FAKE NEWS DETECTION

AAKRITI UPADHYAY
(001309523)
aupadhyay@albany.edu
Department of Computer Science

***Abstract*— The project explores the application of data mining techniques to the detection of fake news. The data was pre-processed for the crawled tweets from Twitter dataset based on the query and geo-location using Stream API and Rest API. The classification of fake news in the tweets has been achieved using ten-folder cross-validation method in linear support vector machine. The use of linear SVM has prevented over fitting of data and has shown quite acceptable results for the classified fake news in the new tweets.**

Keywords—Support Vector Machine; Fake News; sentiment analysis; Tweets; news verification; fraud

I. INTRODUCTION

The term *fake news* is a fabricated news with an intent to manipulate someone or something by giving a misleading or deceive information to gain attention or damage reputation meeting financial or political benefits. This disinformation is created with the objective to align with the audience's point of view because such content is not likely to be questioned or discounted.

Deceptive news, such as fake news, phony press releases and hoaxes, may be misleading or even harmful, especially when they are disconnected from their original sources and contexts. "A 2012 report of Pew Internet Research on the future of big data argues that even though by 2020 big data is likely to have a transformational effect on our knowledge and understanding of the world, there is also danger from inaccurate or false information (called "distribution of harms"). Occasionally reports of non-existent, surreal, alarming events have been taken seriously. For instance, "Jack Warner, the former FIFA vice president, has apparently been taken in by a spoof article from the satirical website The Onion" after The Onion had suggested that the FIFA corruption scandal would result in a 2015 Summer Cup in the US [1].

Facebook has been the epicenter of much critique following media attention [3]. They have already implemented a feature for users to flag fake news on the site. However, it is clear from their public announcements that they are actively researching their ability to distinguish these articles in an automated way. Indeed, it is not an easy task. A given algorithm must be politically unbiased, since fake news exists on both ends of the spectrum and also give equal balance to legitimate news sources on either ends of the spectrum. In addition, the question of legitimacy is a difficult one: what makes a news article 'legitimate'? Can this be determined in an objective way?

In 2016, the prominence of disinformation within American political discourse was the subject of substantial attention, particularly following the surprise election of President Trump. The term 'fake news' became common parlance for the issue, particularly to describe factually incorrect and misleading articles published mostly for the purpose of making money through page views [3].

The proliferation of user-generated content, and Computer Mediated Communication (CMC) technologies such as blogs, Twitter, and other social media have the capacity of news delivery mechanisms on a mass scale— yet much of the information is of questionable veracity. Establishing the reliability of information online is a daunting but critical challenge. Four decades of deception detection research has helped us learn about how well humans are able detect lies in text. The findings show we are not so good at it. In fact, just 4% better than chance, based on a meta-analysis of more than 200 experiments. This problem has led researchers and technical developers to look at several automated ways of assessing the truth value of potentially deceptive text based on the properties of the content and the patterns of computer-mediated communication [11].

It is impossible to be absolutely certain about the legitimacy of and intention behind an application or transaction. Given the reality, the best cost effective option is to tease out possible evidences of fraud from the available data using mathematical algorithms [2].

In a research poll, it is indicated that 64% of US adults said that "made-up news" has caused a "great deal of confusion" about the facts of current events. It caused doubts in the minds of US citizens and it made difficult to distinguish between fake news and legitimate news. In this project, I have used data mining approach to analysis the data quality of collected tweets from twitter dataset and visualized it to understand the trend of word frequency in the tweets. Linear SVM model has been used to classify tweets that are fake news from legitimate news using 10-folder cross-validation method.

II. RELATED WORK

In paper [2], Phua et.al., defines the professional fraudster, formalises the main types and subtypes of known fraud, and presents the nature of data evidence collected within affected industries. They categorised, compared and summarised relevant data mining-based fraud detection methods and techniques in published academic and industrial research. The paper discusses about techniques such as neural networks, Bayesian networks, support vector machine (SVM) and

decision trees and its application on supervised or unsupervised training dataset for fraud detection within business context. It also highlights new directions from related adversarial data mining fields/applications such as epidemic/outbreak detection, insider trading, intrusion detection, money laundering, spam detection, and terrorist detection.

In paper [1], Rubin et.al., discusses three types of fake news, each in contrast to genuine serious reporting, suggesting that there are at least three distinct sub-tasks in fake news detection: a) fabrication, b) hoaxing and c) satire detection. The paper separates the task of fake news detection into three, by type of fake: a) serious fabrications (uncovered in mainstream or participant media, yellow press or tabloids); b) large-scale hoaxes; c) humorous fakes (news satire, parody, game shows). These tasks holds an important development at an intersection of LIS, NLP, big data and journalism for fake news detection (of the three identified types).

In paper [11], Conroy et.al, provides a typology of several varieties of veracity assessment methods emerging from two major categories – linguistic cue approaches (with machine learning), and network analysis approaches. Authors proposes a hybrid approach that combines linguistic cue and machine learning, with network-based behavioural data for the implementation of a fake news detection tool. In Linguistic approach, the content of deceptive messages is extracted and analyzed to associate language patterns with deception whereas in Network approach, the network information such as message metadata or structured knowledge network queries has been harnessed to provide aggregate deception measures. Both forms typically incorporate machine learning techniques for training classifiers to suit the analysis. Support Vector Machines (SVM) and Naïve Bayesian models has been used to classify fake news based on the given sentiment scores and sets of word and category frequencies in the training data.

In the research work [3], Dyson and Golab, has applied NLP technique to the detection of "fake news". Authors uses dataset from Signal Media and a list of sources from OpenSources.co, they apply term frequency-inverse document frequency (TF-IDF) of bi-grams and probabilistic context free grammar (PCFG) detection to a corpus of about 11,000 articles. Authors implemented various classification models such as Support Vector Machines (SVM), Stochastic Gradient Descent (SGD), Gradient Boosting (GB), Bounded Decision Trees (DT) and Random Forests (RF). The performance of these models is evaluated by training on three feature sets: a) Bigram Term Frequency-Inverse Document Frequency; b) Normalized frequency of parsed syntactical dependencies; and c) union of above both sets.

III. Proposed Approach

The processed twitter data was extracted for the feature words and frequency of each feature was computed against each tweet. The results were formatted into DataFrame table and stored in csv (comma separated values) file format. The table consists of "reference websites" in the tweets as the objects and query words as the feature words.

The data was visualized using generated table for the word frequency in each tweet. The visualization plots generated are discussed below.

A. Bar plot

Figure 1 shows the frequency of each feature words: Clinton, Donald, Fake, Hillary, Obama, Russian, Trump, hillary and news for the processed tweets. It can be inferred from the graph that the top three most occurring words in the tweets are Trump, news and Fake which indicate that most of the tweets are related to the query.

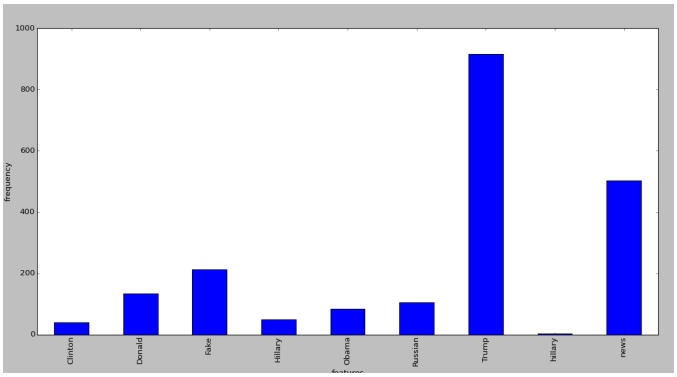

Fig. 1. Frequency of each feature word

B. Histogram

Figure 2 shows five bins having the number of feature words lying within the range of each bin. It can be inferred from the graph that there are four feature words that lie in the range [0-100), two feature words in the range [100-200), one feature in the range [200-300), one feature word in [500-600) and one feature word in [900-1000).

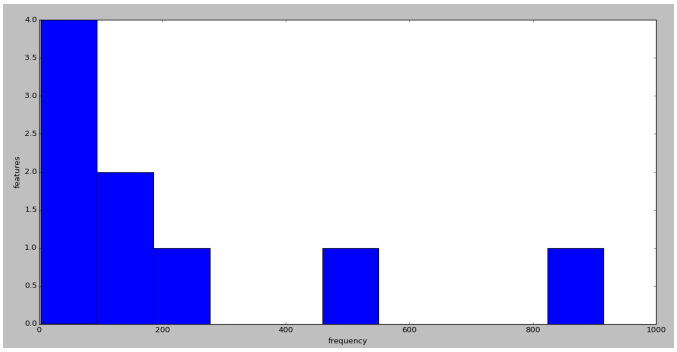

Fig. 2. Frequency range for feature words

C. Box plot

Table I shows the percentile values for all 9 feature words. It can be inferred from the table that most of the features have 10, 25, 50 and 75 percentiles as value '0'. Hence, this results in merging the box for each percentile to 0. In the case of feature words "Trump" and "news", they have 75 percentile values as 1 due to which only 10, 25 and 50 percentile box merges to 0. for feature word "Russian", the 90 percentile value reaches the highest value in y-axis i.e. 3, as shown in the Figure 3.

TABLE I. PERCENTILE VALUES FOR EACH FEATURE WORDS

Percentile	Clinton	Donald	Fake	Hillary	hillary	Trump	Obama	Russian	news
10 percentile	0	0	0	0	0	0	0	0	0
25 percentile	0	0	0	0	0	0	0	0	0
50 percentile	0	0	0	0	0	0	0	0	0
75 percentile	0	0	0	0	0	1	0	0	1
90 percentile	1	2	2	1	1	2	2	3	2

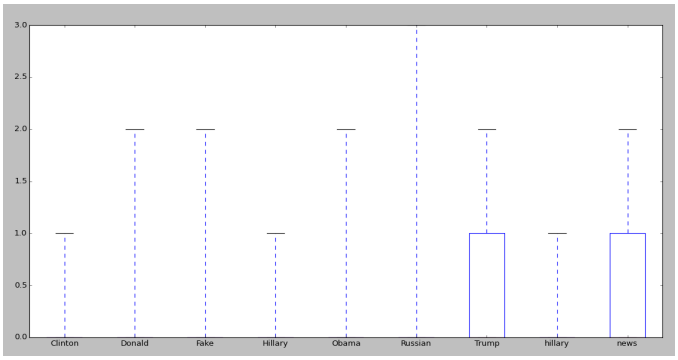

Fig. 3. Percentile range for each feature word

D. *Scatter plot*

Figure 4 shows the broader lines for the values 0 and 1 as most of the features have highest occurrence of these two values in the data table. Each star with different colour represents specific feature words. The occurrence of value 2 is shown by different colour stars for each feature. Only feature word "Russian" has the occurrence of value 3 which is shown in the higher limit of the y-axis as two small stars.

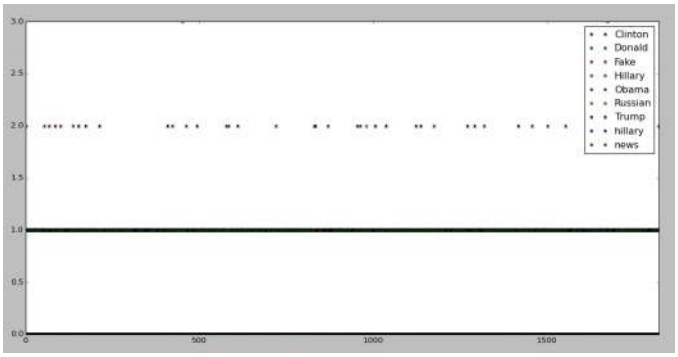

Fig. 4. Occurence values for each Feature word

E. *Density map*

Figure 5, each colored line refer to specific feature word. The figure shows the highest density occurrence of value 0 for all the feature words among 1823 tweets. The density of value 1 is quite lower than the density value of 0 as it can be inferred in the figure.

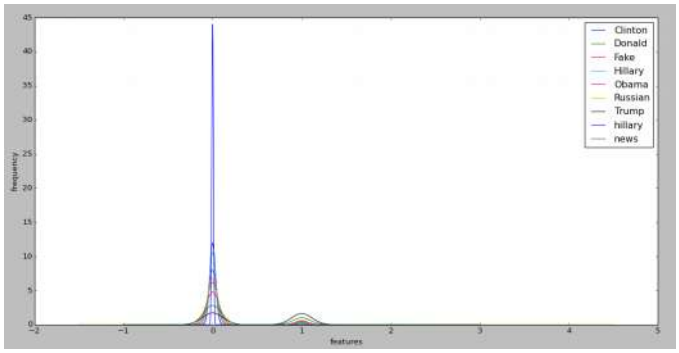

Fig. 5. Highest density occurence values for feature words

F. *Correlation matrix plot*

Figure 6 shows a correlation matrix to giving similarities between each feature words. It can be inferred from the figure that the words do not show similarity with other rather than themselves except for the group of words "Hillary" and "Clinton" that have similar occurrence in most cases in the tweets.

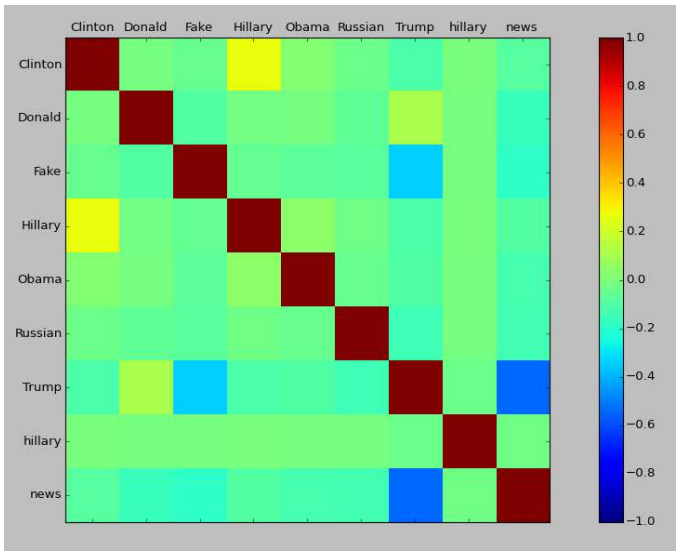

Fig. 6. Similarities between feature words

G. *Word cloud*

Figure 7 shows the word cloud generated from [8]. It can be inferred from the figure that the most frequently appearing word is "Trump" and second frequently appearing word is "news".

Fig. 7. Feature word cloud

The statistical values such as mean, standard deviation, minimum and maximum were calculated for each feature words from the data table as shown in Table II.

TABLE II. STATISTICAL MEASURES FOR FEATURE WORDS

Words	Mean	Standard Deviation	Maximum	Minimum
Clinton	0.021942	0.146534	1	0
Donald	0.073505	0.267269	2	0
Fake	0.116840	0.329748	2	0
Hillary	0.027427	0.163370	1	0
Obama	0.046626	0.213482	2	0
Russian	0.058146	0.249958	3	0
Trump	0.501920	0.522671	2	0
hillary	0.001646	0.040544	1	0
news	0.275919	0.461595	2	0

These feature words were used to generate supervised training dataset by manually labeling the tweets as legitimate news and fake news. The supervised dataset is used to train the classification model in order to distinguish fake news tweets from all other available tweets. The advantage of the supervised training dataset is that since it has been generated by manually labeling the tweets, it is expected to have minimum or negligible human error whereas the weakest point of manually labeling the data is that it becomes a tedious job for 1000 or more tweets.

Linear support vector machine has been used as classification model to avoid over fitting of the data. Using 10-folder cross validation, top 10 features were extracted. The goal of cross-validation is to estimate the expected level of fit of a model to a data such that it is independent of the data that were used to train the model. It can be used to estimate any quantitative measure of fit that is appropriate for the data and model.

The extracted features from the training dataset are

Features = {"trump's": 0, 'good': 4, 'trump': 2, 'news:': 3, '#news': 1, 'comey': 5, 'fake': 6, 'new': 7, 'news':8, 'donald':9}

The training data consist of binary classification of legitimate news and fake news as 0 and 1 respectively. In this binary classification scenario, cross-validation predicts correctly or incorrectly for each tweet from the validation set (the top feature words). Randomly selected 100 tweets from the available collected tweets were taken as testing data to the classification model which separates fake news tweets from legitimate news tweets by an optimal separating hyper plane.

IV. SYSTEM DESIGN AND IMPLEMENTATION

This section takes a deeper coverage of the system design and implemented methods during the development of the project. Each subsection describes the tools, methodology and programming language used to achieve the required goal of the project.

A. Architecture

The hardware and software architecture of the system has been discussed below .

1) Hardware Specification:

a) Processor: Intel core i7 processor.

b) Operating System: Windows 10 (64 bit).

c) Random Access Memory: 8 GB RAM.

2) Software specification:

a) Platform: Python 2.7.

b) Tool: JetBrains PyCharm Community, Edition 2016.3.2 .

B. Dataset

Data was collected from social networking site Twitter, much like Facebook even it is an epicenter for spreading news among different human networks. Rest search API and Stream API has been used to collect data from Twitter. The streaming process gets the input tweets and performs any parsing, filtering, and/or aggregation needed before storing the result to a data store. Connecting to the streaming API requires keeping a persistent HTTP connection open. The streaming API give low latency access to twitter's global stream of tweet data than Rest API.

Twitter search package has been used to search using Rest API. The rest API connects to the Twitter for short interval of time (may be in seconds) and returns more precision tweet data based on the given query.

The tweets with any mention of the name of the source has been scrubbed. Because the reliable/unreliable classification is determined at the source level, this step is necessary to ensure the model doesn't just learn the mappings from known sources to labels. Twitter handles, hashtags and email addresses (which often show up in tweets) have been stripped off from the tweets during pre-processing process.

The quality of collected data was evaluated using three parameters :

a) *API recall* :
It is the measure of total number of retrieved tweets (M) to the total number of tweets in the Twitter space that match given query (N).

b) *Quality recall* :
It is the measure of total number of positive tweets (A) to the total number of retrieved tweets (M).

c) *Quality Precision* :
It is the measure of total number of positive tweets in M i.e. (A) to the total number of positive tweets in the whole Twitter space (A + B + C).

In this project, 20000 historical tweets have been collected for each user from stored user screen names list. The data collected was categorized as biased and unbiased data. Biased data are the set of tweets collected using query and geocode whereas unbiased data are the set of randomly collected data from Twitter. The two datasets were compared to calculate below parameters:

- Total tweets collected from Twitter, D = 20000
- Crawled Tweets from query, M = 698
- Total tweets that match query, N = 1823
- Positive tweets in M, A = 101
- Positive tweets in N not in M, B = 90
- Positive tweets in D not in M and N, C = 804

TABLE III. DATA QUALITY FOR COLLECTED TWEETS

Measures	Values
API Recall	0.382
Quality Recall	0.145
Quality Precision	0.102

C. Major Components

Four major data mining components used in this project are :

1) *Data Collection* :
Tweets from Twitter dataset had been collected using the below query and geo-location

query = "*Elections OR Donald OR Fake OR Hillary OR Obama OR Russian OR Trump OR Fakenews OR news*"

geocode = *39.8, -95.583068847656, 2500km* (reaches the lower 48 states of the USA with a radius of 2500km)

2) *Data Preprocessing* :
The collected tweets were preprocessed to remove redundant, meaningless and unrelated data from the tweets. The preprocessed tweets were used to extract feature words related to query. The frequency of feature words against each tweet is stored in a nested dictionary where keys are the urls (reference website) in the tweet and values are the list of word frequency.

3) *Data Exploration* :
The nested dictionary is converted into python package pandas DataFrame table and stored in csv file format. The generated table was used to plot visualization graph for the feature words using python matplotlib package.

4) *Classification* :
The preprocessed data was used to manually label 520 tweet texts as fake news or legitimate news out of which 260 were labeled fake as '1' and 260 were labeled legitimate as '0'. In order to compensate the equality of both types of news, few texts had been added from the sites listed in [7]. These labeled tweets were used as training data for classification model.

Support vector machine has been used as classification model to distinguish between the fake news tweets from legitimate news tweets using optimal separating hyper plane in the testing data.

D. Experimental Results

SVM classified 3 tweets out of 100 tweets as fake news for the testing data.
The model accuracy is really low to approximately 0.3025 because the size of the training data was small i.e. 520 tweets. It is observed that on increasing the training data size results in better accuracy of SVM model. With the limited time period, I couldn't test the accuracy of predicted values for larger training datasets.

The best parameters of SVM model are given as
Best parameters of SVM = {'kernel': 'linear', 'C': 1.0, 'verbose': False, 'probability': False, 'degree': 3, 'shrinking': True, 'max_iter': -1, 'random_state': None, 'tol': 0.001, 'cache_size': 200, 'coef0': 0.0, 'gamma': 0.0, 'class_weight': None}.

References

[1] Rubin, Victoria L., Yimin Chen, and Niall J. Conroy. "*Deception detection for news: three types of fakes*." Proceedings of the Association for Information Science and Technology 52.1 (2015): 1-4.

[2] Phua, Clifton, et al. "A comprehensive survey of data mining-based fraud detection research." arXiv preprint arXiv:1009.6119 (2010).

[3] (2017, March 15). Aldengolab/fake-news-detection. Retrieved April 02, 2017, from https://github.com/aldengolab/fake-news-detection.

[4] Wakefield, J. (2016, December 02). Fake news detector plug-in developed. Retrieved February 25, 2017, from http://www.bbc.com/news/technology-38181158.

[5] B.S. Detector. (n.d.). Retrieved April 02, 2017, from http://bsdetector.tech/

[6] Examples: Searching Twitter by location and with specific keywords. (n.d.). Retrieved March 30, 2017, from http://thoughtfaucet.com/search-twitter-by-location/examples/

[7] List of fake news websites. (2017, May 13). Retrieved May 14, 2017, from https://en.wikipedia.org/wiki/List_of_fake_news_websites#List_of_fake_news_sites.

[8] Z. (n.d.). Free online word cloud generator and tag cloud creator. Retrieved February 21, 2017, from http://www.wordclouds.com/

[9] Visualization. (n.d.). Retrieved February 21, 2017, from http://pandas.pydata.org/pandas-docs/version/0.18.1/visualization.html.

[10] Visualize Machine Learning Data in Python With Pandas. (2016, September 21). Retrieved February 21, 2017, from http://machinelearningmastery.com/visualize-machine-learning-data-python-pandas/

[11] Conroy, Niall J., Victoria L. Rubin, and Yimin Chen. "Automatic deception detection: methods for finding fake news." *Proceedings of the Association for Information Science and Technology* 52.1 (2015): 1-4.

[12] Fake news. (2017, May 14). Retrieved May 14, 2017, from https://en.wikipedia.org/wiki/Fake_news.

[13] Cross-validation (statistics). (2017, May 02). Retrieved May 14, 2017, from https://en.wikipedia.org/wiki/Cross-validation_(statistics).