FAKE NEWS DETECTION
USING
NATURAL LANGUAGE PROCESSING

PROJECT REPORT

BY

Saurabh Sanyam Jaiswal (CSE/490/19063)

submitted to

Indian Institute of Information Technology, Kalyani

for 4th Year Project

Bachelor of Technology

In

Computer Science and Engineering

Nov, 2023

Certificate

This is to certify that project report entitled "**Fake News Detection Using Natural Language Processing**" **being submitted by Saurabh Sanyam Jaiswal (Reg No. 490)** undergraduate student in the Department of Computer Science and Engineering, Indian Institute of Information Technology Kalyani, West Bengal, 741235, India, for the award of Bachelor of Technology in Computer Science and Engineering, is an original research work carried by him under my supervision and guidance.

The project has fulfilled all the requirements as per the regulations of the Indian Institute of Information Technology Kalyani and in my opinion, has reached the standards needed for submission. The work, techniques and results presented have not been submitted to any other university or institute for the award of any other degree or diploma.

.....

(Dr. Pratik Chakraborty)

Assistant Professor

Department of Computer Science and Engineering

Indian Institute of Information Technology Kalyani

Kalyani, W.B.-741235, India.

Declaration

We hereby declare that the work being presented in this project entitled **Fake News Detection Using Natural Language Processing**, submitted to Indian Institute of Information Technology Kalyani in partial fulfillment for the award of the degree of Bachelor of Technology in Computer Science and Engineering during the period from August 2023 to October 2023 under the supervision of Dr. Pratik Chakraborty, Department of Computer Science and Engineering, Indian Institute of Information Technology Kalyani, West Bengal - 741235, India, does not contain any classified information.

Name of the Candidate: Saurabh Sanyam Jaiswal (Reg No. 490)

Name of the Department: Computer Science and Information Engineering

Institute Name: Indian Institute of Information Technology Kalyani

Date: 25/11/2023

Acknowledgment

First of all I would like to take this opportunity to thank my supervisor Dr. Pratik Chakraborty without whose efforts this project would not have been possible. I am grateful to him for guiding me towards the project wherever possible. I am most grateful to the Department of Computer Science and Engineering, IIIT Kalyani, India, for providing me with this wonderful opportunity to complete my 4th-year project.

IIIT Kalyani Saurabh Sanyam Jaiswal (Reg No. 490)

Date : 25/11/2023

Abstract

Fake news. Misinformation. Disinformation. Wrong information. These are several interchangeable terms that are used to describe articles, stories, or in general, information that appears to be legitimate news on the internet or other propagatable media. They are usually disseminated with a motivation to influence opinions or political views, or merely as a joke, causing indirect or direct negative consequences. This project report discusses the various approaches I have researched, evaluated, and employed by using natural language processing (NLP), and machine learning to address the fake news problem.

Content

Chapter 1

Introduction

This chapter gives a brief introduction about the most widely used field of study “Natural Language Processing(NLP)”. Natural Language Processing (NLP) is a branch of artificial intelligence (AI) that focuses on the interaction between computers and humans through natural language. It enables computers to understand, interpret, and generate human language in a way that is both meaningful and contextually relevant.

1.1) Natural Language Processing

Natural language processing (NLP) is a branch of artificial intelligence (AI) that enables computers to comprehend, generate, and manipulate human language. Natural language processing has the ability to interrogate the data with natural language text or voice. NLP is the core technology behind virtual assistants, such as the Siri, Cortana, or Alexa. When we ask questions of these virtual assistants, NLP is what enables them to not only understand the user’s request but to also respond in natural language.

NLP applies both to written text and speech and can be applied to all human languages. Other examples of tools powered by NLP include web search, email spam filtering, automatic translation of text or speech, document summarization, sentiment analysis, and grammar/spell checking.

1.2) Application of NLP

It has various applications that too in various fields. Some of them are listed below:

- Chatbots
- Autocomplete in Search Engines
- Void Assistants
- Grammer Checker
- Email Classification and Filtering
- Fake News Detection

There are many left to list as it is a very wide topic and here in this project, we have used one of the application i.e. Fake News Detection.

1.3) Concept

The concept of this project starts by firstly focusing on evaluating currently developed and available natural language processing pipelines (n-grams, count vectorization, Bag of Words, etc.) and classification models (Naïve Bayes, SVM, Random Forest) based on set evaluation metrics. With recall, precision, accuracy, and F1-scores, we would be able to analyze and develop baseline creations to initiate a benchmark of the classification models which can be used to train and test the data on and implement improved variations of data processing pipelines on the chosen dataset, efficient machine learning algorithms which are able to handle huge data pre-processing or NLP pipelines.

1.4) Aims

The aim of this Project is as follows:

1. Build an apt NLP pipeline for the chosen dataset

2. Understanding why currently available NLP and machine learning models and classifiers may or may not be as effective in distinguishing between fake and real news articles

3. Exploring the cost-to-benefit ratios of implementing and developing new and improved NLP pipelines and ML classifiers on the chosen dataset

Chapter 2

Design

Before examining the project design, let us first have a refresh on the aim and motivation behind this Final Year Project idea.

Aim:

To achieve an improved, highly accurate fake news detection model that can be deployed and delivered as a viable, robust solution for users by employing the use of NLP, machine learning research, and implementations.

2.1) Project Relevant Techniques

There are 3 main techniques that are instrumental to developing an accurate model to address the fake news detection problem I have posed in this project:

1. Feature Extraction
2. Classification Algorithm Analysis
3. Model Performance Evaluation

Feature Extraction:

There are several methods we can use to engineer features into vectors from the raw dataset; as inputs for model construction:

1. Count Vectorization
2. Word Embedding
3. Linguistical Analysis using NLP

4. TF-IDF Vectorization

5. Topic Modelling

The relevant tools for the above techniques can be generally found in the sklearn library based on Python.

Classification Algorithms:

In order to develop a software module that is able to perform fake news detection, a very important step is to execute the text classification phase. There is a huge pool of machine learning models that we can use to train classifiers with the use of the feature data we have extracted in the precedent phase.

Machine Learning Classifier Examples:

1. Support Vector Machine (SVM) Classifier

2. Naïve Bayes Classifier

3. Passive Aggressive Classifier

Model Performance Evaluation:

The standard library sklearn, can achieve the performance evaluation and the effectiveness of the implemented model, using the sklearn.metrics module. It enables us to calculate and plot confusion matrices, calculate accuracy, recall, precision, and F1 scores.

2.2) Project Methodology

Dataset Acquisition and Analysis:

The dataset I am using for this project has been ethically considered and appropriately sourced from Kaggle under the public domain (Kajal Yadav [Kaggle]. 2020).

1. Text corpus of news articles and metadata scraped across 600 webpages
2. Approximately 10K news article data and metadata
3. 6 feature attributes: News Headline, News URL, Source, Stated On, Date, Label
4. Multi-Class Labelling: False, Pants on Fire, True, Mostly True, Barely True, Half True

Once this step is accomplished, the project design is going to require the dataset to be split into 2 pipeline versions.

1. Non-web scraped, news headline Fake-Real News Dataset
2. Web scraped, news headline + news content Fake-Real News Dataset

Dataset Pre-Processing:

There are going to be 2 main pre-processing pipelines using NLP and other text data cleaning techniques to prepare the raw data to be input into classification models and workflows for this project.

1. SimpleNLP
2. Extensive Data Cleanse NLP (EDC-NLP)

The SimpleNLP pipeline is going to represent a rudimentary as-needed basis of preprocessing characteristics, with simple case conversions, lemmatization, and stopword removals.

As for the Extensive Data Cleanse NLP (EDC-NLP) pipeline, it is going to be designed according to the linguistic characteristics of the news headlines and news bodies found in the dataset I acquired in the previous step. This is such that the data cleaning process can efficiently tackle the data point necessities for text classification while preserving the context of the news data.

The analysis of the news headline and news content features of the dataset has posed some pre-processing requirements as follows:

1. Removal of →

    a. Links, Whitespaces, Newlines, Tabs, Accented Characters, Special Characters, Stopwords, HTML tags

    b. Conversion of resultant text to lowercase text

2. Reduction of →

    a. Repetitive Characters, Punctuations

3. Expansion of contracted words

4. Spelling Correction using Autocorrection Python module

Feature Analysis:

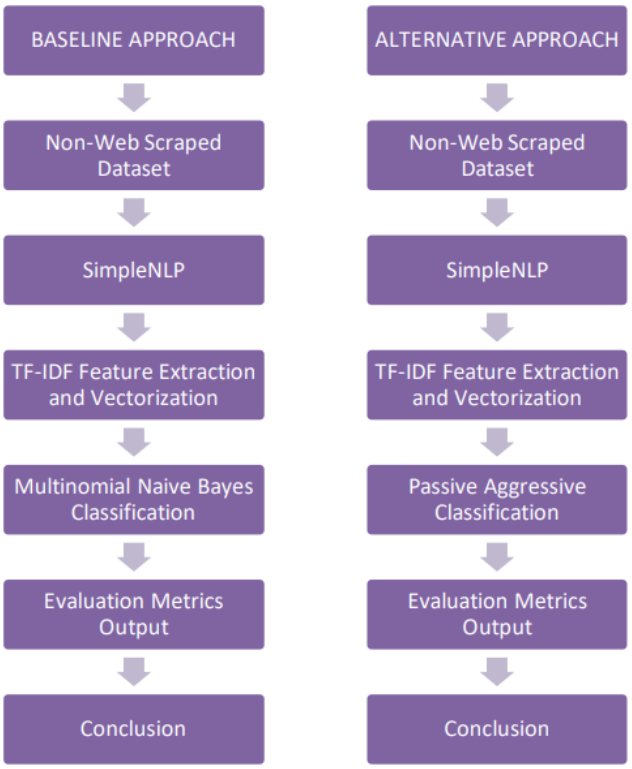
In this step, I will be employing the relevant feature extraction techniques I have explained in the Relevant Techniques section of this chapter, to select the relevant data types and features to be trained and classified as part of model implementation for dependency evaluation.

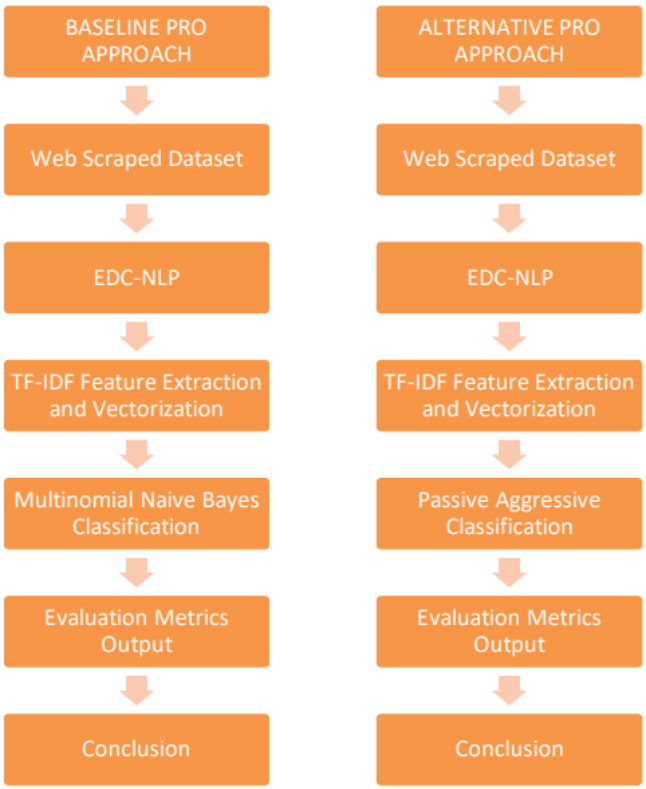
My chosen feature extraction techniques comprise:

1. TF-IDF Vectorization
2. Count Vectorization
3. Linguistical Analysis using NLP

2.3) Fake News Detection Workflow Design

There are going to be 4 main text classification workflows in this project, aimed at optimizing the effectiveness of using NLP and machine learning at detecting fake news or information.





The fundamental workflow for each of the proposed model approaches is as follows:

1. Importation of relevant libraries
2. Dataset Pre-Processing
3. Feature Extraction
4. Input Generation
5. Model Construction and Architecture Analysis
6. Model Prediction
7. Model Performance Evaluation

Chapter 3

Implementation

Before we dive into the project implementation I have developed as part of the Fake News Detection project, let us see the outline of how I have approached my build.

I also want to note here that I have broken down my Google Colab into Chapters for readability. I have implemented 2 NLP workflows and 4 models, to the fake news detection problem I am trying to address with this project.

3.1) Dataset Preparation

Dataset Acquisition:

I started off with the dataset acquisition process from Kaggle under the public domain (Kajal Yadav [Kaggle]. 2020).

Acquired Dataset (Raw):

	News_Headline	Link_Of_News	Source	Stated_On	Date	Label
0	Says Osama bin Laden endorsed Joe Biden	https://www.politifact.com/factchecks/2020/jun...	Donald Trump Jr.	June 18, 2020	June 19, 2020	FALSE
1	CNN aired a video of a toddler running away fr...	https://www.politifact.com/factchecks/2020/jun...	Donald Trump	June 18, 2020	June 19, 2020	pants-fire
2	Says Tim Tebow □kneeled in protest of abortion...	https://www.politifact.com/factchecks/2020/jun...	Facebook posts	June 12, 2020	June 19, 2020	FALSE
3	□Even so-called moderate Democrats like Joe Bi...	https://www.politifact.com/factchecks/2020/jun...	Paul Junge	June 10, 2020	June 19, 2020	barely-true
4	"Our health department, our city and our count...	https://www.politifact.com/factchecks/2020/jun...	Jeanette Kowalik	June 14, 2020	June 18, 2020	TRUE
5	Says before he planned a rally on June 19 □nob...	https://www.politifact.com/factchecks/2020/jun...	Donald Trump	June 17, 2020	June 18, 2020	pants-fire
6	California□s registered independent voters □wi...	https://www.politifact.com/factchecks/2020/jun...	Facebook posts	June 6, 2020	June 18, 2020	FALSE
7	□Antifa now banging on residents□ doors in Sea...	https://www.politifact.com/factchecks/2020/jun...	Facebook posts	June 14, 2020	June 18, 2020	FALSE
8	□President Obama and Vice President Biden neve...	https://www.politifact.com/factchecks/2020/jun...	Donald Trump	June 16, 2020	June 18, 2020	FALSE
9	Says these □elite□ figures are on house arrest...	https://www.politifact.com/factchecks/2020/jun...	Facebook posts	June 16, 2020	June 18, 2020	pants-fire
10	Says Nate McMurray said that □gun owners are p...	https://www.politifact.com/factchecks/2020/jun...	Chris Jacobs	June 13, 2020	June 18, 2020	mostly-true

Next, I ensured to understand the characteristics and information present in the dataset, before I commenced on the Exploratory Data Analysis (EDA) and pre-processing steps.

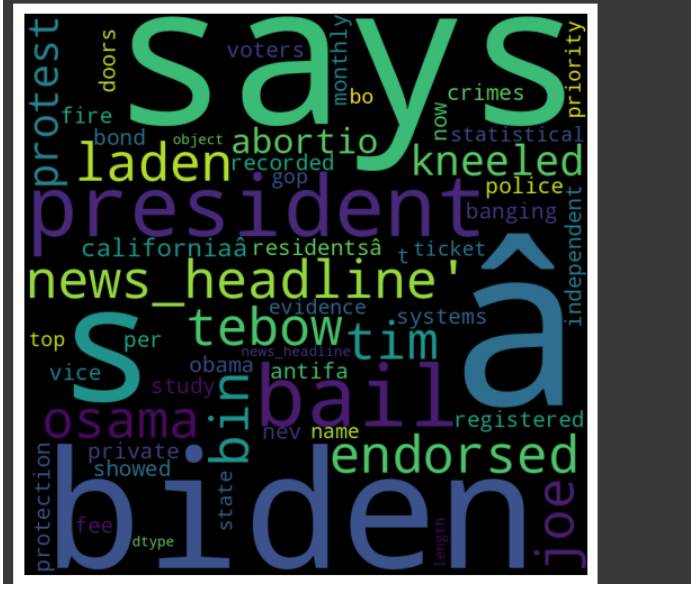
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9960 entries, 0 to 9959
Data columns (total 6 columns):
 #   Column         Non-Null Count  Dtype
---  ------         --------------  -----
 0   News_Headline  9960 non-null   object
 1   Link_Of_News   9960 non-null   object
 2   Source         9960 non-null   object
 3   Stated_On      9960 non-null   object
 4   Date           9960 non-null   object
 5   Label          9960 non-null   object
dtypes: object(6)
memory usage: 467.0+ KB
```

I then proceeded to analyze the characteristics of the Fake-Real news dataset by using Exploratory Data Analysis methods, which include:

Word Clouds:

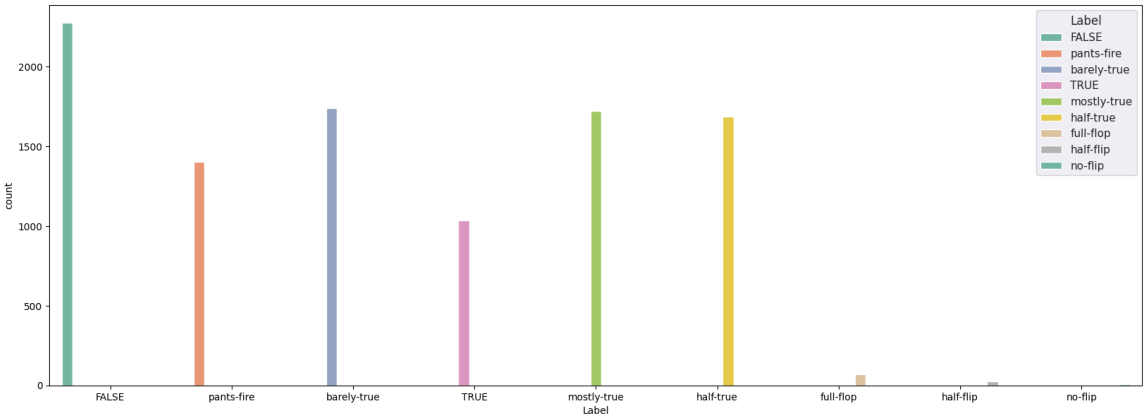
```
#False News
gen_wordcloud('FALSE')
gen_wordcloud('pants-fire')


#Real News
gen_wordcloud('TRUE')
gen_wordcloud('barely-true')
gen_wordcloud('mostly-true')
gen_wordcloud('half-true')
```

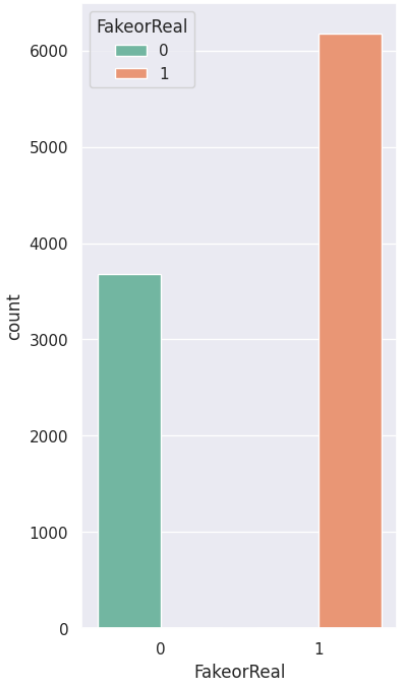


Label Distributions:

Multi-Class Labelled Format:



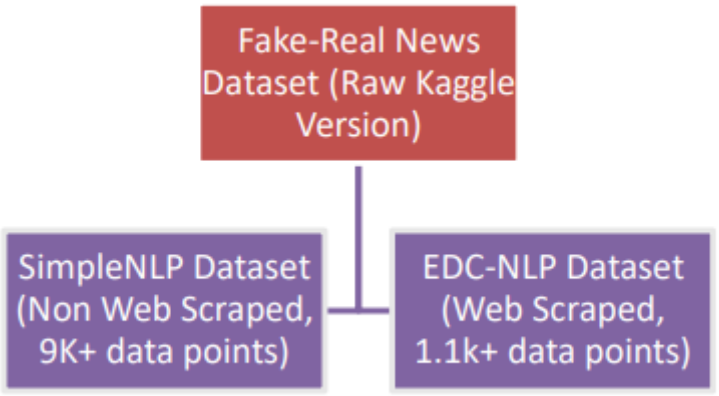
Binary Labelled Format (Post Conversion):



Since the dataset is of a multi-labelled classification nature, with 6 main labels to describe the legitimacy of the news articles, I have decided to convert the dataset into a binary classified dataset with 0 to represent fake news and 1 to represent real news. This also simplifies the machine learning classification process, as we already have 10K+ data points to train and test through.

3.2) Dataset Split, Web Scraping – Splitting the dataset for variational usage

The next step in the implementation, would be splitting and distinguishing the raw dataset into 2 separate datasets as can be understood from the below figure -



Simple NLP Dataset:

	News_Headline	FakeorReal
0	Says Osama bin Laden endorsed Joe Biden	0
1	Says Tim Tebow □kneeled in protest of abortion...	0
2	California□s registered independent voters □wi...	0
3	□Antifa now banging on residents□ doors in Sea...	0
4	□President Obama and Vice President Biden neve...	0
...	...	...
9850	"Not one tax has been raised since I've been g...	1
9851	"Seventy-four percent of Rhode Islanders suppo...	1
9852	Says Wendy Davis, "born into difficult circums...	1
9853	Recent record-low water levels in Lake Michiga...	1
9854	A private school tax break in the Wisconsin st...	1

9855 rows × 2 columns

As we can see from the above dataframe, this dataset consists of just the news headline and binary information of the news headline, with 9855 data points.

EDC-NLP Dataset:

	News_Headline	Link_Of_News	FakeorReal
0	Says Osama bin Laden endorsed Joe Biden	https://www.politifact.com/factchecks/2020/jun...	0
1	Says Tim Tebow □kneeled in protest of abortion...	https://www.politifact.com/factchecks/2020/jun...	0
2	California□s registered independent voters □wi...	https://www.politifact.com/factchecks/2020/jun...	0
3	□Antifa now banging on residents□ doors in Sea...	https://www.politifact.com/factchecks/2020/jun...	0
4	□President Obama and Vice President Biden neve...	https://www.politifact.com/factchecks/2020/jun...	0
...	...	...	...
8528	"Nearly 7 million Floridians have pre-existing...	https://www.politifact.com/factchecks/2018/sep...	1
8529	Says Mike DeWine took $40,000 from ECOT and "d...	https://www.politifact.com/factchecks/2018/sep...	1
8530	"When two judges said it was illegal to fire a...	https://www.politifact.com/factchecks/2018/sep...	1
8531	Says Andrew Gillum "wants to abolish ICE and d...	https://www.politifact.com/factchecks/2018/sep...	1
8532	"Bruce Ohr's wife, Nellie Ohr, worked for Fusi...	https://www.politifact.com/factchecks/2018/sep...	1

1163 rows × 3 columns

This dataset contains 1163 datapoints, as to reduce the program execution time to complete the project withing the acceptable duration.

The screenshot below shows the dataframe after the web-scraping process has been executed.

	Final_News_Content	FakeorReal
0	Says Osama bin Laden endorsed Joe Biden\n\n\n\...	0
1	Says Tim Tebow □kneeled in protest of abortion...	0
2	California□s registered independent voters □wi...	0
3	□Antifa now banging on residents□ doors in Sea...	0
4	□President Obama and Vice President Biden neve...	0
...	...	...
8528	"Nearly 7 million Floridians have pre-existing...	1
8529	Says Mike DeWine took $40,000 from ECOT and "d...	1
8530	"When two judges said it was illegal to fire a...	1
8531	Says Andrew Gillum "wants to abolish ICE and d...	1
8532	"Bruce Ohr's wife, Nellie Ohr, worked for Fusi...	1

1163 rows × 2 columns

As can be observed, I have taken the scraped HTML text content (the news article body) of each news article, and concatenated the columns to make a 'Final_News_Content' column. This column contains information on the news headline + news body content, hence providing us with richer textual data for classification uses.

3.3) SimpleNLP, Lemmatization, n-gram analysis

SimpleNLP:

Now for the pre-processing proper. I utilized simple regex expressions, case conversion functions, lemmatization, and stopword removal libraries to process the data.

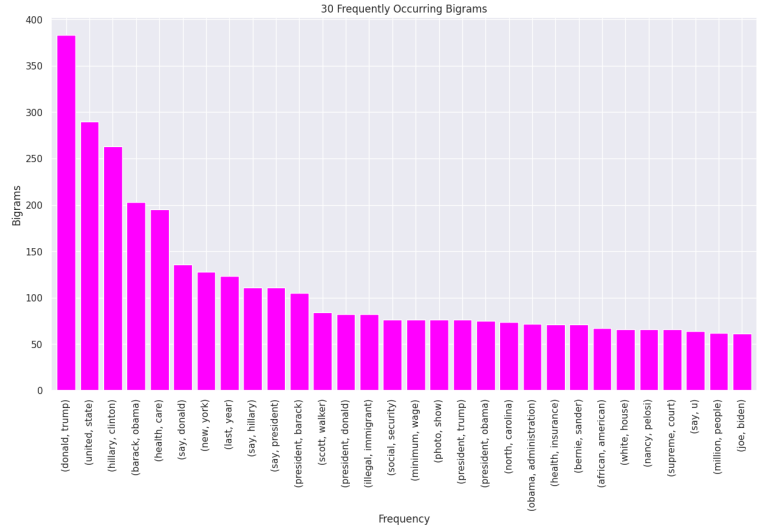
Lemmatization:

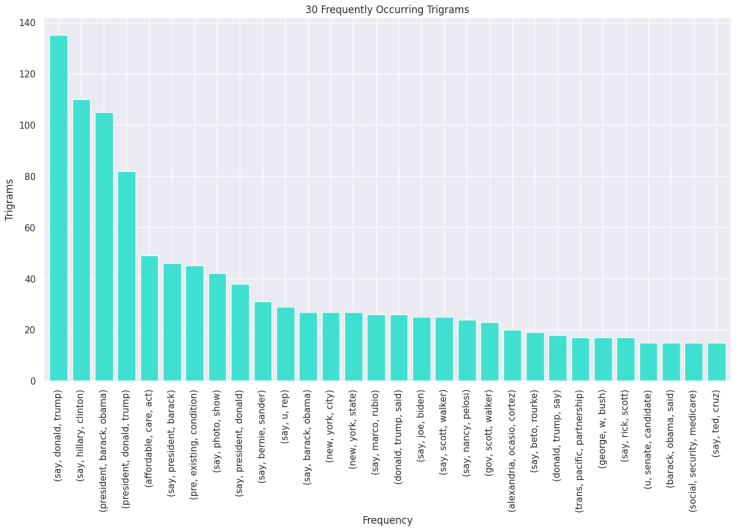
Next, I performed lemmatization to cleanse the news headline data present in the dataset. Lemmatization is usually referred to as the process of converting an evolved word into its root word with the consideration of the context of the word in a sentence or text corpus, and the characteristics of the text corpus.

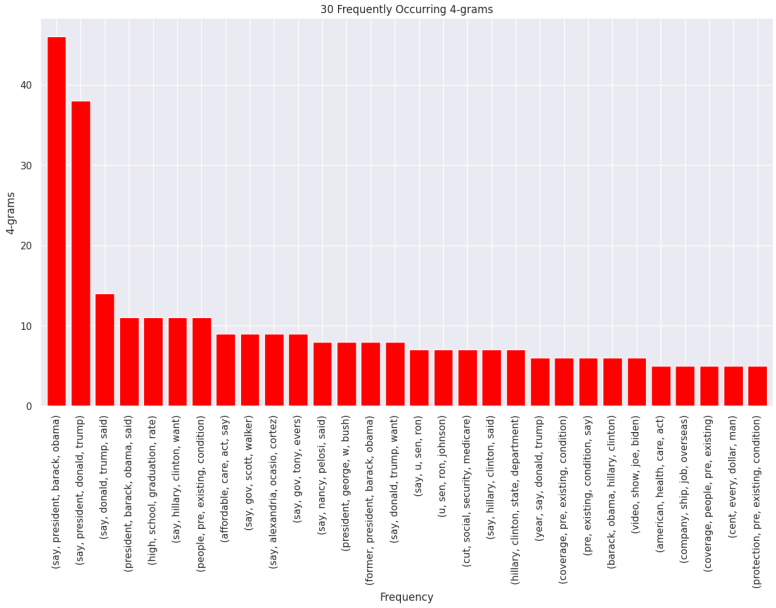
```
['say osama bin laden endorsed joe biden',
 'say tim tebow kneeled protest abortion national anthem praised fan model american',
 'california registered independent voter able vote republican come',
 'antifa banging resident door seattle demanding food supply get house get vandalized',
 'president obama vice president biden never even tried fix police reform eight year period',
 'prison boxer jack johnson invented patented first wrench white people insulted calling monkey wrench',
 'nazi germany hermann g ring worked defund eliminate police department would interfere brown shirt',
 'point defunding police minneapolis minnesota obama settled million islamics want sharia law',
 'according website black life matter inc charity full fledged corporation location',
 'say photo beaten woman aracely henriquez pregnant woman george floyd assaulted armed robbery',
 'real sense oklahoma flattened curve number case oklahoma declined precipitously',
 'free horse thoroughbred horse need home go sugarcreek sat slaughter gentleman died due covid son want noth
 'say joe biden called antifa courageous american',
 'owner taco bell said animal need shot black ppl',
 'donald trump recruiting excited enthusiastic minority actor actress appear campaign rally tulsa okla',
 'corona virus claim black belt chuck norris dead',
```

n-gram Analysis:

I also conducted an n-gram analysis of the resultant lemmatized text corpus to gain a better understanding of the common bigrams, trigrams, and 4-grams that are present in the news headline that is being textually classified.


30 Frequently Occurring Bigrams
Bigrams
Frequency
(donald, trump)
(united, state)
(hillary, clinton)
(barack, obama)
(health, care)
(say, donald)
(new, york)
(last, year)
(say, hillary)
(say, president)
(president, barack)
(scott, walker)
(president, donald)
(illegal, immigrant)
(social, security)
(minimum, wage)
(photo, show)
(president, trump)
(president, obama)
(north, carolina)
(obama, administration)
(health, insurance)
(bernie, sander)
(african, american)
(white, house)
(nancy, pelosi)
(supreme, court)
(say, u)
(million, people)
(joe, biden)


30 Frequently Occurring Trigrams
Trigrams
Frequency
(say, donald, trump)
(say, hillary, clinton)
(president, barack, obama)
(president, donald, trump)
(affordable, care, act)
(say, president, barack)
(pre, existing, condition)
(say, photo, show)
(say, president, donald)
(say, bernie, sander)
(say, u, rep)
(say, barack, obama)
(new, york, city)
(new, york, state)
(say, marco, rubio)
(donald, trump, said)
(say, joe, biden)
(say, scott, walker)
(say, nancy, pelosi)
(gov, scott, walker)
(alexandria, ocasio, cortez)
(say, beto, rourke)
(donald, trump, say)
(trans, pacific, partnership)
(george, w, bush)
(say, rick, scott)
(u, senate, candidate)
(barack, obama, said)
(social, security, medicare)
(say, ted, cruz)


30 Frequently Occurring 4-grams
4-grams
Frequency
(say, president, barack, obama)
(say, president, donald, trump)
(say, donald, trump, said)
(president, barack, obama, said)
(high, school, graduation, rate)
(say, hillary, clinton, want)
(people, pre, existing, condition)
(affordable, care, act, say)
(say, gov, scott, walker)
(say, alexandria, ocasio, cortez)
(say, gov, tony, evers)
(say, nancy, pelosi, said)
(president, george, w, bush)
(former, president, barack, obama)
(say, donald, trump, want)
(say, u, sen, ron)
(u, sen, ron, johnson)
(cut, social, security, medicare)
(say, hillary, clinton, said)
(hillary, clinton, state, department)
(year, say, donald, trump)
(coverage, pre, existing, condition)
(pre, existing, condition, say)
(barack, obama, hillary, clinton)
(video, show, joe, biden)
(american, health, care, act)
(company, ship, job, overseas)
(coverage, people, pre, existing)
(cent, every, dollar, man)
(protection, pre, existing, condition)

3.4) SimpleNLP Workflows + Classification Approaches

Baseline Performance Approach: SimpleNLP + Tf-Idf + Multinomial Naïve Bayes Classification Model

I have implemented the Term Frequency – Inverse Document Frequency (TF-IDF) measure approach to retrieve the textual information present in the news headlines found in the dataset. This approach aids in providing a quantified representation of linguistical information found in the text corpus. Using the TfidfVectorizer module, I was able to convert the textual data into a vector representation to retrieve the information presented in the dataset as a classifiable entity.

```
#Implement the TfidfVectorizer module
tfidf_vectorize = TfidfVectorizer()

#Fit the X data of all the headlines into an array format
X_idf = tfidf_vectorize.fit_transform(res_corpus).toarray()
#Assign the values of the 'FakeorReal' column in the headlines dataset to the y data
y_idf = final_fyp_newsdata['FakeorReal']
```

The standardized train-test data split was done to prepare the x, y training data and x, y testing data.

```
#Split up the data into training sets and testing sets to be used for text classification
X_train_idf, X_test_idf, y_train_idf, y_test_idf = train_test_split(X_idf, y_idf, test_size = 0.33, random_state = 0)
```

The Multinomial Naïve Bayes (NB) classifier was used in this text classification pipeline to evaluate the baseline performance of the model I implemented.

```
#Implementing the multinomial NB classifier
NB_classify_tfidf = MultinomialNB()

#Fitting the X_train and y_train data into the classifier for training
NB_classify_tfidf.fit(X_train_idf, y_train_idf)
```

```
▾ MultinomialNB
MultinomialNB()
```

Alternative Performance Approach: SimpleNLP + Tf-Idf + Passive Aggressive Classification Model

An alternative approach I chose to experiment with was using the TF-IDF vectorized data to input into the Passive Aggressive Classifier found in the sklearn library.

```
##Implementing the passive aggressive classifier
PaggroClassify = PassiveAggressiveClassifier(max_iter = 300)
#Fitting the X_train and y_train data into the passive aggressive classifier for training
PaggroClassify.fit(X_train_paggro, y_train_paggro)

#Predict the news labels for the test data using the passive aggressive classifier
y_pred_paggro = PaggroClassify.predict(X_test_paggro)
```

3.5) Extensive Data Cleanse NLP (EDC-NLP)

At this stage, we can now commence on a more sophisticated and comprehensive data cleaning pipeline which will be able to format the textual information found in the web scraped dataset for fake news detection uses.

I have implemented several functions to effectively clean the text corpus of common linguistic hindrances that may cause errors and pose as inaccurate contextual indicators on the news headlines being classified. These functions will be collectively implemented into 2 driver functions representing an improved NLP pipeline which I have dubbed the EDC-NLP pipeline as part of this project.

The EDC-NLP pipeline handles the following text cleaning, and data pre-processing operations:

1. Removing newlines and tabs
2. Removing HTML tags
3. Removing links
4. Removing accented characters

5. Removing extra whitespaces

6. Removing special characters

7. Converting the text to lowercase

This is the EDC-NLP, web-scraped dataset before the EDC-NLP pre-processing is executed.

	Final_News_Content	FakeorReal
0	Says Osama bin Laden endorsed Joe Biden\n\n\n\...	0
1	Says Tim Tebow □kneeled in protest of abortion...	0
2	California□s registered independent voters □wi...	0
3	□Antifa now banging on residents□ doors in Sea...	0
4	□President Obama and Vice President Biden neve...	0
...	...	...
8528	"Nearly 7 million Floridians have pre-existing...	1
8529	Says Mike DeWine took $40,000 from ECOT and "d...	1
8530	"When two judges said it was illegal to fire a...	1
8531	Says Andrew Gillum "wants to abolish ICE and d...	1
8532	"Bruce Ohr's wife, Nellie Ohr, worked for Fusi...	1

1163 rows × 2 columns

This is the resultant dataset, which will be used as an input dataframe for the PRO text classification workflows as part of this Fake News Detection project implementation.

	Processed_Headline	Final_News_Content	FakeorReal
0	say obama bin lade endorse joe biden politifac...	Says Osama bin Laden endorsed Joe Biden\n\n\n\...	0
1	say tim need protest abortion national anthem ...	Says Tim Tebow □kneeled in protest of abortion...	0
2	california register independent voters wil abl...	California□s registered independent voters □wi...	0
3	anti bang residents settle , demand & supply	□Antifa now banging on residents□ doors in Sea...	0
4	president obama vice president biden never eve...	□President Obama and Vice President Biden neve...	0
...	...	...	...
8528	nearly 7 million floridians pre exist conditio...	"Nearly 7 million Floridians have pre-existing...	1
8529	say mike define tok $ 40,000 eco nothing onlin...	Says Mike DeWine took $40,000 from ECOT and "d...	1
8530	two judge say legal fire teacher view pornogra...	"When two judges said it was illegal to fire a...	1
8531	say andrew film want abolish ice believe type ...	Says Andrew Gillum "wants to abolish ICE and d...	1
8532	bruce 's wife , nelle , work fusion gps firm h...	"Bruce Ohr's wife, Nellie Ohr, worked for Fusi...	1

1163 rows × 3 columns

3.6) PRO Approaches (EDC-NLP)

Baseline PRO Approach:

The overall implementation for the Baseline PRO approach is similar to the original baseline approach with only 1 difference.

It was trained on the EDC-NLP Web Scraped dataset.

Alternative PRO Approach:

The overall implementation for the Alternative PRO approach is similar to the original alternative approach with only 1 difference.

It was trained on the EDC-NLP Web Scraped dataset.

Chapter 4

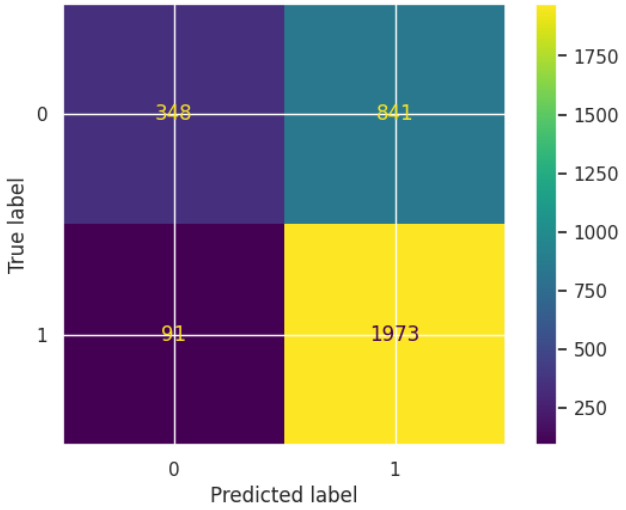
Evaluation

This section basically deals with the graphical representation of the data we got from the model implementation. Using this graphical representation, one can do the analysis of the efficiency and accuracy very well.

4.1) Model Performance Evaluation

Baseline Model Evaluation (SimpleNLP + TF-IDF + Multinomial Naïve Bayes Classifier)

The evaluation results, which have been represented as a confusion matrix are as follows:

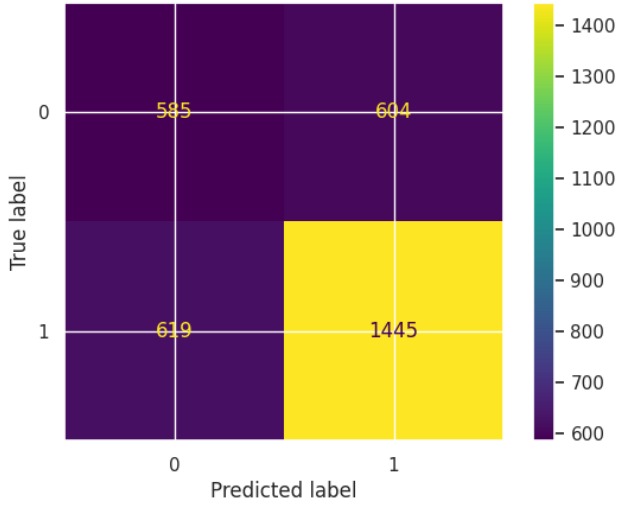


The below table shows the relevant metric scores which have been calculated in the evaluation stage of this workflow, to better assess the performance of the baseline model.

Evaluation Metric	ACCURACY	PRECISION	RECALL
Score	71.35%	70.11%	95.59%

Alternative Model Evaluation (SimpleNLP + TF-IDF + Passive Aggressive Classifier)

The evaluation results, which have been represented as a confusion matrix are as follows:

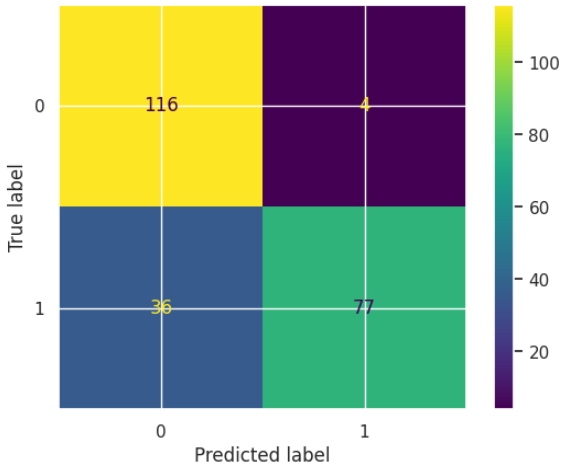


The below table shows the relevant metric scores which have been calculated in the evaluation stage of this workflow, to better assess the performance of the alternative model.

Evaluation Metric	ACCURACY	PRECISION	RECALL
Score	62.4%	70.52%	70.01%

Baseline PRO Model Evaluation (EDC-NLP + TF-IDF + Multinomial Naïve Bayes Classifier)

The evaluation results, which have been represented as a confusion matrix are as follows:

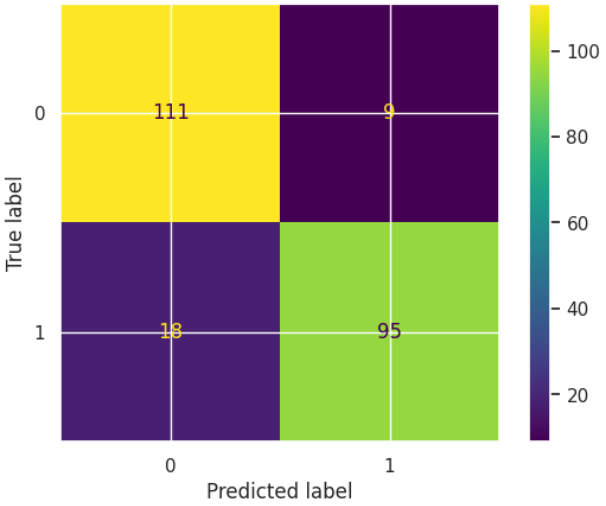


The below table shows the relevant metric scores which have been calculated in the evaluation stage of this workflow, to better assess the performance of the baseline PRO model.

Evaluation Metric	ACCURACY	PRECISION	RECALL
Score	82.83%	95.06%	68.14%

Alternative PRO Model Evaluation (EDC-NLP + TF-IDF + Passive Aggressive Classifier)

The evaluation results, which have been represented as a confusion matrix are as follows:



The below table shows the relevant metric scores which have been calculated in the evaluation stage of this workflow, to better assess the performance of the alternative PRO model..

Evaluation Metric	ACCURACY	PRECISION	RECALL
Score	88.41%	91.35%	84.07%

4.2) Project Results Overview

The table below summarizes the evaluation metrics scores of all the Fake News Detection approaches taken in this project.

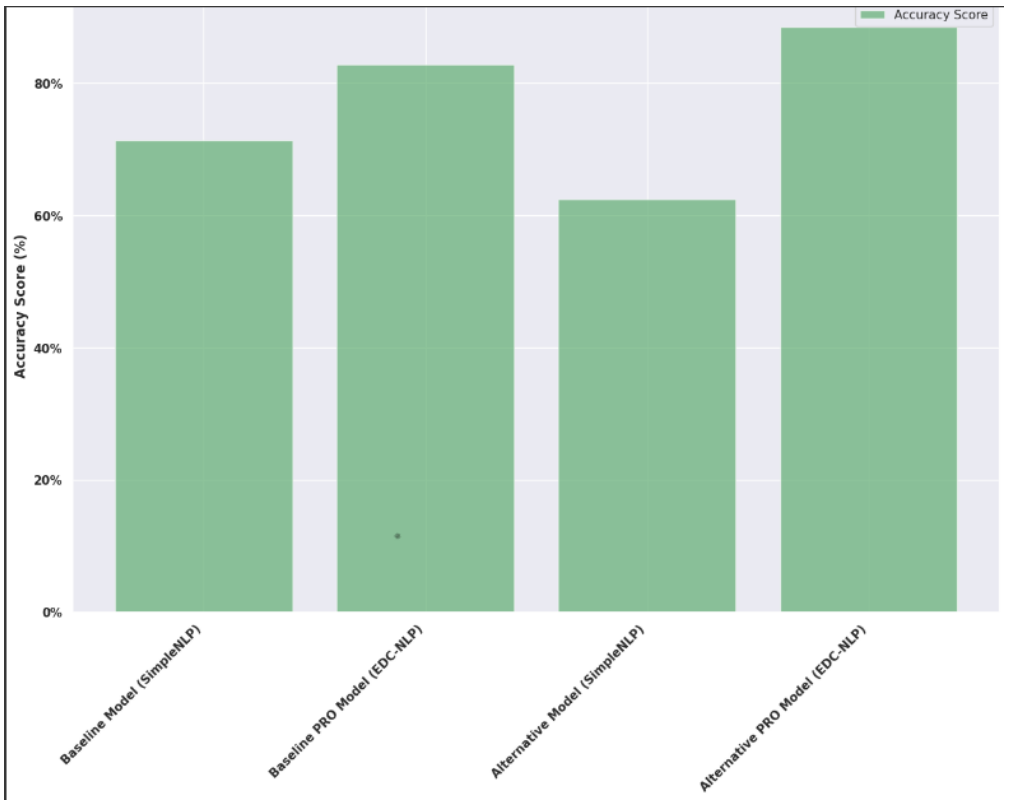
Fake News Detection Approach	Pre Processing / Dataset Used	Accuracy Score (%)	Precision Score (%)	Recall Score (%)
Baseline (tf-idf + multNB classifier)	SimpleNLP	71.35%	70.11%	95.59%
Alternative (tf-idf + passive aggressive classifier	SimpleNLP	62.4%	70.52%	70.01%
Baseline PRO	EDC-NLP	82.83%	95.06%	68.14%
Alternative PRO	EDC-NLP	88.41%	91.35%	84.07%

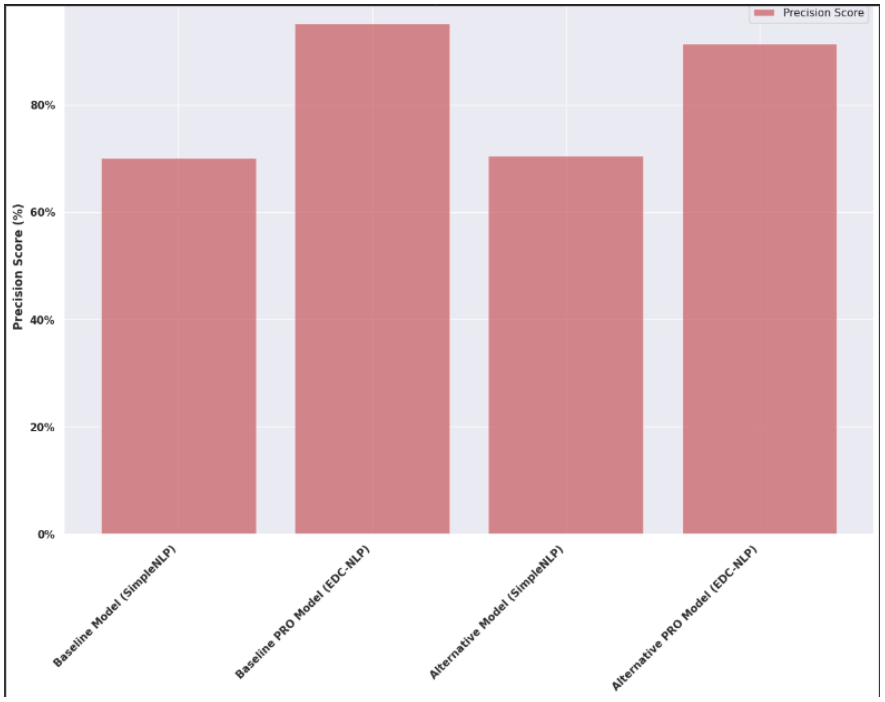
As we can observe, both the pro models show a significant improvement in their Accuracy and Precision Scores when used with EDC-NLP dataset. The difference between the highest and lowest accuracy scores stands at 26.01%. Hence it can be concluded that to observe a ~25% jump in accuracy rates, data cleansed by properly implemented NLP pipelines such as the EDC-NLP is better in Fake News Detection applications.

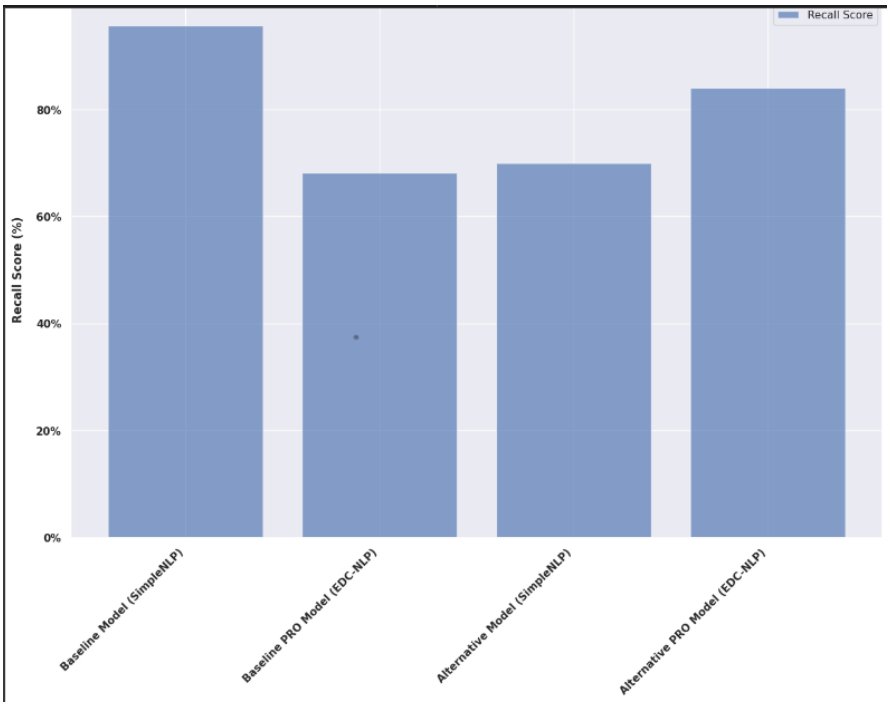
The precision scores of the models indicate the rate at which the model can identify real news as real amongst its overall prediction outputs. A general trend in the summary depicts that models that have used the EDC-NLP pipeline have higher precision scores.

The lowest precision score stands at 70.11% for the baseline model. The difference between the highest and lowest precision score is 24.95%. Hence, we can deduce that to observe a ~25% increase in precision scores, it is better to utilize a sophisticated NLP pipeline with thorough data cleansing operations such as the EDC-NLP pipeline to optimize both accuracy and precision scores in detecting fake news amongst large datasets or data sources.

I have also included bar charts comparing the evaluation metrics of each of the 7 approaches as shown below




Precision Score
Precision Score (%)
80%
60%
40%
20%
0%
Baseline Model (SimpleNLP)
Baseline PRO Model (EDC-NLP)
Alternative Model (SimpleNLP)
Alternative PRO Model (EDC-NLP)


Recall Score
Recall Score (%)
80%
60%
40%
20%
0%
Baseline Model (SimpleNLP)
Baseline PRO Model (EDC-NLP)
Alternative Model (SimpleNLP)
Alternative PRO Model (EDC-NLP)

Chapter 5

Conclusion and Future Scope

To conclude this project after the thorough analysis of the results obtained, it is to be acknowledged that it is very necessary to develop and implement a robust and reliable preprocessing system, such as EDC-NLP for the text data in your dataset. Well-formatted input data goes a long way in how the classification model learns and predicts the test data. It is also worth noting that not all NLP pipelines apply to all datasets, and it is good to have a pipeline that is tailored just right for the dataset you are using. This also helps in the model architecture design process of the classification models to be accurate and efficient in prediction workflows.

Now coming to the future scope of this project or application, from here on the evolution of this project can head in various directions. Let’s look into some noteworthy ideas:

- Investigate the integration of non-textual information, such as images and videos, to create a more comprehensive fake news detection system. Combining text and visual cues can enhance the model's overall accuracy.
- Implement mechanisms for continuous learning to keep the model updated with the evolving nature of misinformation. This could involve periodic retraining with new data and monitoring model performance over time.
- Using better models to improve the detection such as Deep Neural Network and Deep Learning Algorithms.

Bibliography

KAJAL YADAV. 2020 Fake-Real News. Kaggle Dataset Source. https://www.kaggle.com/datasets/c5c1decebd99f153eaf807468727d1df374afd6873d6f49 cc72eab1c9080e9d

SWIRE B, ECKER UK, LEWANDOWSKY S. 2017 The role of familiarity in correcting inaccurate information. Journal of Experimental Psychology. Learning, Memory, and Cognition 43(12): 1948. Crossref. PubMed.

RASHKI HANNAH AND CHOI EUNSOL ET AL. 2017 Truth of varying shades: Analyzing language in fake news and political fact-checking. In Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, pages 2931–2937, Copenhagen, Denmark. Association for Computational Linguistics.

SHU K, WANG S, LIU H. 2019 Beyond news contents: the role of social context for fake news detection. In: Proceedings of the Twelfth ACM International Conference on Web Search and Data Mining. ACM, pp 312–320

TORABI ASR, F., & TABOADA, M. 2019. Big Data and quality data for fake news and misinformation detection. Big Data & Society, 6(1). https://doi.org/10.1177/2053951719843310