A PROJECT REPORT ON

FAKE NEWS DETECTION USING MACHINE LEARNING

A DISSERTATION
SUBMITTED FOR THE MID TERM PROJECT EVALUATION (MTE)
FOR THE COMPLETION OF COURSE MACHINE LEARNING (CO-327)
IN

COMPUTER ENGINEERING

Submitted by:
Sourabh
2K18/CO/355
Vishruth Khare
2K18/CO/393

Under the supervision of
Ms. Juhi Jain

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi- 110042

November, 2020

DECLARATION

I, Sourabh, Roll No. 2K18/CO/355 along with Vishruth Khare, Roll No. 2K18/CO/393 of B.Tech Computer Engineering, hereby declare that the project titled "Fake News Detection Using Machine Learning" which is submitted by us to the Department of Computer Engineering, Delhi Technological University (DTU), Delhi for the mid-term project evaluation (MTE), is unique and not derivative from any source without proper citation.

New Delhi

INDIA

November 16th 2020

ABSTRACT

The dilemma of Fake news has evolved much faster in the current years. Social media has noticeably tainted its reach and impact as a whole. Along with some merits like its stumpy cost, and easy ease of understanding with rapid share of information draws more concentration of people to read news from it. The demerit is, it enables broad increase in counterfeit news, which is nonentity but false in sequence to mislead people. As a result, automation of Fake news detection has turn out to be crucial in order to preserve vigorous online and social media.

In the project, a few Machine learning techniques are engaged to spot the trustworthiness of the news. It uses a term frequency based feature to guide the Algorithms including Support Vector Machine (SVM), Multinomial Naïve Bayes and Passive Aggressive Classifier (PAC). The results were found to be very promising and have scope for more research in the area.

Table of Contents

Declaration 2
Abstract.. 3
List of Contents......... 4
List of Tables 6
List of Figures 7
List of Symbols, Abbreviations 8

CHAPTER 1 INTRODUCTION...9

CHAPTER 2 LITERATURE REVIEW 11
2.1 Impact of Fake news... .11
2.2 Definition and its types 12
2.3 Different types of Detection Methods.................................. 13

CHAPTER 3 METHODOLOGY 15

3.1 Response based detection.. 15

3.2 Flowchart 16

CHAPTER 4 STEPS OF METHOD IMPLEMENTATION…………… 17
4.1 Text Pre Processing.. 18
4.2 Feature generation 20
4.3 Algorithms used for classification 23
4.4 Metrics used to access the Performance of Model 25

CHAPTER 5 RESULTS and ANALYSIS............................. 28
5.1 The Proposed Experiment 28
5.2 Classification Accuracy 29

5.3 Classification Report 30
5.4 Confusion Matrix 31
5.5 Comparison between models.....................................32

Conclusion.. 34

References 36

LIST OF TABLES

Table 4.1- Word Document Matrix

Table 4.2- Confusion Matrix

Table 5.1 Classification accuracy

Table 5.2 Classifier Time

Table 5.3 Vectorizer Time

LIST OF FIGURES

Figure 3.1- Project Flowchart

Figure 4.1-Dataset

Figure 4.2- Text Pre Processing

Figure 4.3- Naïve Bayes

Figure 4.4- Passive Aggressive

Figure 4.5- SVM

Figure 5.1-Vectorizer Time

Figure 5.2 Confusion Matrix for Linear SVM-TFIDF, Split 1

Figure 5.3 Confusion Matrix for Linear SVM-TFIDF, Split 2

Figure 5.4 Confusion Matrix for Linear SVM-TFIDF, Split 3

Figure 5.5-Vectorizer Time Execution

Figure 5.6- Execution Time of Classifier

Figure 5.7- Performance Graph of Classifier

LIST OF ABBREVIATIONS

TFIDF – Term Frequency Inverse Document Frequency
SVM – Support Vector Machine
PAC: Passive Aggressive Classifier
TP: True Positive
FP: False Positive
TN: True Negative
FN: False Negative

Chapter 1

INTRODUCTION

The increased use of social media has led upto, in sequence being generously reachable to all and sundry. With the evolution of internet and spread of social media, it provides a massive quantity of in order however the trustworthiness of information depends ahead several factors. Mammoth sum of in rank is available daily via online along with media, excluding it cannot trouble-free to tell whether the information is a accurate or not. “Fake News” is the expression labeled for kind of untrue information. The fictitious information is a premeditated try with the goal in arrange to spoil or errand an association, entity or individual‟s status or it can be simply with the motive to gain fiscally or politically.

“The aim of project is to develop a model that is able to predict whether the users claim is fake or real.” This project “Fake News Detection System” uses machine learning algorithms and natural language processing techniques. Natural –language processing is an area of 2 computer science and artificial intelligence concerned with connections flanked by computers and human (natural) languages, in particular how to program computers to process and analyze large amounts of natural language data.

Chapter 2

Literature Review

In recent time fake news has become a hot topic of discussion. So I have looked upon few works on the same topic and concluded some of the major topics below.

2.1 Impact of Fake News

Fake news influences people's decisions regarding whom to make your choice for during elections. According to the researchers at the Oxford Internet Institute, in the run up to 2016 US Presidential election, Fake news was prevalent and spread rapidly with the help of social media bots" [1]. "A communal bot refers to an account on social media that is programmed to manufacture content and interact with humans or other wicked bots" [6]. "Studies reveal that these bots influenced the election online discussions largely" [1]. "Fake news hinders serious media coverage and makes it more difficult for journalists to cover important news stories" [2]. "An analysis done by Buzzfeed revealed that the top 20 Fake news stories about the 2016 US Presidential election established more concentration on Facebook than the top 20 election stories from 19 major media outlets" [3].

2.2 Definitions and Types

Fake news is news that is intentionally and verifiably false and has the potential to mislead viewers/readers. There are two important dimensions of this definition: "intention" and "authenticity". First, fake news propagates misinformation that can be verified. Second, most of the fake news is formed with fraudulent intent to deceive community. This definition is widely adopted in recent research analysis [11; 12; 13; 14]. In general, Fake news can be categorized into three groups. In first group - "Actual Fake News", we can put those types of news, which are false and made up by the author of the article. The second group –"Fake news that is actually satire" is created purely to amuse rather than mislead its audience. Therefore, intentionally misleading and deceptive fake news is different from obvious satire or parody. The third group is "Poorly reported news that fits an agenda". This type of news has some real content but is not entirely correct and is designed especially for some political propaganda.

Many researchers have streamlined the types of Fake news to simplify their research. For instance, According to definitions given by [1], there are a few types of news that cannot be called as "Fake"- (1) "Satire news having proper context." (2) "Misinformation that is created unintentionally." (3) Conspiracy theories those are difficult to put in true/false dichotomies. This paper [1] has presented two main aspects of fake news detection problem: “characterization” and “detection”.

2.3 Different types of Detection Methods

1. **Knowledge Based Detection**: It aims to use external sources to fact-check the claims made in the news content. Two typical external sources are open web and knowledge graph. Open web sources are compared to the claims in terms of consistency and frequency [18, 19], whereas Knowledge graph is used to check whether the claims can be inferred from existing facts in graph or not [20, 21, 22].Many fact-checking websites (For eg. AltNews, Snopes, Smhoaxslayer, Boomlive) are using domain experts to determine manually the news veracity. Facebook has recently partnered with Indian fact-checking agency Boomlive to spot false news circulation on its website [23].
2. **Stance based detection**: This technique compares how a series of posts on social media or a group of reputable sources feels about the claim -Agree, Disagree, Neutral or is a unconnected. In [10], the writers old lexical as well as similarity features fed through a multi-layer preceptor

(MLP) with one hidden layer to detect the stance of the articles. They hard-coded reputation score feature (Table 2.1) of various sources based on nationwide research studies. Their model achieved 82% accuracy for pure stance detection on their dataset. Another paper [13] used “wisdom of crowd” feature to improve news verification by discovering conflicting viewpoints in micro blogs with the help of topic model method - Latent Dirichlet Allocation (LDA). Their overall news veracity accuracy reached up to 84%.

3. **Other related works**: In implement Document comparison scrutiny, that calculate the Jaccard match, a far and wide used correspondence evaluate, between a hearsay „n‟ in test set with each news in Fake news instruction set „F‟ and real news instruction set „R‟. The outcome obtain were very shows potential. In [16] the authors have exploited the distribution patterns of in sequence to detect the hoax. Many delve into papers have used different linguistic and word embed features. The most common ones are tf-idf, word2vec, punctuations, ngrams, PCFG.

CHAPTER 3

METHODOLOGY

3.1 Response based detection

It is a projected loom to categorize fake news further precisely by "analyzing the response on such news articles". Accomplishment of same was carried out in five .sub .phases:

1) Collection of data from social. media. platform, Facebook and Twitter

2) Choosing. relevant features for classification and Training the Model

3) Evaluation of different model performance based on extracted lfeatures

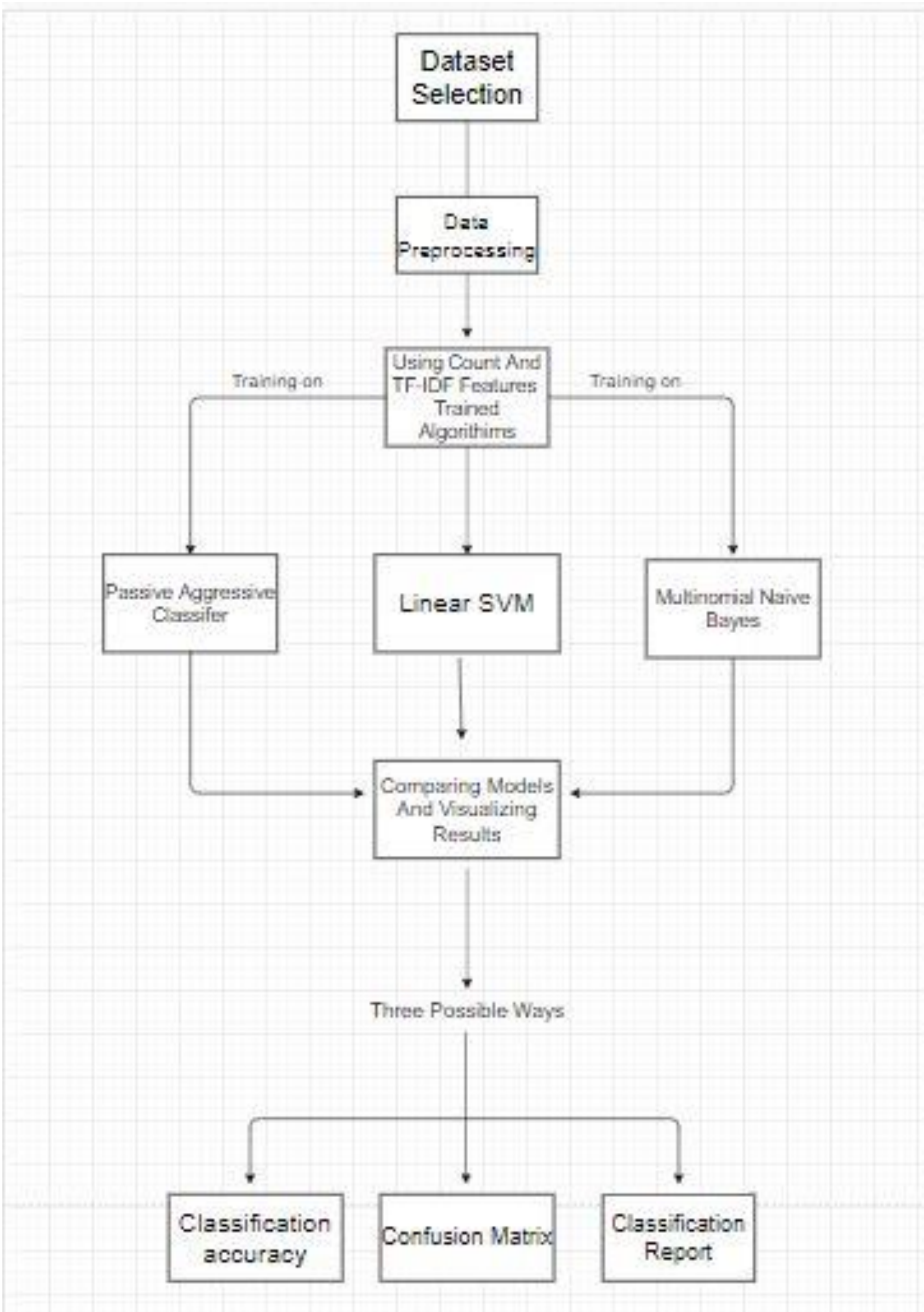
4) Improving presentation

5) Argument and Presentation of results

This project was developed in Python using Sci-kit libraries. Python has a massive set of libraries in addition to extensions, which can be easily used in Machine Learning. Sci-Kit Learn library is the finest foundation for machine learning algorithms where virtually all types of machine learning algorithms are gladly offered for Python, thus easy and quick estimate of ML algorithms is potential.

3.2 Flowchart

Figure 3.1- Project Flowchart



CHAPTER 4

STEPS OF METHOD IMPLEMENTATION

Dataset

The dataset used for the project was downloaded from *https://www.kaggle.com/c/fake-news/data*

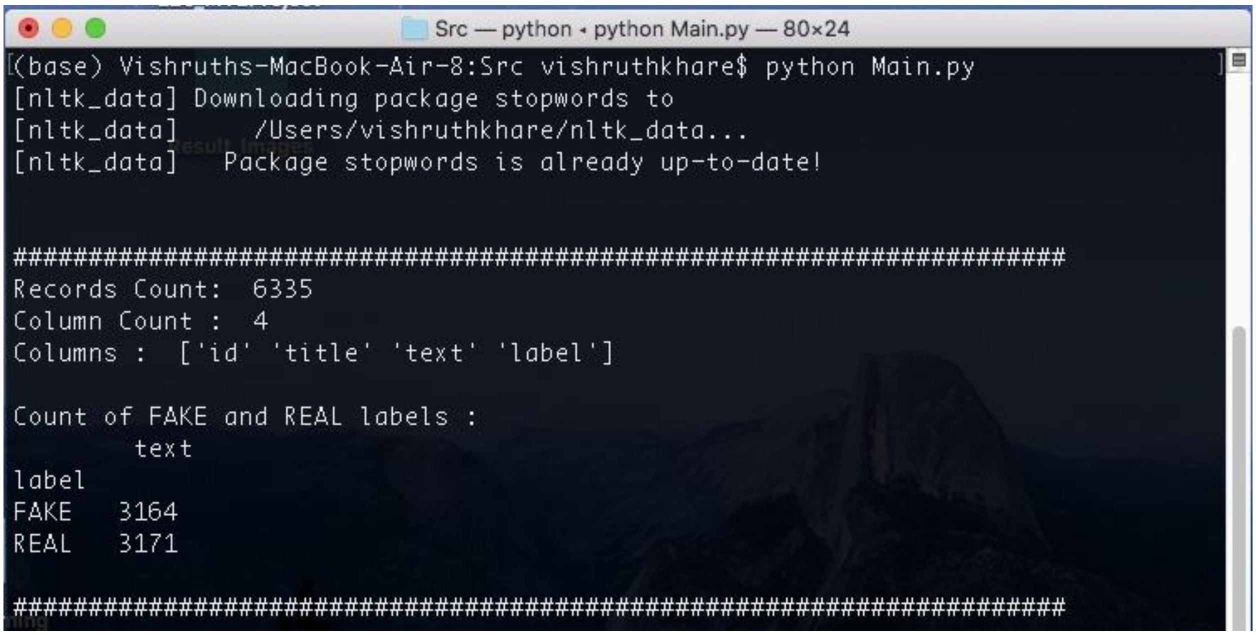
train.csv: A full training dataset with the following attributes:

- id: unique id for a news article
- title: the title of a news article
- text: the text of the article; could be incomplete
- label: a label that marks the article as potentially unreliable

  1: unreliable

  0: reliable

Figure 4.1-Dataset



4.1 Text Pre Processing

The data on social media is extremely amorphous – bulk of it is casual announcement with typing errors, slangs and poor-grammar. For better impending, it is essential to clean the data before it can be used for extrapolative modeling. So, fundamental pre-processing was completed on the training data. This step was comprised of

1. Conversion to Lower case: Initial step was to change the text into lower case, this is done to avoid multiple copies of the same words. For e.g. while ruling the word count, “Basic” and “basic” is full as unlike words.

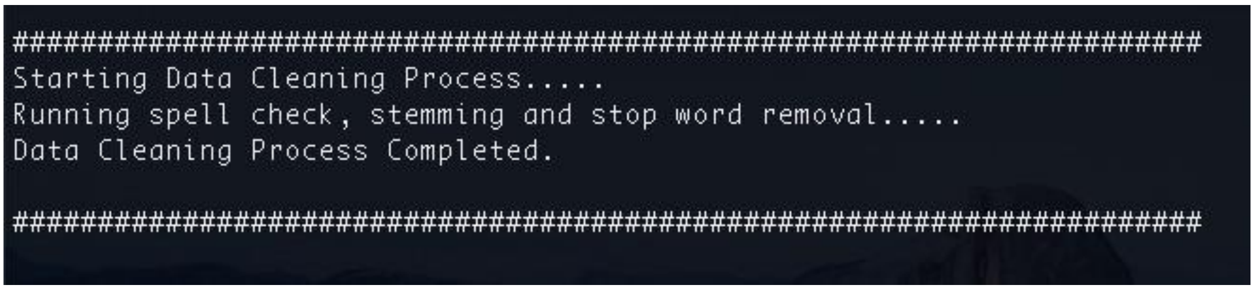
2. Removal of Punctuations: Punctuations do not have a great deal consequence while treating the text data. So, removing them helps to trim down the size by and large text.

3. Stop-words removal: Stop-words are the largest part regularly taking place old words in a quantity. These are for e.g. a, the, of, on, at etc. They routinely classify the constitution of a text and not the context. When treated as feature, it would result in deprived presentation. So, Stop-words are uninvolved from the training data as the part of text onslaught process.

4. Tokenization: It refers to dividing the text into a sequence of words or group of words like bigram, trigram etc. Tokenization was done so that frequency-based vectors values could be obtained for these tokens.

5. Lemmatization: It converts the words into its word root. With the help of a vocabulary, it does morphological analysis to pick up the root word. In this work, Lemmatization was performed to improve the values of frequency-based vectors.

Figure 4.2- Text Pre Processing



4.2 Feature generation

Count Vector

Count Vector can be understood as matrix memo of the dataset, in which rows correspond to the credentials in the quantity, columns symbolize a term from the amount, and cells stand for the count of that scrupulous term in a exacting file. The glossary is fashioned using the inventory of only one of its kind tokens or words in the body.

Example: Let us think three papers in a body C, i.e. D1, D2 and D3 containing the text as below:

D1: It was raining deeply yesterday.

D2: Bad weather caused profound rainfall in London.

D3: Yesterday, London newspapers warned of intense rainfall.

The dictionary can be created with unique words. The unique words identified are:
[Rain, Heavy, Yesterday, Bad, Weather, London, Newspapers, Warned]
No of Documents D = 3
No of Unique words N = 8

Count Matrix represents the occurrence of every term in every document.
The Count matrix M = 3 X 8 is represented below:

Table 4.1- Word Document Matrix

	Rain	Heavy	Yesterday	Bad	Weather	London	Newspaper	Warned
D1	1	1	1	0	0	0	0	0
D2	1	1	0	1	1	1	0	0
D3	1	1	1	0	0	1	1	1

TF-IDF vectors

TF-IDF weight depicts the comparative significance of an expression in the article and entire quantity.

TF stands for Term Frequency: It studies how recurrently an expression appears in a file. As mostly all the file size varies, an expression may become visible more in an extensive sized file, than that in a short one. So the measurement lengthwise of the text often divides Term frequency.

$$TF(t,d) = \frac{Number\ of\ times\ t\ occurs\ in\ a\ document\ 'd'}{Total\ word\ count\ of\ document\ 'd'}$$

IDF stands for Inverse Document Frequency: A word is not of much use if it is present in all the documents. Certain terms like “a”, “an”, “the”, “on”, “of” etc. appear many times in a document but are of little importance. IDF weighs down the importance of these terms and increase the importance of rare ones. The more the value of IDF, the more unique is the word.

$$IDF(t) = \log_e\left(\frac{Total\ number\ of\ documents}{Number\ of\ documents\ with\ term\ t\ in\ it}\right)$$

TF-IDF – Term Frequency-Inverse Document Frequency: TF-IDF works by penalizing the most commonly occurring words by assigning them less weight age while giving high weight age to terms, which are present in the proper subset of the corpus, and has high occurrence in a particular document. It is the product of Term Frequency and Inverse Document Frequency.

$$TFIDF(t,d) = TF(t,d) * IDF(t)$$

TF-IDF is a widely used feature for text classification. In addition, TF-IDF Vectors can be calculated at different levels i.e. Word level and N-gram level, which I have used in this project.

i) Word level TF-IDF: Calculates score for every single term in different documents.

ii) N-gram level TF-IDF: Calculates score for the combination of N terms together in different documents.

4.3 Algorithms used for classification

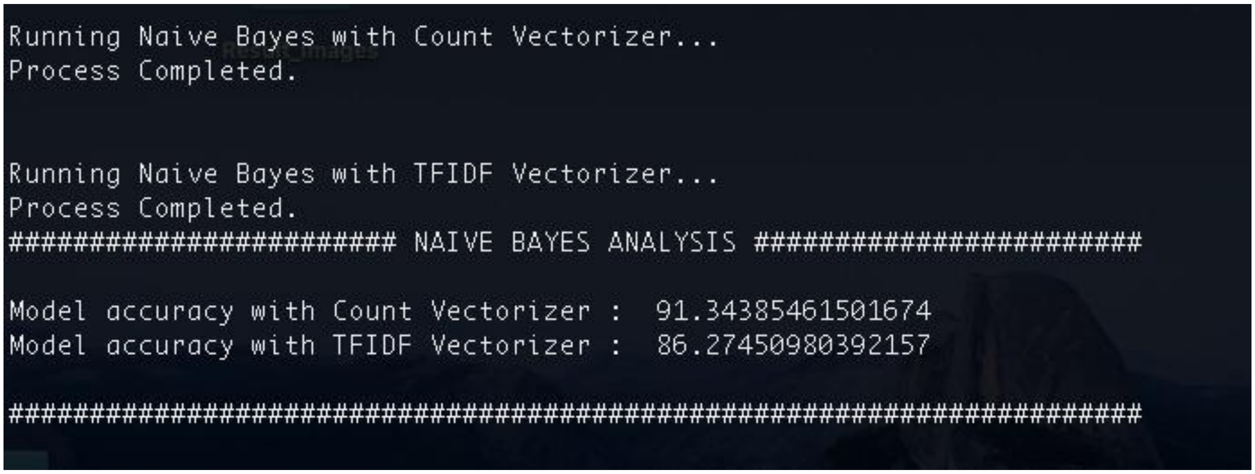
Now we proceed further by training the chosen classifiers. Training of classifiers is done using three specific algorithms Multinomial Naïve Bayes Passive Aggressive Classifier and Linear Support Vector machines.

Naïve Bayes: This classification technique accounts for the presence of a feature in a class to be alone in it's operation thereby providing a way to calculate posterior probability.

$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

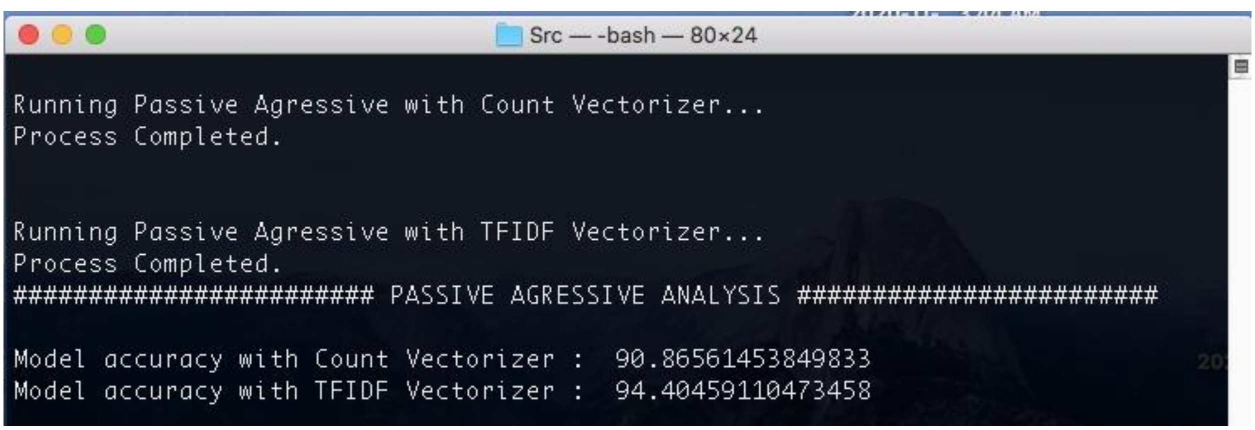
P(c|x)= posterior probability of class given predictor
P(c)= prior probability of class
P(x|c)= likelihood (probability of predictor given class)
P(x) = prior probability of predictor

Figure 4.3- Naïve Bayes



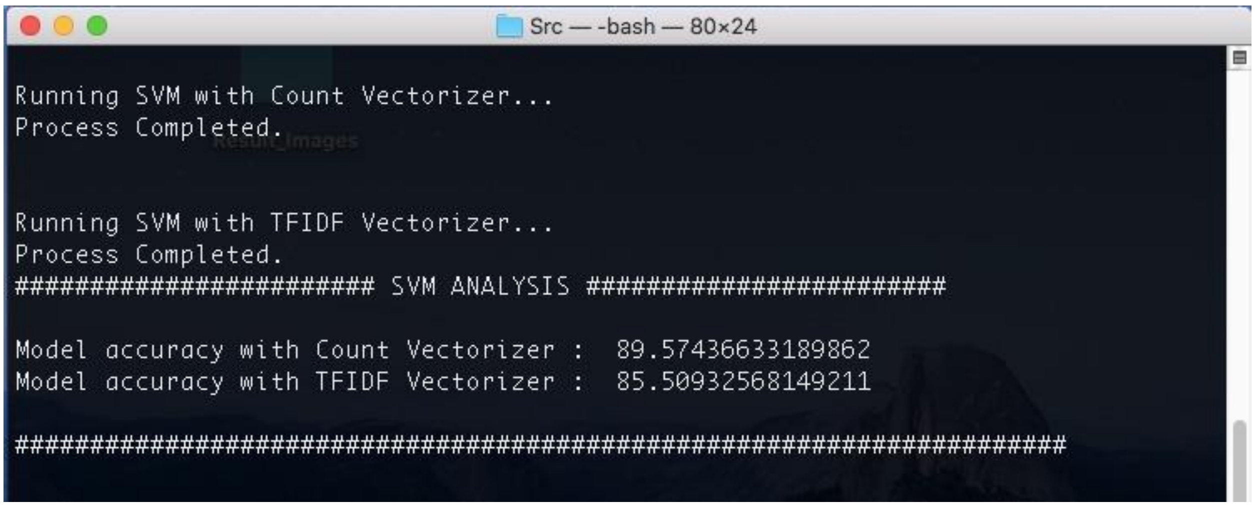
Passive Aggressive Classifier: The Passive Aggressive Algorithm is an online algorithm; perfect to differentiate enormous streams of statistics. It is easier to implement and very fast. It works by taking an example, educating from it and then putting it away.

Figure 4.4- Passive Aggressive



Support Vector Machine: In this algorithm, each fact item is plotted as a point in n-dimensional space. Values of each attribute are the value of all coordinate. It exclusively extracts an unsurpassed promising hyper-plane or a set of hyper-planes in a high dimensional space that segregate two class. Linear kernel is also sometimes used for SVM in this craft.

Figure 4.5- SVM



4.4 Metrics used to access the Performance of Model

This section deals with the tools used to measure the performance of the model. Some of the metrics used in the project are described below.

Classification Accuracy: It is one of the mainstream familiar evaluations metric for organization problems. It can be defined as the number of accurate predication as against the number of whole predictions. However, this metric alone cannot give sufficient in sequence to come to a decision whether the model is a good quality one or not. It is apposite when in attendance are equal numbers of inspection in each class.

Confusion Matrix: It is also known as Error matrix, which are table illustrations that show the recital of the representation. It is special kind of Contingency table having two dimensions- "actual", labeled on x-axis and "predicted" on the y-axis. The cells which are present on table are the number of predictions done with the help of the algorithm.

Table 4.2- Confusion Matrix

Total Instances		Predicted	
		Yes	**No**
Actual	**Yes**	*True Positive*	*False Negative*
	No	*True Negative*	*True Negative*

True Positives: It is correctly predicted positive values.
True Negatives: It is correctly predicted negative values.
False Positives: It is incorrectly predicted negative values as positive values.
False Negatives: It is incorrectly predicted negative values as positive values.

Classification Report: Some platform provides an expediency description when operational on organization troubles which outputs precision, recall, F1 score and support for each class.

Precision: Precision is the ratio of appropriately predicted constructive instances to the entirety predicted constructive instances. High precision means low False Positive rate.

$$Precision = \frac{TP}{TP + FP}$$

Recall (Sensitivity): Recall is the ratio of correctly predicted positive instances to the all instances in actual class - Yes.

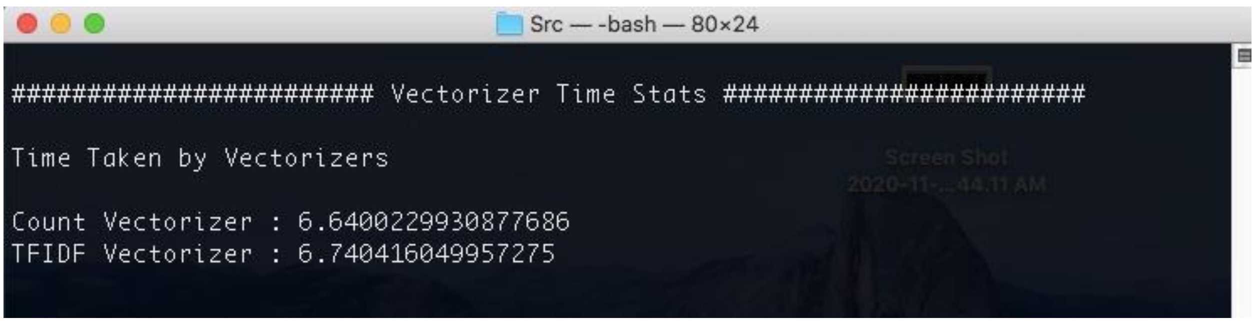
$$Recall = \frac{TP}{TP + FN}$$

Chapter 5

Result and Analysis

To analyze the outcome a few experiments were conducted Count Vectors and using Tf-Idf vectors at Word level and Ngram-level. Correctness was observed and is mentioned ahead for all models.

Figure 5.1-Vectorizer Time



This diagram shows the efficiency of vectorizer.

5.1 The Proposed Experiment

Count Vector and Tf-Idf vector were used to classify the responses at two levels:

- Word level – Only one word was preferred as indication for this experiment.
- N-gram level –The range of N-gram from 1 to 3 i.e. from one word to at most 3.

5.2 Classification Accuracy

After completion classification accuracy was noted. It was found that accuracy at word level performed better than that at N-gram level. The lowest accuracy was at 85.51% for Linear SVM with Tf-Idf at N-gram level while Multinomial and Passive Aggressive Classifier, using Tf-Idf vectors had accuracy above 90%. We know very well that classification accuracy can never be enough to conclude the efficiency of the model. So, other metrics were also performed using Tf-Idf Vectors.

Table 5.1-Classification Accuracy

	Count – Vectorizer	TF-IDF Vectorizer	Remarks
Naïve Bayes	91.34%	86.27%	Simple and Efficient algorithm
Linear SVC	89.57%	85.51%	Worked well for Binary classification
Passive Aggressive	90.86%	94.40%	Highly accurate and but computationally extensive

5.3 Classification Time:

Classifier Time

Table 5.2 Classifier Time

	Count Vectorizer	TF-IDF Vectorizer
Naïve Bayes	0.0727	0.0527
Linear SVC	0.5155	0.1264
Passive Aggressive	0.2913	0.1028

Vectorizer Time

Table 5.3 Vectorizer Time

	Tim (in sec)
Count Vectorizer	6.6400
TF-IDF Vectorizer	6.7404

Precision value for Linear SVM-TFIDF at 94% is higher than PAC-TFIDF, which is 87.3% and Recall values (Sensitivity) was calculated as 92% for both models.

5.4 Confusion Matrix:

Figure 5.2 Confusion Matrix for Naïve Bayes-TFIDF

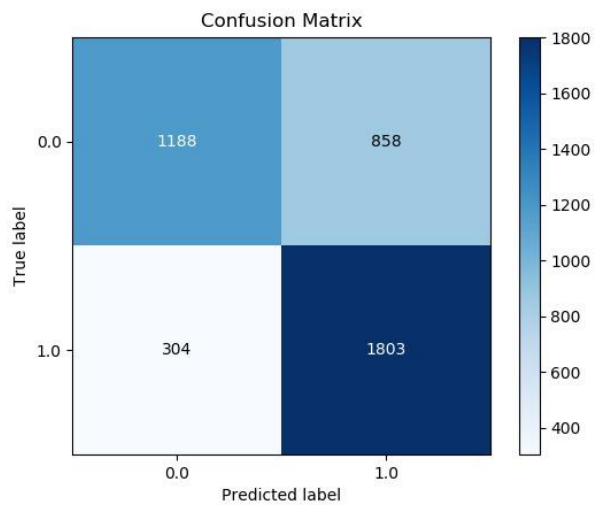


Figure 5.3 Confusion Matrix for Linear SVM-TFIDF

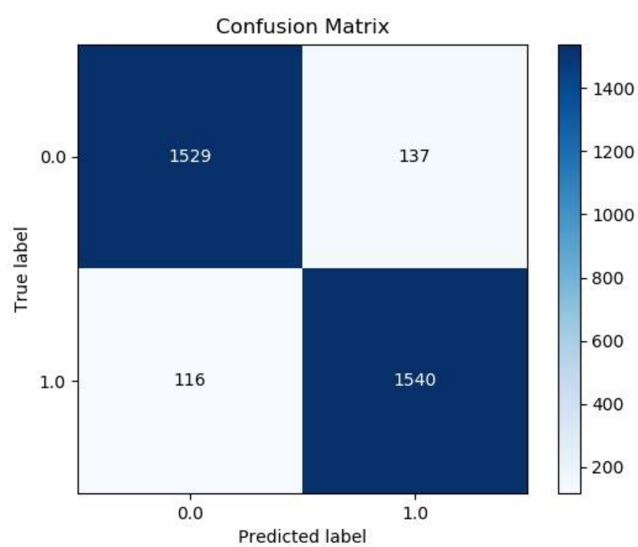
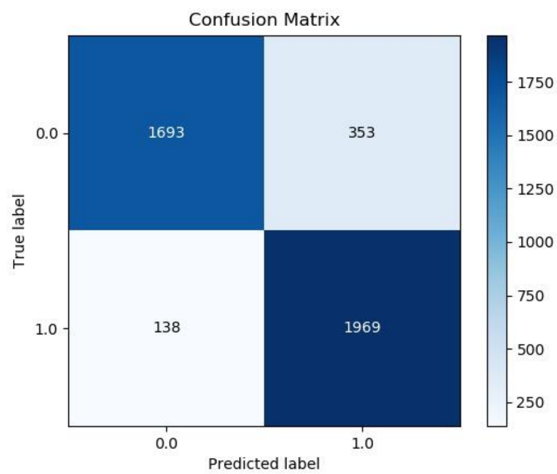


Figure 5.4 Confusion Matrix for Passive Agressive-TFIDF



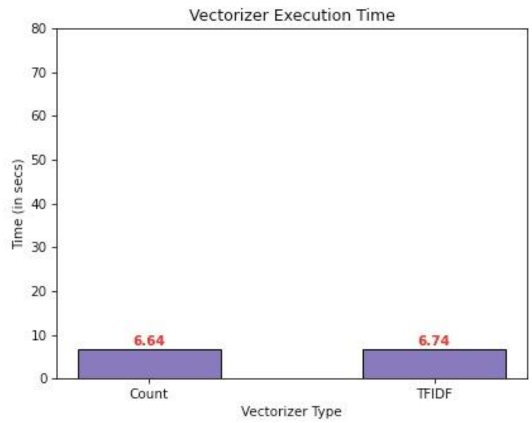
	Precision	Sensitivity	Specificity
Naïve Bayes	0.84	0.90	0.83
LinearSVM	0.89	0.88	0.90
Passive Agressive	0.86	0.92	0.86

5.5 Comparison

This section deals with comparison of count vector and TF-idf vectors on the three algorithm naïve bayes, passive aggressive and linear SVM.

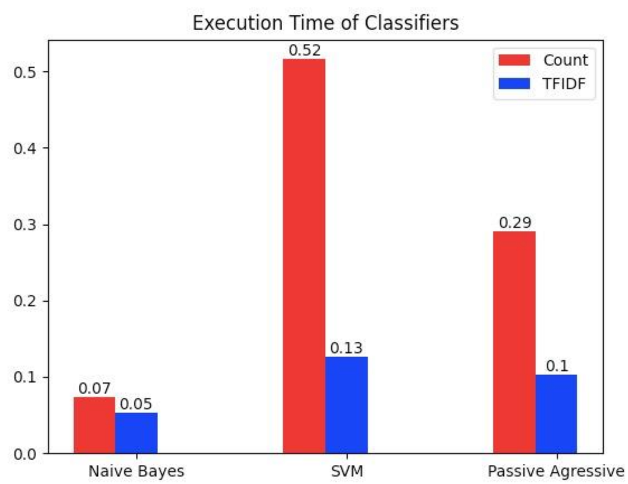
1. Vectorizer Execution Time

Figure 5.5-Vectorizer Time Execution



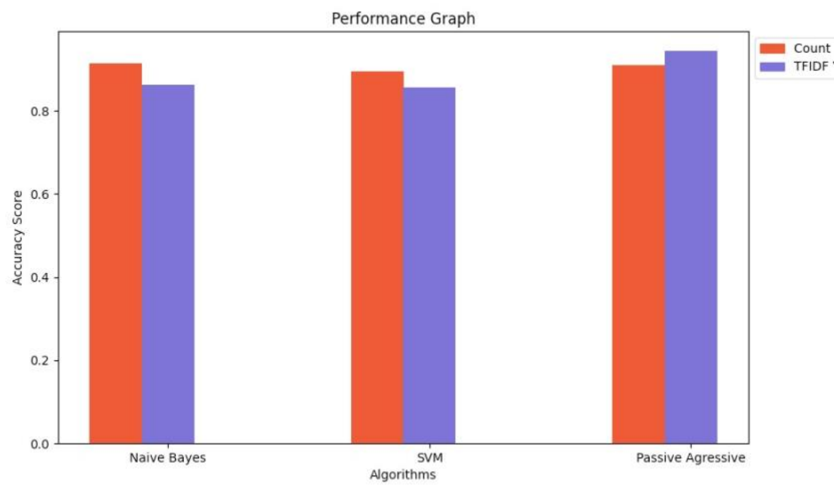
2. Execution Time of Classifiers

Figure 5.6- Execution Time of Classifier



3. Performance Graph

Figure 5.7- Performance Graph of Classifier



Conclusion

Some hypotheses can be made on why same models works very well on one dataset and does not work well on the other one. The first thing we can think of is that the original hypothesis on different styles of writing between fake and reliable news is only verified in one dataset, the Fake News Corpus, and it is the most logical one, as these texts are coming from online newspapers (or pretending to be), and thus capitalize on advertisements for making money. The second dataset, Liar-Liar Corpus is described by its authors as a collection a short sentence coming from various contexts such as political debate, interviews, TV ads and so on, thus it induces a lot of variety in writing style. For instance, it contains a transcription of vocal messages, which have in essence a different style from written one.

The data exploration chapter had already given an insight about this fact, as 2D data projection of the Liar-Liar Corpus shows no clear sign of separation, when Fake News Corpus shows one at the first look.

Future works

Studying fake news recognition only on supervised models on the data has revealed to be insufficient in most of the cases. To solve this setback, the majority of the research emphasizes on supplementary information such as about the author. Most of the flourishing loom would be regular fact checking model, which is much more convincing of a model with some kind of data base, the purpose of the model would then be to haul out information from the data and confirm the information from the database. The only issue with this method would be that the awareness base would need to be persistently and physically renew to keep on up to date.

References

[1] Eni Mustafaraj and Panagiotis Takis Metaxas. The fake news spreading plague: Was it preventable? arXiv preprint arXiv:1703.06988, 2017.

[2] https://en.wikipedia.org/wiki/Fake1_news

[3] https://en.wikipedia.org/wiki/Machine_learning

[4] https://en.wikipedia.org/wiki/Natural_language_processing

[5] FAKE NEWS IDENTIFICATION CS 229: MACHINE LEARNING : GROUP 621 Sohan Mone, Devyani Choudhary, Ayush Singhania

[6]Emilio Ferrara, Onur Varol, Clayton Davis, FilippoMenczer, and Alessandro Flammini. The rise of social bots. Communications of the ACM, 59(7):96{104, 2016.

[7] Carlos Merlo (2017), "Millonario negocio FAKE NEWS", Univision Noticias

[8] Chang, Juju; Lefferman, Jake; Pedersen, Claire; Martz, Geoff (November 29, 2016). "When Fake News Stories Make Real News Headlines". Nightline. ABC News.

[9] Read & Win: Machine Learning - 2019 - Wiley Analytical Science.
https://analyticalscience.wiley.com/do/10.1002/gitlab.19010

[10]/kkkkkkkkkkkkkkkkkkkkkk.txt - Natural language processing https://www.coursehero.com/file/70457569/kkkkkkkkkkkkkkkkkkkkkkktxt/