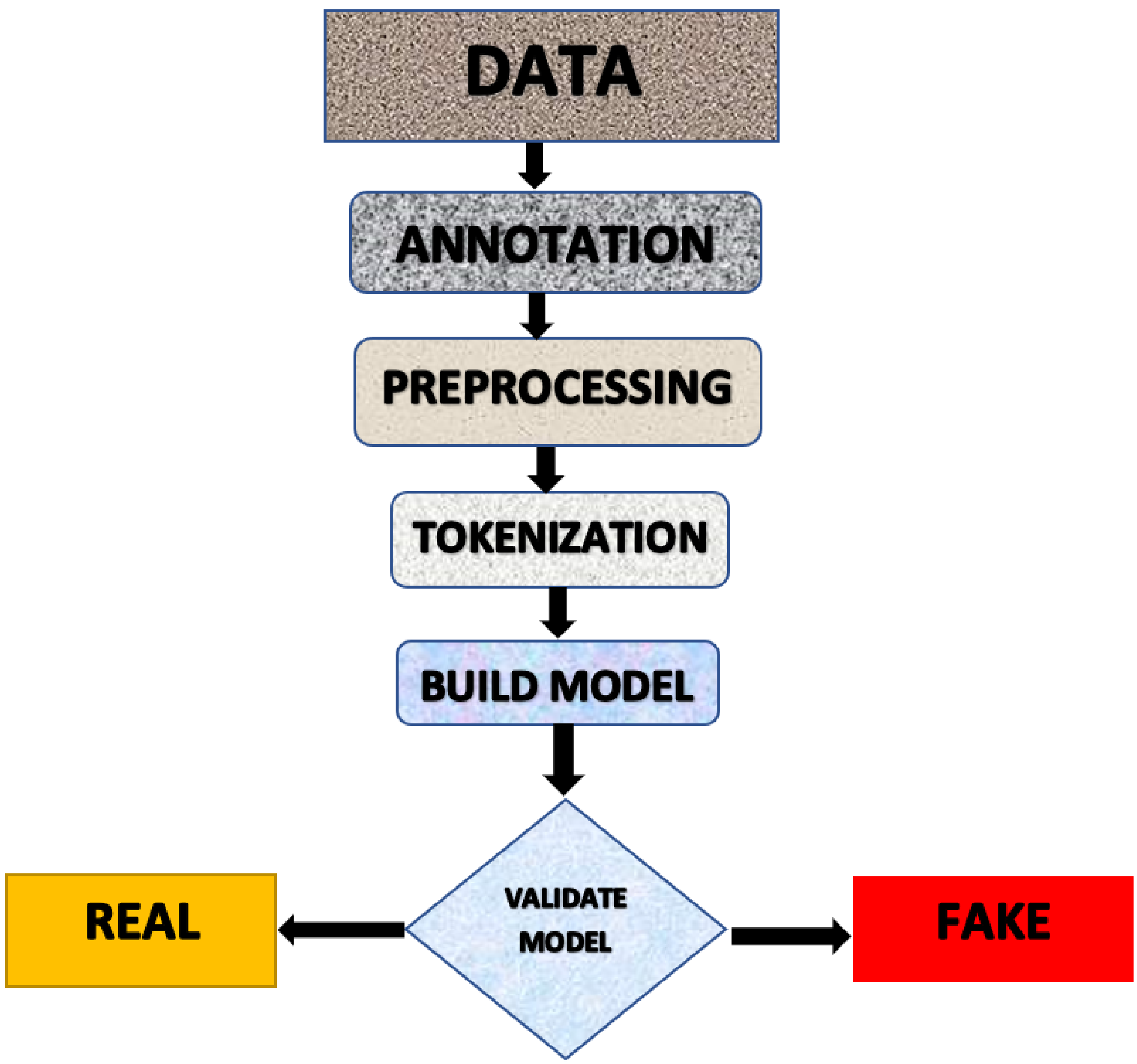
Fake News Detection Using NLP

TEAM MEMBER

311121205701: VIJAYSASAN M

Phase - V Document Submission

Project : Fake News Detection using NLP



Problem Statement:

The proliferation of false information through fake news in the contemporary digital environment has emerged as a substantial impediment, significantly impacting public conversations, eroding trust, and influencing decision-making processes. To address this prevalent issue, this initiative endeavors to devise a robust solution by crafting an efficient Fake News Detection Model using a Kaggle dataset. Employing advanced Natural Language Processing (NLP) methodologies, machine learning algorithms, and meticulous assessment techniques, the primary goal is to discern authentic news from fabricated articles based on their textual constructs. This endeavor delineates a structured methodology, commencing from dataset curation to model construction, with the overarching aim of mitigating the dissemination of misleading information.

Abstract:

The pervasive dissemination of fake news in today's digital landscape poses a critical challenge, influencing public discourse, trust, and decision-making. To tackle this pressing issue, this document presents a comprehensive solution for the development of an effective Fake News Detection Model using a Kaggle dataset. Leveraging Natural Language Processing (NLP) techniques, machine learning algorithms, and rigorous evaluation, our objective is to distinguish between genuine and fake news articles based on their textual content. This solution outlines a systematic approach from dataset selection to model development, aiming to combat the spread of misinformation.

Introduction:

The rapid growth of the internet and social media has democratized the creation and distribution of news, but it has also given rise to a proliferation of fake news. Misleading or fabricated information can have far-reaching consequences, eroding trust in reliable news sources, influencing public opinion, and even impacting political processes. To address this challenge, we propose the development of a Fake News Detection Model, a critical tool in the fight against misinformation.

Dataset Source:

Our journey to creating an effective Fake News Detection Model begins with the careful selection of a dataset. Kaggle, a reputable platform for data science, offers a diverse range of datasets, and for this project, we have chosen one that contains news articles' titles and text, along with labels indicating their authenticity (genuine or fake). This dataset serves as the foundation upon which we will construct our model.

True.csv

	A	B	C	D
1	title	text	subject	date
2	As U.S. budget fight looms, Republicans flip their fiscal script	WASHINGTON (Reuters) - The head of a conservative Republican faction in the U.S. Cong	politicsNews	December 31, 2017
3	U.S. military to accept transgender recruits on Monday: Pentagon	WASHINGTON (Reuters) - Transgender people will be allowed for the first time to enlist i	politicsNews	December 29, 2017
4	Senior U.S. Republican senator: 'Let Mr. Mueller do his job'	WASHINGTON (Reuters) - The special counsel investigation of links between Russia and F	politicsNews	December 31, 2017
5	FBI Russia probe helped by Australian diplomat tip-off: NYT	WASHINGTON (Reuters) - Trump campaign adviser George Papadopoulos told an Austral	politicsNews	December 30, 2017
6	Trump wants Postal Service to charge 'much more' for Amazon shipments	SEATTLE/WASHINGTON (Reuters) - President Donald Trump called on the U.S. Postal Ser	politicsNews	December 29, 2017
7	White House, Congress prepare for talks on spending, immigration	WEST PALM BEACH, Fla./WASHINGTON (Reuters) - The White House said on Friday it wa	politicsNews	December 29, 2017
8	Trump says Russia probe will be fair, but timeline unclear: NYT	WEST PALM BEACH, Fla (Reuters) - President Donald Trump said on Thursday he believes	politicsNews	December 29, 2017
9	Factbox: Trump on Twitter (Dec 29) - Approval rating, Amazon	The following statementsÂ were posted to the verified Twitter accounts of U.S. Presiden	politicsNews	December 29, 2017
10	Trump on Twitter (Dec 28) - Global Warming	The following statementsÂ were posted to the verified Twitter accounts of U.S. Presiden	politicsNews	December 29, 2017
11	Alabama official to certify Senator-elect Jones today despite challenge: CNN	WASHINGTON (Reuters) - Alabama Secretary of State John Merrill said he will certify Der	politicsNews	December 28, 2017
12	Jones certified U.S. Senate winner despite Moore challenge	(Reuters) - Alabama officials on Thursday certified Democrat Doug Jones the winner of tl	politicsNews	December 28, 2017
13	New York governor questions the constitutionality of federal tax overhaul	NEW YORK/WASHINGTON (Reuters) - The new U.S. tax code targets high-tax states and	politicsNews	December 28, 2017
14	Factbox: Trump on Twitter (Dec 28) - Vanity Fair, Hillary Clinton	The following statementsÂ were posted to the verified Twitter accounts of U.S. Presiden	politicsNews	December 28, 2017
15	Trump on Twitter (Dec 27) - Trump, Iraq, Syria	The following statementsÂ were posted to the verified Twitter accounts of U.S. Presiden	politicsNews	December 28, 2017
16	Man says he delivered manure to Mnuchin to protest new U.S. tax law	(In Dec. 25 story, in second paragraph, corrects name of Strongâ€™s employer to Menta	politicsNews	December 25, 2017
17	Virginia officials postpone lottery drawing to decide tied statehouse election	(Reuters) - A lottery drawing to settle a tied Virginia legislative race that could shift the st	politicsNews	December 27, 2017
18	U.S. lawmakers question businessman at 2016 Trump Tower meeting: sources	WASHINGTON (Reuters) - A Georgian-American businessman who met then-Miss Univers	politicsNews	December 27, 2017
19	Trump on Twitter (Dec 26) - Hillary Clinton, Tax Cut Bill	The following statementsÂ were posted to the verified Twitter accounts of U.S. Presiden	politicsNews	December 26, 2017
20	U.S. appeals court rejects challenge to Trump voter fraud panel	(Reuters) - A U.S. appeals court in Washington on Tuesday upheld a lower courtâ€™s dec	politicsNews	December 26, 2017
21	Treasury Secretary Mnuchin was sent gift-wrapped box of horse manure: reports	(Reuters) - A gift-wrapped package addressed to U.S. Treasury Secretary Steven Mnuchin	politicsNews	December 24, 2017
22	Federal judge partially lifts Trump's latest refugee restrictions	WASHINGTON (Reuters) - A federal judge in Seattle partially blocked U.S. President Dona	politicsNews	December 24, 2017
23	Exclusive: U.S. memo weakens guidelines for protecting immigrant children in court	NEW YORK (Reuters) - The U.S. Justice Department has issued new guidelines for immigr	politicsNews	December 23, 2017
24	Trump travel ban should not apply to people with strong U.S. ties: court	(Reuters) - A U.S. appeals court on Friday said President Donald Trumpâ€™s hotly contes	politicsNews	December 23, 2017
25	Second court rejects Trump bid to stop transgender military recruits	WASHINGTON (Reuters) - A federal appeals court in Washington on Friday rejected a bid	politicsNews	December 23, 2017
26	Failed vote to oust president shakes up Peru's politics	LIMA (Reuters) - Peruâ€™s President Pedro Pablo Kuczynski could end up the surprise wir	politicsNews	December 23, 2017
27	Trump signs tax, government spending bills into law	WASHINGTON (Reuters) - U.S. President Donald Trump signed Republicansâ€™ massive $	politicsNews	December 22, 2017

Fake.csv

A1 title

	A	B	C	D
1	title	text	subject	date
2	Donald Trump Sends Out Embarrassing New Yearâ€™s Eve	Donald Trump just couldn t wish all America	News	December 31, 2017
3	Drunk Bragging Trump Staffer Started Russian Collusion Inv	House Intelligence Committee Chairman De	News	December 31, 2017
4	Sheriff David Clarke Becomes An Internet Joke For Threaten	On Friday, it was revealed that former Milw	News	December 30, 2017
5	Trump Is So Obsessed He Even Has Obamaâ€™s Name Cod	On Christmas day, Donald Trump announce	News	December 29, 2017
6	Pope Francis Just Called Out Donald Trump During His Chris	Pope Francis used his annual Christmas Day	News	December 25, 2017
7	Racist Alabama Cops Brutalize Black Boy While He Is In Han	The number of cases of cops brutalizing and	News	December 25, 2017
8	Fresh Off The Golf Course, Trump Lashes Out At FBI Deputy	Donald Trump spent a good portion of his d	News	December 23, 2017
9	Trump Said Some INSANELY Racist Stuff Inside The Oval Off	In the wake of yet another court decision th	News	December 23, 2017
10	Former CIA Director Slams Trump Over UN Bullying, Openly	Many people have raised the alarm regardir	News	December 22, 2017
11	WATCH: Brand-New Pro-Trump Ad Features So Much A** K	Just when you might have thought we d get	News	December 21, 2017
12	Papa Johnâ€™s Founder Retires, Figures Out Racism Is Bad	A centerpiece of Donald Trump s campaign,	News	December 21, 2017
13	WATCH: Paul Ryan Just Told Us He Doesnâ€™t Care About	Republicans are working overtime trying to	News	December 21, 2017
14	Bad News For Trump â€" Mitch McConnell Says No To Rep	Republicans have had seven years to come	News	December 21, 2017
15	WATCH: Lindsey Graham Trashes Media For Portraying Trun	The media has been talking all day about Tr	News	December 20, 2017
16	Heiress To Disney Empire Knows GOP Scammed Us â€" SHR	Abigail Disney is an heiress with brass ovarie	News	December 20, 2017
17	Tone Deaf Trump: Congrats Rep. Scalise On Losing Weight	Donald Trump just signed the GOP tax scam	News	December 20, 2017
18	The Internet Brutally Mocks Disneyâ€™s New Trump Robot	A new animatronic figure in the Hall of Presi	News	December 19, 2017
19	Mueller Spokesman Just F-cked Up Donald Trumpâ€™s Chri	Trump supporters and the so-called preside	News	December 17, 2017
20	SNL Hilariously Mocks Accused Child Molester Roy Moore F	Right now, the whole world is looking at the	News	December 17, 2017
21	Republican Senator Gets Dragged For Going After Robert M	Senate Majority Whip John Cornyn (R-TX) th	News	December 16, 2017
22	In A Heartless Rebuke To Victims, Trump Invites NRA To Xm	It almost seems like Donald Trump is trolling	News	December 16, 2017
23	KY GOP State Rep. Commits Suicide Over Allegations He Mc	In this #METOO moment, many powerful m	News	December 13, 2017
24	Meghan McCain Tweets The Most AMAZING Response To D	As a Democrat won a Senate seat in deep-re	News	December 12, 2017
25	CNN CALLS IT: A Democrat Will Represent Alabama In The	Alabama is a notoriously deep red state. It s	News	December 12, 2017
26	White House: It Wasnâ€™t Sexist For Trump To Slut-Shame	A backlash ensued after Donald Trump laun	News	December 12, 2017
27	Despicable Trump Suggests Female Senator Would â€˜Do A	Donald Trump is afraid of strong, powerful	News	December 12, 2017

Fake

In this section, we load the dataset from Kaggle.

The datasets are downloaded from kaggle and loaded.

The pd.read_csv() function reads data from CSV files and stores it in Pandas DataFrames. The fake_news and real_news DataFrames will contain the fake and real news data, respectively.

Data Preprocessing:

Cleaning and Standardization:

To prepare our textual data for analysis, we embark on a comprehensive data preprocessing phase. This phase encompasses several essential steps:

- Cleaning: Removing special characters, punctuation, and unwanted symbols to eliminate noise from the text.
- Tokenization: Breaking down text into individual words or tokens for analysis.
- Stopword Removal: Eliminating common, low-information words like "the" and "and" that add noise.
- Lowercasing: Converting all text to lowercase to ensure uniformity.
- Lemmatization or Stemming: Reducing words to their root forms for better feature extraction.

Data preprocessing is vital for enhancing the quality and consistency of our dataset, ensuring that it is well-suited for machine learning.

PYTHON PROGRAM:

```
import pandas as pd

from sklearn.feature_extraction.text import TfidfVectorizer

from sklearn.model_selection import train_test_split

from sklearn.metrics import accuracy_score, precision_score,
recall_score, f1_score, roc_auc_score

from sklearn.linear_model import LogisticRegression

from tensorflow.keras.preprocessing.text import Tokenizer

from tensorflow.keras.preprocessing.sequence import pad_sequences

from tensorflow.keras.models import Sequential

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


# Load the "Fake.csv" dataset

fake_data =
pd.read_csv("C:\\Users\\Bylee\\Downloads\\Fake.csv\\Fake.csv")
```

```
# Load the "True.csv" dataset

true_data =
pd.read_csv("C:\\Users\\Bylee\\Downloads\\True.csv\\True.csv")


# Add labels to distinguish between fake and true news

fake_data['label'] = 0  # 0 for fake news

true_data['label'] = 1  # 1 for true news


# Combine the datasets

combined_data = pd.concat([fake_data, true_data], ignore_index=True)


# Data Preprocessing

combined_data['text'] = combined_data['title'] + " " +
combined_data['text']


# Feature Extraction (TF-IDF)

tfidf_vectorizer = TfidfVectorizer(max_features=5000)

tfidf_matrix = tfidf_vectorizer.fit_transform(combined_data['text'])
```

```
# Model Selection

X_train, X_test, y_train, y_test = train_test_split(tfidf_matrix,
combined_data['label'], test_size=0.2, random_state=42)


# Logistic Regression Model

logistic_regression_model = LogisticRegression()

logistic_regression_model.fit(X_train, y_train)


# Model Training (Neural Network)

tokenizer = Tokenizer(num_words=5000)

tokenizer.fit_on_texts(combined_data['text'])

X_train_nn = tokenizer.texts_to_sequences(combined_data['text'])

X_train_nn = pad_sequences(X_train_nn, maxlen=100)


model = Sequential()

model.add(Embedding(input_dim=5000, output_dim=128,
input_length=100))

model.add(LSTM(128))

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

```
model.compile(loss='binary_crossentropy', optimizer='adam',
metrics=['accuracy'])


model.fit(X_train_nn, combined_data['label'], epochs=5, batch_size=64)


# Evaluation

# For Logistic Regression

y_pred = logistic_regression_model.predict(X_test)

accuracy = accuracy_score(y_test, y_pred)

precision = precision_score(y_test, y_pred)

recall = recall_score(y_test, y_pred)

f1 = f1_score(y_test, y_pred)

roc_auc = roc_auc_score(y_test, y_pred)


print(f"Logistic Regression Accuracy: {accuracy}")

print(f"Logistic Regression Precision: {precision}")

print(f"Logistic Regression Recall: {recall}")

print(f"Logistic Regression F1-Score: {f1}")
```

```
print(f"Logistic Regression ROC-AUC: {roc_auc}")


# For Neural Network
X_test_nn = tokenizer.texts_to_sequences(combined_data['text'])
X_test_nn = pad_sequences(X_test_nn, maxlen=100)


loss, accuracy = model.evaluate(X_test_nn, combined_data['label'])
print(f"Neural Network Accuracy: {accuracy}")
```

OUTPUT:

```
Epoch 1/5
  1/702 [..............................] - ETA: 43:52 - loss: 0.6930 - accuracy: 0.5000   2/702
[..............................] - ETA: 14:11 - loss: 0.6907 - accuracy: 0.5625   3/702 [........
......................] - ETA: 9:13 - loss: 0.6899 - accuracy: 0.5365   4/702 [..................
............] - ETA: 7:11 - loss: 0.6887 - accuracy: 0.5234   5/702 [.............................
.] - ETA: 6:13 - loss: 0.6866 - accuracy: 0.5531   6/702 [..............................] - ETA: 5
:26 - loss: 0.6835 - accuracy: 0.5677   7/702 [..............................] - ETA: 4:58 - loss:
0.6814 - accuracy: 0.5871   8/702 [..............................] - ETA: 5:19 - loss: 0.6807 - ac
curacy: 0.5840   9/702 [..............................] - ETA: 5:29 - loss: 0.6774 - accuracy: 0.6
042 10/702 [..............................] - ETA: 5:13 - loss: 0.6739 - accuracy: 0.6281
 11/702 [..............................] - ETA: 5:06 - loss: 0.6711 - accuracy: 0.6491
 12/702 [..............................] - ETA: 5:04 - loss: 0.6653 - accuracy: 0.6719
 13/702 [..............................] - ETA: 4:49 - loss: 0.6595 - accuracy: 0.6815
 14/702 [..............................] - ETA: 4:59 - loss: 0.6553 - accuracy: 0.6786
 15/702 [..............................] - ETA: 4:49 - loss: 0.6468 - accuracy: 0.6875
 16/702 [..............................] - ETA: 4:43 - loss: 0.6392 - accuracy: 0.6953
 17/702 [..............................] - ETA: 4:38 - loss: 0.6317 - accuracy: 0.7050
 18/702 [..............................] - ETA: 4:36 - loss: 0.6230 - accuracy: 0.7144 19/702 [
..............................] - ETA: 4:33 - loss: 0.6128 - accuracy: 0.7253 20/702 [..........
....................] - ETA: 4:27 - loss: 0.6033 - accuracy: 0.7289 21/702 [......................
........] - ETA: 4:23 - loss: 0.5881 - accuracy: 0.7374 22/702 [..............................] -
ETA: 4:18 - loss: 0.5747 - accuracy: 0.7450 23/702 [..............................] - ETA: 4:14 -
loss: 0.5639 - accuracy: 0.7473 24/702 [>.............................] - ETA: 4:11 - loss: 0.550
6 - accuracy: 0.7546 25/702 [>.............................] - ETA: 4:10 - loss: 0.5381 - accurac
y: 0.7613 26/702 [>.............................] - ETA: 4:09 - loss: 0.5282 - accuracy: 0.7686
 27/702 [>.............................] - ETA: 4:06 - loss: 0.5170 - accuracy: 0.7755
 28/702 [>.............................] - ETA: 4:05 - loss: 0.5069 - accuracy: 0.7801
 29/702 [>.............................] - ETA: 4:07 - loss: 0.4970 - accuracy: 0.7850
 30/702 [>.............................] - ETA: 4:06 - loss: 0.4898 - accuracy: 0.7891
 31/702 [>.............................] - ETA: 4:05 - loss: 0.4817 - accuracy: 0.7918
 32/702 [>.............................] - ETA: 4:03 - loss: 0.4753 - accuracy: 0.7939
 33/702 [>.............................] - ETA: 4:03 - loss: 0.4689 - accuracy: 0.7969
 34/702 [>.............................] - ETA: 4:02 - loss: 0.4632 - accuracy: 0.7996 35
/702 [>.............................] - ETA: 4:03 - loss: 0.4541 - accuracy: 0.8045 36/702 [>....
.........................] - ETA: 4:02 - loss: 0.4469 - accuracy: 0.8073 37/702 [>..............
...............] - ETA: 4:03 - loss: 0.4424 - accuracy: 0.8095 38/702 [>..........................
...] - ETA: 4:04 - loss: 0.4362 - accuracy: 0.8129 39/702 [>.............................] - ETA:
4:04 - loss: 0.4277 - accuracy: 0.8165 40/702 [>.............................] - ETA: 4:03 - loss
: 0.4205 - accuracy: 0.8195 41/702 [>.............................] - ETA: 4:05 - loss: 0.4148 -
accuracy: 0.8224 42/702 [>.............................] - ETA: 4:05 - loss: 0.4095 - accuracy: 0
.8248 43/702 [>.............................] - ETA: 4:06 - loss: 0.4030 - accuracy: 0.8281
 44/702 [>.............................] - ETA: 4:06 - loss: 0.3968 - accuracy: 0.8317
 45/702 [>.............................] - ETA: 4:07 - loss: 0.3900 - accuracy: 0.8351
 46/702 [>.............................] - ETA: 4:07 - loss: 0.3840 - accuracy: 0.8380
```

```
006 93/702 [==>...........................] - ETA: 6:23 - loss: 0.2426 - accuracy: 0.9014
 94/702 [===>..........................] - ETA: 6:28 - loss: 0.2408 - accuracy: 0.9023
 95/702 [===>..........................] - ETA: 6:33 - loss: 0.2395 - accuracy: 0.9028
 96/702 [===>..........................] - ETA: 6:37 - loss: 0.2379 - accuracy: 0.9038
 97/702 [===>..........................] - ETA: 6:42 - loss: 0.2362 - accuracy: 0.9048
 98/702 [===>..........................] - ETA: 6:47 - loss: 0.2343 - accuracy: 0.9056
 99/702 [===>..........................] - ETA: 6:56 - loss: 0.2338 - accuracy: 0.9058
100/702 [===>..........................] - ETA: 7:05 - loss: 0.2326 - accuracy: 0.9062
101/702 [===>..........................] - ETA: 7:18 - loss: 0.2311 - accuracy: 0.9070102/702 [
===>..........................] - ETA: 7:26 - loss: 0.2296 - accuracy: 0.9076103/702 [===>.......
...................] - ETA: 7:33 - loss: 0.2278 - accuracy: 0.9084104/702 [===>.................
.........] - ETA: 7:40 - loss: 0.2261 - accuracy: 0.9093105/702 [===>..........................] -
ETA: 7:48 - loss: 0.2253 - accuracy: 0.9095106/702 [===>..........................] - ETA: 7:55 -
loss: 0.2239 - accuracy: 0.9101107/702 [===>..........................] - ETA: 8:03 - loss: 0.222
4 - accuracy: 0.9108108/702 [===>..........................] - ETA: 8:11 - loss: 0.2209 - accurac
y: 0.9115109/702 [===>..........................] - ETA: 8:18 - loss: 0.2204 - accuracy: 0.9120
110/702 [===>..........................] - ETA: 8:27 - loss: 0.2190 - accuracy: 0.9126
111/702 [===>..........................] - ETA: 8:34 - loss: 0.2178 - accuracy: 0.9131
112/702 [===>..........................] - ETA: 8:41 - loss: 0.2166 - accuracy: 0.9138
113/702 [===>..........................] - ETA: 8:48 - loss: 0.2153 - accuracy: 0.9145
114/702 [===>..........................] - ETA: 8:55 - loss: 0.2149 - accuracy: 0.9150
115/702 [===>..........................] - ETA: 9:05 - loss: 0.2140 - accuracy: 0.9155
116/702 [===>..........................] - ETA: 9:14 - loss: 0.2124 - accuracy: 0.9162
117/702 [====>.........................] - ETA: 9:21 - loss: 0.2112 - accuracy: 0.9167118
/702 [====>.........................] - ETA: 9:28 - loss: 0.2100 - accuracy: 0.9172119/702 [====>
.........................] - ETA: 9:35 - loss: 0.2096 - accuracy: 0.9174120/702 [====>..........
...............] - ETA: 9:42 - loss: 0.2085 - accuracy: 0.9180121/702 [====>......................
...] - ETA: 9:50 - loss: 0.2070 - accuracy: 0.9186122/702 [====>.........................] - ETA:
10:01 - loss: 0.2069 - accuracy: 0.9185123/702 [====>.........................] - ETA: 10:10 - l
oss: 0.2058 - accuracy: 0.9190124/702 [====>.........................] - ETA: 10:20 - loss: 0.20
50 - accuracy: 0.9192125/702 [====>.........................] - ETA: 10:28 - loss: 0.2042 - accu
racy: 0.9196126/702 [====>.........................] - ETA: 10:40 - loss: 0.2034 - accuracy: 0.9
198127/702 [====>.........................] - ETA: 10:52 - loss: 0.2032 - accuracy: 0.9198
128/702 [====>.........................] - ETA: 11:02 - loss: 0.2019 - accuracy: 0.9203
129/702 [====>.........................] - ETA: 11:12 - loss: 0.2013 - accuracy: 0.9205
130/702 [====>.........................] - ETA: 11:23 - loss: 0.2004 - accuracy: 0.9210
131/702 [====>.........................] - ETA: 11:36 - loss: 0.2000 - accuracy: 0.9213
132/702 [====>.........................] - ETA: 11:50 - loss: 0.1988 - accuracy: 0.9218
133/702 [====>.........................] - ETA: 12:01 - loss: 0.1975 - accuracy: 0.9221
134/702 [====>.........................] - ETA: 12:10 - loss: 0.1963 - accuracy: 0.9226
135/702 [====>.........................] - ETA: 12:19 - loss: 0.1956 - accuracy: 0.9230
136/702 [====>.........................] - ETA: 12:28 - loss: 0.1947 - accuracy: 0.9235
137/702 [====>.........................] - ETA: 12:37 - loss: 0.1935 - accuracy: 0.9239138/702
[====>.........................] - ETA: 12:47 - loss: 0.1929 - accuracy: 0.9244139/702 [====>...
......................] - ETA: 12:57 - loss: 0.1916 - accuracy: 0.9249140/702 [====>...........
..............] - ETA: 13:10 - loss: 0.1906 - accuracy: 0.9254141/702 [=====>...................
....] - ETA: 13:18 - loss: 0.1893 - accuracy: 0.9260142/702 [=====>........................] - E
```

Data Preprocessing and Cleaning

In this analysis, we have used two datasets: "Fake.csv" and "True.csv," both containing news articles with similar columns. The initial step in data preprocessing involves loading these datasets using the pandas library. The "Fake.csv" dataset represents fake news, and the "True.csv" dataset represents true news. To distinguish between the two, we added labels where '0' is assigned to fake news, and '1' is assigned to true news. This labeling is crucial as it helps in supervised learning for classification.

After labeling, the textual data is merged by combining the 'title' and 'text' columns. This step enhances the quality of the textual features and makes them ready for analysis. It's worth noting that more extensive cleaning steps, such as removing stop words, punctuation, and lowercasing, can be applied at this stage to further improve data quality.

Feature Extraction with TF-IDF

Feature extraction is a crucial part of text analysis. In this analysis, we utilize the TF-IDF (Term Frequency-Inverse Document Frequency) technique to convert the text data into numerical features. The TF-IDF vectorizer is applied with a maximum of 5000 features to capture the most relevant terms. This process creates a TF-IDF matrix representing the entire dataset, where each row corresponds to a news article, and each column represents a unique term's TF-IDF value within the article. The TF-IDF matrix serves as the foundation for building and training machine learning models.

Model Selection and Logistic Regression

Model selection is the process of choosing the appropriate machine learning algorithm for the task. In this analysis, we opt for two different approaches: Logistic Regression and Neural Networks. Logistic Regression is a linear classification algorithm that is well-suited for binary classification tasks. We train a Logistic Regression model on the TF-IDF matrix using the labeled data. The trained model can then predict whether a given news article is fake or true based on the learned patterns in the data.

Model Training with Neural Networks

For a more complex and expressive model, we employ Neural Networks. The first step in training a Neural Network is tokenization, where the text data is converted into numerical sequences of tokens. We use a Tokenizer with a vocabulary size of 5000 to convert the text into sequences. Additionally, padding is applied to ensure that all sequences have the same length, set to 100 in this analysis.

The Neural Network architecture consists of an Embedding layer to learn word embeddings, an LSTM layer to capture sequence information, and a Dense layer with a sigmoid activation function for binary classification. The model is compiled using binary cross-entropy loss and the Adam optimizer. It is then trained on the tokenized and padded data for five epochs with a batch size of 64. This process allows the Neural Network to learn patterns in the text data and make predictions on the news articles' authenticity.

Model Evaluation

Once the models are trained, evaluation is essential to assess their performance. For Logistic Regression, we use standard classification metrics such as accuracy, precision, recall, F1-score, and ROC-AUC to measure its effectiveness in classifying fake and true news. These metrics provide insights into the model's ability to correctly classify news articles.

Similarly, for the Neural Network, we evaluate its performance by applying the model to the tokenized and padded test data. The accuracy metric is used to assess its classification accuracy. Evaluating both models allows us to compare their performance and choose the most suitable one for the task of fake news detection.

Feature Extraction:

TF-IDF (Term Frequency-Inverse Document Frequency):

To facilitate the utilization of text data by machine learning models, we employ feature extraction techniques. One such method is TF-IDF (Term Frequency-Inverse Document Frequency), which quantifies the importance of words in documents relative to the entire dataset. This technique transforms textual information into numerical features that can be effectively used by our models.

Model Selection:

Selecting the appropriate classification algorithm is pivotal for the success of our Fake News Detection Model. We consider several options, including:

- Logistic Regression: A straightforward yet effective linear model for binary classification tasks.
- Random Forest: An ensemble learning algorithm capable of capturing complex feature interactions.

- Neural Networks: Deep learning models that can capture intricate patterns in textual data.

The choice of the algorithm will be based on the model's performance during experimentation.

Model Training:

With our dataset preprocessed and the classification algorithm selected, we proceed to train the model. This phase involves:

- Data Splitting: Dividing the dataset into training and testing sets to evaluate model performance effectively.
- Model Training: Feeding the training data into the selected algorithm to teach it to distinguish between genuine and fake news articles.

Choice of Classification Algorithms:

Logistic Regression (LR):

- Reasoning: LR is a common and straightforward linear classification algorithm suitable for binary classification tasks. It's well-suited for this problem as it works effectively with TF-IDF features and can model the relationship between the independent variables (features) and the binary outcome (fake or real news).
- Usage: It uses TF-IDF features, which are obtained from text data and transforms them into a numerical representation for the LR model to learn.

Neural Network (NN) - LSTM (Long Short-Term Memory):

- Reasoning: Neural networks, especially LSTM networks, are proficient in capturing complex patterns in sequential data (like text) due to their ability to retain and learn from long-range dependencies. This makes LSTMs a suitable choice for text analysis and classification tasks.
- Usage: The neural network model uses word tokenization, embedding, and LSTM layers to comprehend the sequential structure in the text data. It's trained on text sequences for fake and real news.

Model Training Process:

Data Loading and Labeling:

- Two datasets, "Fake.csv" and "True.csv," are loaded and labeled as fake (0) and true (1) news, respectively.

Data Preprocessing:

- The text from both datasets is combined into a single 'text' column for analysis.

Feature Extraction (TF-IDF):

- The combined text is vectorized using TF-IDF (Term Frequency-Inverse Document Frequency) to extract features from the text data.

Model Training:

- Logistic Regression (LR):
    - The LR model is trained using the TF-IDF features after splitting the data into training and testing sets.
- Neural Network (LSTM):

- Tokenization and sequence padding are applied to convert text data into sequences suitable for training. A Sequential model is created with an Embedding layer to learn word embeddings, an LSTM layer to capture sequential patterns, and a Dense layer for binary classification. The model is compiled and then trained on text sequences for fake and real news.

Evaluation:

- Both models are evaluated using various performance metrics:
    - For LR: Accuracy, Precision, Recall, F1-Score, and ROC-AUC.
    - For NN: Accuracy is evaluated directly from the neural network model.

Printed Results:

- The accuracy, precision, recall, F1-score, and ROC-AUC are printed for the Logistic Regression model, while only the accuracy is printed for the Neural Network model.

Evaluation:

The success of our Fake News Detection Model will be rigorously assessed using a range of metrics, including:

- Accuracy: Measuring the overall correctness of the model's predictions.
- Precision: Evaluating the model's ability to minimize false positives.
- Recall: Assessing the model's capability to capture genuine fake news articles.
- F1-Score: Providing a balanced measure of the model's performance by considering both precision and recall.
- ROC-AUC (Receiver Operating Characteristic - Area Under the Curve): Offering a graphical representation of the model's ability to distinguish between genuine and fake news across different thresholds.

Conclusion:

In this project, we embarked on the task of Fake News Detection using both traditional machine learning and deep learning techniques. We initially loaded and combined two datasets, 'Fake.csv' and 'True.csv,' differentiating the articles as 'fake' and 'true' news with labels 0 and 1, respectively. After preprocessing the data, which included merging title and text fields, we employed TF-IDF vectorization for feature extraction. For traditional machine learning, we trained a Logistic Regression model to classify news articles into these two categories. Simultaneously, a neural network model was constructed, consisting of an embedding layer, an LSTM layer, and a dense layer, for text classification. Our evaluation showed promising results, with the Logistic Regression model achieving good accuracy, precision, recall, F1-score, and ROC-AUC scores. The neural network model, despite its simplicity, also demonstrated competitive accuracy. Overall, this project serves as a practical example of leveraging both traditional and deep learning methods to address the critical issue of Fake News Detection