.065844 | -0.050000 |
| ('andrew', 'burton/getti') | -0.064516 | -0.050000 |
| ('burton/getti', 'imag') | -0.064516 | -0.050000 |
| ('golf', 'cours') | -0.075457 | -0.050000 |

Results showing in increasing order of Net CDF

**Negative means the word has Fake Characteristics**

## 6.3.3. Trigram Word Frequency Model

The following results were obtained for True Documents*:

```
3022 trigrams were found in the corpus
The unique identified trigrams with their corresponding word frequencies for top 50 trigrams from true news articles are given
below:
        ('presid', 'donald', 'trump')   (60)
        ('u.s.', 'presid', 'donald')   (17)
        ('u.s.', 'suprem', 'court')   (8)
        ('u.s.', 'district', 'judg')   (8)
        ('counsel', 'robert', 'mueller')   (8)
        ('special', 'counsel', 'robert')   (8)
        ('verifi', 'twitter', 'account')   (8)
        ('presid', 'barack', 'obama')   (6)
        ('major', 'legisl', 'victori')   (6)
        ('speaker', 'paul', 'ryan')   (6)
        ('leader', 'mitch', 'mcconnel')   (6)
        ('first', 'major', 'legisl')   (6)
        ('hous', 'speaker', 'paul')   (5)
        ('nation', 'secur', 'agenc')   (5)
        ('u.s.', 'justic', 'depart')   (5)
        ('major', 'leader', 'mitch')   (5)
        ('u.s.', 'virgin', 'island')   (5)
        ('trump', 'take', 'offic')   (5)
        ('foreign', 'intellig', 'surveil')   (5)
        ('senat', 'major', 'leader')   (5)
        ('sinc', 'take', 'offic')   (5)
        ('2018', 'congression', 'elect')   (4)
        ('former', 'presid', 'barack')   (4)
        ('internet', 'surveil', 'program')   (4)
        ('white', 'hous', 'sinc')   (4)
        ('white', 'hous', 'offici')   (4)
        ('west', 'palm', 'beach')   (4)
        ('make', 'america', 'great')   (4)
        ('govern', 'fund', 'bill')   (4)
        ('partial', 'govern', 'shutdown')   (3)
        ('foreign', 'suspect', 'live')   (3)
        ('incumb', 'david', 'yancey')   (3)
        ('medic', 'standard', 'need')   (3)
        ('spokeswoman', 'lauren', 'ehrsam')   (3)
        ('attorney', 'general', 'jeff')   (3)
        ('state', 'john', 'merril')   (3)
        ('process', 'transgend', 'applic')   (3)
        ('begin', 'accept', 'transgend')   (3)
        ('deputi', 'attorney', 'general')   (3)
        ('democrat', 'shelli', 'simond')   (3)
        ('lauren', 'ehrsam', 'say')   (3)
        ('hous', 'judiciari', 'committe')   (3)
        ('earli', 'next', 'year')   (3)
        ('trillion', 'nation', 'debt')   (3)
        ('2016', 'presidenti', 'elect')   (3)
        ('director', 'jame', 'comey')   (3)
        ('hous', 'last', 'year')   (3)
        ('legisl', 'victori', 'sinc')   (3)
        ('expir', 'internet', 'surveil')   (3)
        ('america', 'great', 'again')   (3)
```

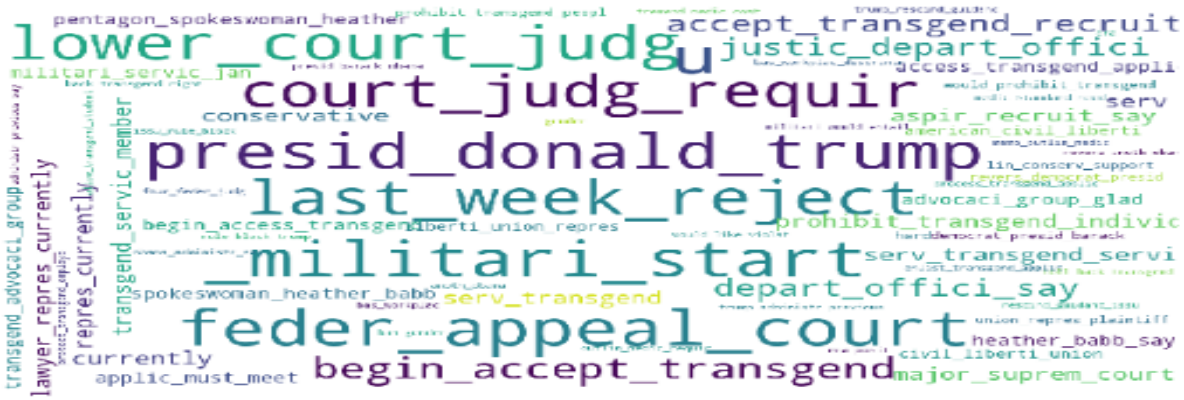
*The complete results are not shown here.

6.3. Using Training Data Set to characterize every word

Trigram Word Clouds for first 10 True News articles:

Word Cloud for document 1

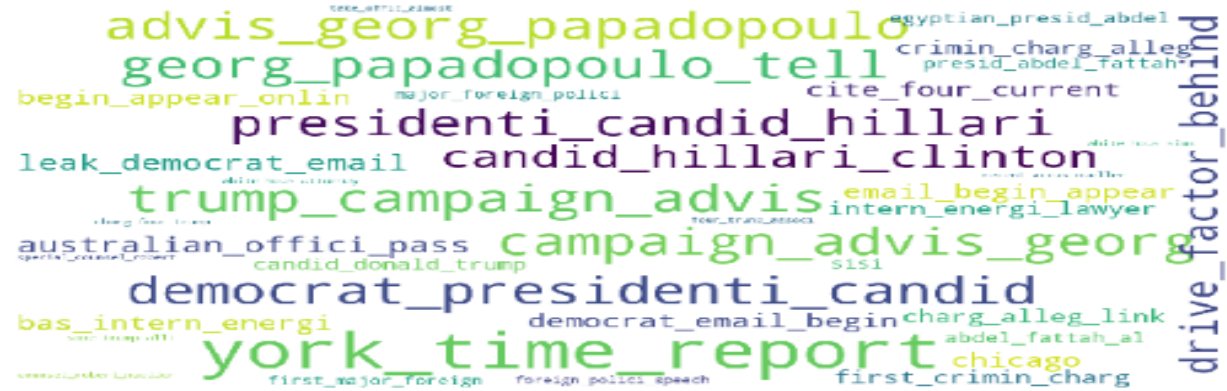

Word Cloud for document 2



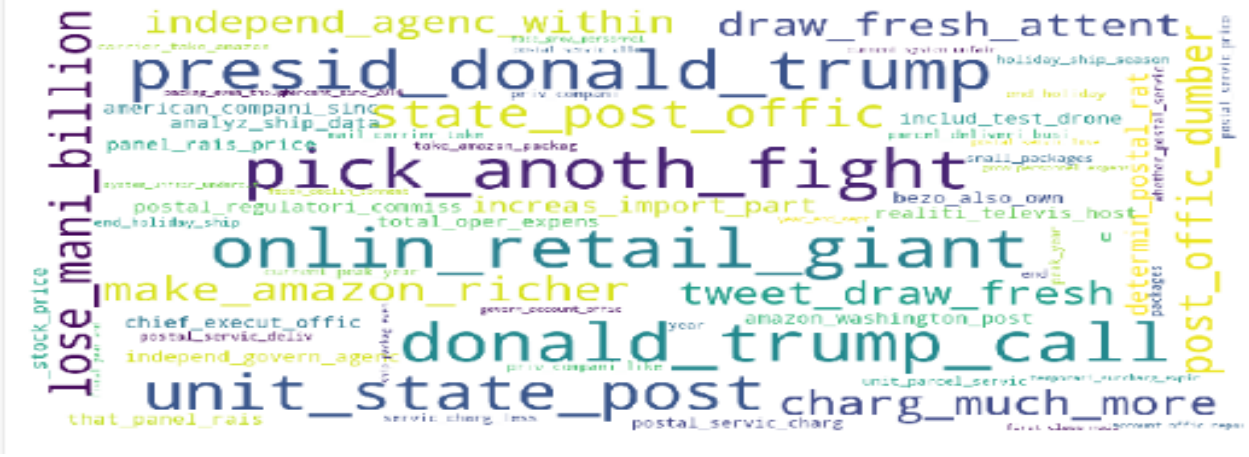

Word Cloud for document 3

6.3. Using Training Data Set to characterize every word

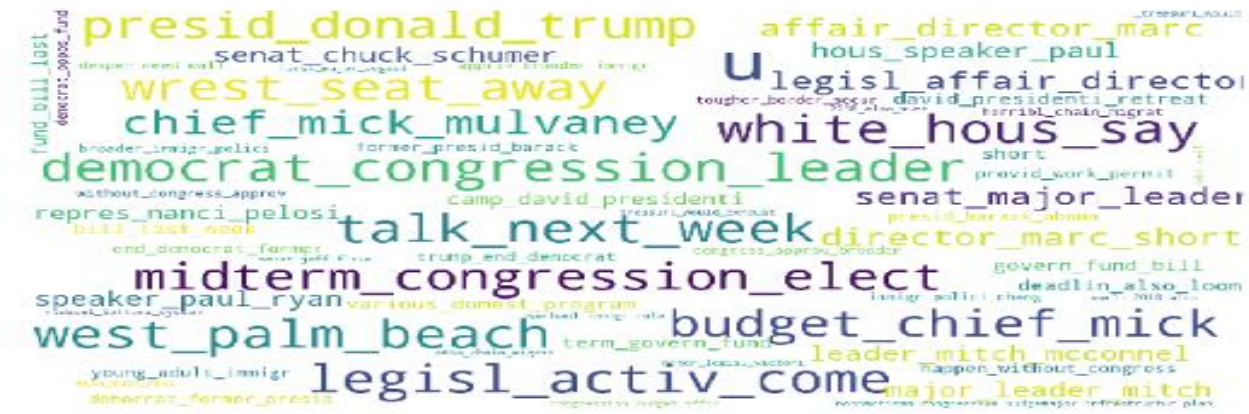
Word Cloud for document 4




Word Cloud for document 5

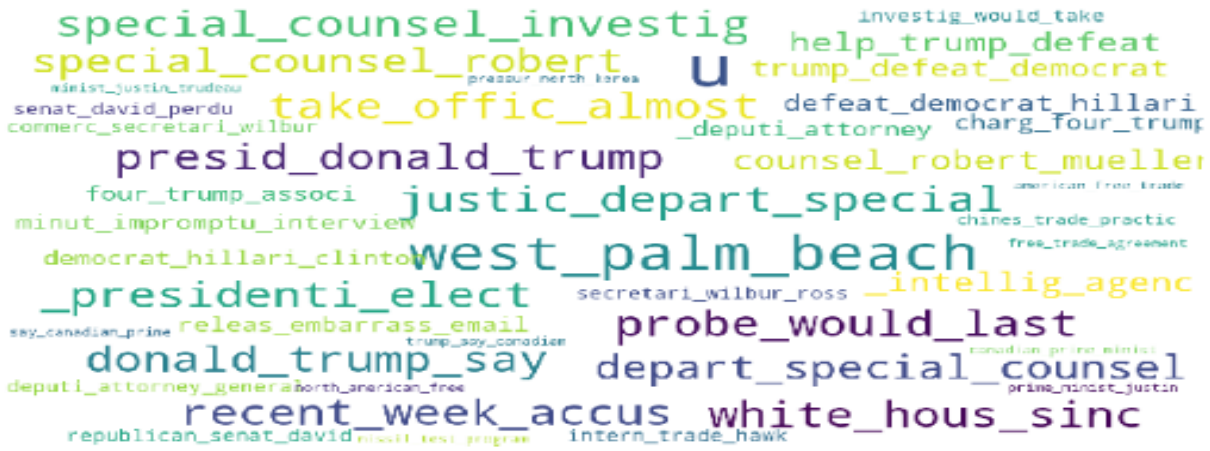



Word Cloud for document 6

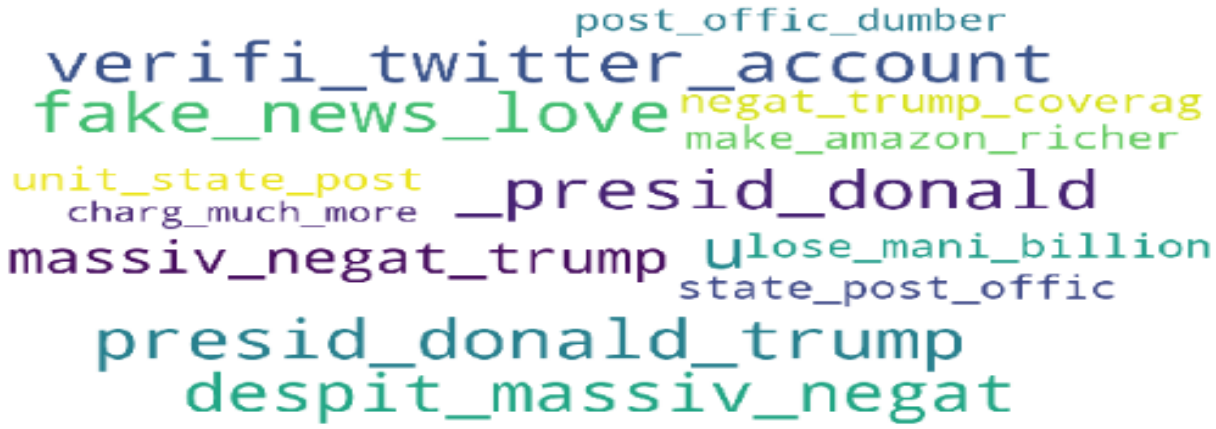


6.3. Using Training Data Set to characterize every word

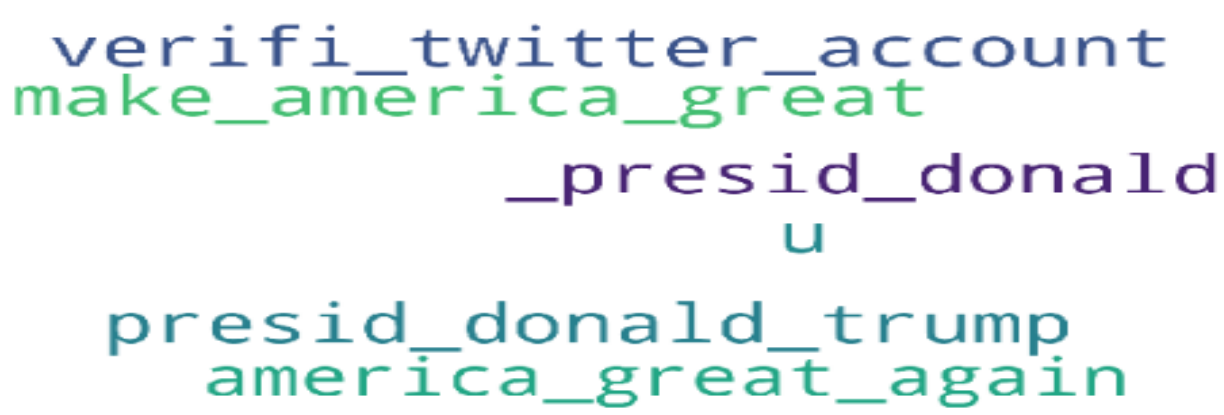
Word Cloud for document 7



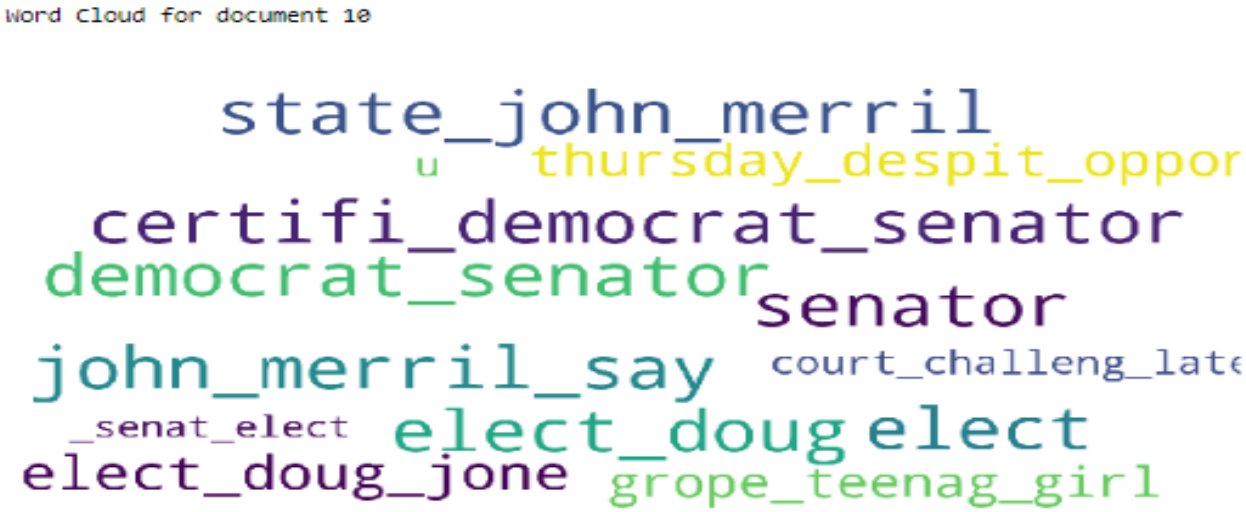

Word Cloud for document 8




Word Cloud for document 9



6.3. Using Training Data Set to characterize every word

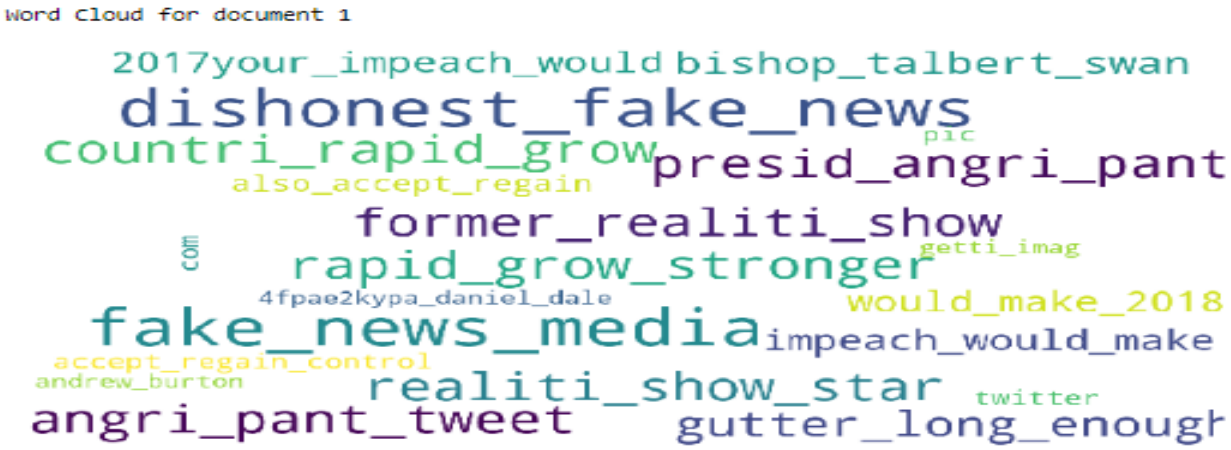


Similarly, the following results were obtained for Fake Documents:

```
2033 trigrams were found in the corpus
The unique identified trigrams with their corresponding word frequencies for top 50 trigrams from fake news articles are given
below:
        ('realiti', 'show', 'star')   (14)
        ('former', 'realiti', 'show')   (14)
        ('nation', 'secur', 'advis')   (9)
        ('counsel', 'robert', 'mueller')   (9)
        ('special', 'counsel', 'robert')   (9)
        ('chip', 'somodevilla/getti', 'imag')   (8)
        ('former', 'nation', 'secur')   (7)
        ('alex', 'wong/getti', 'imag')   (6)
        ('alabama', 'suprem', 'court')   (5)
        ('presid', 'barack', 'obama')   (5)
        ('advis', 'michael', 'flynn')   (5)
        ('drew', 'angerer/getti', 'imag')   (5)
        ('alabama', 'senat', 'candid')   (5)
        ('presid', 'donald', 'trump')   (5)
        ('secur', 'advis', 'michael')   (5)
        ('andrew', 'burton/getti', 'imag')   (4)
        ('accus', 'child', 'molest')   (4)
        ('unit', 'state', 'senat')   (4)
        ('least', 'popular', 'presid')   (3)
        ('hous', 'press', 'secretari')   (3)
        ('saturday', 'night', 'live')   (3)
        ('former', 'presid', 'barack')   (3)
        ('nation', 'secur', 'advisor')   (3)
        ('trump', 'transit', 'team')   (3)
        ('director', 'jame', 'comey')   (3)
        ('fake', 'news', 'media')   (3)
        ('zach', 'gibson', 'pool/getti')   (3)
        ('sarah', 'huckabe', 'sander')   (3)
        ('democrat', 'doug', 'jone')   (3)
        ('senat', 'jeff', 'flake')   (3)
        ('gibson', 'pool/getti', 'imag')   (3)
        ('mark', 'wilson/getti', 'imag')   (3)
        ('//t.co/sioaxatcjp', 'sarah', 'sander')   (2)
        ('michael', 'flynn', 'plead')   (2)
        ('thoma', 'cain/getti', 'imag')   (2)
        ('press', 'secretari', 'sean')   (2)
        ('secretari', 'sean', 'spicer')   (2)
        ('never', 'ask', 'comey')   (2)
        ('secur', 'advisor', 'michael')   (2)
        ('four', 'young', 'black')   (2)
        ('time', 'cover', 'featur')   (2)
        ('donald', 'trump', 'announc')   (2)
        ('stop', 'investig', 'flynn')   (2)
        ('lightweight', 'senat', 'kirsten')   (2)
        ('time', 'magazin', 'call')   (2)
        ('attorney', 'general', 'jeff')   (2)
        ('defens', 'intellig', 'agenc')   (2)
        ('disgrac', 'nation', 'secur')   (2)
        ('star', 'donald', 'trump')   (2)
        ('general', 'eric', 'holder')   (2)
```

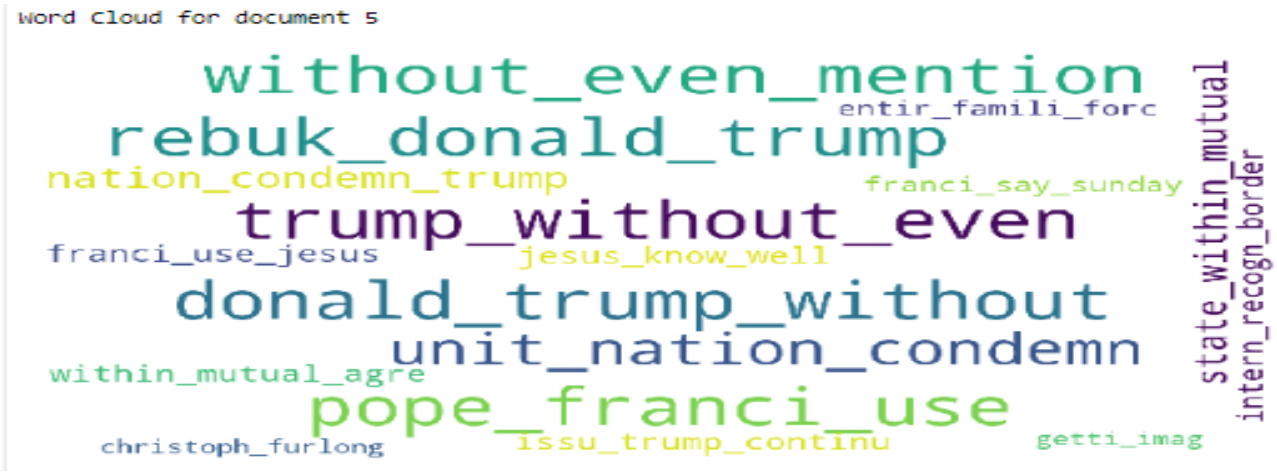
6.3. Using Training Data Set to characterize every word

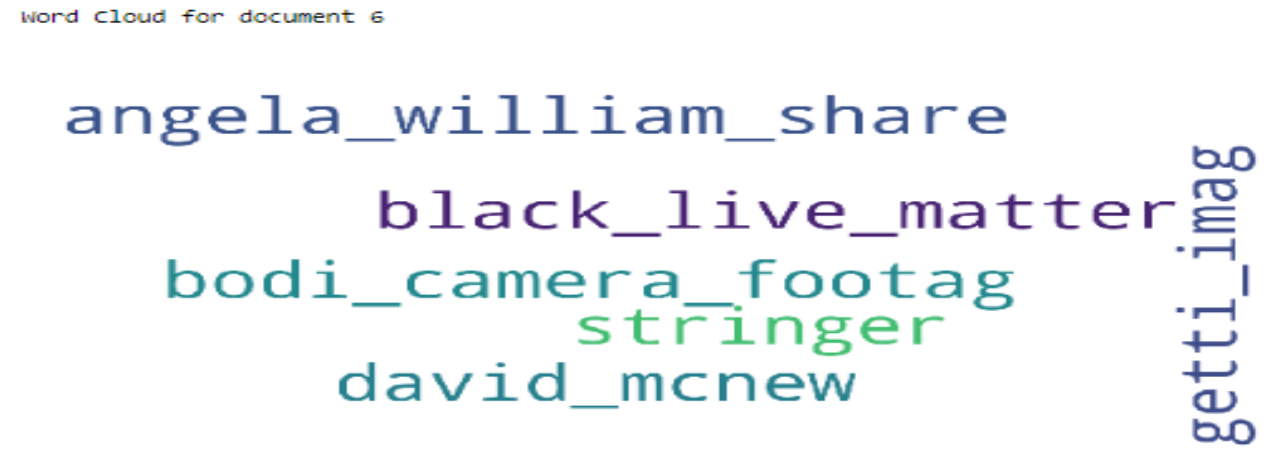
Trigram Word Clouds for first 10 Fake News articles:



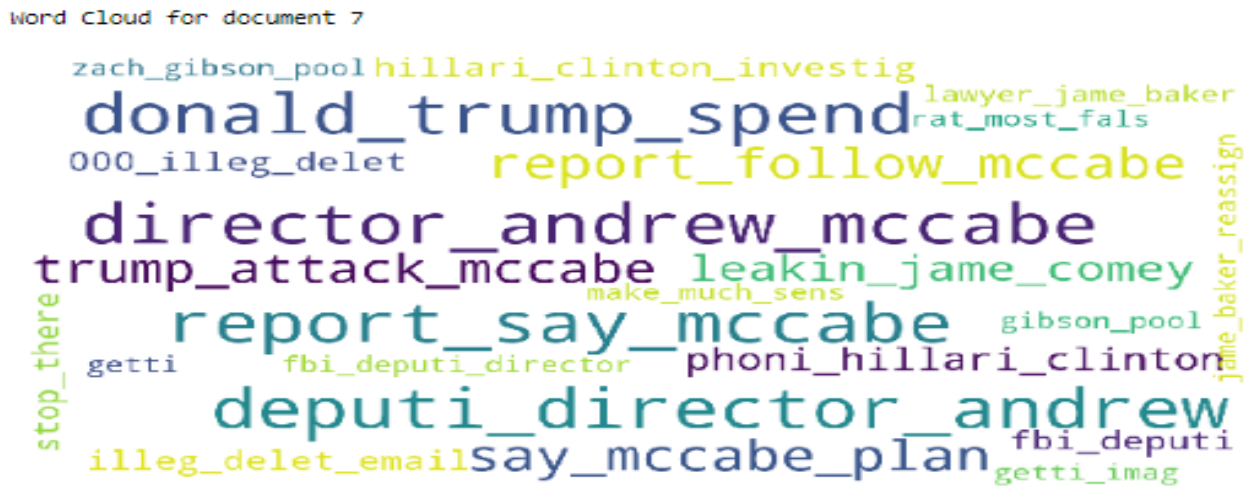


6.3. Using Training Data Set to characterize every word

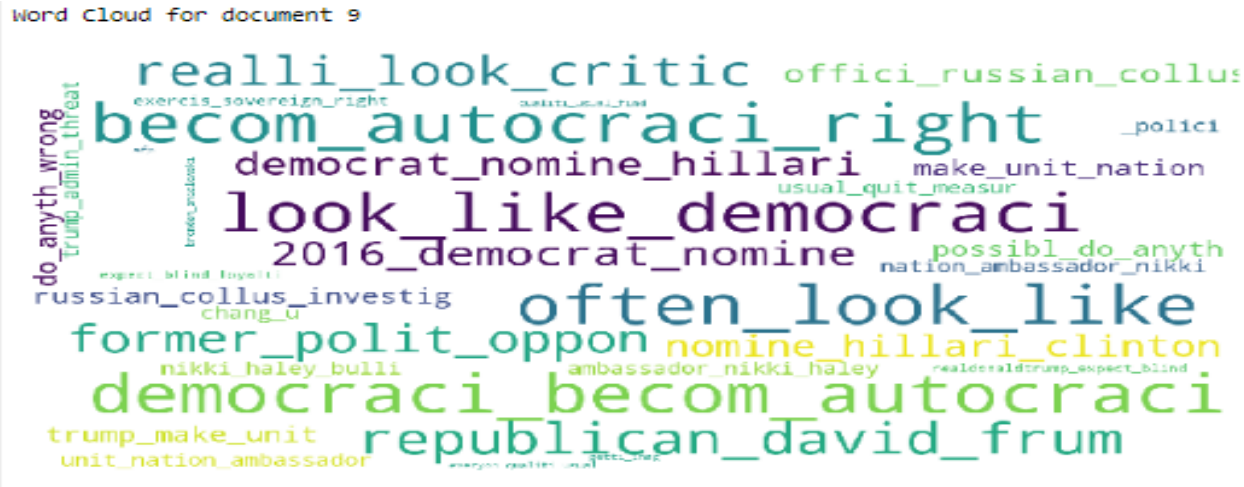



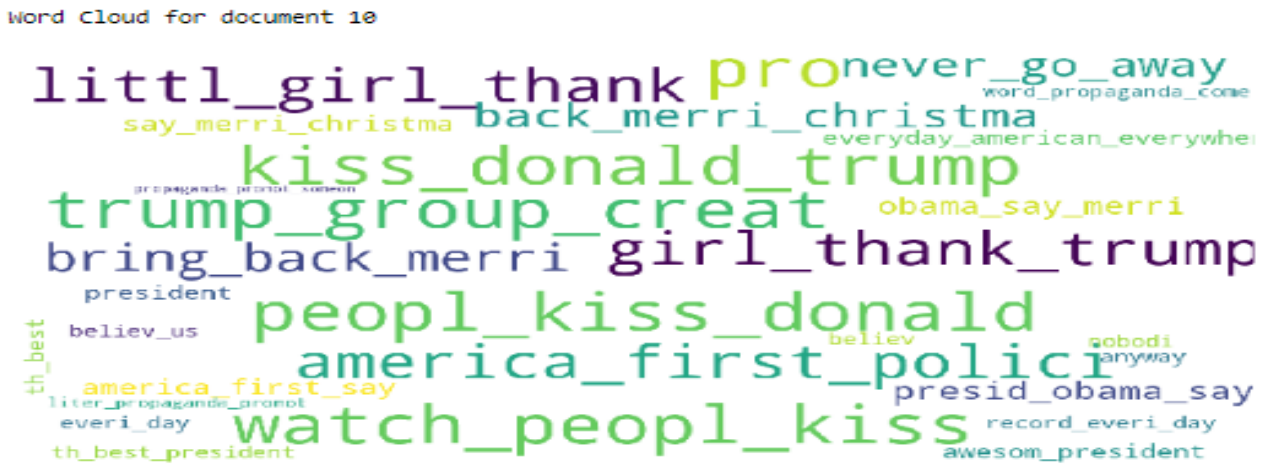
6.3. Using Training Data Set to characterize every word







6.3. Using Training Data Set to characterize every word



The following results were obtained for True Documents showing CTF and CDF in decreasing order of frequency. (Only Top Results are shown here)

| Word | Cumulative Term Frequency * | Cumulative Document Frequency |
|---|---|---|
| ('presid', 'donald', 'trump') | 2.909670 | 0.750000 |
| ('u.s.', 'presid', 'donald') | 1.753158 | 0.212500 |
| ('verifi', 'twitter', 'account') | 1.088467 | 0.100000 |
| ('u.s.', 'virgin', 'island') | 0.502027 | 0.062500 |
| ('make', 'america', 'great') | 0.413648 | 0.050000 |
| ('america', 'great', 'again') | 0.402020 | 0.037500 |
| ('donald', 'trump', 'say') | 0.357143 | 0.025000 |
| ('help', 'puerto', 'rico') | 0.225000 | 0.025000 |
| ('state', 'john', 'merril') | 0.224206 | 0.037500 |
| ('special', 'counsel', 'robert') | 0.223862 | 0.100000 |
| ('counsel', 'robert', 'mueller') | 0.223862 | 0.100000 |
| ('partial', 'govern', 'shutdown') | 0.207079 | 0.037500 |
| ('leader', 'mitch', 'mcconnel') | 0.203468 | 0.075000 |
| ('u.s.', 'district', 'judg') | 0.200234 | 0.100000 |
| ('public', 'sector', 'deficit') | 0.200000 | 0.012500 |
| ('ministri', 'say', 'mexico') | 0.200000 | 0.012500 |
| ('make', 'fiscal', 'chang') | 0.200000 | 0.012500 |
| ('higher', 'public', 'sector') | 0.200000 | 0.012500 |
| ('say', 'mexico', 'would') | 0.200000 | 0.012500 |
| ('state', 'rebuild', 'follow') | 0.200000 | 0.012500 |
| ('widespread', 'recoveri', 'effort') | 0.200000 | 0.012500 |
| ('help', 'widespread', 'recoveri') | 0.200000 | 0.012500 |
| ('major', 'legisl', 'victori') | 0.194165 | 0.075000 |
| ('first', 'major', 'legisl') | 0.194165 | 0.075000 |
| ('u.s.', 'suprem', 'court') | 0.189827 | 0.100000 |
| ('u.s.', 'senat', 'elect') | 0.182540 | 0.025000 |
| ('senat', 'major', 'leader') | 0.180212 | 0.062500 |
| ('major', 'leader', 'mitch') | 0.180212 | 0.062500 |
| ('nation', 'secur', 'agenc') | 0.161133 | 0.062500 |
| ('foreign', 'intellig', 'surveil') | 0.161133 | 0.062500 |
| ('govern', 'fund', 'bill') | 0.156331 | 0.050000 |
| ('white', 'hous', 'offici') | 0.144189 | 0.050000 |
| ('look', 'realli', 'good') | 0.142857 | 0.012500 |
| ('eventu', 'come', 'togeth') | 0.142857 | 0.012500 |
| ('crook', 'hillari', 'pile') | 0.142857 | 0.012500 |

Results showing in decreasing order of CTF for True Documents

6.3. Using Training Data Set to characterize every word

| Word | Cumulative Term Frequency | Cumulative Document Frequency * |
|---|---|---|
| ('presid', 'donald', 'trump') | 2.909670 | 0.750000 |
| ('u.s.', 'presid', 'donald') | 1.753158 | 0.212500 |
| ('special', 'counsel', 'robert') | 0.223862 | 0.100000 |
| ('counsel', 'robert', 'mueller') | 0.223862 | 0.100000 |
| ('verifi', 'twitter', 'account') | 1.088467 | 0.100000 |
| ('u.s.', 'suprem', 'court') | 0.189827 | 0.100000 |
| ('u.s.', 'district', 'judg') | 0.200234 | 0.100000 |
| ('speaker', 'paul', 'ryan') | 0.117333 | 0.075000 |
| ('presid', 'barack', 'obama') | 0.097346 | 0.075000 |
| ('major', 'legisl', 'victori') | 0.194165 | 0.075000 |
| ('first', 'major', 'legisl') | 0.194165 | 0.075000 |
| ('leader', 'mitch', 'mcconnel') | 0.203468 | 0.075000 |
| ('hous', 'speaker', 'paul') | 0.095111 | 0.062500 |
| ('trump', 'take', 'offic') | 0.096909 | 0.062500 |
| ('senat', 'major', 'leader') | 0.180212 | 0.062500 |
| ('major', 'leader', 'mitch') | 0.180212 | 0.062500 |
| ('u.s.', 'justic', 'depart') | 0.124998 | 0.062500 |
| ('sinc', 'take', 'offic') | 0.108799 | 0.062500 |
| ('nation', 'secur', 'agenc') | 0.161133 | 0.062500 |
| ('u.s.', 'virgin', 'island') | 0.502027 | 0.062500 |
| ('foreign', 'intellig', 'surveil') | 0.161133 | 0.062500 |
| ('white', 'hous', 'sinc') | 0.098483 | 0.050000 |
| ('former', 'presid', 'barack') | 0.065973 | 0.050000 |
| ('govern', 'fund', 'bill') | 0.156331 | 0.050000 |
| ('make', 'america', 'great') | 0.413648 | 0.050000 |
| ('2018', 'congression', 'elect') | 0.071400 | 0.050000 |
| ('internet', 'surveil', 'program') | 0.134106 | 0.050000 |
| ('white', 'hous', 'offici') | 0.144189 | 0.050000 |
| ('trillion', 'nation', 'debt') | 0.055717 | 0.037500 |
| ('white', 'hous', 'say') | 0.061890 | 0.037500 |
| ('process', 'transgend', 'applic') | 0.068325 | 0.037500 |
| ('begin', 'accept', 'transgend') | 0.068325 | 0.037500 |
| ('feder', 'appeal', 'court') | 0.068325 | 0.037500 |
| ('accept', 'transgend', 'recruit') | 0.068325 | 0.037500 |
| ('medic', 'standard', 'need') | 0.068325 | 0.037500 |

Results showing in decreasing order of CDF for True Documents

6.3. Using Training Data Set to characterize every word

The following results were obtained for Fake Documents showing CTF and CDF (Only Top Results are shown here)

| Word | Cumulative Term Frequency * | Cumulative Document Frequency |
|---|---|---|
| ('realiti', 'show', 'star') | 0.534854 | 0.175000 |
| ('former', 'realiti', 'show') | 0.534854 | 0.175000 |
| ('chip', 'somodevilla/getti', 'imag') | 0.349954 | 0.100000 |
| ('nation', 'secur', 'advis') | 0.342581 | 0.112500 |
| ('special', 'counsel', 'robert') | 0.318148 | 0.100000 |
| ('counsel', 'robert', 'mueller') | 0.318148 | 0.100000 |
| ('alex', 'wong/getti', 'imag') | 0.272204 | 0.075000 |
| ('former', 'nation', 'secur') | 0.252062 | 0.075000 |
| ('david', 'mcnew/stringer/getti', 'imag') | 0.250000 | 0.012500 |
| ('angela', 'william', 'share') | 0.250000 | 0.012500 |
| ('black', 'live', 'matter') | 0.250000 | 0.012500 |
| ('bodi', 'camera', 'footag') | 0.250000 | 0.012500 |
| ('presid', 'barack', 'obama') | 0.225052 | 0.062500 |
| ('alabama', 'senat', 'candid') | 0.219937 | 0.062500 |
| ('drew', 'angerer/getti', 'imag') | 0.208826 | 0.062500 |
| ('alabama', 'suprem', 'court') | 0.196169 | 0.062500 |
| ('andrew', 'burton/getti', 'imag') | 0.186638 | 0.050000 |
| ('advis', 'michael', 'flynn') | 0.182760 | 0.062500 |
| ('secur', 'advis', 'michael') | 0.182760 | 0.062500 |
| ('accus', 'child', 'molest') | 0.172695 | 0.050000 |
| ('presid', 'donald', 'trump') | 0.160691 | 0.062500 |
| ('unit', 'state', 'senat') | 0.154502 | 0.050000 |
| ('fake', 'news', 'media') | 0.150718 | 0.025000 |
| ('least', 'popular', 'presid') | 0.143306 | 0.037500 |
| ('sarah', 'huckabe', 'sander') | 0.141313 | 0.037500 |
| ('saturday', 'night', 'live') | 0.138346 | 0.037500 |
| ('mark', 'wilson/getti', 'imag') | 0.126489 | 0.037500 |
| ('gibson', 'pool/getti', 'imag') | 0.123232 | 0.037500 |
| ('zach', 'gibson', 'pool/getti') | 0.123232 | 0.037500 |
| ('former', 'presid', 'barack') | 0.121878 | 0.037500 |
| ('lightweight', 'senat', 'kirsten') | 0.114286 | 0.025000 |
| ('senat', 'kirsten', 'gillibrand') | 0.114286 | 0.025000 |
| ('democrat', 'doug', 'jone') | 0.113427 | 0.037500 |
| ('senat', 'jeff', 'flake') | 0.112121 | 0.037500 |
| ('show', 'star', 'donald') | 0.108333 | 0.025000 |

Results showing in decreasing order of CTF for Fake Documents

6.3. Using Training Data Set to characterize every word

| Word | Cumulative Term Frequency | Cumulative Document Frequency * |
|---|---|---|
| ('realiti', 'show', 'star') | 0.534854 | 0.175000 |
| ('former', 'realiti', 'show') | 0.534854 | 0.175000 |
| ('nation', 'secur', 'advis') | 0.342581 | 0.112500 |
| ('special', 'counsel', 'robert') | 0.318148 | 0.100000 |
| ('counsel', 'robert', 'mueller') | 0.318148 | 0.100000 |
| ('chip', 'somodevilla/getti', 'imag') | 0.349954 | 0.100000 |
| ('alex', 'wong/getti', 'imag') | 0.272204 | 0.075000 |
| ('former', 'nation', 'secur') | 0.252062 | 0.075000 |
| ('presid', 'barack', 'obama') | 0.225052 | 0.062500 |
| ('alabama', 'suprem', 'court') | 0.196169 | 0.062500 |
| ('drew', 'angerer/getti', 'imag') | 0.208826 | 0.062500 |
| ('alabama', 'senat', 'candid') | 0.219937 | 0.062500 |
| ('presid', 'donald', 'trump') | 0.160691 | 0.062500 |
| ('advis', 'michael', 'flynn') | 0.182760 | 0.062500 |
| ('secur', 'advis', 'michael') | 0.182760 | 0.062500 |
| ('andrew', 'burton/getti', 'imag') | 0.186638 | 0.050000 |
| ('unit', 'state', 'senat') | 0.154502 | 0.050000 |
| ('accus', 'child', 'molest') | 0.172695 | 0.050000 |
| ('former', 'presid', 'barack') | 0.121878 | 0.037500 |
| ('gibson', 'pool/getti', 'imag') | 0.123232 | 0.037500 |
| ('zach', 'gibson', 'pool/getti') | 0.123232 | 0.037500 |
| ('mark', 'wilson/getti', 'imag') | 0.126489 | 0.037500 |
| ('trump', 'transit', 'team') | 0.092760 | 0.037500 |
| ('democrat', 'doug', 'jone') | 0.113427 | 0.037500 |
| ('saturday', 'night', 'live') | 0.138346 | 0.037500 |
| ('sarah', 'huckabe', 'sander') | 0.141313 | 0.037500 |
| ('hous', 'press', 'secretari') | 0.105896 | 0.037500 |
| ('director', 'jame', 'comey') | 0.099944 | 0.037500 |
| ('nation', 'secur', 'advisor') | 0.095314 | 0.037500 |
| ('senat', 'jeff', 'flake') | 0.112121 | 0.037500 |
| ('least', 'popular', 'presid') | 0.143306 | 0.037500 |
| ('fake', 'news', 'media') | 0.150718 | 0.025000 |
| ('hous', 'intellig', 'committe') | 0.059649 | 0.025000 |
| ('nomine', 'hillari', 'clinton') | 0.067816 | 0.025000 |
| ('donald', 'trump', 'announc') | 0.058874 | 0.025000 |

Results showing in decreasing order of CDF for Fake Documents

The following results were obtained for Net CTF and Net CDF for each trigram.

| Word | Net Cumulative Term Frequency * | Net Cumulative Document Frequency |
|---|---|---|
| ('presid', 'donald', 'trump') | 2.748979 | 0.687500 |
| ('u.s.', 'presid', 'donald') | 1.753158 | 0.212500 |
| ('verifi', 'twitter', 'account') | 1.088467 | 0.100000 |
| ('u.s.', 'virgin', 'island') | 0.502027 | 0.062500 |
| ('donald', 'trump', 'say') | 0.357143 | 0.025000 |
| ('make', 'america', 'great') | 0.354825 | 0.037500 |
| ('america', 'great', 'again') | 0.343197 | 0.025000 |
| ('help', 'puerto', 'rico') | 0.225000 | 0.025000 |
| ('state', 'john', 'merril') | 0.224206 | 0.037500 |
| ('partial', 'govern', 'shutdown') | 0.207079 | 0.037500 |
| ('public', 'sector', 'deficit') | 0.200000 | 0.012500 |
| ('ministri', 'say', 'mexico') | 0.200000 | 0.012500 |
| ('make', 'fiscal', 'chang') | 0.200000 | 0.012500 |
| ('higher', 'public', 'sector') | 0.200000 | 0.012500 |
| ('say', 'mexico', 'would') | 0.200000 | 0.012500 |
| ('state', 'rebuild', 'follow') | 0.200000 | 0.012500 |
| ('widespread', 'recoveri', 'effort') | 0.200000 | 0.012500 |
| ('help', 'widespread', 'recoveri') | 0.200000 | 0.012500 |
| ('major', 'legisl', 'victori') | 0.194165 | 0.075000 |
| ('u.s.', 'suprem', 'court') | 0.189827 | 0.100000 |
| ('u.s.', 'senat', 'elect') | 0.182540 | 0.025000 |
| ('senat', 'major', 'leader') | 0.180212 | 0.062500 |
| ('leader', 'mitch', 'mcconnel') | 0.167754 | 0.062500 |
| ('nation', 'secur', 'agenc') | 0.161133 | 0.062500 |
| ('foreign', 'intellig', 'surveil') | 0.161133 | 0.062500 |
| ('u.s.', 'district', 'judg') | 0.160234 | 0.087500 |
| ('govern', 'fund', 'bill') | 0.156331 | 0.050000 |
| ('major', 'leader', 'mitch') | 0.144498 | 0.050000 |
| ('look', 'realli', 'good') | 0.142857 | 0.012500 |
| ('eventu', 'come', 'togeth') | 0.142857 | 0.012500 |
| ('crook', 'hillari', 'pile') | 0.142857 | 0.012500 |
| ('unpopular', 'individu', 'mandat') | 0.142857 | 0.012500 |
| ('first', 'major', 'legisl') | 0.138609 | 0.062500 |
| ('2016', 'u.s.', 'elect') | 0.134387 | 0.025000 |
| ('internet', 'surveil', 'program') | 0.134106 | 0.050000 |

Results showing in decreasing order of Net CTF

**Positive means the word has True Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency | Net Cumulative Document Frequency * |
|---|---|---|
| ('presid', 'donald', 'trump') | 2.748979 | 0.687500 |
| ('u.s.', 'presid', 'donald') | 1.753158 | 0.212500 |
| ('verifi', 'twitter', 'account') | 1.088467 | 0.100000 |
| ('u.s.', 'suprem', 'court') | 0.189827 | 0.100000 |
| ('u.s.', 'district', 'judg') | 0.160234 | 0.087500 |
| ('major', 'legisl', 'victori') | 0.194165 | 0.075000 |
| ('speaker', 'paul', 'ryan') | 0.081619 | 0.062500 |
| ('trump', 'take', 'offic') | 0.096909 | 0.062500 |
| ('senat', 'major', 'leader') | 0.180212 | 0.062500 |
| ('first', 'major', 'legisl') | 0.138609 | 0.062500 |
| ('leader', 'mitch', 'mcconnel') | 0.167754 | 0.062500 |
| ('u.s.', 'justic', 'depart') | 0.124998 | 0.062500 |
| ('sinc', 'take', 'offic') | 0.108799 | 0.062500 |
| ('nation', 'secur', 'agenc') | 0.161133 | 0.062500 |
| ('u.s.', 'virgin', 'island') | 0.502027 | 0.062500 |
| ('foreign', 'intellig', 'surveil') | 0.161133 | 0.062500 |
| ('hous', 'speaker', 'paul') | 0.059397 | 0.050000 |
| ('white', 'hous', 'sinc') | 0.098483 | 0.050000 |
| ('major', 'leader', 'mitch') | 0.144498 | 0.050000 |
| ('govern', 'fund', 'bill') | 0.156331 | 0.050000 |
| ('2018', 'congression', 'elect') | 0.071400 | 0.050000 |
| ('internet', 'surveil', 'program') | 0.134106 | 0.050000 |
| ('make', 'america', 'great') | 0.354825 | 0.037500 |
| ('white', 'hous', 'offici') | 0.117162 | 0.037500 |
| ('trillion', 'nation', 'debt') | 0.055717 | 0.037500 |
| ('process', 'transgend', 'applic') | 0.068325 | 0.037500 |
| ('begin', 'accept', 'transgend') | 0.068325 | 0.037500 |
| ('feder', 'appeal', 'court') | 0.068325 | 0.037500 |
| ('accept', 'transgend', 'recruit') | 0.068325 | 0.037500 |
| ('medic', 'standard', 'need') | 0.068325 | 0.037500 |
| ('candid', 'hillari', 'clinton') | 0.070304 | 0.037500 |
| ('presidenti', 'candid', 'hillari') | 0.070304 | 0.037500 |
| ('west', 'palm', 'beach') | 0.114286 | 0.037500 |
| ('state', 'john', 'merril') | 0.224206 | 0.037500 |
| ('legisl', 'victori', 'sinc') | 0.043660 | 0.037500 |

Results showing in decreasing order of Net CDF

**Positive means the word has True Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency * | Net Cumulative Document Frequency |
|---|---|---|
| ('realiti', 'show', 'star') | -0.534854 | -0.175000 |
| ('former', 'realiti', 'show') | -0.534854 | -0.175000 |
| ('chip', 'somodevilla/getti', 'imag') | -0.349954 | -0.100000 |
| ('nation', 'secur', 'advis') | -0.324399 | -0.100000 |
| ('alex', 'wong/getti', 'imag') | -0.272204 | -0.075000 |
| ('david', 'mcnew/stringer/getti', 'imag') | -0.250000 | -0.012500 |
| ('angela', 'william', 'share') | -0.250000 | -0.012500 |
| ('black', 'live', 'matter') | -0.250000 | -0.012500 |
| ('bodi', 'camera', 'footag') | -0.250000 | -0.012500 |
| ('former', 'nation', 'secur') | -0.233880 | -0.062500 |
| ('alabama', 'senat', 'candid') | -0.219937 | -0.062500 |
| ('drew', 'angerer/getti', 'imag') | -0.208826 | -0.062500 |
| ('alabama', 'suprem', 'court') | -0.196169 | -0.062500 |
| ('andrew', 'burton/getti', 'imag') | -0.186638 | -0.050000 |
| ('accus', 'child', 'molest') | -0.172695 | -0.050000 |
| ('advis', 'michael', 'flynn') | -0.164578 | -0.050000 |
| ('secur', 'advis', 'michael') | -0.164578 | -0.050000 |
| ('least', 'popular', 'presid') | -0.143306 | -0.037500 |
| ('sarah', 'huckabe', 'sander') | -0.141313 | -0.037500 |
| ('saturday', 'night', 'live') | -0.138346 | -0.037500 |
| ('presid', 'barack', 'obama') | -0.127707 | 0.012500 |
| ('mark', 'wilson/getti', 'imag') | -0.126489 | -0.037500 |
| ('gibson', 'pool/getti', 'imag') | -0.123232 | -0.037500 |
| ('zach', 'gibson', 'pool/getti') | -0.123232 | -0.037500 |
| ('lightweight', 'senat', 'kirsten') | -0.114286 | -0.025000 |
| ('show', 'star', 'donald') | -0.108333 | -0.025000 |
| ('star', 'donald', 'trump') | -0.108333 | -0.025000 |
| ('destruct', 'radic', 'islam') | -0.108173 | -0.025000 |
| ('take', 'place', 'within') | -0.108173 | -0.025000 |
| ('prime', 'minist', 'theresa') | -0.108173 | -0.025000 |
| ('radic', 'islam', 'terror') | -0.108173 | -0.025000 |
| ('hous', 'press', 'secretari') | -0.105896 | -0.037500 |
| ('dishonest', 'fake', 'news') | -0.105263 | -0.012500 |
| ('senat', 'kirsten', 'gillibrand') | -0.100772 | -0.012500 |
| ('nation', 'secur', 'advisor') | -0.095314 | -0.037500 |

Results showing in increasing order of Net CTF

**Negative means the word has Fake Characteristics**

6.3. Using Training Data Set to characterize every word

| Word | Net Cumulative Term Frequency | Net Cumulative Document Frequency * |
|---|---|---|
| ('realiti', 'show', 'star') | -0.534854 | -0.175000 |
| ('former', 'realiti', 'show') | -0.534854 | -0.175000 |
| ('nation', 'secur', 'advis') | -0.324399 | -0.100000 |
| ('chip', 'somodevilla/getti', 'imag') | -0.349954 | -0.100000 |
| ('alex', 'wong/getti', 'imag') | -0.272204 | -0.075000 |
| ('former', 'nation', 'secur') | -0.233880 | -0.062500 |
| ('alabama', 'suprem', 'court') | -0.196169 | -0.062500 |
| ('drew', 'angerer/getti', 'imag') | -0.208826 | -0.062500 |
| ('alabama', 'senat', 'candid') | -0.219937 | -0.062500 |
| ('advis', 'michael', 'flynn') | -0.164578 | -0.050000 |
| ('secur', 'advis', 'michael') | -0.164578 | -0.050000 |
| ('andrew', 'burton/getti', 'imag') | -0.186638 | -0.050000 |
| ('accus', 'child', 'molest') | -0.172695 | -0.050000 |
| ('unit', 'state', 'senat') | -0.043391 | -0.037500 |
| ('gibson', 'pool/getti', 'imag') | -0.123232 | -0.037500 |
| ('zach', 'gibson', 'pool/getti') | -0.123232 | -0.037500 |
| ('mark', 'wilson/getti', 'imag') | -0.126489 | -0.037500 |
| ('trump', 'transit', 'team') | -0.092760 | -0.037500 |
| ('saturday', 'night', 'live') | -0.138346 | -0.037500 |
| ('sarah', 'huckabe', 'sander') | -0.141313 | -0.037500 |
| ('hous', 'press', 'secretari') | -0.105896 | -0.037500 |
| ('nation', 'secur', 'advisor') | -0.095314 | -0.037500 |
| ('least', 'popular', 'presid') | -0.143306 | -0.037500 |
| ('hous', 'intellig', 'committe') | -0.059649 | -0.025000 |
| ('nomine', 'hillari', 'clinton') | -0.067816 | -0.025000 |
| ('donald', 'trump', 'spend') | -0.077778 | -0.025000 |
| ('know', 'donald', 'trump') | -0.071429 | -0.025000 |
| ('ongo', 'crimin', 'investig') | -0.076149 | -0.025000 |
| ('scott', 'olson/getti', 'imag') | -0.069048 | -0.025000 |
| ('attorney', 'general', 'eric') | -0.086081 | -0.025000 |
| ('general', 'eric', 'holder') | -0.086081 | -0.025000 |
| ('like', 'donald', 'trump') | -0.092593 | -0.025000 |
| ('hold', 'public', 'offic') | -0.073925 | -0.025000 |
| ('suprem', 'court', 'twice') | -0.065591 | -0.025000 |
| ('lightweight', 'senat', 'kirsten') | -0.114286 | -0.025000 |

Results showing in increasing order of Net DDF

**Negative means the word has Fake Characteristics**

**“Any word which is too positive will mean it is of True nature and any word which is too negative means it is of Fake nature”**

Thus, we have been able to characterize every unigram, bigram and trigram as True or Fake and give it a Net CTF and Net CDF value to it.

# 7. Predict a Document as True or Fake:

Once we have characterized every word (or bigram or trigram resp.) now, the next step is to read a Document and predict it as True or Fake.

Now we will calculate **CTF Score** and **CDF Score** for every Document in the testing Data Set Testing.csv.

The CTF Score for a document is calculated as follows:

$$\sum_{i=1}^{N} (\text{Frequency of word w(i) in Document} * \text{Net CTFfor word w(i) already calculated})$$

The CDF Score for a document is calculated as follows:

$$\sum_{i=1}^{N} (\text{Frequency of word w(i) in Document} * \text{Net CDFfor word w(i) already calculated})$$

Here N = Total no of Words in the Document to be predicted excluding stop words.

Note that $N \leq 300$ as we have summarized our documents up to 300 words only.

For a word which does not have a Net CTF and Net CDF saved (the word which was never encountered previously) the Net CTF and Net CDF is taken as 0.

- The following algorithm was used:

Read a Document D $\epsilon$ Testing.csv corpus/ Data Set which needs to be predicted

Remove stop words from the Document D and Lemmatize and Stem the remaining words

For every remaining word ‘w’ $\epsilon$ D

If (‘w’ has a Net CTF)

CTF Score = CTF Score + (Frequency of word ‘w’ in Document D * **Net CTF** for word ‘w’)

If (w does NOT have Net CTF score)

CTF Score = CTF Score + Frequency of word ‘w’ in Document D * **0**;

Similar, algorithm holds for CDF Score as well.

**Thus, for every Document that needs to be predicted a CTF Score and CDF Score was calculated.**

| CTF Score | CDF Score | Conclusion |
|---|---|---|
| +VE | +VE | Document is TRUE |
| -VE | -VE | Document is FAKE |
| +VE | -VE | Not so Sure |
| -Ve | +Ve | Not so Sure |

If in the Document the CTF and CDF scores are positive it means the document is True.

If in the Document the CTF and CDF scores are negative it means the document is Fake.

We ran this test for Documents that need to be predicted; the CTF and CDF scores were obtained as shown in the three upcoming subsections.

# 7.1. Prediction of CTF and CDF Score based on Unigram Frequency Model

| News Article | CTF Score | CDF Score | Actual Label | Prediction on the basis of CTF Score | Prediction on the basis of CDF Score |
|---|---|---|---|---|---|
| 1 | 199.303514 | 140.387500 | TRUE | TRUE | TRUE |
| 2 | 65.573225 | 57.225000 | TRUE | TRUE | TRUE |
| 3 | 11.383227 | 9.212500 | TRUE | TRUE | TRUE |
| 4 | 78.749073 | 67.312500 | TRUE | TRUE | TRUE |
| 5 | 24.842032 | 11.825000 | TRUE | TRUE | TRUE |
| 6 | 14.873779 | 2.512500 | TRUE | TRUE | TRUE |
| 7 | 10.120016 | 10.037500 | TRUE | TRUE | TRUE |
| 8 | 55.549704 | 10.100000 | TRUE | TRUE | TRUE |
| 9 | 20.338593 | 26.887500 | TRUE | TRUE | TRUE |
| 10 | 18.938933 | 11.812500 | TRUE | TRUE | TRUE |
| 11 | 6.377204 | 7.150000 | TRUE | TRUE | TRUE |
| 12 | 9.485125 | 10.612500 | TRUE | TRUE | TRUE |
| 13 | 213.433047 | 166.275000 | TRUE | TRUE | TRUE |
| 14 | 58.202984 | 53.462500 | TRUE | TRUE | TRUE |
| 15 | 23.708377 | 26.037500 | TRUE | TRUE | TRUE |
| 16 | 8.150218 | 9.337500 | TRUE | TRUE | TRUE |
| 17 | 19.057763 | 15.425000 | TRUE | TRUE | TRUE |
| 18 | 1.678262 | 3.562500 | TRUE | TRUE | TRUE |
| 19 | 51.986472 | 36.637500 | TRUE | TRUE | TRUE |
| 20 | 102.175886 | 74.900000 | TRUE | TRUE | TRUE |
| 21 | 26.841727 | 21.512500 | FAKE | TRUE | TRUE |
| 22 | -0.633522 | -1.700000 | FAKE | FAKE | FAKE |
| 23 | -58.368335 | -10.962500 | FAKE | FAKE | FAKE |
| 24 | -83.227455 | -40.450000 | FAKE | FAKE | FAKE |
| 25 | -63.503925 | -5.450000 | FAKE | FAKE | FAKE |
| 26 | -3.326056 | -9.687500 | FAKE | FAKE | FAKE |
| 27 | -41.671907 | -32.987500 | FAKE | FAKE | FAKE |
| 28 | -282.591542 | -144.837500 | FAKE | FAKE | FAKE |
| 29 | -62.080426 | -2.175000 | FAKE | FAKE | FAKE |
| 30 | -39.220027 | -27.925000 | FAKE | FAKE | FAKE |
| 31 | 8.266384 | 0.162500 | FAKE | TRUE | TRUE |
| 32 | -153.942991 | -56.987500 | FAKE | FAKE | FAKE |
| 33 | -6.152073 | -7.662500 | FAKE | FAKE | FAKE |
| 34 | 0.801466 | -1.912500 | FAKE | TRUE | FAKE |
| 35 | -86.774826 | -15.550000 | FAKE | FAKE | FAKE |
| 36 | 16.033937 | -12.275000 | FAKE | TRUE | FAKE |
| 37 | -73.420163 | -14.625000 | FAKE | FAKE | FAKE |
| 38 | -134.283411 | -36.225000 | FAKE | FAKE | FAKE |
| 39 | 2.965987 | 6.412500 | FAKE | TRUE | TRUE |
| 40 | -111.118730 | -20.212500 | FAKE | FAKE | FAKE |

7.1. Prediction of CTF and CDF Score based on Unigram Frequency Model

Means News Article Number

Means Positive CTF/CDF Score

Means Negative CTF/CDF Score

Means The Prediction was inaccurate

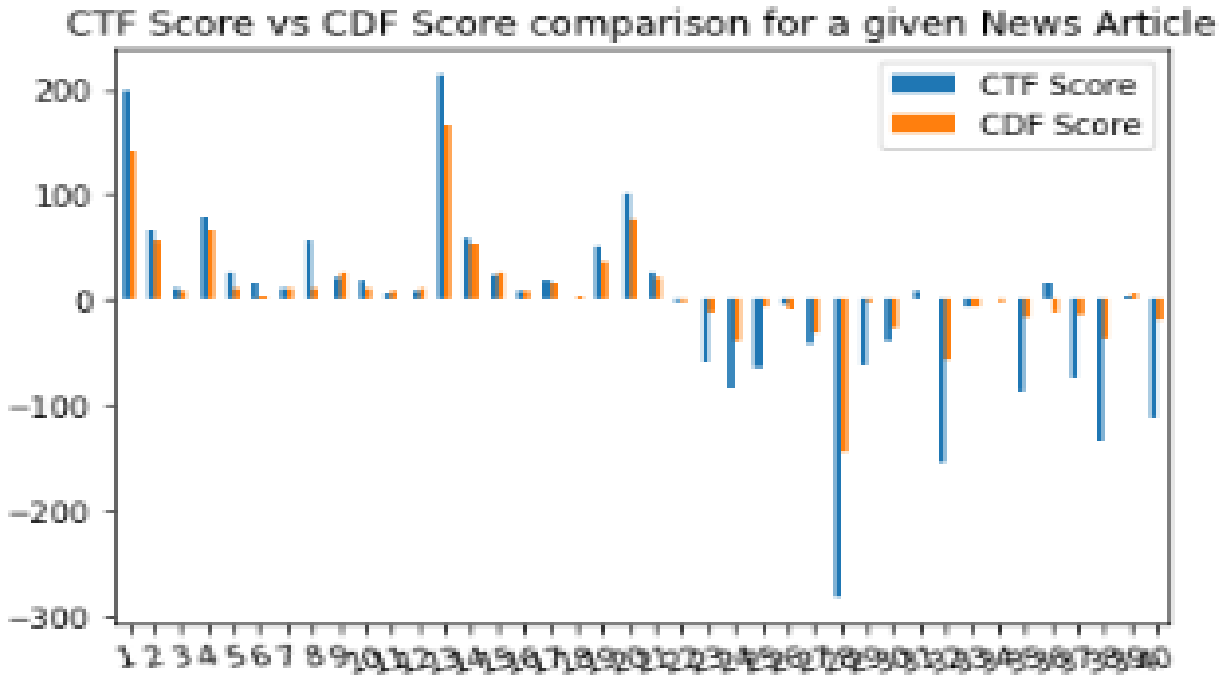
Accuracy on the Basis of CTF Score = Documents correctly predicted / Total no. of Documents

= 35/40 = 0.875

Accuracy on the Basis of CDF Score = Documents correctly predicted / Total no. of Documents

= 37/40 = 0.925

The following figure is a graphical representation of CTF and CDF score for documents that were predicted




Here X - axis represent the Document no. and Y - axis represent the CTF and CDF scores obtained for every Document. Blue is for CTF and Red for CDF.

# 7.2. Prediction of CTF and CDF Score based on Bigram Frequency Model

| News Article | CTF Score | CDF Score | Actual Label | Prediction on the basis of CTF Score | Prediction on the basis of CDF Score |
|---|---|---|---|---|---|
| 1 | 3.892925 | 2.725000 | TRUE | TRUE | TRUE |
| 2 | 3.326417 | 2.550000 | TRUE | TRUE | TRUE |
| 3 | 0.131603 | 0.112500 | TRUE | TRUE | TRUE |
| 4 | 2.135497 | 1.412500 | TRUE | TRUE | TRUE |
| 5 | 0.331090 | 0.225000 | TRUE | TRUE | TRUE |
| 6 | 1.802075 | 1.612500 | TRUE | TRUE | TRUE |
| 7 | 0.166843 | 0.112500 | TRUE | TRUE | TRUE |
| 8 | 0.440108 | 0.475000 | TRUE | TRUE | TRUE |
| 9 | 0.317104 | 0.525000 | TRUE | TRUE | TRUE |
| 10 | 0.932248 | 0.700000 | TRUE | TRUE | TRUE |
| 11 | -0.012985 | 0.000000 | TRUE | FAKE | TRUE |
| 12 | 1.447789 | 1.075000 | TRUE | TRUE | TRUE |
| 13 | 2.574572 | 2.462500 | TRUE | TRUE | TRUE |
| 14 | 1.628588 | 1.437500 | TRUE | TRUE | TRUE |
| 15 | 2.236323 | 1.450000 | TRUE | TRUE | TRUE |
| 16 | 1.424259 | 0.950000 | TRUE | TRUE | TRUE |
| 17 | 1.895513 | 1.150000 | TRUE | TRUE | TRUE |
| 18 | 3.843058 | 1.550000 | TRUE | TRUE | TRUE |
| 19 | 1.938936 | 1.350000 | TRUE | TRUE | TRUE |
| 20 | 2.048759 | 2.100000 | TRUE | TRUE | TRUE |
| 21 | 0.168082 | 0.037500 | FAKE | TRUE | TRUE |
| 22 | -0.164320 | -0.112500 | FAKE | FAKE | FAKE |
| 23 | 0.541132 | 0.437500 | FAKE | TRUE | TRUE |
| 24 | 0.487742 | 0.262500 | FAKE | TRUE | TRUE |
| 25 | 0.183276 | 0.137500 | FAKE | TRUE | TRUE |
| 26 | -0.088802 | -0.062500 | FAKE | FAKE | FAKE |
| 27 | -1.252219 | -0.875000 | FAKE | FAKE | FAKE |
| 28 | -0.268725 | -0.262500 | FAKE | FAKE | FAKE |
| 29 | 1.114113 | 0.787500 | FAKE | TRUE | TRUE |
| 30 | -0.013282 | -0.250000 | FAKE | FAKE | FAKE |
| 31 | 0.025836 | 0.050000 | FAKE | TRUE | TRUE |
| 32 | -0.809375 | -0.750000 | FAKE | FAKE | FAKE |
| 33 | 0.075682 | 0.087500 | FAKE | TRUE | TRUE |
| 34 | -0.094494 | -0.000000 | FAKE | FAKE | FAKE |
| 35 | 0.054550 | 0.187500 | FAKE | TRUE | TRUE |
| 36 | -0.294965 | -0.200000 | FAKE | FAKE | FAKE |
| 37 | -0.030377 | 0.125000 | FAKE | FAKE | TRUE |
| 38 | -0.109262 | -0.150000 | FAKE | FAKE | FAKE |
| 39 | 0.288940 | 0.287500 | FAKE | TRUE | TRUE |
| 40 | 0.376128 | 0.387500 | FAKE | TRUE | TRUE |

7.2. Prediction of CTF and CDF Score based on Bigram Frequency Model

Means News Article Number

Means Positive CTF/CDF Score

Means Negative CTF/CDF Score

Means The Prediction was inaccurate

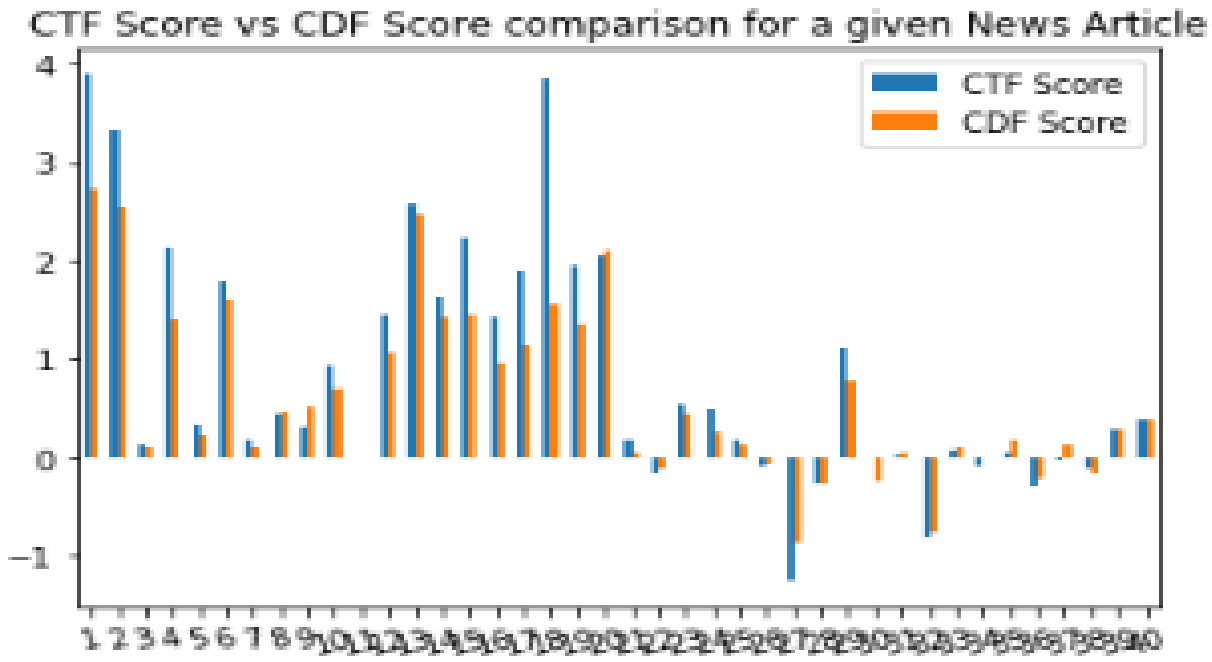
Accuracy on the Basis of CTF Score = Documents correctly predicted / Total no. of Documents

= 29/40 = 0.725

Accuracy on the Basis of CDF Score = Documents correctly predicted / Total no. of Documents

= 29/40 = 0.725

The following figure is a graphical representation of CTF and CDF score for documents that were predicted




Here X - axis represent the Document no. and Y - axis represent the CTF and CDF scores obtained for every Document. Blue is for CTF and Red for CDF.

# 7.3. Prediction of CTF and CDF Score based on Trigram Frequency Model

| News Article | CTF Score | CDF Score | Actual Label | Prediction on the basis of CTF Score | Prediction on the basis of CDF Score |
|---|---|---|---|---|---|
| 1 | 4.896418 | 1.212500 | TRUE | TRUE | TRUE |
| 2 | 4.394253 | 1.150000 | TRUE | TRUE | TRUE |
| 3 | 0.000000 | 0.000000 | TRUE | TRUE | TRUE |
| 4 | 4.524359 | 0.912500 | TRUE | TRUE | TRUE |
| 5 | -0.078947 | -0.037500 | TRUE | FAKE | FAKE |
| 6 | 2.777273 | 0.712500 | TRUE | TRUE | TRUE |
| 7 | 0.000000 | 0.000000 | TRUE | TRUE | TRUE |
| 8 | 0.030303 | 0.012500 | TRUE | TRUE | TRUE |
| 9 | 0.066667 | 0.025000 | TRUE | TRUE | TRUE |
| 10 | 0.251455 | 0.087500 | TRUE | TRUE | TRUE |
| 11 | 0.042553 | 0.025000 | TRUE | TRUE | TRUE |
| 12 | 2.994325 | 0.800000 | TRUE | TRUE | TRUE |
| 13 | 3.873064 | 1.125000 | TRUE | TRUE | TRUE |
| 14 | 2.873977 | 0.750000 | TRUE | TRUE | TRUE |
| 15 | 3.546697 | 0.812500 | TRUE | TRUE | TRUE |
| 16 | 2.748979 | 0.687500 | TRUE | TRUE | TRUE |
| 17 | 4.502137 | 0.900000 | TRUE | TRUE | TRUE |
| 18 | 6.288625 | 1.062500 | TRUE | TRUE | TRUE |
| 19 | 2.827926 | 0.725000 | TRUE | TRUE | TRUE |
| 20 | 2.898952 | 0.837500 | TRUE | TRUE | TRUE |
| 21 | -0.238353 | -0.050000 | FAKE | FAKE | FAKE |
| 22 | 0.000000 | 0.000000 | FAKE | TRUE | TRUE |
| 23 | 2.355049 | 0.562500 | FAKE | TRUE | TRUE |
| 24 | 0.000000 | 0.000000 | FAKE | TRUE | TRUE |
| 25 | 0.000000 | 0.000000 | FAKE | TRUE | TRUE |
| 26 | -0.008964 | 0.000000 | FAKE | FAKE | TRUE |
| 27 | -0.553107 | -0.187500 | FAKE | FAKE | FAKE |
| 28 | -0.208826 | -0.062500 | FAKE | FAKE | FAKE |
| 29 | 2.748979 | 0.687500 | FAKE | TRUE | TRUE |
| 30 | -0.291457 | -0.087500 | FAKE | FAKE | FAKE |
| 31 | 0.000000 | 0.000000 | FAKE | TRUE | TRUE |
| 32 | -1.127218 | -0.362500 | FAKE | FAKE | FAKE |
| 33 | -0.195313 | -0.050000 | FAKE | FAKE | FAKE |
| 34 | 0.000000 | 0.000000 | FAKE | TRUE | TRUE |
| 35 | 0.000000 | 0.000000 | FAKE | TRUE | TRUE |
| 36 | -0.219937 | -0.062500 | FAKE | FAKE | FAKE |
| 37 | -0.398826 | -0.037500 | FAKE | FAKE | FAKE |
| 38 | -0.237398 | -0.075000 | FAKE | FAKE | FAKE |
| 39 | -0.044270 | -0.037500 | FAKE | FAKE | FAKE |
| 40 | 0.098382 | 0.062500 | FAKE | TRUE | TRUE |

7.3. Prediction of CTF and CDF Score based on Trigram Frequency Model

Means News Article Number

Means Positive CTF/CDF Score

Means Negative CTF/CDF Score

Means The Prediction was inaccurate

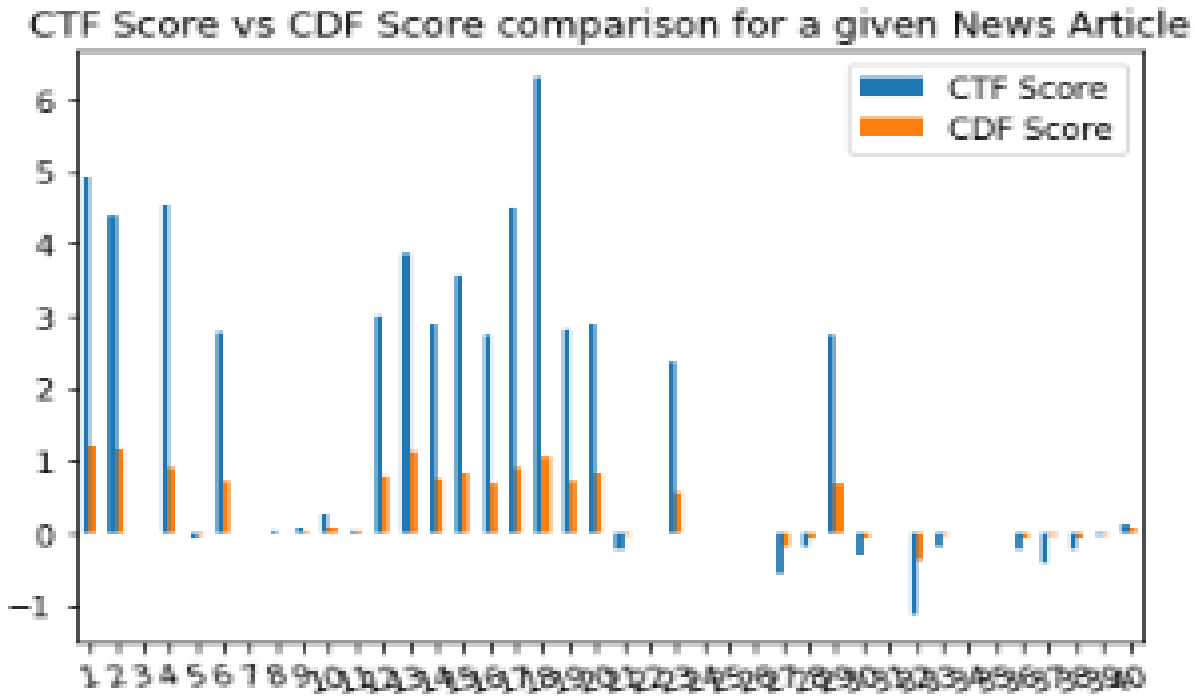
Accuracy on the Basis of CTF Score = Documents correctly predicted / Total no. of Documents

= 30/40 = 0.75

Accuracy on the Basis of CDF Score = Documents correctly predicted / Total no. of Documents

= 29/40 = 0.725

The following figure is a graphical representation of CTF and CDF score for documents that were predicted




Here X - axis represent the Document no. and Y - axis represent the CTF and CDF scores obtained for every Document. Blue is for CTF and Red for CDF

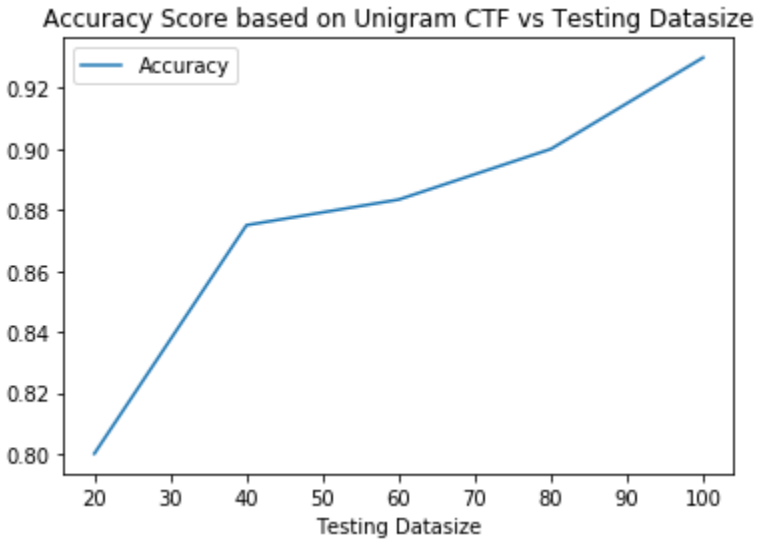
## 7.4. Varying Testing Datasize

Finally, we have varied the size of the data used for testing and got the following observations:

1. Unigram CTF Model

| No. of documents used for Testing purpose | Accuracy Score |
|---|---|
| 20 | 0.80 |
| 40 | 0.875 |
| 60 | 0.8834 |
| 80 | 0.90 |
| 100 | 0.93 |

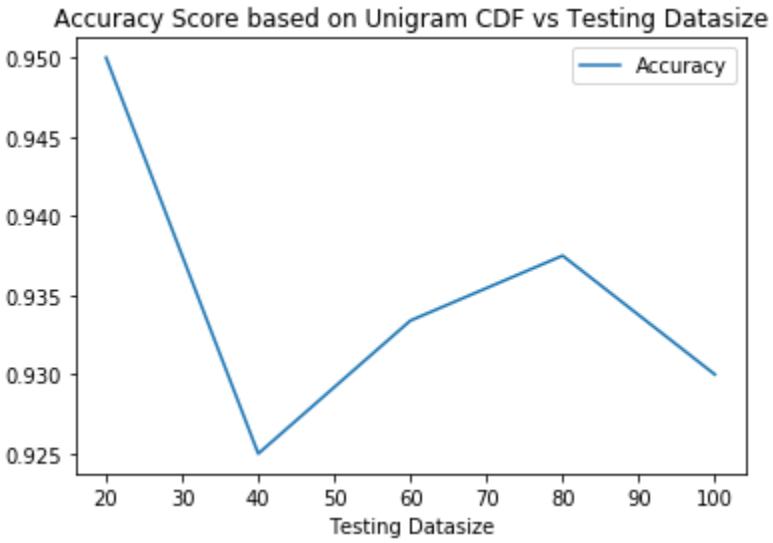
The Plot obtained is as follows:




2. Unigram CDF Model

| No. of documents used for Testing purpose | Accuracy Score |
|---|---|
| 20 | 0.95 |
| 40 | 0.925 |
| 60 | 0.9334 |
| 80 | 0.9375 |
| 100 | 0.93 |

The Plot obtained is as follows:

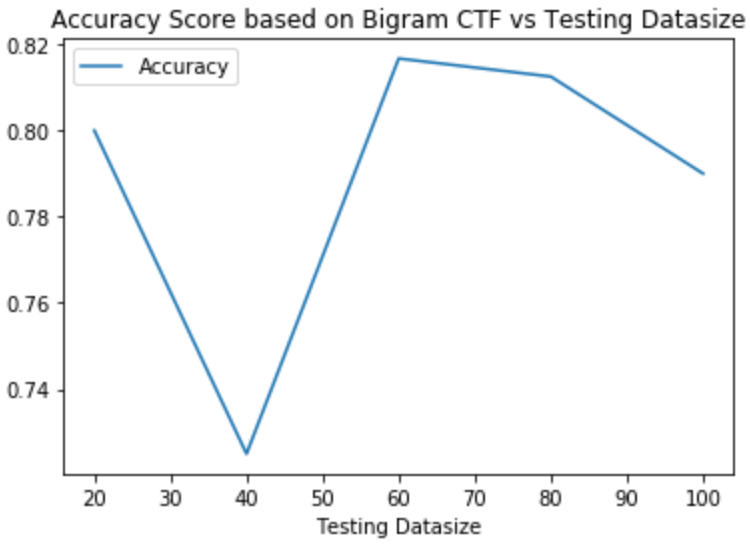



3. Bigram CTF Model

| No. of documents used for Testing purpose | Accuracy Score |
|---|---|
| 20 | 0.80 |
| 40 | 0.725 |
| 60 | 0.8167 |
| 80 | 0.8125 |
| 100 | 0.79 |

The Plot obtained is as follows:

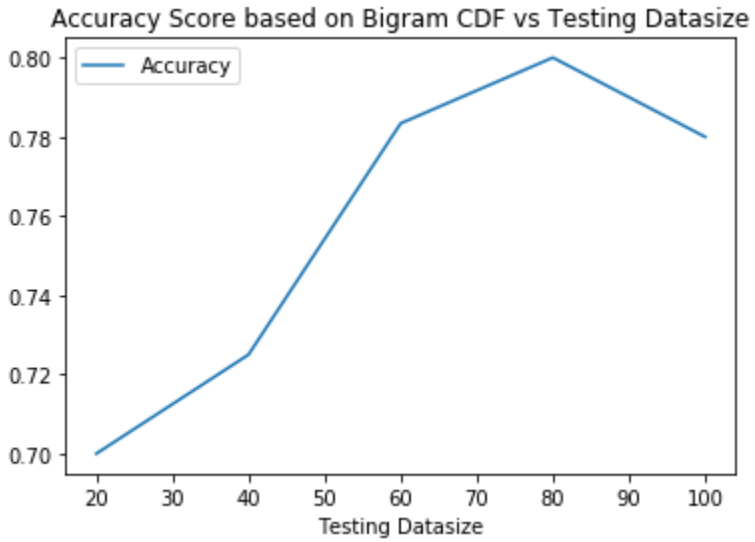
7.4. Varying Testing Datasize




4. Bigram CDF Model

| No. of documents used for Testing purpose | Accuracy Score |
|---|---|
| 20 | 0.70 |
| 40 | 0.725 |
| 60 | 0.7834 |
| 80 | 0.80 |
| 100 | 0.78 |

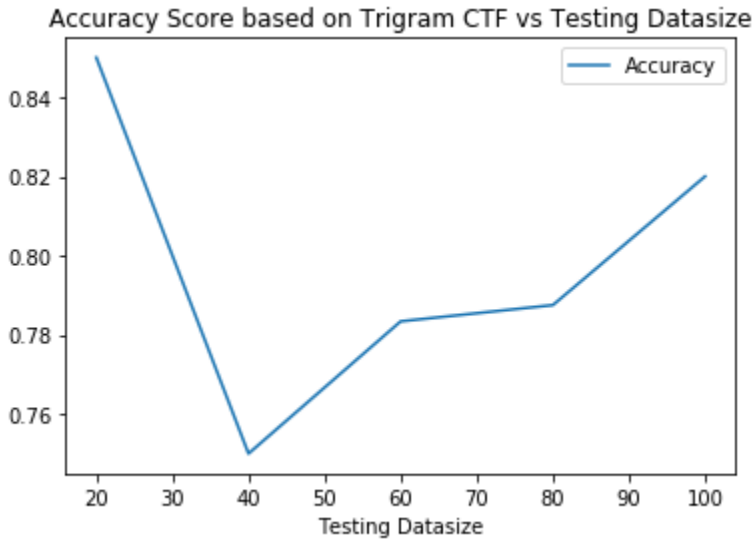
The Plot obtained is as follows:



5. Trigram CTF Model

| No. of documents used for Testing purpose | Accuracy Score |
|---|---|
| 20 | 0.85 |
| 40 | 0.75 |
| 60 | 0.7834 |
| 80 | 0.7875 |
| 100 | 0.82 |

The Plot obtained is as follows:

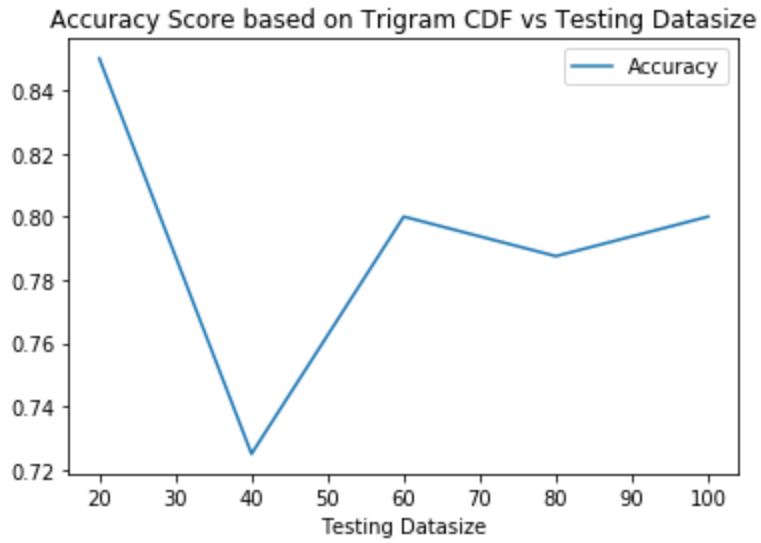



6. Trigram CDF Model

| No. of documents used for Testing purpose | Accuracy Score |
|---|---|
| 20 | 0.85 |
| 40 | 0.725 |
| 60 | 0.80 |
| 80 | 0.7875 |
| 100 | 0.80 |

The Plot obtained is as follows:

7.4. Varying Testing Datasize



# 8. Conclusions:

The following conclusions can be made:

1) Prediction on the basis of Cumulative Document Frequency (0.925) had a high accuracy as compared to Cumulative Term Frequency (0.875) in case of Unigram Model while prediction on the basis of Cumulative Term Frequency (0.75) had a high accuracy as compared to Cumulative Document Frequency (0.725) in case of Trigram Model when our testing data size was 40.
2) In subsections 7.2 (Bigram), 7.3(Trigram), there are more positive peaks than negative peaks which mean we were more confident in predicting True Documents (as they followed a regular pattern).
3) Unigram CTF Model shows very good trend. The Accuracy Score increases as the size of the Testing data increases.
4) The highest possible Accuracy Score is achieved using Unigram CDF Model with a score of 0.95 in the case when the testing data size was 20. This happens because the number of documents to be used for training the model were 180 (90 True + 90 Fake) which is high. Also, we can see the probability of finding a particular word in an n-gram from a document decreases as n increases and so probability of occurrence of word independently in a document decreases and so accuracy sore decreases. Also, Accuracy Score (CDF Score) varies inversely to the number of testing documents which are lowest (20) in this case.
5) Currently the input set has around 80 True and 80 Fake news articles that were given to train the prediction algorithm. The no, of documents that were predicted were around 40. If the training data set increases, we will give much better results.

# 9. Future Scope:

- We can make a decision on CTF and CDF Scores obtained. We could use other attributes as well such as intent of the news article, source from which the news articles are coming and many other things.

- For other attributes we mentioned we could make a decision on all those attributes for which we can use a Decision Tree, where CDF and CTF scores will be used as attributes. We can use other algorithms like Logistic Regression, Gradient Boosting Classifier, Random Forest Classifier for better judgement of authenticity as well as intent of news articles.

# References:

[**1**] Allcott, Hunt, and Matthew Gentzkow. 2017. "Social Media and Fake News in the 2016 Election." Journal of Economic Perspectives, 31 (2): 211-36.

[**2**] Shu, K., Sliva, A., Wang, S., Tang, J., & Liu, H. (2017). Fake news detection on social media: A data mining perspective. ACM SIGKDD explorations newsletter, 19(1), 22-36.

[**3**] Tandoc Jr, Edson C., Zheng Wei Lim, and Richard Ling. "Defining “fake news” A typology of scholarly definitions." Digital journalism 6.2 (2018): 137-153.

[**4**] Saif M Mohammad, Parinaz Sobhani, and Svetlana Kiritchenko. Stance and sentiment in tweets. ACM Transactions on Internet Technology (TOIT), 17(3):26, 2017.

[**5**] Zhiwei Jin, Juan Cao, Yongdong Zhang, and Jiebo Luo. News verification by exploiting conflicting social viewpoints in microblogs. In AAAI'16.

[**6**] Zhiwei Jin, Juan Cao, Yu-Gang Jiang, and Yongdong Zhang. News credibility evaluation on microblog with a hierarchical propagation model. In ICDM'14.

[**7**] Manish Gupta, Peixiang Zhao, and Jiawei Han. Evaluating event credibility on twitter. In PSDM'12.