Kathmandu University

Department of Computer Science and Engineering

Dhulikhel, Kavrepalanchowk

A Project Report On

"Fake News Detection"

[Code No: COMP 308]

For partial fulfillment of Third Year/ First Semester in Computer Science/Engineering

Submitted by:

Bipin Bohora(10)

Jasmin Karki (23)

Divash Ranabhat (42)

Kamal Shrestha (49)

Submitted to:

Dr. Gajendra Sharma

Associate Professor

Project Coordinator

Department of Computer Science and Engineering

Date of Submission: 23 July 2019

Bonafide Certificate

This project work on “Fake News Detection” is the bonafide work of

Bipin Bohora(10)

Jasmin Karki (23)

Divash Ranabhat (42)

Kamal Shrestha (49)

who carried out the project work under my supervision.

Project Supervisor

Dr. Prakash Poudyal

Department of Computer Science and Engineering

Acknowledgement

First of all, we would like to express our heartfelt gratitude to our project supervisor Dr. Prakash Poudyal for his support during the project and co-operating with us to help us carry our project smoothly. His experiences in the field have been a great asset for our project. His encouraging words and working techniques have made this project a successful one.

Our deepest appreciation to all those who provided us the possibility to complete this project. We extend our gratitude to all who have deliberately or unknowingly added a brick in the completion of this project. We enjoyed the duration of the work studying different modules and creating this project.

Abstract

The digital information age has generated new outlets for content creators to publish so-called “fake news”, a new form of propaganda that is intentionally designed to mislead the reader. The widespread use of fake news has a high potential for extremely negative impacts on an individual and society. The first step of such tasks would be to classify claims associated based on the credibility of the news source. So it’s compulsion not to mislead the reader and spread the rumor. A promising solution that has come up recently is to use machine learning to detect patterns in the news articles and sources. We trained a machine to identify the articles and declare it fake if it is supposed to mislead the reader. This system uses various algorithms such as SVM, Naive Bayes, Logistic Regression, k-NN on semantic, syntactic and lexical features to predict whether given news is real or fake with 95% accuracy.

Keywords: Fake News, Fake News Detection, word2vec, doc2vec, Cosine Similarity

List of Abbreviation

ML: Machine Learning

LSVM: Linear Support Vector Machine

LR: Logistics Regression

k-NN: k Nearest Neighbor

NLP: Natural Language Processing

Table of Contents

List of Figure

List of Table

Chapter 1: Introduction

There was a time when if anyone needed any news, he or she would wait for the next-day newspaper. However, with the growth of online newspapers who update news almost instantly, people have found a better and faster way to be informed of the matter of his/her interest. Nowadays social-networking systems, online news portals, and other online media have become the main sources of news through which interesting and breaking news are shared at a rapid pace. However, many news portals serve special interest by feeding with distorted, partially correct, and sometimes imaginary news that is likely to attract the attention of a target group of people. Fake news has become a major concern for being destructive sometimes spreading confusion and deliberate disinformation among the people.

News printed in a newspaper or posted on a social media platform can be classified as “FAKE” if it consists of false information or contradicts with the factual data. Based on the research carried out by Claire Wilde, fake news can be defined with the help of seven variations including Satire or Parody, False Connection, misleading Content, False Context, Imposter Content, Manipulated Content and Fabricated Content. With social media being one of the essential basic human needs, it has also become a platform for peoples with false intention to spread negative and falsified information over it. Fake news is one of the burning modern issues, which has had its effect in the financial sector for losses of millions of dollars, political sector for illicit selection of government, manipulation of common people for information, disruption in many celebrity careers and so on. Few examples of fake news and how they are viewed as are ‘Clickbait’, ‘Propaganda’, ‘Satire/parody’, ‘Sloppy Journalism’, ‘Misleading headings’, ‘Biased or slanted news’. These are the features of fake news and may help to identify and avoid instances of fake news.

The International Federation of Library Associations and Institutions (IFLA) published a summary in diagram form to assist people in recognizing fake news that is:

1. Consider the source (to understand its mission and purpose)
2. Read beyond the headline (to understand the whole story)

3. Check the authors (to see if they are real and credible)
4. Assess the supporting sources (to insure they support the claims)
5. Check the date of publication (to see if the story is relevant and up to date)
6. Ask if it is a joke (to determine if it is meant to be satire)
7. Review your own biases (to see if they are affecting your judgement)
8. Ask experts (to get confirmation from independent people with knowledge)

These are the solutions used by humans to identify whether the news is real or fake. An easier and more efficient way to do so is by using Artificial Intelligence and Machine Learning which would drastically reduce the human time spent to identify the news and all of its related features. We used standards of Natural Language Processing for data processing and feature extraction followed by cosine similarity, sentiment analysis, word movers distance, and so on.

1.1 Background

With the high increase in interaction of people in social networking sites, online news portals there is no guarantee of whether the given information is real or fake. Fake news and lack of trust in the media are growing problems with huge ramifications in our society. Fake news, defined by the New York Times as "a made-up story with an intention to deceive" 1, often for a secondary gain, is arguably one of the most serious challenges facing the news industry today. Sarcastic articles, articles that are blatantly false, satirical, that provide a truthful event but then make some false interpretations, that are pseudoscientific or really just opinion pieces disguised as news or that comprised of mostly tweets and quotes from other people can be considered as Fake News. Facebook has been at the epicenter of much critique following media attention.

This problem of widespread fake news can be solved by understanding what fake news is and with the help of natural language processing of machine learning.

1.2 Objectives

- To find the calculated prediction of how likely the news is falsified.
- To compare and report the results from multiple different model implementations, and present an analysis of the findings.

1.3 Motivation and Significance

At present, different organization, Medias, individuals are creating fake articles, news and links to get attraction and popularity of their brand, product, page, group and site through advertising and click bait. This has been a serious issue as people start believing in wrong information and further spread rumor which is completely misleading the society, falsified news seems to outperform the real one.

“Fake News” is the Collins “Word of the Year” for 2017. Nowadays, Most of the fake news outrage has been centered on the political arena, security, crime, accident statistics, etc. worldwide. There are numerous cases where people believed in fake news and changed their decisions. From a country’s election of the president to the person accused of rape, form a scandalized celebrity tabloid to a case of social harassment, falsified information has led many innocents being accused and harmed, merely knowing the consequences of the crime. Whether the cause be a properly trained journalist or ill minded approach, any sort of falsified news has the potential to cause racisms, harassment, intimidation and damage to reputation. So with the proper categorization of news being faked or factual, this project will provide a crucial tool for us all to tackle the existence of widely and abruptly spread fake news content. Also with the recognition of such content, proper actions can be taken from the concerned authorities, to nullify the content.

1.4 Features

This machine learning project resonates with the following features:

- The model will be able to predict whether the given news is real or fake, given its headline and description.

Chapter 2: Literature Review

Many research works and systems have been developed related to fake news detection. Different methods have been proposed by different researchers.

(Leah C. Windsor) Windsor, Cai, Cupit proposed expectations “Real news will have more syntactic complexity, more concrete words, less narrativity, more deep cohesion and less referential cohesion.”, “Fake news will have more syntactic simplicity, more abstract words, more narrativity, less deep cohesion and more referential cohesion.”, “Fake news text will display features of deceptive language”, “Real news will display features of honest language.” They used Linguistic Inquiry and Word Count (LIWC), singular value decomposition (SVD), t-Stochastic Neighbor Embedding (t-SNE) to meet their expectations.

(Junaed Younus Khan, 2019) Khan, Khondaker, Iqbal, Afroz have compared the performance of traditional machine learning and neural network based deep learning models. They achieved 94% accuracy on combined corpus for Naïve Bayes, with n-gram and also concluded that the addition of sentiment features along with lexical features doesn’t significantly improve accuracy. However, SVM and LR models performed better for them than other traditional models.

(Urja, 2017) Urja have used POS Tagging, sentiment analysis and likelihood models. They found out the use of certain words differentiated between real and bad news. They also used N-grams, punctuation, generality and, the use of LIAR dataset resulted in 49.03% accuracy for statements with Gradient Boosting, 50.16% with Logistic Regression, 77.57% with Extra Trees.

(Vivek Singh, 2017)Many research works concluded high use of sentiment analysis for deception detection in the news article and its body. Conroy, Rubin, Chen, and Cornwell hypothesized expanding the possibilities of word-level analysis by measuring the utility of features like part of speech frequency, and semantic categories such as generalizing terms, positive and negative polarity (sentiment analysis).

Chapter 3: Procedures and Methods

3.1 System Specifications

3.1.1 Software Specifications

- Programming language: Python3, Django-web framework
- Operating System: Windows or Linux
- Tools: Anaconda Prompt, Jupyter Notebook, Weka Tool

3.1.2 Hardware Specifications

Processor: Intel i3-8th Generation Processor

RAM: 8GB, 16GB Recommended

3.2 System Diagrams

3.2.1 Block Diagram

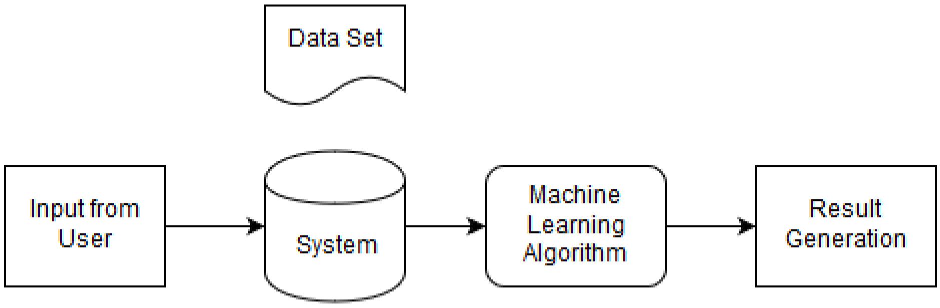

Figure 3.2.1: Block Diagram

3.2.2 Use Case Diagram

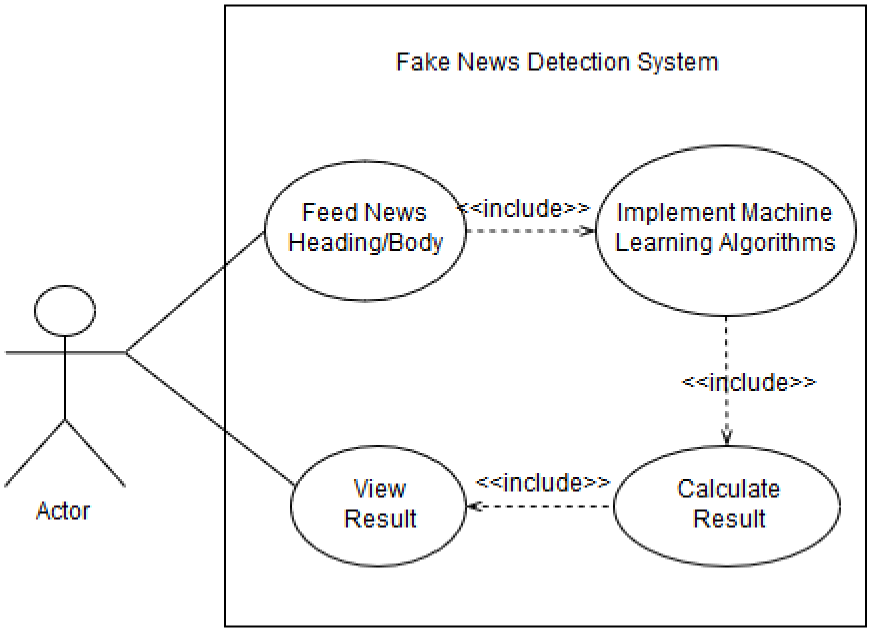

Figure 3.2.2: Use Case Diagram

3.2.3 Flow Chart

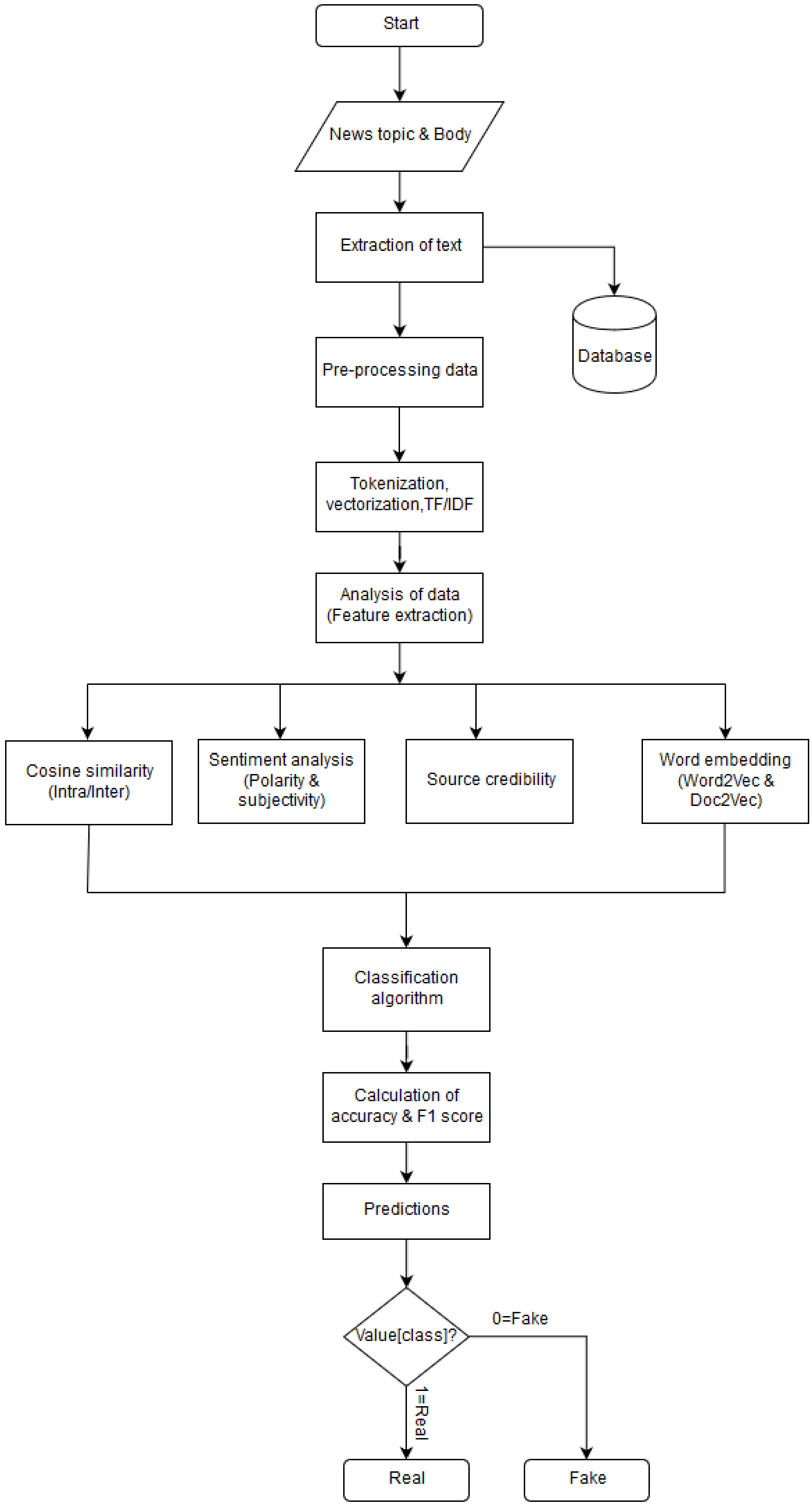

Figure 3.2.3: Flow Chart

3.3 Methodology

3.3.1 Data Collection and Preprocessing

The development of a model required an extensive collection of real and fake news based on the textual features stored in those collected information. There is no such thing as availability of real and accurate news in the most general cases because the data is actually a relative term in reference to the publication date or constant updates in the news and also in our case the actual date of extraction of the data from the website. We acquired a dataset containing a collection of fake news (13000+) scrapped from webhose.io in a period of 30 days, which were flagged fake, bullshit or conspiracy using the chrome extension BS Detector (by Daniel Sieradski), extracted by Megan Risdal, Kaggle (Risdal). This dataset of fake news was much preferred in comparison to other collections as many projects based on the same objective were carried out using these collections. Similarly, a dataset for collection of real news were self-prepared using the data scraping using the official APIs provided by the news company themselves. A website called News API was used to acquire name-tags for different publishers of English news, which was then used to extract the daily news using the company's API. This method of scraping was carried out for a period of 10 days collecting an approximate of 10000+ real news data. Both of the datasets were combined and subjected to various pre-processing techniques, by which we were able to obtain our golden dataset for the project (18000+ after pre-processing).Elimination of null/missing values, foreign language, use of extensive special characters to English language, and so on were the basic pre-processing techniques carried out.

3.3.2 Attributes Selection

The main aim of our project is to analyze texts, paragraphs, titles, sources and so on to make some sense of what the actual text is saying. Briefly saying we have to analyze the string values for each entry of news which is to be categorized either fake or real, but any machine learning algorithm only takes categorical numerical values as its input to develop a model and make predictions, which is a bummer. So, one of the most challenging jobs was to select some very appropriate attributes that would successfully extract necessary numerical features to develop a model. Selection of features was divided in to two categorical forms: Lexical & Syntactical Analysis of the text and Semantic Analysis of the text.

Generally, real news published by authenticated and prestigious source have several layers of editing and factual checks so there is very few chances of grammatical and factual errors like incorrect spaces between words, after comma, after full stop, randomly capitalized words, extensive sentence to word ratios and so on, based on which lexical and syntactical feature extraction was done. It included the following features:

1. **Nouns Count**

   It is the count of proper nouns in the given text. Proper nouns were categorized as keywords which were then used for modeling word2vec and doc2vec models.

2. **Punctuation Count**

   It is the count of punctuation marks in the given text. Generally, fake news has more punctuation due to the use of fake dialogues, statements.

3. **Capitalized Words Count**

   It is the total capital word count in the given text. Fake news tends to incorrectly capitalize words in between the text.

4. **Incorrect Spaces between Words**

   It is the count of incorrect number of spaces (more than 1) between any two words. In real news, there is single space between words.

5. **Incorrect Space after Comma**

   It is the count of incorrect spaces after comma. There is always single space after comma in English Grammar.

6. **Incorrect Space after full stop**

   It is the count of incorrect spaces after full stop. There is always a single space after full stop in English Grammar.

7. **Incorrect Full Stop at the end of title**

   Fake News tends to use full stop at the end of news heading. It is to check whether full stop is present or not.

8. **Sentence to Word Ratio**

   Real News tend to have less words in a sentence in compare to fake news as it is rigorously edited by professional editors.

Similarly, the semantic features of the news included:

1. **Words Count with CountVectorizer and TF-IDF:**

   The CountVectorizer provides a simple way to both tokenize a collection of text documents and build a vocabulary of known words, but also to encode new documents using that vocabulary. It can be simply be considered as a matrix containing all the unique words and their frequency, for every sentence or even documents. Similarly, TF_IDF short for term frequency-inverse document frequency is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. The TF-IDF value increases proportionally to the number of times a word appears in the document and is offset by the number of documents in the corpus that contain the word, which helps to adjust for the fact that some words appear more frequently in general. Similar to CountVectorizer, TFIDF also calculates values on the basis of term frequency but in countvectorizer common words like “a”, ”the”, ”and”, ”an” etc will appear most of the time and other words that carry the topic information of your document will be less frequent. For instance if you’re training a classifier to identify documents related to AI you don’t want it to learn words like “a” and “the” because they’ll be in every single document (both related to AI and not related documents).The way to combat this problem is to use TF-IDF. What TF-IDF does is it balances out the term frequency (how often the word appears in the document) with its inverse document frequency (how often the term appears across all documents in the data set). This means that words like “a” and “the” will have very low scores as they’ll appear in all documents in your set. Rarer words like for instance “machine learning” will be very common in just a handful of documents which talk about computer science or AI. TF-IDF will give higher scores to these words and thus they’ll be the ones that the model identifies as important and tries to learn.

Hence TF IDF was more preferred to CountVectorizer for calculating cosine similarity between the news elements and also news.

Variants of term frequency (tf) weight

weighting scheme	tf weight
binary	$0, 1$
raw count	$f_{t,d}$
term frequency	$f_{t,d} \Big/ \sum_{t' \in d} f_{t',d}$
log normalization	$\log(1 + f_{t,d})$
double normalization 0.5	$0.5 + 0.5 \cdot \frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$
double normalization K	$K + (1 - K) \frac{f_{t,d}}{\max_{\{t' \in d\}} f_{t',d}}$

Table 3.3.2.1: TF Table

Variants of inverse document frequency (idf) weight

weighting scheme	idf weight ($n_t = \lvert\{d \in D : t \in d\}\rvert$)
unary	1
inverse document frequency	$\log \frac{N}{n_t} = -\log \frac{n_t}{N}$
inverse document frequency smooth	$\log\left(\frac{N}{1 + n_t}\right)$
inverse document frequency max	$\log\left(\frac{\max_{\{t' \in d\}} n_{t'}}{1 + n_t}\right)$
probabilistic inverse document frequency	$\log \frac{N - n_t}{n_t}$

Table 3.3.2.2: IDF Table

2. **Cosine Similarity:** One of the most important characteristics of Natural Language Processing (NLP) and text comparison is the cosine similarity between two documents or simply the similarity between two documents. Previously obtaining the two vectorical forms of a news text in the form of TF-IDF for two comparative documents, we can calculate the cosine similarity using:

$$\cos\theta = \frac{d1.d2}{|d1||d2|}$$

Cosine Similarity was calculated to measure similarity of texts between:

- News Headline and News Description for single News
- News Headline of a given news and News Headline of Several Real news (in our context at minimum of 10 real news were used to compare, but in ideal comparison case preferably with at least 100 real news)
- Same News Headline was now compared to Fake news in the similar fashion i.e. Specific News Headline Vs Fake News Headline
- Similar procedure was carried out for News Body i.e.
- Specific News Description Vs Real News Description
- Specific News Description Vs Fake News Description

Cosine Similarity was also measured in two parameters:

- Ratio of similarity: Ratio of total similarity of a specific news with the total availability of news
- Numerical Similarity: Ratio of total similar news having similarity greater or equal to 5 to the total availability of news.

3. **Source Credibility:**

It is pretty evident from the study of sources of news to distinguish whether the news is real or probably fake. It can tautologically be said that the news appearing from a reputed and well managed source like a prestigious news company is most probably real news because these companies compulsorily have several layers of editorials and factual checking for every news every day, which is a clear indication of news being not fake. It was very vital for us to categorize and numerically classify our news sources in terms of their source and their credibility to being and authenticate news. In lack of any proper rankings, measurement criteria and source authenticity, we were forced to use the biggest publisher on Facebook ranking table provided every month by Facebook, based on the tallied score of total likes, shares, comments, and reactions for that given month which listed 23 trusted sources.

4. Sentiment Analysis:

Sentiment Analysis was carried out using TextBlob feature of standard NLP, which measures the sentiment of the given sentence (in our context, news) in terms of two defining parameter:

- Polarity: Polarity is a float value within the range [-1.0 to 1.0] where 0 indicates neutral, +1 indicates a very positive sentiment and -1 represents a very negative sentiment.
- Subjectivity: Subjectivity is a float value within the range [0.0 to 1.0] where 0.0 is very objective and 1.0 is very subjective. Subjective sentence expresses some personal feelings, views, beliefs, opinions, allegations, desires, beliefs, suspicions, and speculations whereas Objective sentences are factual.

5. Word Embeddings using Word2vec:

Word embedding is one of the most popular representations of document vocabulary. It is capable of capturing context of a word in a document, semantic and syntactic similarity, relation with other words etc and word2vec is one of the most important and popular techniques to learn it. When it comes to natural language processing systems traditionally it treats words as discrete atomic symbols, and therefore ‘cat’ may be represented as Id537 and ‘dog’ as Id143.These encodings are arbitrary, and provide no useful information to the system regarding the relationships

that may exist between the individual symbols. This means that the model can leverage very little of what it has learned about ‘cats’ when it is processing data about ‘dogs’ (such that they are animals, four-legged, pets, etc.). One of the ways of representing a word in terms of vector across an axis of words and their closeness can be achieved by word2vec modeling.
We were able to model word2vec modeling in two numbers of ways:

- Use of words and sentences available to us in our dataset
- Standard Data Dumps from Google and Wikipedia

Using these two techniques a model was prepared which was used to calculate the Word Mover’s Distance between two sentences (in our context preferably the news title and its description).It allows us to assess the “distance” between two sentences using word2vec to measure the similarity, which by convention is: Two similar documents will have a *high* similarity score and a small distance; two very different documents will have *low* similarity score, and a large distance.
This calculation was carried out for sentences referring to :

- News Headline Vs News Description
- Specific News Headline V News Headline from a Real News (Number of real news taken here is 10 due to hardware restrictions but ideally should be taken 100+)
- Specific News Headline V News Headline from a Fake News
- Similar two attributes for News Description

6. WordEmbeddings Using Doc2vec:

Similar to word2vec, document to vector was also modeled out using the available texts from news descriptions and headlines from datasets and iterated to develop a model using 10 epochs. No any data dumps were used in this process as the calculation burden would be very large to handle. Rather than WMDistance, cosine similarity is more preferable for this technique of calculating the similarity between any two given texts. In addition to that the actual vectorized doc2vec value of the sentence was also used in prediction. Similar to comparisons made above, the following comparisons were made referencing doc2vec model:

- News Headline V News Description
- Doc2 vec of Specific News Headline
- Doc2 vec of Specific News Description

3.3.3 Algorithms Implemented

With the integration of all these attributes in reference to their source, URL, News Headline, News Description, four major models or Machine Learning were used to develop their respective model based on the processed dataset prepared and eventually make predictions, calculate accuracy of predictions, F1 score and other metrics of calculations. Models were prepared by splitting the dataset into 70-30 training and testing split, with random state of 42.Four of the models used are:

a. Naive Bayes Classification Algorithm- Multi Class Classifier
    - Gaussian Naive Bayes
    - Bernoulli Naive Bayes
b. Linear Support Vector Machine (random state=0, tol=1e-5)
c. KNN Algorithm
d. Logistic Regression

3.3.4 Results and Prediction

Naive Bayes, KNN Algorithm, Logistic Regression and Linear Support Vector Machine models from sklearn were used to make final predictions and

Results were evaluated in several stages of development of model based on the classification of attributes:

- Results from the Lexical and Syntactic Attributes
- Results from Semantic Attributes
- Results from Word2vec Modelling
- Results from Doc2vec Modelling
- Combined Results

S.N.	Attributes	Naive Bayes Algorithm (Gaussian NB)		Linear Support Vector Machine (LSVM)		KNN Algorithm		Logistic Regression	
		Accuracy	F_1	Accuracy	F_1	Accuracy	F_1	Accuracy	F_1
1	Lexical and Syntactic Attributes	92.56%	0.93	91.93%	0.92	94.96%	0.95	94.63%	0.95
2	Logical Semantic Attributes	58.53%	0.48	83%	0.84	82.28%	0.82	82.23%	0.82
3	Word Embedding: Doc2Vec	91.57%	0.92	92.04%	0.92	91.59%	0.92	91.59%	0.92
4	Word Embedding: Word2vec	73%	0.74	64.26%	0.63	78.08%	0.78	78.08%	0.78
5	Combined Results	93.66%	0.94	94.391%	0.94	95.09%	0.95	95.09%	0.95

Table 3.3.4: Results from all algorithms

3.4 Data Visualization

Cummulative Attributes	Naïve Bayes	Linear SVM	KNN	Logistic Regression
Lexical/Syntactical	0.93	0.92	0.95	0.95
Semantic	0.93	0.94	0.95	0.95
Doc2vec	0.94	0.94	0.95	0.95
Combined Attributes	0.94	0.94	0.95	0.95

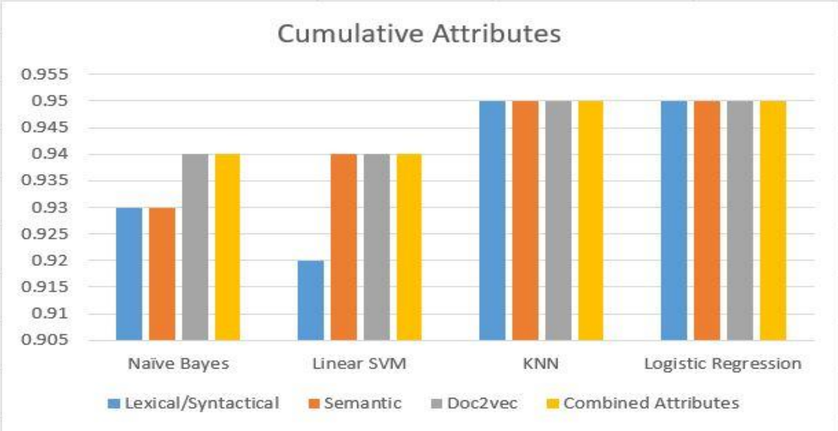

Figure 1.4.1: Accuracy for Cumulative Attributes

Cummulative Attributes	Naïve Bayes	Linear SVM	KNN	Logistic Regression
Lexical/Syntactical	92.77	92.02	94.65	94.65
Semantic	93.34	92.02	94.65	94.65
Doc2vec	93.64	94.46	95.04	95.04
Word2vec	93.66	94.39	95.09	95.09

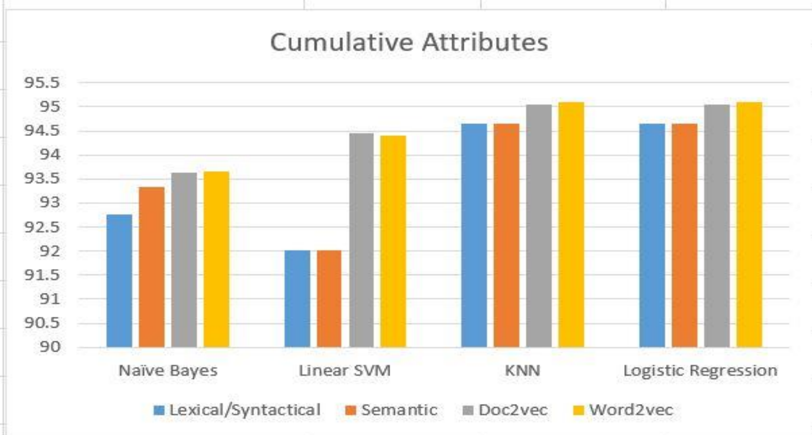

Figure 3.4.2: F1 Score for Cumulative Attributes

Attributes	Naïve Bayes	Linear SVM	KNN	Logistic Regression
Combined Attributes	93.66	94.39	95.09	95.09

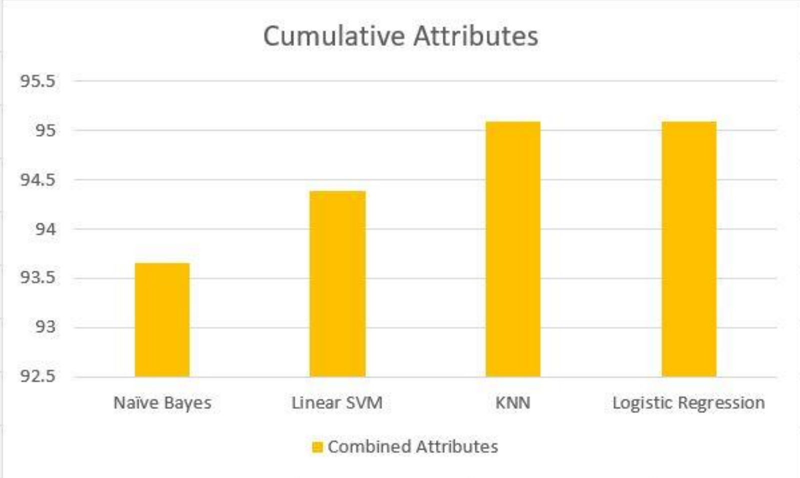

Figure 3.4.3: Final Accuracy for Cumulative Attributes

Chapter 4: Discussion on the Achievements

The main objective of our project was to successfully predict whether given news with its headline and its description is real or fake on the basis of different features selected using varied standard techniques of Natural Language Processing. With reference to various related works, previously done on this subject using similar attributes using similar model fitting algorithms, they were able to obtain a maximum accuracy of 67% using Support vector machine (SVM) using lexical analysis whereas 66% using lexical and sentiment analysis. In modeling using Logistic regression, they were able to attain an accuracy of 67% and 86% using Naïve Bayes Algorithm and 71% accuracy using k-NN algorithm.

Similarly, using the similar techniques we were able to attain an accuracy of 93% using Naïve Bayes, 95% using LSVM and 95% using k-NN algorithm for our dataset containing a total of 18000 data entries with an F1 score of 0.91-0.95 among the algorithms.

On using some additional attributes like word embedding: wod2vec and doc2vec we were able to increase the accuracy to 94-95% on every algorithm we tested out, with 0.94-0.95 F1 score between them. Comparisons among the news headline and the news descriptions were made in the previously mentioned fashion (in Methodology) for more detailed analysis and detailed feature extraction which resulted in a total of 34 attributes including the lexical and semantic analysis but in references to the previous works done, not more than 6-7 attributes were used, which were pretty much straight forward comparison of features.

Chapter 5: Conclusion and Recommendation

5.1 Limitations

- The project limits in making sufficient comparisons of a specific news in references to Real and Fake news (only 10 iterative comparisons were made but ideally at least 100+ comparisons should be made) to determine the real authenticity of news
- With lack of sufficient processing speed, data dumps from Google and Wikipedia were not used, which limited the model in comparison to previous works
- Credibility of source was based upon the rank of publishers from Facebook, because of the availability of properly ranked news companies which limited our project to provide proper source credibility
- With the lack of use of neural networks and deep learning techniques, calculation semantics of the news was not carried, which limited the project efficiency.

5.2 Future Enhancements

This project can be further enhanced in future by:

- Comparison of fake news with its corresponding real news from different news source can be compared simultaneously.
- We can implement deep learning and convoluted neural network techniques to improve the efficiency and accuracy of results.
- Proper use of Source credibility values can be computed and implemented.
- Recommendation of URLs containing the given news can be made.

References

1. Ahmed H, T. I. (2017). Detecting opinion spams and fake news using text classification. *WILEY*.

2. Bajaj, S. (n.d.). *"The Pope Has a New Baby!" Fake News Detection Using Deep Learning.*

3. C.Fan. (n.d.). *Classifying Fake News.*

4. Junaed Younus Khan, M. T. (2019). A Benchmark Study on Machine Learning Methods for Fake News Detection. *Cornell University*.

5. Kai Shu, A. S. (2017). Fake News Detection on Social Media: A Data Mining Perspective. *ACM SIGKDD Explorations Newsletter*.

6. Leah C. Windsor, Z. ". (n.d.). *Syntactic, Semantic, and Topics: The Cognitive Framework of Fake News.*

7. Risdal, M. (n.d.). *Getting Real about Fake News*. Retrieved April 18, 2019, from Kaggle: https://www.kaggle.com/mrisdal/fake-news/version/1

8. Urja. (2017). *The Linguistic Features of Fake News Headlines and Statements.* University of Amsterdam .

9. Victoria L. Rubin, N. J. (n.d.). Fake News or Truth? Using Satirical Cues to Detect Potentially Misleading News. *Workshop on Computational Approaches to Deception Detection.* San Diego: Association for Computational Linguistics: Human Language Technologies.

10. Vivek Singh, R. (2017). *Automated Fake News Detection Using Linguistic Analysis and Machine Learning.* 10.13140/RG.2.2.16825.67687.