FAKE NEWS DETECTION USING NLP

TEAM MEMBERS

611721104073-SABITHRIJA.S

611721104058-PAPETHA.D

PHASE2 SUBMISSION DOCUMENT

**PROJECT:**

**-FAKE NEWS DETECTION USING NLP**



**INTRODUCTION:**

Fake news detection using Natural Language Processing (NLP) in deep learning involves training a model to distinguish between real and fake news articles based on their textual content. Here's a simplified step-by-step process:

1. \*Data Collection\*: Gather a dataset of label news articles, where each article is label as either real or fake.

2. \*Data Preprocessing\*:

- Tokenization: Break down the text into individual words or tokens.

- Stop word Removal: Eliminate common words (e.g., "the", "is") that don't carry much information.

- Lemmatization/Stemming: Reduce words to their base or root form.

3. \*Feature Extraction\*:

- Convert the processed text into numerical form that can be fed into a neural network. This is usually done using techniques like TF-IDF (Term Frequency-Inverse Document Frequency) or word embeddings (e.g., Word2Vec, Glove).

4. \*Model Architecture\*:

- Design a deep learning architecture suitable for text classification. Common choices include:

- Recurrent Neural Networks (RNNs) or Long Short-Term Memory Networks (LSTMs) for sequential data.

- Convolutional Neural Networks (CNNs) with 1D convolutions for text.

5. \*Training\*:

- Split the dataset into training and validation sets.

- Feed the processed text data into the model and train it using appropriate loss functions (e.g., binary cross-entropy).

6. \*Validation and Testing\*:

- Evaluate the model on a validation set to tune hyperparameters.

- Finally, test the model on a separate test set to get an unbiased evaluation.

7. \*Model Evaluation\*:

- Metrics like accuracy, precision, recall, and F1-score are used to assess the performance of the model

Remember, the quality and diversity of your dataset, as well as the architecture and hyperparameters of your model, play crucial roles in the success of this task. Additionally, it's improved results

**CONTENT FOR PROJECT PHASE2**:

Consider exploring the advanced regression techniques like gradient boosting or XGBoost for

Fake news detection

***D ATA SOURCE:***

**A good** data source for fake news detection using NLP should be accurate, complete , covering the of

Accessible.

Dataset link:( https://www.kaggle.com/datasets/clmentbisaillon/fake -news-dataset)

|  |  |  |
| --- | --- | --- |
| Donald Trump Sends Out Embarrassing New Year’s Eve Message; This is Disturbing | Donald Trump just couldn t wish all Americans a Happy New Year and leave it at that. Instead, he had... | December 30, 2017 |
| Drunk Bragging Trump Staffer Started Russian Collusion Investigation | House Intelligence Committee Chairman Devin Nunes is going to have a bad day. He s been under the as.. | December 29, 2017 |
| Trump Is So Obsessed He Even Has Obama’s Name Coded Into His Website (IMAGES) | On Christmas day, Donald Trump announced that he would be back to work the following day, but he i... | December 25 2017 |
| Pope Francis Just Called Out Donald Trump During His Christmas Speech | Pope Francis used his annual Christmas Day message to rebuke Donald Trump without even mentioning hi... | December 24, 2017 |

***DATA COLLECTION AND PROCESSING:***

1. \*Data Collection\*: - \*Source Selection\*: Identify a diverse range of sources such as news websites, social media platforms, blogs, and forums. Ensure a mix of credible and unreliable sources.

- \*Web Scraping\*: Use web scraping techniques to collect text data from these sources. Tools like Beautiful Soup or Scrapy can be helpful.

- \*APIs\*: Utilize APIs provided by social media platforms (e.g., Twitter, Facebook) to access publicly available data.

- \*RSS Feeds\*: Subscribe to and collect data from RSS feeds of news websites.

2. \*Data processing\*:

- \*Text Extraction\*: Extract the main text content from web pages or social media posts while removing irrelevant elements (e.g., advertisements, sidebars)

- \*Language Detection\*: Identify the language of the text and discard content in languages you're not analyzing.

- \*Text Cleaning\*: Remove HTML tags, special characters, and unwanted symbols.

- \*Tokenization\*: Break text into individual words or tokens.

- \*Stopword Removal\*: Eliminate common words that don't carry much information (e.g., "and," "the"). - \*Lemmatization/Stemming\*: Reduce words to their base form.

**ADVANCED REGRESSION TECHNIQUES**:

Detecting fake news using advanced regression techniques in combination with Natural Language Processing (NLP) and Deep Learning is an interesting and challenging problem. Here's a high-level outline of how you might approach it:

1. \*Data Collection and Preprocessing\*:- Gather a diverse dataset containing both real and fake news articles. - Preprocess the text data: tokenization, stopword removal, lowercasing, etc.

2. \*Feature Extraction\*: - Use NLP techniques to extract relevant features from the text. This might include techniques like TF-IDF, word embeddings (e.g., Word2Vec, GloVe), or even more advanced methods like BERT embeddings.

3. \*Labeling and Training Data Preparation\*: - Label the dataset, designating each article as real or fake. - Split the dataset into training, validation, and test sets.

4. \*Regression Model\*: - Choose an advanced regression algorithm. Gradient Boosted Trees (e.g., XGBoost, LightGBM), Support Vector Machines (SVM), or even more complex models like Random Forest Regressors can be suitable candidates.

5. \*Model Training\*: - Train the regression model on the extracted features. The target variable will be a continuous value indicating the likelihood of an article being fake.

6. \*Evaluation\*: - Use appropriate evaluation metrics like Mean Absolute Error (MAE), Mean Squared Error (MSE), or custom metrics to assess the performance of your regression model.

7. \*Thresholding\*: - Define a threshold on the regression output to classify articles as real or fake. For example, if the output is above a certain value, classify it as fake; otherwise, classify it as real.

8. \*Deep Learning Component\*: - Optionally, you can enhance the model's performance by incorporating a Deep Learning component. This could involve using recurrent neural networks (RNNs), convolutional neural networks (CNNs), or transformers (e.g., BERT) for feature extraction.

9. \*Ensemble Techniques\*: - Consider using ensemble methods to combine the predictions from different models

**MODEL EVALUTION AND SECTION:**

--Evaluating fake news detection models typically involves using metrics like accuracy, precision, recall, F1-score, and possibly area under the receiver operating characteristic curve (AUC-ROC). These metrics help assess the performance of a model in distinguishing between real and fake news.

--Accuracy measures the overall correctness of predictions, while precision focuses on the proportion of true positive predictions out of all positive predictions. Recall measures the proportion of true positive predictions out of all actual positives. F1-score balances precision and recall. AUC-ROC provides an aggregate measure of a model's ability to distinguish between classes.

--It's important to use a diverse dataset for evaluation and consider the potential bias in the data. Additionally, cross-validation techniques like k-fold validation can help assess model robustness. Keep in mind that no model is perfect, and trade-offs between different evaluation metrics may need to be made.

**PROGRAM CODE:**

Certainly! Here's a simplified Python code outline using the TensorFlow and Keras libraries for fake news detection using NLP in a deep learning model. Please note that this is a basic example and you may need to adapt it to your specific dataset and requirements.

python

# Import necessary libraries

Import numpy as np

import pandas as pd

from sklearn.model\_selection import train\_test\_split

from tensorflow.keras.models import Sequential

from tensorflow.keras.layers import Dense, Embedding, LSTM, SpatialDropout1D

from tensorflow.keras.preprocessing.text import Tokenizer

from tensorflow.keras.preprocessing.sequence import pad\_sequences

# Load your dataset (ensure it has columns for 'text' and 'label')

# For example, you can use pandas to load a CSV file:

# df = pd.read\_csv('your\_dataset.csv')

# Data Preprocessing

max\_words = 5000 # Maximum number of words in your vocabulary

max\_len = 100 # Maximum length of each input sequence

tokenizer = Tokenizer(num\_words=max\_words)

tokenizer.fit\_on\_texts(df['text'])

X = tokenizer.texts\_to\_sequences(df['text'])

X = pad\_sequences(X, maxlen=max\_len)

# Split data into training and testing sets

X\_train, X\_test, y\_train, y\_test = train\_test\_split(X, df['label'], test\_size=0.2, random\_state=42)

# Define the model architecture

model = Sequential()

model.add(Embedding(max\_words, 128, input\_length=max\_len))

model.add(SpatialDropout1D(0.2))

model.add(LSTM(100))

model.add(Dense(1, activation='sigmoid'))

# Compile the model

model.compile(loss='binary\_crossentropy', optimizer='adam', metrics=['accuracy'])

# Train the model

model.fit(X\_train, y\_train, epochs=5, batch\_size=64, validation\_data=(X\_test, y\_test))

# Evaluate the model

loss, accuracy = model.evaluate(X\_test, y\_test)

print(f'Accuracy: {accuracy\*100:.2f}%')

# Now you can use this model to make predictions on new data

# For example:

# new\_data = ['This is a news article.', 'This is a fake news article.']

# new\_data\_sequences = tokenizer.texts\_to\_sequences(new\_data)

# new\_data\_padded = pad\_sequences(new\_data\_sequences, maxlen=max\_len)

# predictions = model.predict(new\_data\_padded)

# print(predictions)

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

import nltk

import re

import string

from sklearn.model\_selection import train\_test\_split

from sklearn.metrics import classification\_report

import keras

from keras.preprocessing import text,sequence

from keras.models import Sequential

from keras.layers import Dense,Embedding,LSTM,Dropout

import warnings

warnings.filterwarnings('ignore')

import os

for dirname, \_, filenames **in** os.walk('/kaggle/input'):

for filename **in** filenames:

print(os.path.join(dirname, filename))

/kaggle/input/fake-and-real-news-dataset/True.csv

/kaggle/input/fake-and-real-news-dataset/Fake.csv

# **Load and Check Data**

In [2]:

real\_data = pd.read\_csv('/kaggle/input/fake-and-real-news-dataset/True.csv')

fake\_data = pd.read\_csv('/kaggle/input/fake-and-real-news-dataset/Fake.csv')

In [3]:

real\_data.head()

Out[3]:

|  | title | text | subject | date |
| --- | --- | --- | --- | --- |
| 0 | As U.S. budget fight looms, Republicans flip t... | WASHINGTON (Reuters) - The head of a conservat... | politicsNews | December 31, 2017 |
| 1 | U.S. military to accept transgender recruits o... | WASHINGTON (Reuters) - Transgender people will... | politicsNews | December 29, 2017 |
| 2 | Senior U.S. Republican senator: 'Let Mr. Muell... | WASHINGTON (Reuters) - The special counsel inv... | politicsNews | December 31, 2017 |
| 3 | FBI Russia probe helped by Australian diplomat... | WASHINGTON (Reuters) - Trump campaign adviser ... | politicsNews | December 30, 2017 |
| 4 | Trump wants Postal Service to charge 'much mor... | SEATTLE/WASHINGTON (Reuters) - President Donal... | politicsNews | December 29, 2017 |

In [4]:

fake\_data.head()

Out[4]:

|  | title | text | subject | date |
| --- | --- | --- | --- | --- |
| 0 | Donald Trump Sends Out Embarrassing New Year’... | Donald Trump just couldn t wish all Americans ... | News | December 31, 2017 |
| 1 | Drunk Bragging Trump Staffer Started Russian ... | House Intelligence Committee Chairman Devin Nu... | News | December 31, 2017 |
| 2 | Sheriff David Clarke Becomes An Internet Joke... | On Friday, it was revealed that former Milwauk... | News | December 30, 2017 |
| 3 | Trump Is So Obsessed He Even Has Obama’s Name... | On Christmas day, Donald Trump announced that ... | News | December 29, 2017 |
| 4 | Pope Francis Just Called Out Donald Trump Dur... | Pope Francis used his annual Christmas Day mes... | News | December 25, 2017 |

In [5]:

*#add column*

real\_data['target'] = 1

fake\_data['target'] = 0

In [6]:

real\_data.tail()

Out[6]:

|  | title | text | subject | date | target |
| --- | --- | --- | --- | --- | --- |
| 21412 | 'Fully committed' NATO backs new U.S. approach... | BRUSSELS (Reuters) - NATO allies on Tuesday we... | worldnews | August 22, 2017 | 1 |
| 21413 | LexisNexis withdrew two products from Chinese ... | LONDON (Reuters) - LexisNexis, a provider of l... | worldnews | August 22, 2017 | 1 |
| 21414 | Minsk cultural hub becomes haven from authorities | MINSK (Reuters) - In the shadow of disused Sov... | worldnews | August 22, 2017 | 1 |
| 21415 | Vatican upbeat on possibility of Pope Francis ... | MOSCOW (Reuters) - Vatican Secretary of State ... | worldnews | August 22, 2017 | 1 |
| 21416 | Indonesia to buy $1.14 billion worth of Russia... | JAKARTA (Reuters) - Indonesia will buy 11 Sukh... | worldnews | August 22, 2017 | 1 |

In [7]:

*#Merging the 2 datasets*

data = pd.concat([real\_data, fake\_data], ignore\_index=True, sort=False)

data.head()

Out[7]:

|  | title | text | subject | date | target |
| --- | --- | --- | --- | --- | --- |
| 0 | As U.S. budget fight looms, Republicans flip t... | WASHINGTON (Reuters) - The head of a conservat... | politicsNews | December 31, 2017 | 1 |
| 1 | U.S. military to accept transgender recruits o... | WASHINGTON (Reuters) - Transgender people will... | politicsNews | December 29, 2017 | 1 |
| 2 | Senior U.S. Republican senator: 'Let Mr. Muell... | WASHINGTON (Reuters) - The special counsel inv... | politicsNews | December 31, 2017 | 1 |
| 3 | FBI Russia probe helped by Australian diplomat... | WASHINGTON (Reuters) - Trump campaign adviser ... | politicsNews | December 30, 2017 | 1 |
| 4 | Trump wants Postal Service to charge 'much mor... | SEATTLE/WASHINGTON (Reuters) - President Donal... | politicsNews | December 29, 2017 | 1 |

In [8]:

data.isnull().sum()

Out[8]:

title 0

text 0

subject 0

date 0

target 0

dtype: int64

# **Visualization**

**1.Count of Fake and Real Data**

In [9]:

print(data["target"].value\_counts())

fig, ax = plt.subplots(1,2, figsize=(19, 5))

g1 = sns.countplot(data.target,ax=ax[0],palette="pastel");

g1.set\_title("Count of real and fake data")

g1.set\_ylabel("Count")

g1.set\_xlabel("Target")

g2 = plt.pie(data["target"].value\_counts().values,explode=[0,0],labels=data.target.value\_counts().index, autopct='**%1.1f%%**',colors=['SkyBlue','PeachPuff'])

fig.show()

0 23481

1 21417

Name: target, dtype: int64

**2.Distribution of The Subject According to Real and Fake Data**

In [10]:

print(data.subject.value\_counts())

plt.figure(figsize=(10, 5))

ax = sns.countplot(x="subject", hue='target', data=data, palette="pastel")

plt.title("Distribution of The Subject According to Real and Fake Data")

politicsNews 11272

worldnews 10145

News 9050

politics 6841

left-news 4459

Government News 1570

US\_News 783

Middle-east 778

Name: subject, dtype: int64

Out[10]:

Text(0.5, 1.0, 'Distribution of The Subject According to Real and Fake Data')

# **Data Cleaning**

In [11]:

data['text']= data['subject'] + " " + data['title'] + " " + data['text']

del data['title']

del data['subject']

del data['date']

data.head()

Out[11]:

|  | text | target |
| --- | --- | --- |
| 0 | politicsNews As U.S. budget fight looms, Repub... | 1 |
| 1 | politicsNews U.S. military to accept transgend... | 1 |
| 2 | politicsNews Senior U.S. Republican senator: '... | 1 |
| 3 | politicsNews FBI Russia probe helped by Austra... | 1 |
| 4 | politicsNews Trump wants Postal Service to cha... | 1 |

In [12]:

first\_text = data.text[10]

first\_text

Out[12]:

'politicsNews Jones certified U.S. Senate winner despite Moore challenge (Reuters) - Alabama officials on Thursday certified Democrat Doug Jones the winner of the state’s U.S. Senate race, after a state judge denied a challenge by Republican Roy Moore, whose campaign was derailed by accusations of sexual misconduct with teenage girls. Jones won the vacant seat by about 22,000 votes, or 1.6 percentage points, election officials said. That made him the first Democrat in a quarter of a century to win a Senate seat in Alabama. The seat was previously held by Republican Jeff Sessions, who was tapped by U.S. President Donald Trump as attorney general. A state canvassing board composed of Alabama Secretary of State John Merrill, Governor Kay Ivey and Attorney General Steve Marshall certified the election results. Seating Jones will narrow the Republican majority in the Senate to 51 of 100 seats. In a statement, Jones called his victory “a new chapter” and pledged to work with both parties. Moore declined to concede defeat even after Trump urged him to do so. He stood by claims of a fraudulent election in a statement released after the certification and said he had no regrets, media outlets reported. An Alabama judge denied Moore’s request to block certification of the results of the Dec. 12 election in a decision shortly before the canvassing board met. Moore’s challenge alleged there had been potential voter fraud that denied him a chance of victory. His filing on Wednesday in the Montgomery Circuit Court sought to halt the meeting scheduled to ratify Jones’ win on Thursday. Moore could ask for a recount, in addition to possible other court challenges, Merrill said in an interview with Fox News Channel. He would have to complete paperwork “within a timed period” and show he has the money for a challenge, Merrill said. “We’ve not been notified yet of their intention to do that,” Merrill said. Regarding the claim of voter fraud, Merrill told CNN that more than 100 cases had been reported. “We’ve adjudicated more than 60 of those. We will continue to do that,” he said. Republican lawmakers in Washington had distanced themselves from Moore and called for him to drop out of the race after several women accused him of sexual assault or misconduct dating back to when they were teenagers and he was in his early 30s. Moore has denied wrongdoing and Reuters has not been able to independently verify the allegations. '

## Removal of HTML Contents

**First, let's remove HTML content.**

In [13]:

pip install bs4

Collecting bs4

Downloading bs4-0.0.1.tar.gz (1.1 kB)

Collecting beautifulsoup4

Downloading beautifulsoup4-4.9.3-py3-none-any.whl (115 kB)

|████████████████████████████████| 115 kB 1.3 MB/s

Collecting soupsieve>1.2

Downloading soupsieve-2.2.1-py3-none-any.whl (33 kB)

Building wheels for collected packages: bs4

Building wheel for bs4 (setup.py) ... - \ done

Created wheel for bs4: filename=bs4-0.0.1-py3-none-any.whl size=1273 sha256=2bea095cbbbc5fb6fc44736f40fce54b119a54eba4fa1dbedd43deddc70fda9b

Stored in directory: /root/.cache/pip/wheels/0a/9e/ba/20e5bbc1afef3a491f0b3bb74d508f99403aabe76eda2167ca

Successfully built bs4

Installing collected packages: soupsieve, beautifulsoup4, bs4

Successfully installed beautifulsoup4-4.9.3 bs4-0.0.1 soupsieve-2.2.1

Note: you may need to restart the kernel to use updated packages.

In [14]:

from bs4 import BeautifulSoup

soup = BeautifulSoup(first\_text, "html.parser")

first\_text = soup.get\_text()

first\_text

Out[14]:

'politicsNews Jones certified U.S. Senate winner despite Moore challenge (Reuters) - Alabama officials on Thursday certified Democrat Doug Jones the winner of the state’s U.S. Senate race, after a state judge denied a challenge by Republican Roy Moore, whose campaign was derailed by accusations of sexual misconduct with teenage girls. Jones won the vacant seat by about 22,000 votes, or 1.6 percentage points, election officials said. That made him the first Democrat in a quarter of a century to win a Senate seat in Alabama. The seat was previously held by Republican Jeff Sessions, who was tapped by U.S. President Donald Trump as attorney general. A state canvassing board composed of Alabama Secretary of State John Merrill, Governor Kay Ivey and Attorney General Steve Marshall certified the election results. Seating Jones will narrow the Republican majority in the Senate to 51 of 100 seats. In a statement, Jones called his victory “a new chapter” and pledged to work with both parties. Moore declined to concede defeat even after Trump urged him to do so. He stood by claims of a fraudulent election in a statement released after the certification and said he had no regrets, media outlets reported. An Alabama judge denied Moore’s request to block certification of the results of the Dec. 12 election in a decision shortly before the canvassing board met. Moore’s challenge alleged there had been potential voter fraud that denied him a chance of victory. His filing on Wednesday in the Montgomery Circuit Court sought to halt the meeting scheduled to ratify Jones’ win on Thursday. Moore could ask for a recount, in addition to possible other court challenges, Merrill said in an interview with Fox News Channel. He would have to complete paperwork “within a timed period” and show he has the money for a challenge, Merrill said. “We’ve not been notified yet of their intention to do that,” Merrill said. Regarding the claim of voter fraud, Merrill told CNN that more than 100 cases had been reported. “We’ve adjudicated more than 60 of those. We will continue to do that,” he said. Republican lawmakers in Washington had distanced themselves from Moore and called for him to drop out of the race after several women accused him of sexual assault or misconduct dating back to when they were teenagers and he was in his early 30s. Moore has denied wrongdoing and Reuters has not been able to independently verify the allegations. '

## Removal of Punctuation Marks and Special Characters

**Let's now remove everything except uppercase / lowercase letters using Regular Expressions.**

In [15]:

first\_text = re.sub('\[[^]]\*\]', ' ', first\_text)

first\_text = re.sub('[^a-zA-Z]',' ',first\_text) *# replaces non-alphabets with spaces*

first\_text = first\_text.lower() *# Converting from uppercase to lowercase*

first\_text

Out[15]:

'politicsnews jones certified u s senate winner despite moore challenge reuters alabama officials on thursday certified democrat doug jones the winner of the state s u s senate race after a state judge denied a challenge by republican roy moore whose campaign was derailed by accusations of sexual misconduct with teenage girls jones won the vacant seat by about votes or percentage points election officials said that made him the first democrat in a quarter of a century to win a senate seat in alabama the seat was previously held by republican jeff sessions who was tapped by u s president donald trump as attorney general a state canvassing board composed of alabama secretary of state john merrill governor kay ivey and attorney general steve marshall certified the election results seating jones will narrow the republican majority in the senate to of seats in a statement jones called his victory a new chapter and pledged to work with both parties moore declined to concede defeat even after trump urged him to do so he stood by claims of a fraudulent election in a statement released after the certification and said he had no regrets media outlets reported an alabama judge denied moore s request to block certification of the results of the dec election in a decision shortly before the canvassing board met moore s challenge alleged there had been potential voter fraud that denied him a chance of victory his filing on wednesday in the montgomery circuit court sought to halt the meeting scheduled to ratify jones win on thursday moore could ask for a recount in addition to possible other court challenges merrill said in an interview with fox news channel he would have to complete paperwork within a timed period and show he has the money for a challenge merrill said we ve not been notified yet of their intention to do that merrill said regarding the claim of voter fraud merrill told cnn that more than cases had been reported we ve adjudicated more than of those we will continue to do that he said republican lawmakers in washington had distanced themselves from moore and called for him to drop out of the race after several women accused him of sexual assault or misconduct dating back to when they were teenagers and he was in his early s moore has denied wrongdoing and reuters has not been able to independently verify the allegations '

## Removal of Stopwords

**Let's remove stopwords like is,a,the... Which do not offer much insight.**

In [16]:

nltk.download("stopwords")

from nltk.corpus import stopwords

*# we can use tokenizer instead of split*

first\_text = nltk.word\_tokenize(first\_text)

[nltk\_data] Downloading package stopwords to /usr/share/nltk\_data...

[nltk\_data] Package stopwords is already up-to-date!

In [17]:

first\_text = [ word for word **in** first\_text if **not** word **in** set(stopwords.words("english"))]

## Lemmatization

**Lemmatization to bring back multiple forms of same word to their common root like 'coming', 'comes' into 'come'.**

In [18]:

lemma = nltk.WordNetLemmatizer()

first\_text = [ lemma.lemmatize(word) for word **in** first\_text]

first\_text = " ".join(first\_text)

first\_text

Out[18]:

'politicsnews jones certified u senate winner despite moore challenge reuters alabama official thursday certified democrat doug jones winner state u senate race state judge denied challenge republican roy moore whose campaign derailed accusation sexual misconduct teenage girl jones vacant seat vote percentage point election official said made first democrat quarter century win senate seat alabama seat previously held republican jeff session tapped u president donald trump attorney general state canvassing board composed alabama secretary state john merrill governor kay ivey attorney general steve marshall certified election result seating jones narrow republican majority senate seat statement jones called victory new chapter pledged work party moore declined concede defeat even trump urged stood claim fraudulent election statement released certification said regret medium outlet reported alabama judge denied moore request block certification result dec election decision shortly canvassing board met moore challenge alleged potential voter fraud denied chance victory filing wednesday montgomery circuit court sought halt meeting scheduled ratify jones win thursday moore could ask recount addition possible court challenge merrill said interview fox news channel would complete paperwork within timed period show money challenge merrill said notified yet intention merrill said regarding claim voter fraud merrill told cnn case reported adjudicated continue said republican lawmaker washington distanced moore called drop race several woman accused sexual assault misconduct dating back teenager early moore denied wrongdoing reuters able independently verify allegation'

## Perform it for all the examples

**We performed the steps for a single example. Now let's perform it for all the examples in the data.**

In [19]:

*#Removal of HTML Contents*

def remove\_html(text):

soup = BeautifulSoup(text, "html.parser")

return soup.get\_text()

*#Removal of Punctuation Marks*

def remove\_punctuations(text):

return re.sub('\[[^]]\*\]', '', text)

*# Removal of Special Characters*

def remove\_characters(text):

return re.sub("[^a-zA-Z]"," ",text)

*#Removal of stopwords*

def remove\_stopwords\_and\_lemmatization(text):

final\_text = []

text = text.lower()

text = nltk.word\_tokenize(text)

for word **in** text:

if word **not** **in** set(stopwords.words('english')):

lemma = nltk.WordNetLemmatizer()

word = lemma.lemmatize(word)

final\_text.append(word)

return " ".join(final\_text)

*#Total function*

def cleaning(text):

text = remove\_html(text)

text = remove\_punctuations(text)

text = remove\_characters(text)

text = remove\_stopwords\_and\_lemmatization(text)

return text

*#Apply function on text column*

data['text']=data['text'].apply(cleaning)

In [20]:

data.head()

Out[20]:

|  | text | target |
| --- | --- | --- |
| 0 | politicsnews u budget fight loom republican fl... | 1 |
| 1 | politicsnews u military accept transgender rec... | 1 |
| 2 | politicsnews senior u republican senator let m... | 1 |
| 3 | politicsnews fbi russia probe helped australia... | 1 |
| 4 | politicsnews trump want postal service charge ... | 1 |

## Let's make some visualization with new data.

### 1.WordCloud for Real News

In [21]:

from wordcloud import WordCloud,STOPWORDS

plt.figure(figsize = (15,15))

wc = WordCloud(max\_words = 500 , width = 1000 , height = 500 , stopwords = STOPWORDS).generate(" ".join(data[data.target == 1].text))

plt.imshow(wc , interpolation = 'bilinear')

Out[21]:

<matplotlib.image.AxesImage at 0x7f6934fd2750>

### 2.WordCloud for Fake News

In [22]:

plt.figure(figsize = (15,15))

wc = WordCloud(max\_words = 500 , width = 1000 , height = 500 , stopwords = STOPWORDS).generate(" ".join(data[data.target == 0].text))

plt.imshow(wc , interpolation = 'bilinear')

Out[22]:

<matplotlib.image.AxesImage at 0x7f6934fdd050>

### Number of words in each text

In [23]:

fig,(ax1,ax2)=plt.subplots(1,2,figsize=(12,8))

text\_len=data[data['target']==0]['text'].str.split().map(lambda x: len(x))

ax1.hist(text\_len,color='SkyBlue')

ax1.set\_title('Fake news text')

text\_len=data[data['target']==1]['text'].str.split().map(lambda x: len(x))

ax2.hist(text\_len,color='PeachPuff')

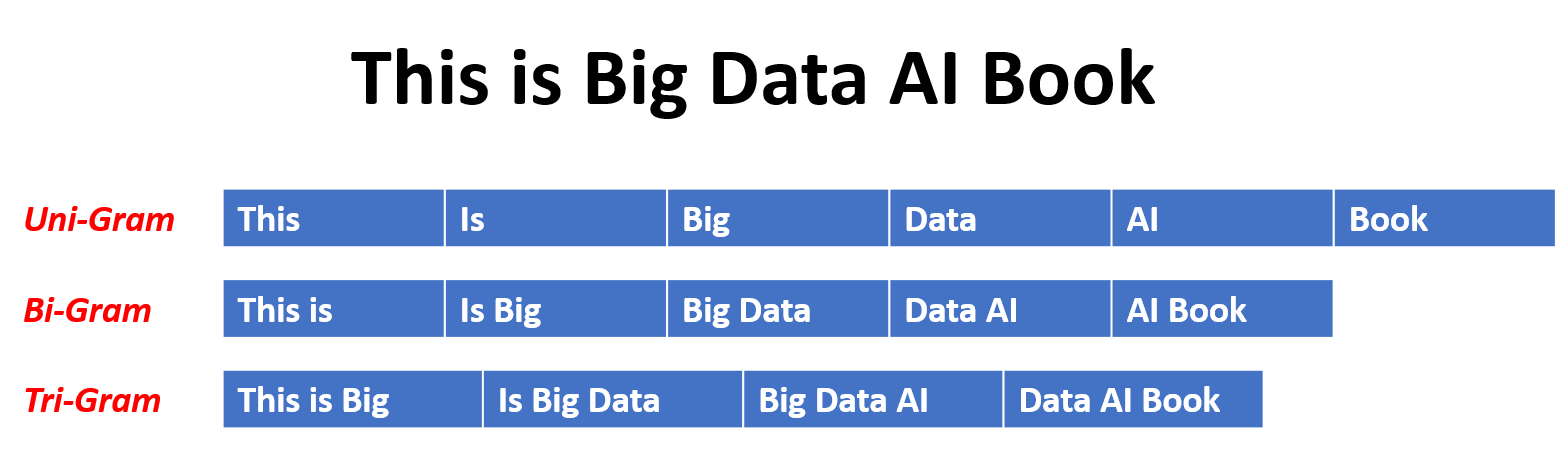
ax2.set\_title('Real news text')

fig.suptitle('Words in texts')

plt.show()

**The number of words seems to be a bit different. 500 words are most common in real news category while around 250 words are most common in fake news category.**

## N-Gram Analysis



In [24]:

texts = ' '.join(data['text'])

In [25]:

string = texts.split(" ")

In [26]:

def draw\_n\_gram(string,i):

n\_gram = (pd.Series(nltk.ngrams(string, i)).value\_counts())[:15]

n\_gram\_df=pd.DataFrame(n\_gram)

n\_gram\_df = n\_gram\_df.reset\_index()

n\_gram\_df = n\_gram\_df.rename(columns={"index": "word", 0: "count"})

print(n\_gram\_df.head())

plt.figure(figsize = (16,9))

return sns.barplot(x='count',y='word', data=n\_gram\_df)

## Unigram Analysis

In [27]:

draw\_n\_gram(string,1)

word count

0 (trump,) 149603

1 (said,) 133030

2 (u,) 78516

3 (state,) 62726

4 (president,) 58790

Out[27]:

<AxesSubplot:xlabel='count', ylabel='word'>

## Bigram Analysis

In [28]:

draw\_n\_gram(string,2)

word count

0 (donald, trump) 25203

1 (united, state) 18943

2 (white, house) 16296

3 (hillary, clinton) 10217

4 (new, york) 9305

Out[28]:

<AxesSubplot:xlabel='count', ylabel='word'>

## Trigram Analysis

In [29]:

draw\_n\_gram(string,3)

word count

0 (president, donald, trump) 6830

1 (pic, twitter, com) 6185

2 (featured, image, via) 6029

3 (president, barack, obama) 3911

4 (getty, image, news) 3575

Out[29]:

<AxesSubplot:xlabel='count', ylabel='word'>

## Modeling

## Train Test Split

In [30]:

X\_train, X\_test, y\_train, y\_test = train\_test\_split(data['text'], data['target'], random\_state=0)

## Tokenizing

* **Tokenizing Text -> Repsesenting each word by a number**
* **Mapping of orginal word to number is preserved in word\_index property of tokenizer**

### Lets keep all news to 300, add padding to news with less than 300 words and truncating long ones

In [31]:

max\_features = 10000

maxlen = 300

In [32]:

tokenizer = text.Tokenizer(num\_words=max\_features)

tokenizer.fit\_on\_texts(X\_train)

tokenized\_train = tokenizer.texts\_to\_sequences(X\_train)

X\_train = sequence.pad\_sequences(tokenized\_train, maxlen=maxlen)

In [33]:

tokenized\_test = tokenizer.texts\_to\_sequences(X\_test)

X\_test = sequence.pad\_sequences(tokenized\_test, maxlen=maxlen)

## Training LSTM Model

In [34]:

batch\_size = 256

epochs = 10

embed\_size = 100

In [35]:

model = Sequential()

*#Non-trainable embeddidng layer*

model.add(Embedding(max\_features, output\_dim=embed\_size, input\_length=maxlen, trainable=False))

*#LSTM*

model.add(LSTM(units=128 , return\_sequences = True , recurrent\_dropout = 0.25 , dropout = 0.25))

model.add(LSTM(units=64 , recurrent\_dropout = 0.1 , dropout = 0.1))

model.add(Dense(units = 32 , activation = 'relu'))

model.add(Dense(1, activation='sigmoid'))

model.compile(optimizer=keras.optimizers.Adam(lr = 0.01), loss='binary\_crossentropy', metrics=['accuracy'])

In [36]:

model.summary()

Model: "sequential"

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Layer (type) Output Shape Param #

=================================================================

embedding (Embedding) (None, 300, 100) 1000000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lstm (LSTM) (None, 300, 128) 117248

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

lstm\_1 (LSTM) (None, 64) 49408

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense (Dense) (None, 32) 2080

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

dense\_1 (Dense) (None, 1) 33

=================================================================

Total params: 1,168,769

Trainable params: 168,769

Non-trainable params: 1,000,000

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

In [37]:

history = model.fit(X\_train, y\_train, validation\_split=0.3, epochs=10, batch\_size=batch\_size, shuffle=True, verbose = 1)

Epoch 1/10

93/93 [==============================] - 268s 3s/step - loss: 0.5514 - accuracy: 0.7044 - val\_loss: 1.2749 - val\_accuracy: 0.5668

Epoch 2/10

93/93 [==============================] - 261s 3s/step - loss: 0.3611 - accuracy: 0.8452 - val\_loss: 0.2542 - val\_accuracy: 0.8987

Epoch 3/10

93/93 [==============================] - 263s 3s/step - loss: 0.2870 - accuracy: 0.8763 - val\_loss: 0.2555 - val\_accuracy: 0.8998

Epoch 4/10

93/93 [==============================] - 264s 3s/step - loss: 0.2686 - accuracy: 0.8857 - val\_loss: 0.2131 - val\_accuracy: 0.9171

Epoch 5/10

93/93 [==============================] - 264s 3s/step - loss: 0.2209 - accuracy: 0.9162 - val\_loss: 0.1326 - val\_accuracy: 0.9435

Epoch 6/10

93/93 [==============================] - 263s 3s/step - loss: 0.1733 - accuracy: 0.9389 - val\_loss: 0.1308 - val\_accuracy: 0.9392

Epoch 7/10

93/93 [==============================] - 267s 3s/step - loss: 0.0712 - accuracy: 0.9695 - val\_loss: 0.0389 - val\_accuracy: 0.9860

Epoch 8/10

93/93 [==============================] - 269s 3s/step - loss: 0.0414 - accuracy: 0.9843 - val\_loss: 0.0402 - val\_accuracy: 0.9838

Epoch 9/10

93/93 [==============================] - 276s 3s/step - loss: 0.0418 - accuracy: 0.9842 - val\_loss: 0.0400 - val\_accuracy: 0.9875

Epoch 10/10

93/93 [==============================] - 270s 3s/step - loss: 0.0317 - accuracy: 0.9886 - val\_loss: 0.0454 - val\_accuracy: 0.9828

## Analysis After Training

In [38]:

print("Accuracy of the model on Training Data is - " , model.evaluate(X\_train,y\_train)[1]\*100 , "%")

print("Accuracy of the model on Testing Data is - " , model.evaluate(X\_test,y\_test)[1]\*100 , "%")

1053/1053 [==============================] - 101s 96ms/step - loss: 0.0393 - accuracy: 0.9843

Accuracy of the model on Training Data is - 98.42603802680969 %

351/351 [==============================] - 34s 97ms/step - loss: 0.0397 - accuracy: 0.9840

Accuracy of the model on Testing Data is - 98.39643836021423 %

In [39]:

plt.figure()

plt.plot(history.history["accuracy"], label = "Train")

plt.plot(history.history["val\_accuracy"], label = "Test")

plt.title("Accuracy")

plt.ylabel("Acc")

plt.xlabel("epochs")

plt.legend()

plt.show()

In [40]:

plt.figure()

plt.plot(history.history["loss"], label = "Train")

plt.plot(history.history["val\_loss"], label = "Test")

plt.title("Loss")

plt.ylabel("Acc")

plt.xlabel("epochs")

plt.legend()

plt.show()

In [41]:

pred = model.predict\_classes(X\_test)

print(classification\_report(y\_test, pred, target\_names = ['Fake','Real']))

precision recall f1-score support

Fake 1.00 0.97 0.98 5858

Real 0.97 1.00 0.98 5367

accuracy 0.98 11225

macro avg 0.98 0.98 0.98 11225

weighted avg 0.98 0.98 0.98 11225

**CONCLUSION AND FUTURE WORK FOR PROJECT PHASE2:**

**PROJECT CONCLUSION:**

To conclude, fake news detection is a critical area in the battle against misinformation. Current approaches employ machine learning models, natural language processing techniques, and dataset curation to identify deceptive content. However, there are challenges, such as evolving tactics used by purveyors of fake news.

**FUTURE WORK:**

Future work in this field could focus on several key areas. Firstly, enhancing model robustness by incorporating multi-modal information (text, images, videos) for a more comprehensive analysis. Additionally, improving the interpretability of models to gain insight into their decision-making process is crucial. Research on adversarial attacks and defenses is also essential to stay ahead of those trying to bypass detection systems.

Moreover, creating larger and more diverse annotated datasets, especially for non-English languages, will help address the issue of generalization. Collaborative efforts between academia, industry, and governments are necessary to combat this global problem effectively.

Incorporating real-time detection capabilities, developing user-friendly browser extensions, and educating the public about fake news indicators are areas ripe for exploration. Finally, ongoing vigilance and adaptability to emerging threats will be key in the continuous fight against fake neW.