INTERNSHIP PROJECT REPORT

ON

Fake or Real News Detection

Presented By

Pradnya Shende

Executive Summary:

The advent of the World Wide Web and the rapid adoption of social media platforms paved the way for information dissemination that has never been witnessed in the human history before. The news media evolved from newspapers, tabloids, and magazines to a digital form such as online news platforms, blogs, social media feeds and other digital media formats. It became easier for consumers to acquire the latest news at their fingertips. These social media platforms in their current state are extremely powerful and useful for their ability to allow users to discuss and share ideas and debate over issues such as democracy, education and health. However, such platforms are also used with a negative perspective by certain entities commonly for monetary gain and in other cases for creating biased opinions, manipulating mindsets, and spreading satire or absurdity. The phenomenon is commonly known as fake news.

There has been a rapid increase in the spread of fake news in the last decade. And area affected by fake news is the financial markets where a rumour can have disastrous consequences and may bring the market to a halt.

Our ability to take a decision relies mostly on the type of information we consume and our world view is shaped on the basis of information we digest. There is increasing evidence that consumers have reacted absurdly to news that later proved to be fake. One recent case is the spread of novel corona virus, where fake reports spread over the Internet about the origin, nature, and behaviour of the virus. The situation worsened as more people read about the fake contents online. Identifying such news online is a daunting task.

Fortunately, there are a number of computational techniques that can be used to mark certain articles as fake on the basis of their textual content. The World Wide Web contains data in diverse formats such as documents, videos, and audios. News published online in an unstructured format is relatively difficult to detect and classify as this strictly requires human expertise. However, computational techniques such as natural language processing (NLP) can be used to detect anomalies that separate a text article that is deceptive in nature from articles that are based on facts.

So, here we used Logistic Regression for detecting fake news and real news.

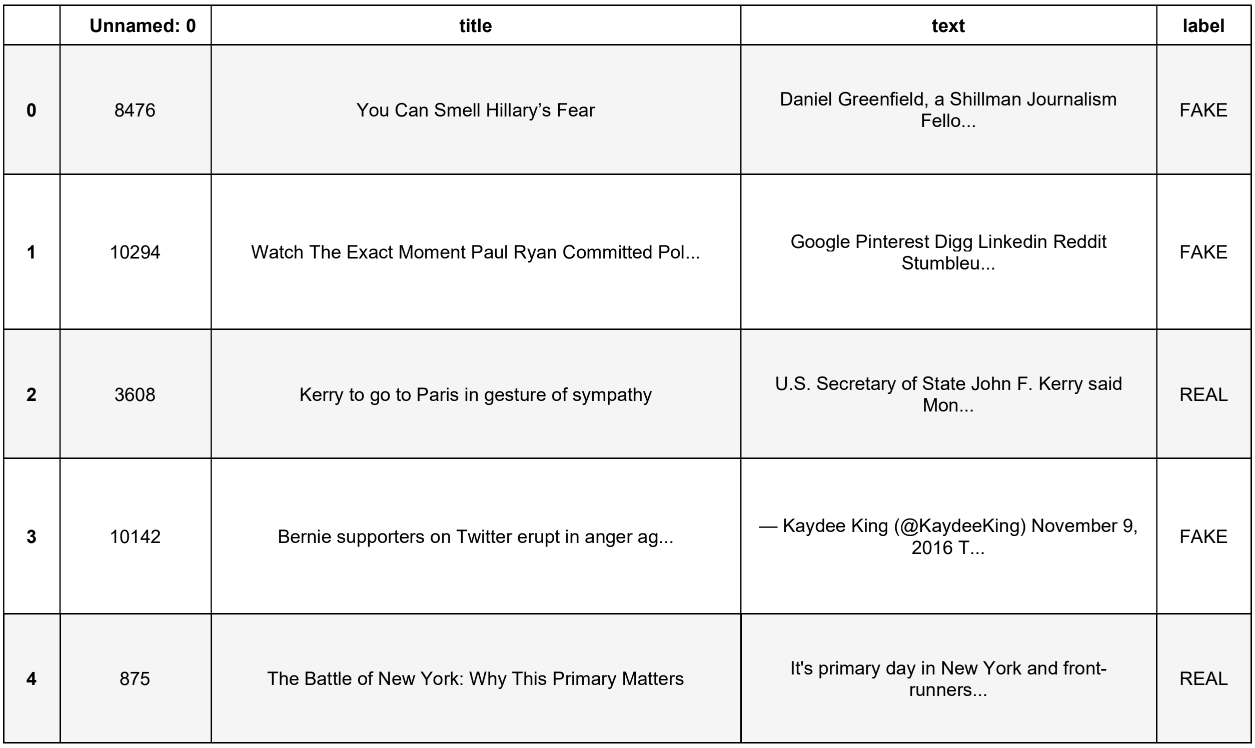
Dataset:

To create Fake and real news predictor, we obtained the dataset from World Wide Web. The dataset contains 6335 observations (rows) and 3 features (columns). The dataset contains 2 categorical features. Title and text are categorical variable. And Our target variable is Label (binary form).

First, dataset is loaded and also, we loaded a range of libraries for general visualization to understand data and deal with unbalanced data.

The following table 1 shows top 5 entries of dataset:

Table 1. Dataset Overview



	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary’s Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

- This are top 5 records consist of 3 variables. Here we have all categorical columns
- Also, we have 1 Target variable and 2 predictor variables
- Label is target variable and other are predictors

Let’s check information about the dataframe.

	Column	Non-Null	Count	Dtype
0	title	6335	non-null	object
1	text	6335	non-null	object
2	label	6335	non-null	object

- Data dimension: There is total 6335 observations and 3 variables
- Data types used in dataset is object
- There is no missing value in whole data

Data Pre-processing:

Removal of unwanted variables

Following variable have been removed from the dataset:

- Unnamed: 0

Missing value treatment:

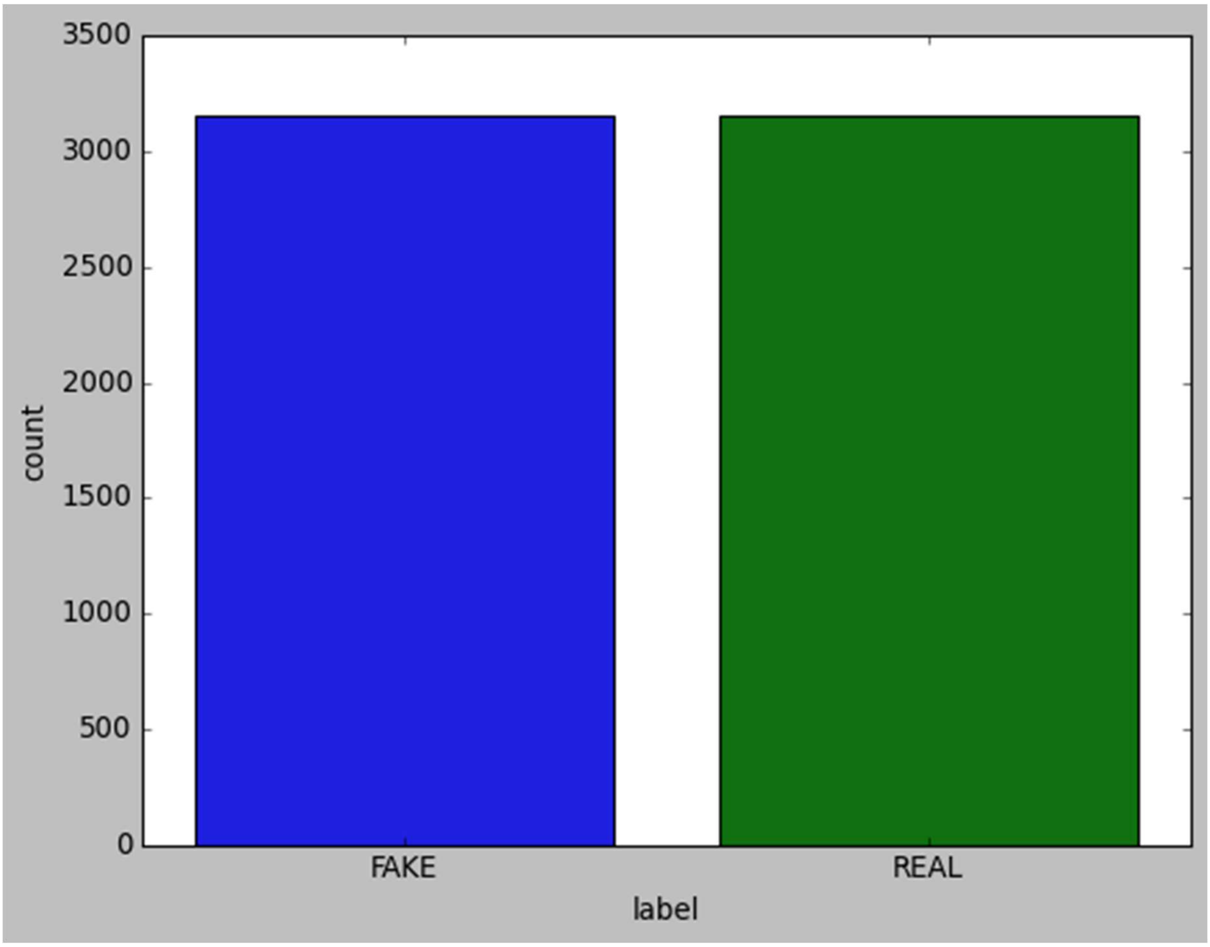
As we don’t have null values in our dataset. So, there is no need of missing value treatment.

Removing Duplicates:

There are 29 duplicated rows in our dataset. So, we dropped all the duplicated rows.

Now, we’ll try to reduce those 2 columns to only 1 column since it will be easier for us to train the model. For that we’ll combine the title and the text columns into one, naming it as content. We can drop the other columns as they don’t have much effect on determining whether the article is fake or not. This step will leave us with 2 columns - content and label.

Count Plot of Target Variable



So, here the distribution is slightly skewed towards the Real news.

Stemming:

Now coming to the stemming part, it basically is the process of reducing a word to its word stem that affixes to suffixes and prefixes or to the roots of words**.**

- First, we use the re package and remove everything that is not a letter (lower or uppercase letters).
- We then convert every uppercase letter to a lower one.

- We then split each sentence into a list of words.

- Then we use the stemmer and stem each word which exists in the column and remove every English stopword present in the list.

- We then join all these words which were present in the form of a list and convert them back into a sentence.

- Finally, we return the stemmed_content which has been pre-processed.

Then We will apply stemming function to our dataset. And next step is to name our input and output features.

Our last preprocessing step would be to transform our textual X to numerical so that our ML model can understand it and can work with it. This is where **TfidfVectorizer** comes into play.

TF-IDF is a measure of originality of a word by comparing the number of times a word appears in a doc with the number od docs the word appears in.

Splitting the Dataset

This means that we have divided our dataset into 80% as training set and 20% as test set. *Stratify = y* implies that we have made sure that the division into train-test sets have around equal distribution of either classes (0 and 1 or Real and Fake). random_state = 1 will guarantee that the split will always be the same.

Now, we will train the model. Let's check the accuracy of our training set predictions.

Accuracy score of the training data: 0.95
Accuracy score of the test data: 0.92
So, test accuracy is also pretty good.

Building a System

Finally, to make this model useful we need to make a system. Taking a sample out of the test-set (I took the fourth sample) and checking our prediction for this sample**.**

With this we have built a system as well.