جامعة أم القرى
UMM AL-QURA UNIVERSITY

# Data Analysis 2 Project

# Duration: 1st Semester

# Task 2 (Fake news)

***Submitted By:***


| | |
|---|---|
| ***Reem Ghazi Alosaimi*** | ***444001268*** |
| ***Remas Majed Almalki*** | ***444002871*** |

# Introduction:

Text analysis involves a collection of techniques and algorithms designed to derive meaningful information from text data. Rather than being a single approach, it encompasses a variety of methods that share a common goal: to transform unstructured text into structured data, enabling further analysis. To begin, let's consider a dataset of textual information.

As one of the most straightforward yet effective approaches, text analysis facilitates the rapid development of machine learning models that can process and interpret large volumes of text data efficiently. This process is highly valuable for various applications such as sentiment analysis, topic modeling, information extraction, and document classification.

In text analysis, the data typically involves high dimensionality, as each word or phrase can represent a unique feature. These methods are extensively used for tasks like spam filtering, sentiment detection, and categorizing reviews or articles. The advantage of text analysis lies in its ability to quickly process and analyze large datasets, making predictions and extracting insights with ease, even when the data contains thousands of features.

Text analysis models work by estimating the likelihood that a specific word or set of words is associated with a particular category or sentiment. They use probabilistic and statistical techniques to identify patterns, relationships, and trends within the text data. Although it is common to assume independence between words or phrases, in practice, this is rarely the case. However, by leveraging probabilistic algorithms and machine learning models, text analysis remains a robust and efficient solution for processing complex and high-dimensional text data.

# Data Features:

| | id | title | author | text | label |
|---|---|---|---|---|---|
| 0 | 0 | House Dem Aide: We Didn't Even See Comey's Let... | Darrell Lucus | House Dem Aide: We Didn't Even See Comey's Let... | 1 |
| 1 | 1 | FLYNN: Hillary Clinton, Big Woman on Campus - ... | Daniel J. Flynn | Ever get the feeling your life circles the rou... | 0 |
| 2 | 2 | Why the Truth Might Get You Fired | Consortiumnews.com | Why the Truth Might Get You Fired October 29, ... | 1 |
| 3 | 3 | 15 Civilians Killed In Single US Airstrike Hav... | Jessica Purkiss | Videos 15 Civilians Killed In Single US Airstr... | 1 |
| 4 | 4 | Iranian woman jailed for fictional unpublished... | Howard Portnoy | Print \nAn Iranian woman has been sentenced to... | 1 |
| 5 | 5 | Jackie Mason: Hollywood Would Love Trump if He... | Daniel Nussbaum | In these trying times, Jackie Mason is the Voi... | 0 |
| 6 | 6 | Life: Life Of Luxury: Elton John's 6 Favorite ... | NaN | Ever wonder how Britain's most iconic pop pian... | 1 |
| 7 | 7 | Benoît Hamon Wins French Socialist Party's Pre... | Alissa J. Rubin | PARIS — France chose an idealistic, traditi... | 0 |
| 8 | 8 | Excerpts From a Draft Script for Donald Trump'... | NaN | Donald J. Trump is scheduled to make a highly ... | 0 |
| 9 | 9 | A Back-Channel Plan for Ukraine and Russia, Co... | Megan Twohey and Scott Shane | A week before Michael T. Flynn resigned as nat... | 0 |

# ANALYSIS:

## Explore the Dataset:

We reviewed the size and dimensions of the data so that we knew the number of articles or news, the title, the content, etc. and the names of the columns present, and we concluded that the data contains 5000 news items and each news item contains four columns: title, text, author, date. Then we split the text in each row into a list of words using the str.split() function in the text column.

Then we counted the number of words in each row, i.e. in each article, using the str.len() function. The final result will be a string containing the number of words for each article in the text column. Then we calculated the descriptive statistics for the length of the texts (i.e. the number of words) in the text column, such as: count, mean, std, min, max. They were also applied to calculating the length of the titles.

The statistics for the training and testing sets are as follows:

The text attribute has a higher word count with an average of 760 words and 75% having more than 1000 words. The title attribute is a short statement with an average of 12 words, and 75% of them are around 15 words. Our experiment would be with both text and title together

# ANALYSIS:

The news is classified into different categories including (1: Unreliable, 0: Reliable) print(news_d.label.value_counts()) This function counts and displays the number of cases (number of rows) that contain each category (label) in the label column

We have imported NLTK, which is a famous platform for developing Python applications that interact with human language. Next, we import re for regex. We import stopwords from nltk.corpus. When working with words, particularly when considering semantics, we sometimes need to eliminate common words that do not add any significant meaning to a statement, such as "but", "can", "we", etc. PorterStemmer is used to perform stemming words with NLTK. Stemmers strip words of their morphological affixes, leaving the word stem solely. We import WordNetLemmatizer() from NLTK library for lemmatization. Lemmatization is much more effective than stemming. It goes beyond word reduction and evaluates a language's whole lexicon to apply morphological analysis to words, with the goal of just removing inflectional ends and returning the base or dictionary form of a word, known as the lemma. stopwords.words('english') allow us to look at the list of all the English stop words supported by NLTK. remove_unused_c() function is used to remove the unused columns. We impute null values with None using the null_process() function. Inside the function

# ANALYSIS:

clean_dataset(), we call remove_unused_c() and null_process() functions. This function is responsible for data cleaning. To clean text from unused characters, we have created the clean_text() function. For preprocessing, we will use only stop word removal. We created the nltk_preprocess() function for that purpose

()ps = PorterStemmer  is the process of converting words to their roots or base forms. WordNetLemmatizer is also used in the lemmatization process, which is a process similar to Stemming but more precise. It converts words to their base forms based on linguistic rules (such as converting "best" to "good").

Stop words are also used to remove unnecessary words during processing. A counter is used to create a dictionary of stop words and is used to count the frequency of these words or to quickly check for their presence

# Modification:

Some data cleaning functions are used such as

def remove_unused_c(df,column_n=remove_c) is used : to remove unimportant columns from the data frame and def null_process(feature_df) is used to process missing values. Text cleaning functions are used including:def clean_text(text) is used to clean the text from unnecessary elements such as links, punctuation marks and unwanted characters.

NLTK text processing functions include:

def nltk_preprocess(text) This function performs the following steps to process text using NLTK:

Clean text using clean_text(text)

Split text into a list of words using text).split()

Remove unwanted words: Words in stopwords_dict are removed

# Dataset after cleaning and preprocessing step:

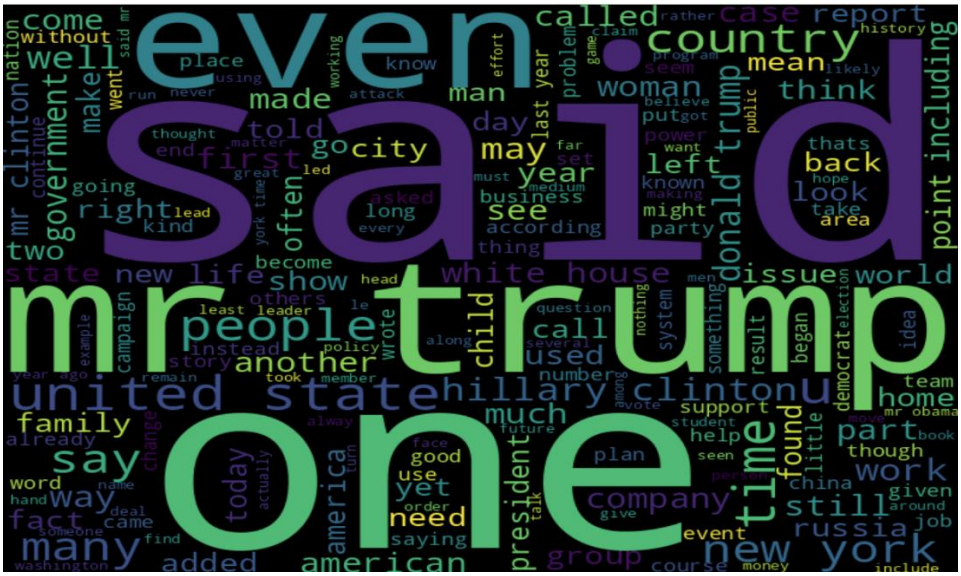
| | title | text | label |
|---|---|---|---|
| 0 | house dem aide didnt even see comeys letter ja... | house dem aide didnt even see comeys letter ja... | 1 |
| 1 | flynn hillary clinton big woman campus breitbart | ever get feeling life circle roundabout rather... | 0 |
| 2 | truth might get fired | truth might get fired october 29 2016 tension ... | 1 |
| 3 | 15 civilian killed single u airstrike identified | video 15 civilian killed single u airstrike id... | 1 |
| 4 | iranian woman jailed fictional unpublished sto... | print iranian woman sentenced six year prison ... | 1 |
| 5 | jackie mason hollywood would love trump bombed... | trying time jackie mason voice reason week exc... | 0 |
| 6 | life life luxury elton john 6 favorite shark p... | ever wonder britain iconic pop pianist get lon... | 1 |
| 7 | benoît hamon win french socialist party presid... | paris france chose idealistic traditional cand... | 0 |
| 8 | excerpt draft script donald trump qampa black ... | donald j trump scheduled make highly anticipat... | 0 |
| 9 | backchannel plan ukraine russia courtesy trump... | week michael flynn resigned national security ... | 0 |

# ANALYSIS:

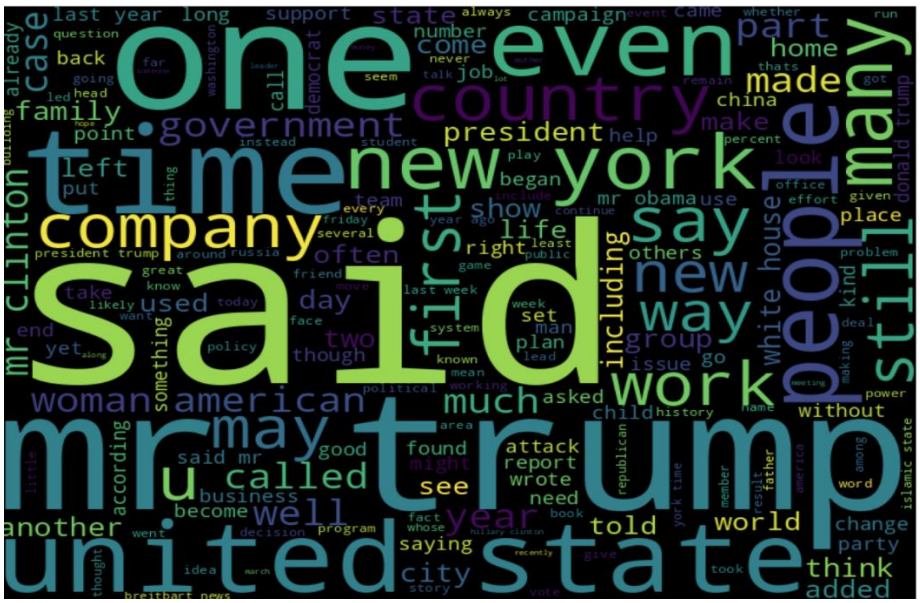
Here the WordCloud library is imported to create a word cloud. STOPWORDS is also imported, which is a list of common words that can be excluded from the word cloud (such as "the", "is", "and"), and matplotlib.pyplot is imported, which is a library used to create graphs and charts and will be used here to display the word cloud.
text_cloud = wordcloud.generate(' '.join(df['text']))
Here the word cloud is generated using the texts in the text column of the df dataset, all the texts in the text column are combined into a single text string using ' '.join(), where a space is added between each text to form a large string containing all the words, wordcloud.generate() analyzes the most frequent words in it to create a word cloud, where the most frequent words are displayed in a larger size,
plt.imshow(text_cloud) Here we display the word cloud using imshow() to display it as an image  true_n = ' '.join(df[df['label']==0]['text'])
Here the dataset df is filtered to select only trustworthy news, which have label = 0
df[df['label']==0]: This statement selects all rows in the dataframe df where label = 0 (i.e. trustworthy news)
['text']: After filtering out trustworthy news, only the text column of these rows is accessed.

join(...): All trustworthy texts are combined into one large text string, with a space between each text
Result: true_n contains only trustworthy news texts in the form of a long text string , and has been applied to fake news as well

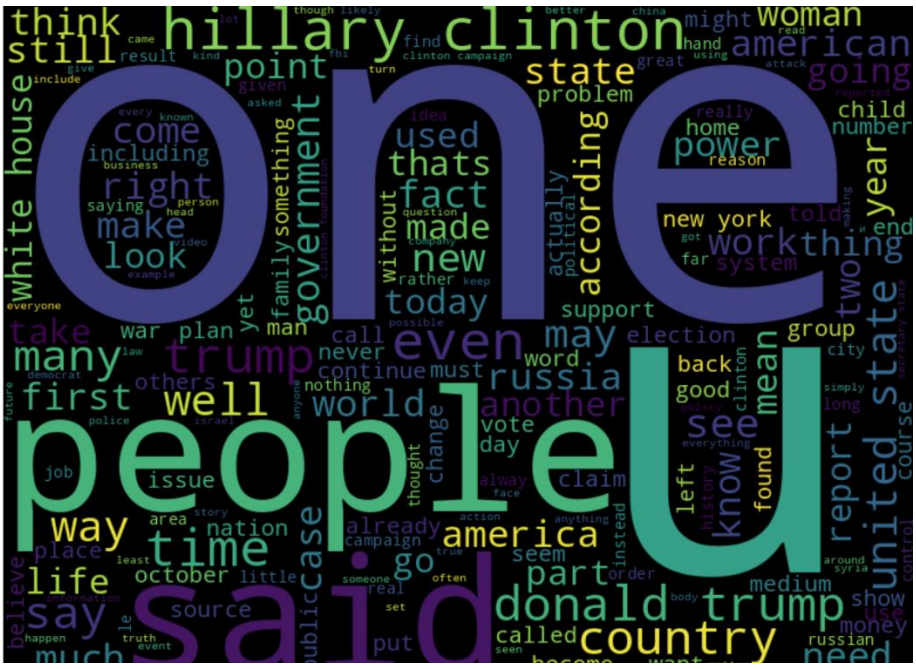
- Most frequently repeated words



- Reliable news




- Fake news



# ANALYSIS:

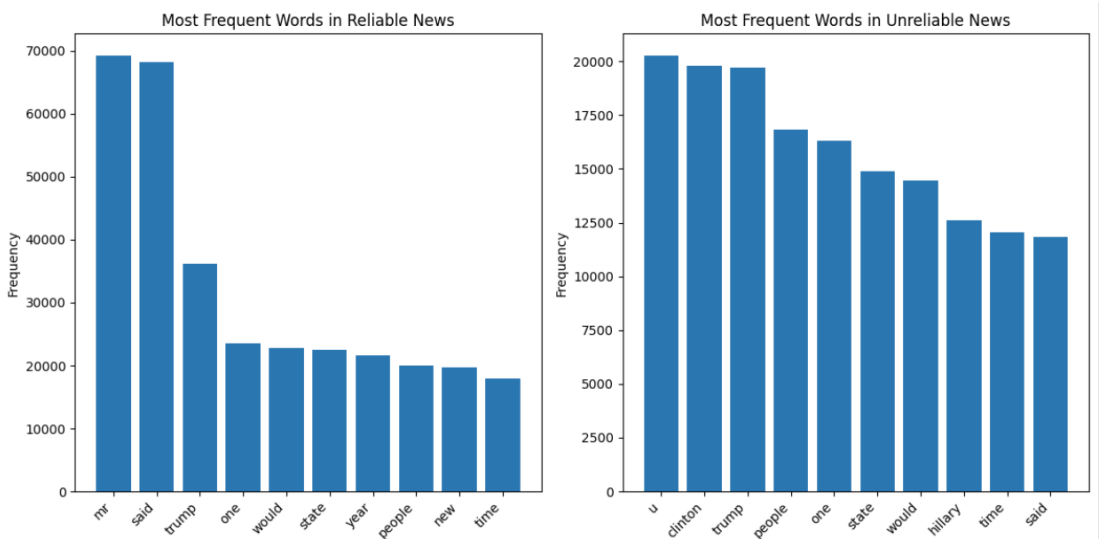
We have created a new dataframe named dt by copying data from the variable df which contains previously prepared or cleaned data.
dt['label'] = df['label']
This line resets or adds a label column in the new DataFrame dt If the label column already exists in dt, the values are updated, If it does not exist the column is added automatically.

We used lambda to remove Stopwords (common, unimportant words) from the text in the text column of the dt data frame, and then saved the resulting text (without Stopwords) in a new column called no_sw.

(From import counter groups) It is a powerful tool used to count the frequency of items in a given list or text, (get_most_frequent_words)
This function counts the most frequent words in a text string and makes a list of the most frequent words with the number of times they appear. Each text is split into words using split(), and then the words are added to the ()list all_words using extend.
The final result is that all_words contains all the words from the text.

dt[dt['label'] == 0]['no_sw']: This part filters only reliable news from the dt data frame, and then takes the text without stop words from the no_sw column.
Result: The most frequent words in the reliable news are stored in the variable most_frequent_reliable.

We then drew two vertical charts: one to show the most frequent words in reliable news, and the other to show the most frequent words in unreliable news. The charts are displayed in parallel so that the most frequent words in each type of news can be compared, which helps in understanding the different linguistic patterns between reliable and unreliable news.



# ANALYSIS:

We remove the most frequent words from the texts in the no_sw column of the dt data frame. The goal is to improve the quality of the texts by removing words that appear frequently and may not be useful for analysis, after previously removing the stopwords. A new column is created in the data frame called (wo_stopfreq) that contains the texts after removing the most frequent words.

We load the WordNet database and it uses a lemmatization tool to trace the words in the wo_stopfreq column back to their roots, then saves the results in a new column wo_stopfreq_lem.

| wo_stopfreq | wo_stopfreq_lem |
|---|---|
| house dem aide didnt even see comeys letter ja... | house dem aide didnt even see comeys letter ja... |
| ever get feeling life circle roundabout rather... | ever get feeling life circle roundabout rather... |
| truth might get fired october 29 2016 tension ... | truth might get fired october 29 2016 tension ... |
| video 15 civilian killed single airstrike iden... | video 15 civilian killed single airstrike iden... |

# ANALYSIS:

The text in wo_stopfreq_lem is split into distinct words (tokens) using word_tokenize.
These lyrics are saved in a new column called tokenized_text.
A RegexpTokenizer object is created with a regular expression (regex) that is used to extract words that contain only letters and numbers (a-z, A-Z, and 0-9).
The regular expression r'[a-zA-Z0-9]+' means to search for a sequence of only letters or numbers.
fit_transform: This function learns from texts to convert each text into a numerical representation using Bag of Words, where each word represents a column in an array, and each row represents a document or sentence. The value in the cell is the number of times the word appears in the text.
The result is a sparse array (an array containing many zero values) called text_counts, which represents the frequency of words in the texts.

# ANALYSIS:

We train a Complement Naive Bayes model to classify the data, and then evaluate the model using:
Accuracy to measure the percentage of correct predictions.
A Confusion Matrix to illustrate the errors the model makes for each class.
A Classification Report that contains precision, recall, and F1 score for each class, and was also applied to MultinomialNB() and BernoulliNB()

```
ComplementNB model accuracy is 89.81%
-------------------------------------------------
Confusion Matrix:
      0     1
0  2044    88
1   336  1692
-------------------------------------------------
Classification Report:
              precision    recall  f1-score   support

           0       0.86      0.96      0.91      2132
           1       0.95      0.83      0.89      2028

    accuracy                           0.90      4160
   macro avg       0.90      0.90      0.90      4160
weighted avg       0.90      0.90      0.90      4160
```

```
MultinominalNB model accuracy is 90.48%
-------------------------------------------------
Confusion Matrix:
      0     1
0  2044    88
1   308  1720
-------------------------------------------------
Classification Report:
              precision    recall  f1-score   support

           0       0.87      0.96      0.91      2132
           1       0.95      0.85      0.90      2028

    accuracy                           0.90      4160
   macro avg       0.91      0.90      0.90      4160
weighted avg       0.91      0.90      0.90      4160
```

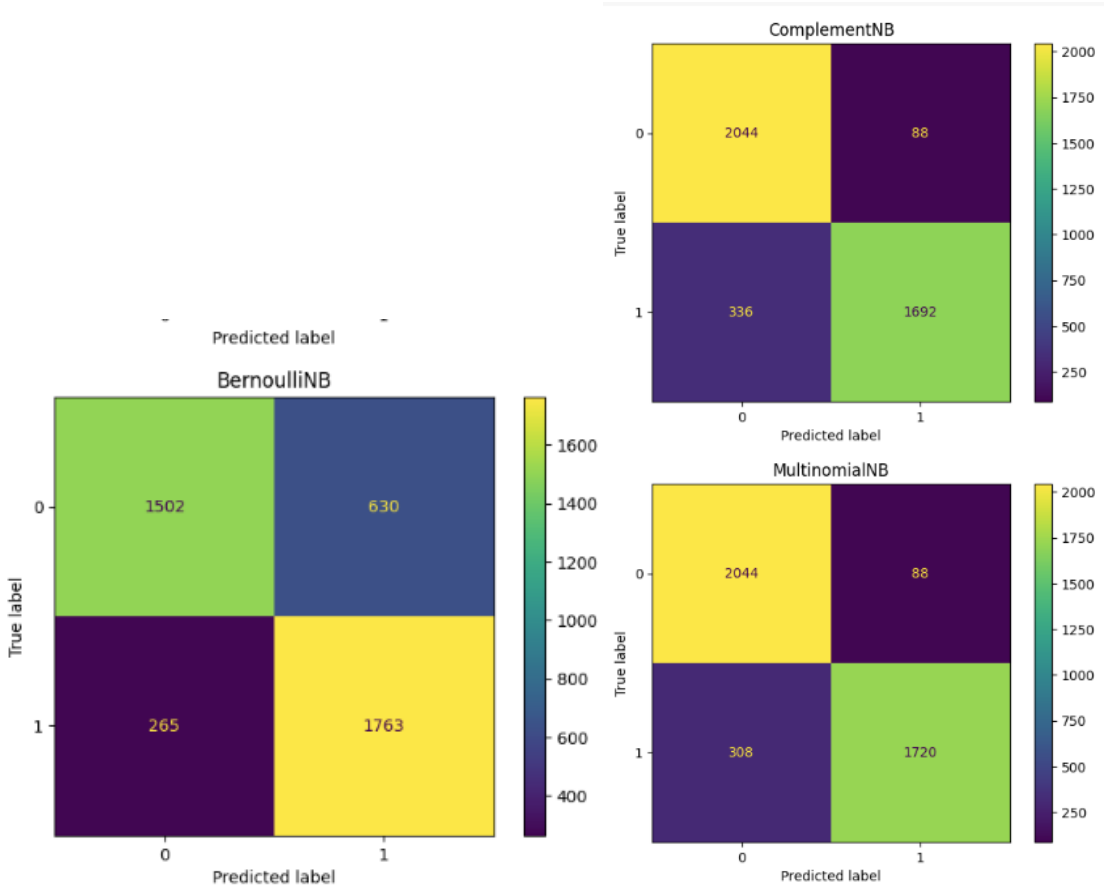
```
BernoulliNB model accuracy = 78.49%
-------------------------------------------------
Confusion Matrix:
      0     1
0  1502   630
1   265  1763
-------------------------------------------------
Classification Report:
              precision    recall  f1-score   support

           0       0.85      0.70      0.77      2132
           1       0.74      0.87      0.80      2028

    accuracy                           0.78      4160
   macro avg       0.79      0.79      0.78      4160
weighted avg       0.79      0.78      0.78      4160
```

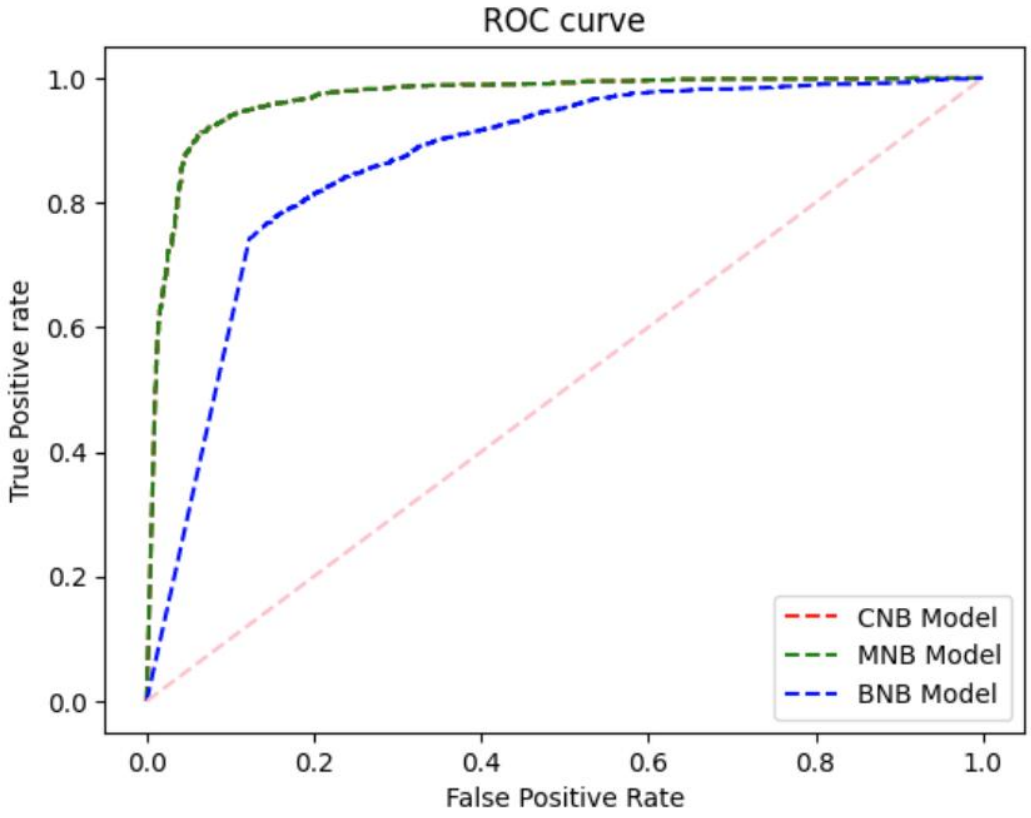
# ANALYSIS:

It displays the confusion matrix for several models including k = [CNB, MNB, BNB] and each confusion matrix is generated based on the expected results of the model versus the actual results using the test data X_test and y_test.



# ANALYSIS:

ROC curves are calculated and displayed for three Naive Bayes models (CNB, MNB, BNB) and compare these models with the results of random predictions. The ROC curve measures the performance of models across a range of thresholds, measuring the rate of true positives versus the rate of false positives.



## Conclusion:

The report focuses on text analysis using a set of techniques and algorithms to discover patterns and classify news into reliable and unreliable news:

.1Text data analysis and the use of appropriate tools:
The report provides an in-depth look at how to process text data, including segmenting texts, cleaning data from unnecessary words, and processing words using morphological analysis such as Stemming and Lemmatization.
The focus is on the use of well-known libraries such as NLTK, which provides an understanding of how to develop models capable of handling and classifying text data.

.2News filtering and classification:
News classification: The report relies on text classification techniques to determine whether news is reliable or not. Companies or organizations that rely on content review can use this technique to automatically filter out unreliable news.
Using machine learning: The report shows how to train models using algorithms such as Naive Bayes to analyze and classify news. These models can be used in practical applications such as detecting fake news or analyzing content.

.3Extracting the most frequent words:

Word analysis techniques such as WordCloud and frequency counting have been used to find out the most frequently used words in reliable and unreliable news. This information can be useful for analyzing the languages used in fake news versus real news.

.4Evaluating models and analyzing their performance:
Confusion matrix and classification reports are used to analyze the performance of models, which helps assess how accurate the models are in classifying news. Companies can improve their models based on these evaluations.

ROC curves provide a comparison between the performance of different models, which helps in choosing the most accurate model for use in practical applications.

.5Linguistic pattern analysis:
The report provides an analysis of linguistic patterns in reliable versus unreliable news. These results can be used to improve AI models that rely on understanding linguistic context and identifying unreliable content.

Overall benefit:
Improved content classification: With these techniques, content can be classified effectively and accurately, which helps in making better decisions about news and textual content.

Improving editorial strategies: The report provides a better understanding of the use of words in reliable and

unreliable news, helping journalists and editors improve content writing. Developing AI models: Intelligent models can be developed to classify news and detect fake news, helping to reduce the spread of misinformation.

جامعة أم القرى
UMM AL-QURA UNIVERSITY