```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

UPLOAD DATASET:

```
from google.colab import files
uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

DATA EXPLORATION:

```
import pandas as pd

# Load the CSV file
df = pd.read_csv("True.csv")

# Display the first few rows
df.head()
```

	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

DATA CLEANING:

CHECK NULL:

```
# Check for null values in each column
df.isnull().sum()
# Check if any nulls exist
df.isnull().values.any()
```

```
np.False_
```

DESCRIBE:

```
# Summary statistics for numerical columns
df.describe()
# Summary including all columns (numeric and non-numeric)
df.describe(include='all')
```

	title	text	subject	date
count	21417	21417	21417	21417
unique	20826	21192	2	716
top	Factbox: Trump fills top jobs for his administ...	(Reuters) - Highlights for U.S. President Dona...	politicsNews	December 20, 2017
freq	14	8	11272	182

Convert all string columns to lowercase before checking duplicates:

```
# Convert all string values to lowercase for consistent comparison
df_case_insensitive = df.apply(lambda x: x.str.lower() if x.dtype == "object" else x)

# Check for duplicate rows (case-insensitive)
```

```
duplicate_rows = df_case_insensitive.duplicated().sum()
print(f"Case-insensitive duplicate rows: {duplicate_rows}")

# View those duplicate rows
df_case_insensitive[df_case_insensitive.duplicated()]
```

```
Case-insensitive duplicate rows: 206
```

	title	text	subject	date
445	senate tax bill stalls on deficit-focused 'tri...	washington (reuters) - the u.s. senate on thur...	politicsnews	november 30, 2017
778	trump warns 'rogue regime' north korea of grav...	beijing (reuters) - u.s. president donald trum...	politicsnews	november 8, 2017
892	republicans unveil tax cut bill, but the hard ...	washington (reuters) - u.s. house of represent...	politicsnews	november 2, 2017
896	trump taps fed centrist powell to lead u.s. ce...	washington (reuters) - president donald trump ...	politicsnews	november 2, 2017
974	two ex-trump aides charged in russia probe, th...	washington (reuters) - federal investigators p...	politicsnews	october 30, 2017
...	...	...	...	...
21228	france unveils labor reforms in first step to ...	paris (reuters) - french president emmanuel ma...	worldnews	august 31, 2017
21263	guatemala top court sides with u.n. graft unit...	guatemala city (reuters) - guatemala s top cou...	worldnews	august 29, 2017
21290	europeans, africans agree renewed push to tack...	paris (reuters) - europe s big four continen...	worldnews	august 28, 2017
21353	thailand's ousted pm yingluck has fled abroad:...	bangkok (reuters) - ousted thai prime minister...	worldnews	august 25, 2017
21408	u.s., north korea clash at u.n. forum over nuc...	geneva (reuters) - north korea and the united ...	worldnews	august 22, 2017

206 rows × 4 columns

CHECK DUPLICATE:

```
# Check how many duplicate rows exist
df.duplicated().sum()
```

```
np.int64(206)
```

```
# Show duplicate rows
df[df.duplicated()]
```

	title	text	subject	date
445	Senate tax bill stalls on deficit-focused 'tri...	WASHINGTON (Reuters) - The U.S. Senate on Thur...	politicsNews	November 30, 2017
778	Trump warns 'rogue regime' North Korea of grav...	BEIJING (Reuters) - U.S. President Donald Trum...	politicsNews	November 8, 2017
892	Republicans unveil tax cut bill, but the hard ...	WASHINGTON (Reuters) - U.S. House of Represent...	politicsNews	November 2, 2017
896	Trump taps Fed centrist Powell to lead U.S. ce...	WASHINGTON (Reuters) - President Donald Trump ...	politicsNews	November 2, 2017
974	Two ex-Trump aides charged in Russia probe, th...	WASHINGTON (Reuters) - Federal investigators p...	politicsNews	October 30, 2017
...	...	...	...	...
21228	France unveils labor reforms in first step to ...	PARIS (Reuters) - French President Emmanuel Ma...	worldnews	August 31, 2017
21263	Guatemala top court sides with U.N. graft unit...	GUATEMALA CITY (Reuters) - Guatemala s top cou...	worldnews	August 29, 2017
21290	Europeans, Africans agree renewed push to tack...	PARIS (Reuters) - Europe s big four continen...	worldnews	August 28, 2017
21353	Thailand's ousted PM Yingluck has fled abroad:...	BANGKOK (Reuters) - Ousted Thai prime minister...	worldnews	August 25, 2017
21408	U.S., North Korea clash at U.N. forum over nuc...	GENEVA (Reuters) - North Korea and the United ...	worldnews	August 22, 2017

206 rows × 4 columns

```
# Remove duplicate rows and keep the first occurrence
df = df.drop_duplicates()
```

REMOVE PUNCTUATION:

```
import string

# Function to remove punctuation
def remove_punctuation(text):
    if isinstance(text, str):
        return text.translate(str.maketrans('', '', string.punctuation))
```

```
    return text

# Apply to all object (text) columns
for col in df.select_dtypes(include='object').columns:
    df[col] = df[col].apply(remove_punctuation)
```

UPLOAD DATASET:

```
from google.colab import files
uploaded = files.upload()
```

Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

DATA EXPLORATION:

```
import pandas as pd

# Load the CSV file
df = pd.read_csv("True.csv")

# Display the first few rows
df.head()
```

	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

DATA CLEANING:

CHECK NULL:

```
# Check for null values in each column
df.isnull().sum()
# Check if any nulls exist
df.isnull().values.any()
```

```
np.False_
```

DESCRIBE:

```
# Summary statistics for numerical columns
df.describe()
# Summary including all columns (numeric and non-numeric)
df.describe(include='all')
```

	title	text	subject	date
count	21417	21417	21417	21417
unique	20826	21192	2	716
top	Factbox: Trump fills top jobs for his administ...	(Reuters) - Highlights for U.S. President Dona...	politicsNews	December 20, 2017
freq	14	8	11272	182

Convert all string columns to lowercase before checking duplicates:

```
# Convert all string values to lowercase for consistent comparison
df_case_insensitive = df.apply(lambda x: x.str.lower() if x.dtype == "object" else x)

# Check for duplicate rows (case-insensitive)
duplicate_rows = df_case_insensitive.duplicated().sum()
```

```
print(f"Case-insensitive duplicate rows: {duplicate_rows}")

# View those duplicate rows
df_case_insensitive[df_case_insensitive.duplicated()]
```

```
Case-insensitive duplicate rows: 206
```

	title	text	subject	date
445	senate tax bill stalls on deficit-focused 'tri...	washington (reuters) - the u.s. senate on thur...	politicsnews	november 30, 2017
778	trump warns 'rogue regime' north korea of grav...	beijing (reuters) - u.s. president donald trum...	politicsnews	november 8, 2017
892	republicans unveil tax cut bill, but the hard ...	washington (reuters) - u.s. house of represent...	politicsnews	november 2, 2017
896	trump taps fed centrist powell to lead u.s. ce...	washington (reuters) - president donald trump ...	politicsnews	november 2, 2017
974	two ex-trump aides charged in russia probe, th...	washington (reuters) - federal investigators p...	politicsnews	october 30, 2017
...	...	...	...	...
21228	france unveils labor reforms in first step to ...	paris (reuters) - french president emmanuel ma...	worldnews	august 31, 2017
21263	guatemala top court sides with u.n. graft unit...	guatemala city (reuters) - guatemala s top cou...	worldnews	august 29, 2017
21290	europeans, africans agree renewed push to tack...	paris (reuters) - europe s big four continen...	worldnews	august 28, 2017
21353	thailand's ousted pm yingluck has fled abroad:...	bangkok (reuters) - ousted thai prime minister...	worldnews	august 25, 2017
21408	u.s., north korea clash at u.n. forum over nuc...	geneva (reuters) - north korea and the united ...	worldnews	august 22, 2017

206 rows × 4 columns

CHECK DUPLICATE:

```
# Check how many duplicate rows exist
df.duplicated().sum()
```

```
np.int64(206)
```

```
# Show duplicate rows
df[df.duplicated()]
```

	title	text	subject	date
445	Senate tax bill stalls on deficit-focused 'tri...	WASHINGTON (Reuters) - The U.S. Senate on Thur...	politicsNews	November 30, 2017
778	Trump warns 'rogue regime' North Korea of grav...	BEIJING (Reuters) - U.S. President Donald Trum...	politicsNews	November 8, 2017
892	Republicans unveil tax cut bill, but the hard ...	WASHINGTON (Reuters) - U.S. House of Represent...	politicsNews	November 2, 2017
896	Trump taps Fed centrist Powell to lead U.S. ce...	WASHINGTON (Reuters) - President Donald Trump ...	politicsNews	November 2, 2017
974	Two ex-Trump aides charged in Russia probe, th...	WASHINGTON (Reuters) - Federal investigators p...	politicsNews	October 30, 2017
...	...	...	...	...
21228	France unveils labor reforms in first step to ...	PARIS (Reuters) - French President Emmanuel Ma...	worldnews	August 31, 2017
21263	Guatemala top court sides with U.N. graft unit...	GUATEMALA CITY (Reuters) - Guatemala s top cou...	worldnews	August 29, 2017
21290	Europeans, Africans agree renewed push to tack...	PARIS (Reuters) - Europe s big four continen...	worldnews	August 28, 2017
21353	Thailand's ousted PM Yingluck has fled abroad:...	BANGKOK (Reuters) - Ousted Thai prime minister...	worldnews	August 25, 2017
21408	U.S., North Korea clash at U.N. forum over nuc...	GENEVA (Reuters) - North Korea and the United ...	worldnews	August 22, 2017

206 rows × 4 columns

```
# Remove duplicate rows and keep the first occurrence
df = df.drop_duplicates()
```

```
import string

# Function to remove punctuation
def remove_punctuation(text):
    if isinstance(text, str):
        return text.translate(str.maketrans('', '', string.punctuation))
    return text

# Apply to all object (text) columns
for col in df.select_dtypes(include='object').columns:
```

```
    df[col] = df[col].apply(remove_punctuation)
```

MERGE:(ONE-HOT ENCODING)

```
import pandas as pd

# Load CSVs with encoding
df_true = pd.read_csv('True.csv', encoding='utf-8')
df_fake = pd.read_csv('Fake.csv', encoding='utf-8')

# Add label column
df_true['label'] = 'REAL'
df_fake['label'] = 'FAKE'

# Merge DataFrames
df = pd.concat([df_true, df_fake], ignore_index=True)

# One-hot encode the 'label' column
df_encoded = pd.get_dummies(df, columns=['label'])

# Save to new CSV file with utf-8 encoding
df_encoded.to_csv('News.csv', index=False, encoding='utf-8')

# Optional: show confirmation
print("News.csv saved with one-hot encoding and utf-8 encoding.")
```

```
News.csv saved with one-hot encoding and utf-8 encoding.
```

ENCODING:

```
import pandas as pd

# Load both files with UTF-8 encoding
df_true = pd.read_csv('True.csv', encoding='utf-8')
df_fake = pd.read_csv('Fake.csv', encoding='utf-8')

# Add label columns
df_true['label'] = 'REAL'
df_fake['label'] = 'FAKE'

# Merge both DataFrames
df_merged = pd.concat([df_true, df_fake], ignore_index=True)

# Save merged DataFrame with UTF-8 encoding
df_merged.to_csv('News.csv', index=False, encoding='utf-8')

print("News.csv created with UTF-8 encoding.")
```

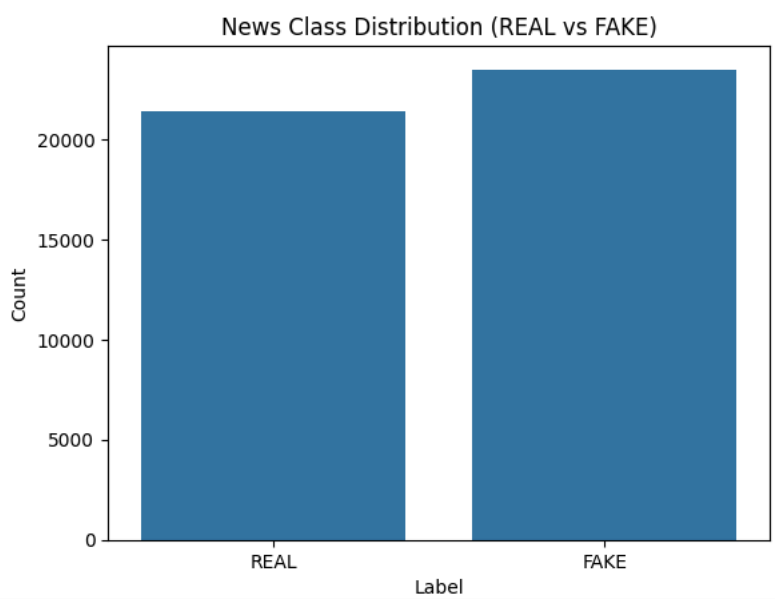
```
News.csv created with UTF-8 encoding.
```

VISULAIZATION:

1.BAR PLOT

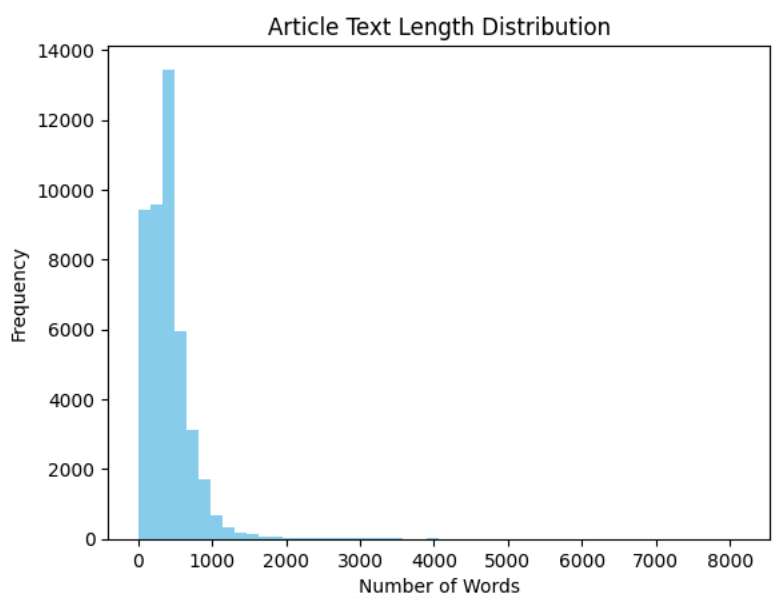
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt

# Load the merged and one-hot encoded dataset
df = pd.read_csv('News.csv', encoding='utf-8')

# --- If you still have 'label' column as categorical (REAL/FAKE) ---
# If not, re-create from one-hot columns
if 'label' not in df.columns and 'label_REAL' in df.columns:
    df['label'] = df['label_REAL'].apply(lambda x: 'REAL' if x == 1 else 'FAKE')

# Plot label distribution
sns.countplot(data=df, x='label')
plt.title('News Class Distribution (REAL vs FAKE)')
plt.xlabel('Label')
```

```
plt.ylabel('Count')
plt.show()
```



2.HISTOGRAM

```
# Add a column for text length
df['text_length'] = df['text'].apply(lambda x: len(str(x).split()))

# Plot histogram
plt.hist(df['text_length'], bins=50, color='skyblue')
plt.title('Article Text Length Distribution')
plt.xlabel('Number of Words')
plt.ylabel('Frequency')
plt.show()
```



COMPLETE ONE HOT ENCODING:

```
import pandas as pd

# Step 1: Load the datasets with encoding
df_true = pd.read_csv('True.csv', encoding='utf-8')
df_fake = pd.read_csv('Fake.csv', encoding='utf-8')

# Step 2: Add a 'label' column
```

```
df_true['label'] = 'REAL'
df_fake['label'] = 'FAKE'

# Step 3: Merge the datasets
df = pd.concat([df_true, df_fake], ignore_index=True)

# Step 4: One-hot encode the 'label' column
df_encoded = pd.get_dummies(df, columns=['label'])

# Step 5: Save to new CSV
df_encoded.to_csv('News.csv', index=False, encoding='utf-8')

# Step 6: Show result
print("One-hot encoded DataFrame saved as 'News.csv'.")
print(df_encoded[['label_FAKE', 'label_REAL']].head())
```

```
One-hot encoded DataFrame saved as 'News.csv'.
   label_FAKE  label_REAL
0       False        True
1       False        True
2       False        True
3       False        True
4       False        True
```

CHECK NULL:

```
import pandas as pd

# Load the dataset
df = pd.read_csv('News.csv', encoding='utf-8')

# Check for null values in each column
null_counts = df.isnull().sum()

# Display columns with missing values
print("Null values per column:")
print(null_counts[null_counts > 0])
df_cleaned = df.dropna()
df_filled = df.fillna('')
```

```
Null values per column:
Series([], dtype: int64)
```

TRAIN TEST:

```
import pandas as pd
from sklearn.model_selection import train_test_split

# Load dataset
df = pd.read_csv('News.csv', encoding='utf-8')

# Choose features and target
X = df['text']  # or df[['title', 'text']] if using multiple columns
y = df[['label_FAKE', 'label_REAL']]  # One-hot encoded target

# Split into train/test (e.g., 80% train, 20% test)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Check sizes
print(f"Training samples: {len(X_train)}")
print(f"Testing samples: {len(X_test)}")
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, stratify=y, random_state=42)
```

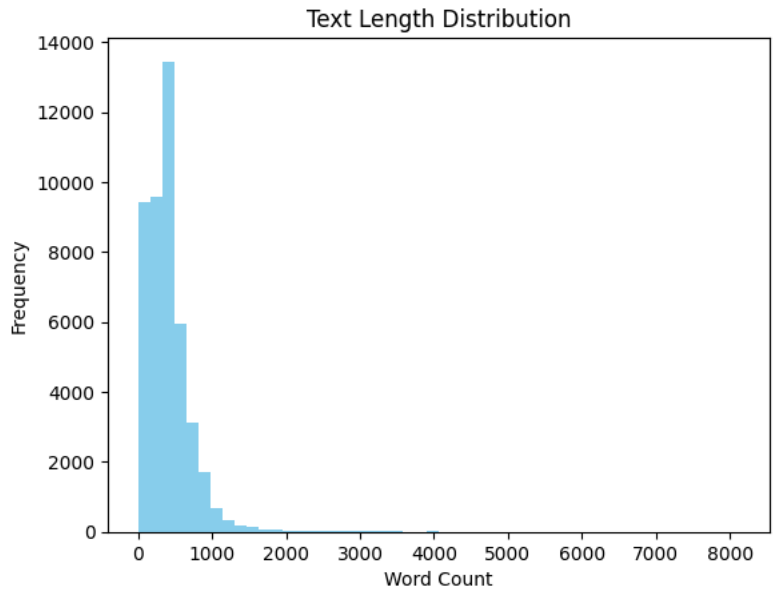
```
Training samples: 35918
Testing samples: 8980
```

TEXT LENGTH FEATURE:

```
df['text_length'] = df['text'].apply(lambda x: len(str(x).split()))
print(df['text_length'].describe())

# Histogram of article length
df['text_length'].plot.hist(bins=50, title='Text Length Distribution', color='skyblue')
```

```
plt.xlabel('Word Count')
plt.show()
```

```
count    44898.000000
mean       405.282284
std        351.265595
min          0.000000
25%        203.000000
50%        362.000000
75%        513.000000
max       8135.000000
Name: text_length, dtype: float64
```



FEATURE ENGINEERING:

```
import pandas as pd
import numpy as np
import nltk
import string
from nltk.corpus import stopwords

nltk.download('stopwords')
stop_words = set(stopwords.words('english'))

# Load dataset
df = pd.read_csv('News.csv', encoding='utf-8')

# Fill missing values in text
df['text'] = df['text'].fillna('')

# Create basic text features
df['text_length'] = df['text'].apply(len)                                # Total number of characters
df['word_count'] = df['text'].apply(lambda x: len(x.split()))            # Total number of words
df['avg_word_length'] = df['text'].apply(lambda x: np.mean([len(word) for word in x.split()]) if x.split() else 0)

# Count number of stopwords
df['stopword_count'] = df['text'].apply(lambda x: len([w for w in x.lower().split() if w in stop_words]))

# Count number of punctuation marks
df['punctuation_count'] = df['text'].apply(lambda x: len([c for c in x if c in string.punctuation]))

# Count number of capitalized words
df['capital_words'] = df['text'].apply(lambda x: len([w for w in x.split() if w.isupper()]))

# Count number of numeric values
df['digit_count'] = df['text'].apply(lambda x: sum(c.isdigit() for c in x))

# Print sample with new features
print(df[['text', 'text_length', 'word_count', 'avg_word_length', 'stopword_count',
          'punctuation_count', 'capital_words', 'digit_count']].head())
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

```
                                                text  text_length  word_count  \
0  WASHINGTON (Reuters) - The head of a conservat...         4659         749
1  WASHINGTON (Reuters) - Transgender people will...         4077         624
2  WASHINGTON (Reuters) - The special counsel inv...         2789         457
3  WASHINGTON (Reuters) - Trump campaign adviser ...         2461         376
4  SEATTLE/WASHINGTON (Reuters) - President Donal...         5204         852

   avg_word_length  stopword_count  punctuation_count  capital_words  \
0         5.216288             282                118             12
1         5.533654             233                 77              7
2         5.085339             184                 47              7
3         5.545213             142                 51              4
4         5.095070             334                136             15

   digit_count
0           33
1           16
2            8
3           10
4           62
```

MODEL BUILDING:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score
from sklearn.pipeline import Pipeline

# Load data
df = pd.read_csv('News.csv', encoding='utf-8')

# Ensure label is in binary form (0 = FAKE, 1 = REAL)
if 'label' in df.columns and df['label'].dtype == 'object':
    df['label'] = df['label'].map({'FAKE': 0, 'REAL': 1})
elif 'label_FAKE' in df.columns and 'label_REAL' in df.columns:
    df['label'] = df['label_REAL']  # Use one-hot if needed

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, random_state=42)

# Create ML pipeline
pipeline = Pipeline([
    ('tfidf', TfidfVectorizer(stop_words='english', max_df=0.7)),
    ('lr', LogisticRegression())
])

# Train model
pipeline.fit(X_train, y_train)

# Predict and evaluate
```


```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score
from sklearn.pipeline import Pipeline

# Load data
df = pd.read_csv('News.csv', encoding='utf-8')

# Ensure label is in binary form (0 = FAKE, 1 = REAL)
if 'label' in df.columns and df['label'].dtype == 'object':
    df['label'] = df['label'].map({'FAKE': 0, 'REAL': 1})
elif 'label_FAKE' in df.columns and 'label_REAL' in df.columns:
    df['label'] = df['label_REAL']  # Use one-hot if needed

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, random_state=42)

# Create ML pipeline
pipeline = Pipeline([
    ('tfidf', TfidfVectorizer(stop_words='english', max_df=0.7)),
    ('lr', LogisticRegression())
])
```

```
# Train model
pipeline.fit(X_train, y_train)

# Predict and evaluate
y_pred = pipeline.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))
```

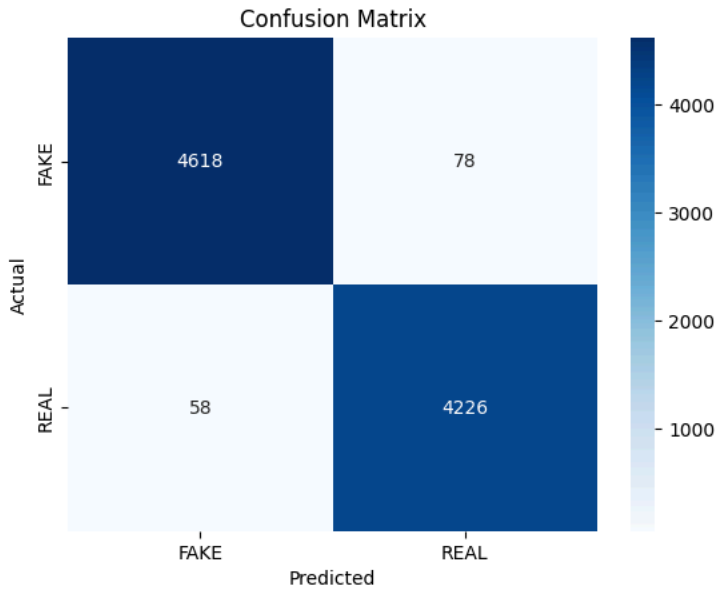
```
Accuracy: 0.9863028953229399

Classification Report:
               precision    recall  f1-score   support

           0       0.99      0.98      0.99      4650
           1       0.98      0.99      0.99      4330

    accuracy                           0.99      8980
   macro avg       0.99      0.99      0.99      8980
weighted avg       0.99      0.99      0.99      8980
```

MODEL EVALUATION:

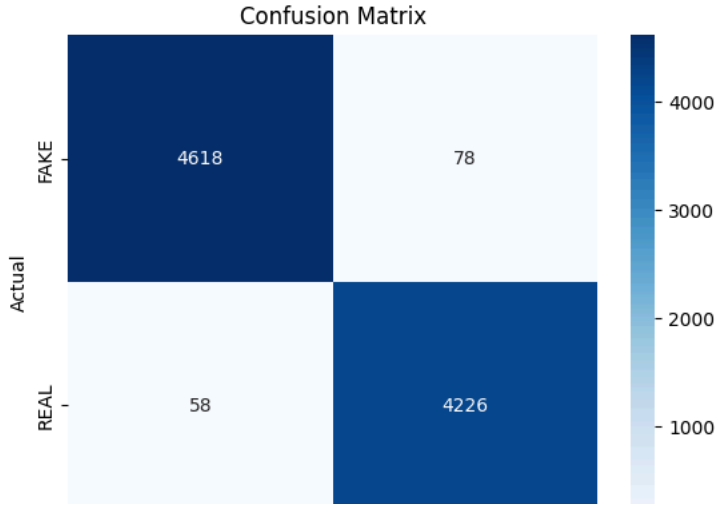
```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

# Load data
df = pd.read_csv('News.csv', encoding='utf-8')
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score
from sklearn.pipeline import Pipeline

# Load data
df = pd.read_csv('News.csv', encoding='utf-8')

# Ensure label is in binary form (0 = FAKE, 1 = REAL)
if 'label' in df.columns and df['label'].dtype == 'object':
    df['label'] = df['label'].map({'FAKE': 0, 'REAL': 1})
elif 'label_FAKE' in df.columns and 'label_REAL' in df.columns:
    df['label'] = df['label_REAL']  # Use one-hot if needed

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, random_state=42)

# Create ML pipeline
pipeline = Pipeline([
    ('tfidf', TfidfVectorizer(stop_words='english', max_df=0.7)),
    ('lr', LogisticRegression())
])

# Train model
pipeline.fit(X_train, y_train)

# Predict and evaluate
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
import seaborn as sns
import matplotlib.pyplot as plt

# Load data
df = pd.read_csv('News.csv', encoding='utf-8')
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, accuracy_score
```

```
from sklearn.pipeline import Pipeline

# Load data
df = pd.read_csv('News.csv', encoding='utf-8')

# Ensure label is in binary form (0 = FAKE, 1 = REAL)
if 'label' in df.columns and df['label'].dtype == 'object':
    df['label'] = df['label'].map({'FAKE': 0, 'REAL': 1})
elif 'label_FAKE' in df.columns and 'label_REAL' in df.columns:
    df['label'] = df['label_REAL']  # Use one-hot if needed

# Split dataset
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, random_state=42)

# Create ML pipeline
pipeline = Pipeline([
    ('tfidf', TfidfVectorizer(stop_words='english', max_df=0.7)),
    ('lr', LogisticRegression())
])

# Train model
pipeline.fit(X_train, y_train)

# Predict and evaluate
y_pred = pipeline.predict(X_test)
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))

# Split
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, random_state=42, stratify=df['label'])

# TF-IDF
tfidf = TfidfVectorizer(stop_words='english', max_df=0.7)
X_train_vec = tfidf.fit_transform(X_train)
X_test_vec = tfidf.transform(X_test)

# Model
model = LogisticRegression()
model.fit(X_train_vec, y_train)
y_pred = model.predict(X_test_vec)

# Evaluation
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))

# Confusion matrix
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['FAKE', 'REAL'], yticklabels=['FAKE', 'REAL'])
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Confusion Matrix')
plt.show()
X_train, X_test, y_train, y_test = train_test_split(df['text'], df['label'], test_size=0.2, random_state=42, stratify=df['label'])

# TF-IDF
tfidf = TfidfVectorizer(stop_words='english', max_df=0.7)
X_train_vec = tfidf.fit_transform(X_train)
X_test_vec = tfidf.transform(X_test)

# Model
model = LogisticRegression()
model.fit(X_train_vec, y_train)
y_pred = model.predict(X_test_vec)

# Evaluation
print("Accuracy:", accuracy_score(y_test, y_pred))
print("\nClassification Report:\n", classification_report(y_test, y_pred))

# Confusion matrix
cm = confusion_matrix(y_test, y_pred)
sns.heatmap(cm, annot=True, fmt='d', cmap='Blues', xticklabels=['FAKE', 'REAL'], yticklabels=['FAKE', 'REAL'])
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Confusion Matrix')
plt.show()
```

```
Accuracy: 0.9863028953229399

Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      4650
           1       0.98      0.99      0.99      4330

    accuracy                           0.99      8980
   macro avg       0.99      0.99      0.99      8980
weighted avg       0.99      0.99      0.99      8980

Accuracy: 0.9848552338530067

Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      4696
           1       0.98      0.99      0.98      4284

    accuracy                           0.98      8980
   macro avg       0.98      0.98      0.98      8980
weighted avg       0.98      0.98      0.98      8980
```




```
Accuracy: 0.9848552338530067

Classification Report:
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      4696
           1       0.98      0.99      0.98      4284

    accuracy                           0.98      8980
   macro avg       0.98      0.98      0.98      8980
weighted avg       0.98      0.98      0.98      8980
```



DEPLOYMENT:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score
import joblib

# Step 1: Load and prepare data
# Ensure the 'News.csv' file used here is the one generated after one-hot encoding.
df = pd.read_csv('News.csv', encoding='utf-8')

# Check if 'label_REAL' column exists before using it
if 'label_REAL' in df.columns:
    # Create the binary 'label' column from 'label_REAL'
    df['label'] = df['label_REAL'].apply(lambda x: 1 if x == 1 else 0)  # 1 = REAL, 0 = FAKE

    # Step 2: Split data
    # Assuming 'text' is your feature column and 'label' is your target column
    X = df['text']
    y = df['label']

    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42, stratify=y)

    # Step 3: Vectorization
    tfidf = TfidfVectorizer(stop_words='english', max_df=0.7)
    X_train_vec = tfidf.fit_transform(X_train)
    X_test_vec = tfidf.transform(X_test)

    # Step 4: Train model
    model = LogisticRegression()
    model.fit(X_train_vec, y_train)

    # Step 5: Save model and vectorizer
    joblib.dump(model, 'news_model.pkl')
    joblib.dump(tfidf, 'tfidf_vectorizer.pkl')

    # Step 6: Load and predict (simulate deployment)
    loaded_model = joblib.load('news_model.pkl')
    loaded_vectorizer = joblib.load('tfidf_vectorizer.pkl')
    sample_text = ["The government confirmed the new policy on climate change."]
    sample_vec = loaded_vectorizer.transform(sample_text)
    prediction = loaded_model.predict(sample_vec)

    # Output result
    print("Prediction (1 = REAL, 0 = FAKE):", prediction[0])
else:
    print("Error: 'label_REAL' column not found in the DataFrame. Please ensure 'News.csv' is correctly generated with one-hot encoding.")
```

```
Error: 'label_REAL' column not found in the DataFrame. Please ensure 'News.csv' is correctly generated with one-hot encoding.
```