FAKE NEWS PREDICTION

- The Fake News Prediction Dataset features both real and fake news, providing a basis for predictive modeling to identify misinformation. With columns including Title, Text, and Label (Fake or Real), it addresses the pervasive issue of false or misleading information in news.
- The dataset supports efforts to enhance information integrity, combat fake news, and promote media literacy.

In [165…
```
import pandas as pd
import numpy as np
# from wordcloud import WordCloud
import seaborn as sns
import matplotlib.pyplot as plt
# from textblob import TextBlob
```

In [166…
```
# Load dataset
df = pd.read_csv('news.csv')
df
```

Out[166]:

	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL
...	...	...	...	...
6330	4490	State Department says it can't find emails fro...	The State Department told the Republican Natio...	REAL
6331	8062	The 'P' in PBS Should Stand for 'Plutocratic' ...	The 'P' in PBS Should Stand for 'Plutocratic' ...	FAKE
6332	8622	Anti-Trump Protesters Are Tools of the Oligarc...	Anti-Trump Protesters Are Tools of the Oligar...	FAKE
6333	4021	In Ethiopia, Obama seeks progress on peace, se...	ADDIS ABABA, Ethiopia —President Obama convene...	REAL
6334	4330	Jeb Bush Is Suddenly Attacking Trump. Here's W...	Jeb Bush Is Suddenly Attacking Trump. Here's W...	REAL

6335 rows × 4 columns

In [167…
```
df.head()
```

Out[167]:

	Unnamed: 0	title	text	label
0	8476	You Can Smell Hillary's Fear	Daniel Greenfield, a Shillman Journalism Fello...	FAKE
1	10294	Watch The Exact Moment Paul Ryan Committed Pol...	Google Pinterest Digg Linkedin Reddit Stumbleu...	FAKE
2	3608	Kerry to go to Paris in gesture of sympathy	U.S. Secretary of State John F. Kerry said Mon...	REAL
3	10142	Bernie supporters on Twitter erupt in anger ag...	— Kaydee King (@KaydeeKing) November 9, 2016 T...	FAKE
4	875	The Battle of New York: Why This Primary Matters	It's primary day in New York and front-runners...	REAL

In [168…
```
print("Number of rows",df.shape[0])
print("Number of columns",df.shape[1])
```

```
Number of rows 6335
Number of columns 4
```

In [169…
```
df.columns
```

Out[169]:
```
Index(['Unnamed: 0', 'title', 'text', 'label'], dtype='object')
```

In [170…
```
df.isnull().sum()
```

Out[170]:
```
Unnamed: 0    0
title         0
text          0
label         0
dtype: int64
```

In [171…
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6335 entries, 0 to 6334
Data columns (total 4 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   Unnamed: 0  6335 non-null   int64 
 1   title       6335 non-null   object
 2   text        6335 non-null   object
 3   label       6335 non-null   object
dtypes: int64(1), object(3)
memory usage: 198.1+ KB
```

In [172…
```
df.describe()
```

Out[172]:

	Unnamed: 0
count	6335.000000
mean	5280.415627
std	3038.503953
min	2.000000
25%	2674.500000
50%	5271.000000
75%	7901.000000
max	10557.000000

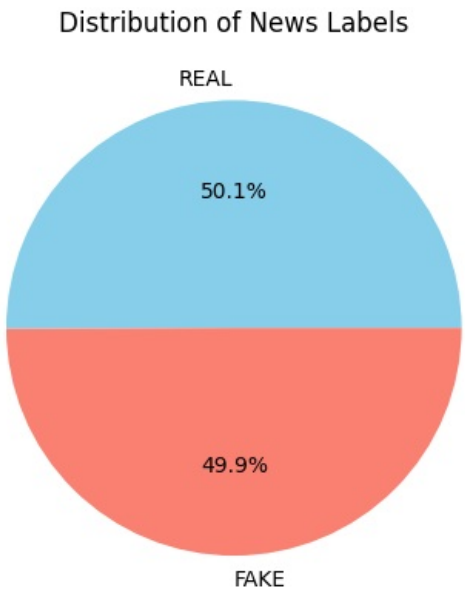
In [173…
```
df.dtypes
```

Out[173]:
```
Unnamed: 0     int64
title         object
text          object
label         object
dtype: object
```

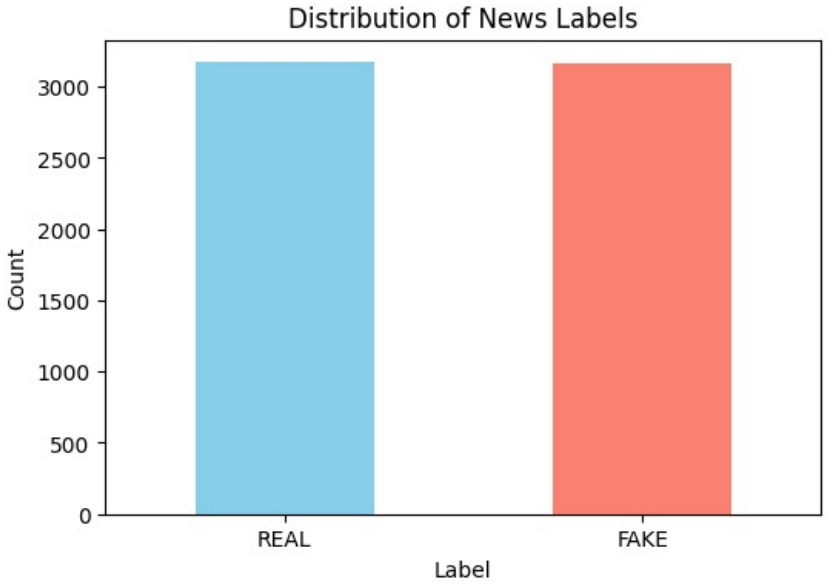
In [174…
```
# Value counts of labels
label_counts = df['label'].value_counts()
print(label_counts)
```

```
label
REAL    3171
FAKE    3164
Name: count, dtype: int64
```

In [175…
```
# Visualization of label count using a pie plot
label_counts.plot(kind='pie', autopct='%1.1f%%', colors=['skyblue', 'salmon'])
plt.title('Distribution of News Labels')
plt.ylabel('')
plt.show()
```



In [176…
```
# Plotting the distribution of labels using a bar plot
plt.figure(figsize=(6, 4))
df['label'].value_counts().plot(kind='bar', color=['skyblue', 'salmon'])
plt.title('Distribution of News Labels')
plt.xlabel('Label')
plt.ylabel('Count')
plt.xticks(rotation=0)
plt.show()
```

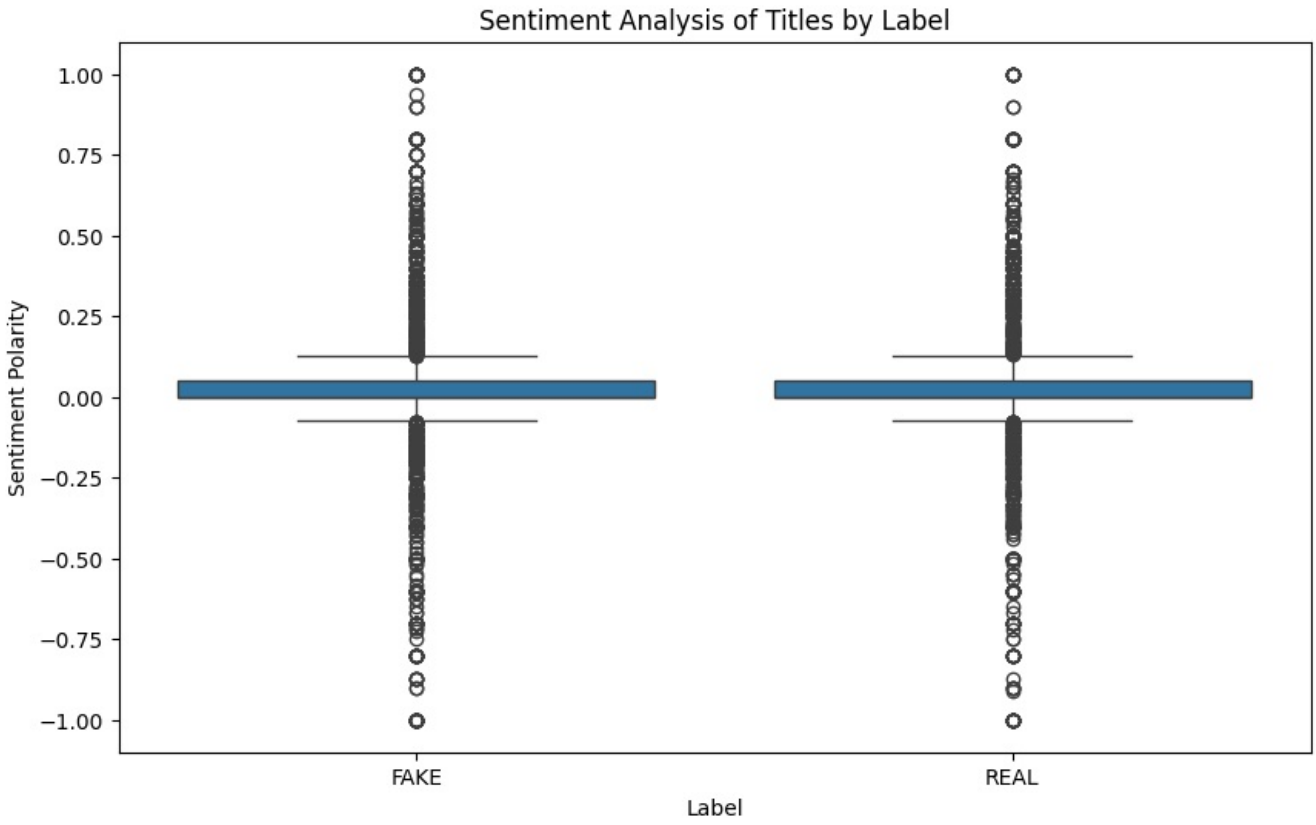


In [177…
```python
# Function to generate word cloud
def generate_wordcloud(text, title):
    wordcloud = WordCloud(width=800, height=400, background_color='white').generate(text)
    plt.figure(figsize=(10, 6))
    plt.imshow(wordcloud, interpolation='bilinear')
    plt.axis('off')
    plt.title(title)
    plt.show()

# Generate word clouds for fake and real news titles and text
fake_titles = ' '.join(df[df['label'] == 'Fake']['title'].values)
if fake_titles:
    generate_wordcloud(fake_titles, 'Word Cloud for Fake News Titles')
else:
    print("No fake news titles found.")

real_titles = ' '.join(df[df['label'] == 'Real']['title'].values)
if real_titles:
    generate_wordcloud(real_titles, 'Word Cloud for Real News Titles')
else:
    print("No real news titles found.")

fake_text = ' '.join(df[df['label'] == 'Fake']['text'].values)
if fake_text:
    generate_wordcloud(fake_text, 'Word Cloud for Fake News Text')
else:
    print("No fake news text found.")

real_text = ' '.join(df[df['label'] == 'Real']['text'].values)
if real_text:
    generate_wordcloud(real_text, 'Word Cloud for Real News Text')
else:
    print("No real news text found.")
```

```
No fake news titles found.
No real news titles found.
No fake news text found.
No real news text found.
```

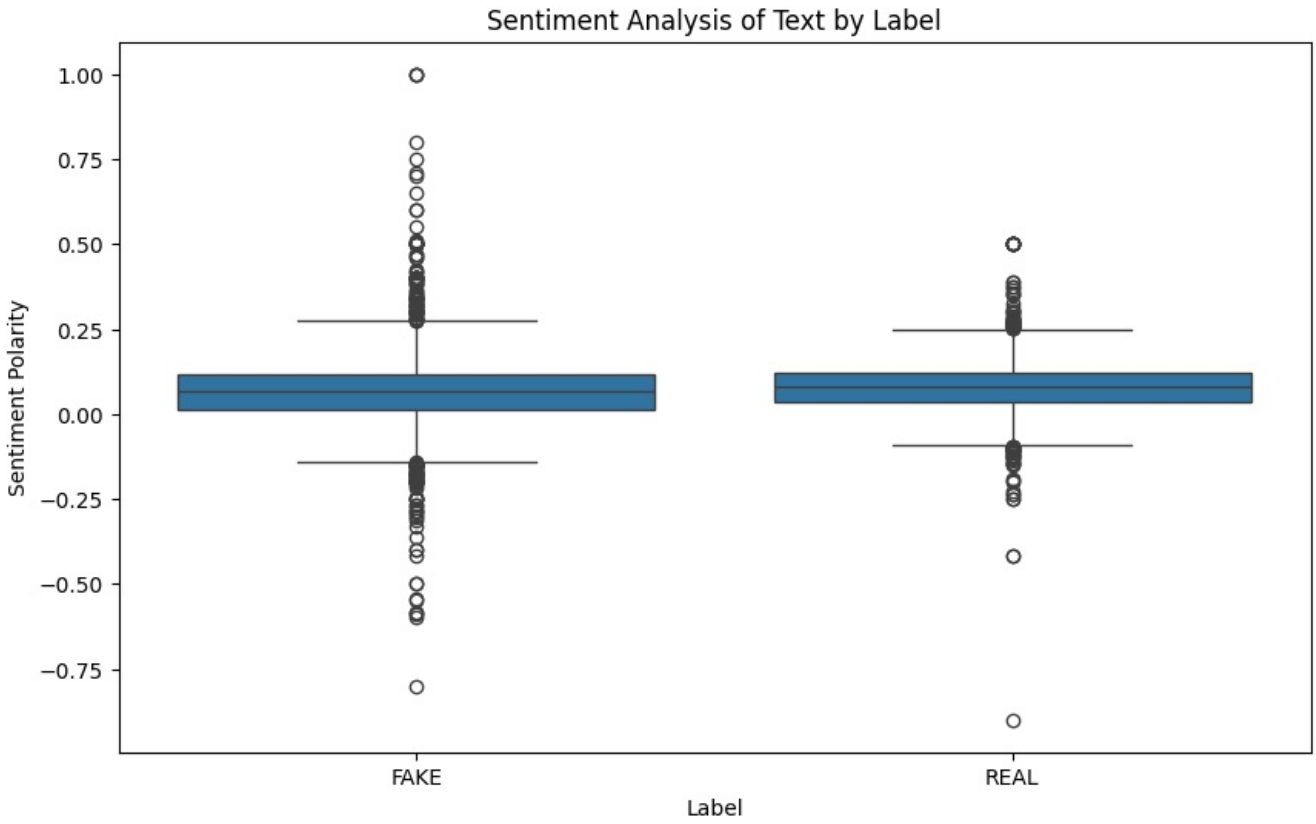
In [178…
```python
# Function to calculate sentiment polarity
def calculate_sentiment(text):
    blob = TextBlob(text)
    return blob.sentiment.polarity

# Apply sentiment analysis to titles and text
df['title_sentiment'] = df['title'].apply(calculate_sentiment)
df['text_sentiment'] = df['text'].apply(calculate_sentiment)
```

In [179…
```python
# Visualization of sentiment analysis
plt.figure(figsize=(10, 6))
sns.boxplot(x='label', y='title_sentiment', data=df)
plt.title('Sentiment Analysis of Titles by Label')
plt.xlabel('Label')
plt.ylabel('Sentiment Polarity')
plt.show()
```



In [180…

```
plt.figure(figsize=(10, 6))
sns.boxplot(x='label', y='text_sentiment', data=df)
plt.title('Sentiment Analysis of Text by Label')
plt.xlabel('Label')
plt.ylabel('Sentiment Polarity')
plt.show()
```



In [ ]:

In [ ]:

In [ ]: