NAME : Henjesh K O

USN : 1SV21CS035

6th SEM ‘A’sec

fake-news-detectection

June 27, 2024

[ ]:
```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

[ ]:
```
import pandas as pd
import numpy as nm
from sklearn.model_selection import train_test_split as ttp
from sklearn.metrics import classification_report  # Import from the correct␣
↪module
import re
import string
import matplotlib.pyplot as plt
```

[ ]:
```
data_true=pd.read_csv("/content/drive/MyDrive/fake_detection/True.csv")
data_fake=pd.read_csv("/content/drive/MyDrive/fake_detection/Fake.csv")
```

[ ]:
```
data_true.shape, data_fake.shape
```

[ ]:
```
((21417, 4), (23481, 4))
```

[ ]:
```

```

[ ]:
```

```

[ ]:
```
data_true_manual_testing = data_true.tail(10)
for i in range(21416,21416,-1):
    data_true.drop([i], axis=0, inplace=True)
data_fake_manual_testing = data_fake.tail(10)
for i in range(21416,21416,-1):
    data_fake.drop([i], axis=0, inplace=True)
```

[ ]:
```
data_manual_testing= pd.
↪concat([data_fake_manual_testing,data_true_manual_testing], axis=0)
data_manual_testing.to_csv("manual_testing.csv")
```

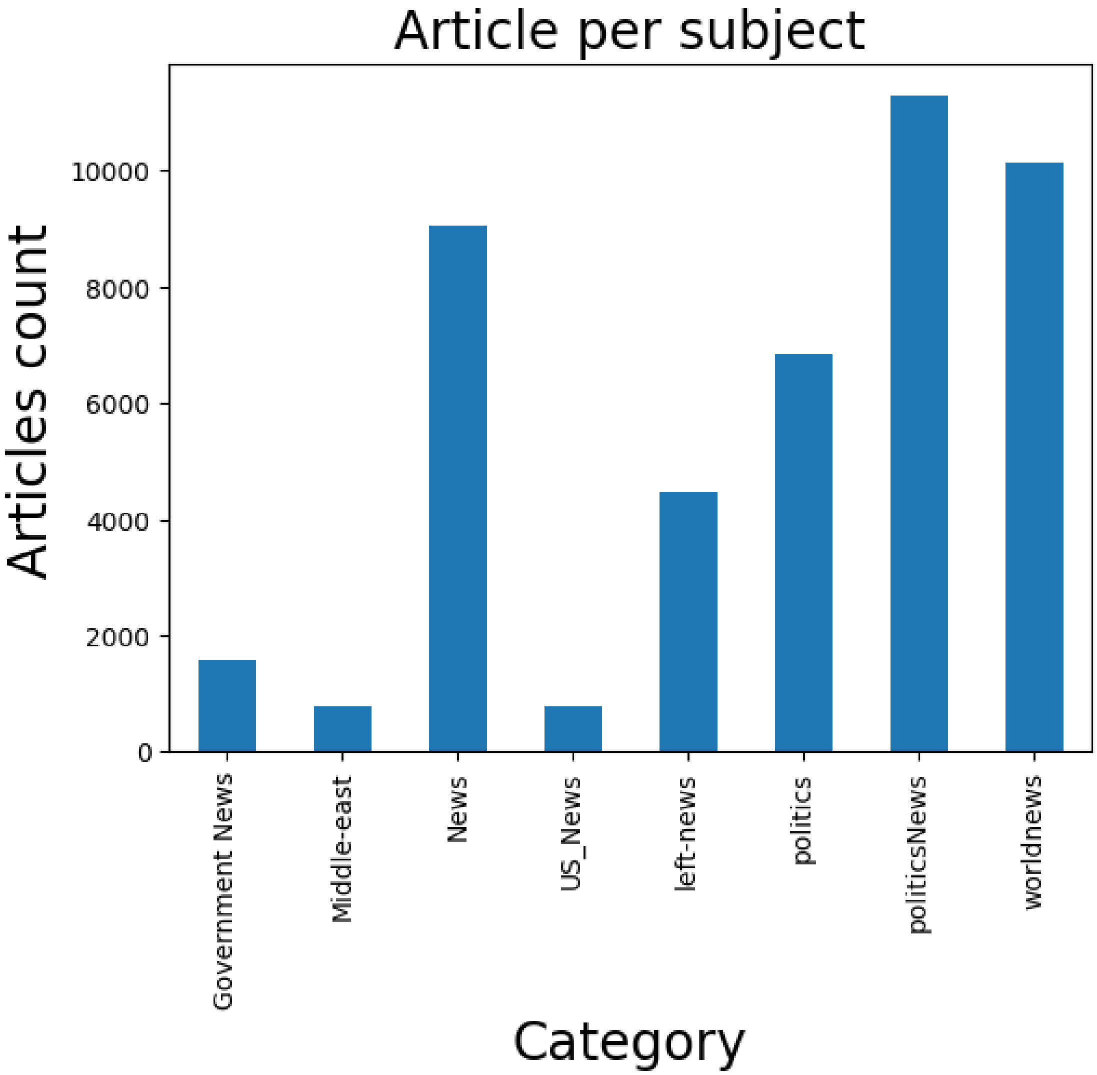
[ ]:
```
data_merge=pd.concat([data_true,data_fake], axis=0)
data_merge.head(10)
```

[ ]:
```
                                               title  \
0  As U.S. budget fight looms, Republicans flip t...
1  U.S. military to accept transgender recruits o...
2  Senior U.S. Republican senator: 'Let Mr. Muell...
3  FBI Russia probe helped by Australian diplomat...
4  Trump wants Postal Service to charge 'much mor...
5  White House, Congress prepare for talks on spe...
6  Trump says Russia probe will be fair, but time...
7  Factbox: Trump on Twitter (Dec 29) - Approval ...
8         Trump on Twitter (Dec 28) - Global Warming
9  Alabama official to certify Senator-elect Jone...

                                                text       subject  \
0  WASHINGTON (Reuters) - The head of a conservat...  politicsNews
1  WASHINGTON (Reuters) - Transgender people will...  politicsNews
2  WASHINGTON (Reuters) - The special counsel inv...  politicsNews
3  WASHINGTON (Reuters) - Trump campaign adviser ...  politicsNews
4  SEATTLE/WASHINGTON (Reuters) - President Donal...  politicsNews
5  WEST PALM BEACH, Fla./WASHINGTON (Reuters) - T...  politicsNews
6  WEST PALM BEACH, Fla (Reuters) - President Don...  politicsNews
7  The following statements were posted to the ve...  politicsNews
8  The following statements were posted to the ve...  politicsNews
9  WASHINGTON (Reuters) - Alabama Secretary of St...  politicsNews

                 date  class
0  December 31, 2017      1
1  December 29, 2017      1
2  December 31, 2017      1
3  December 30, 2017      1
4  December 29, 2017      1
5  December 29, 2017      1
6  December 29, 2017      1
7  December 29, 2017      1
8  December 29, 2017      1
9  December 28, 2017      1
```

[ ]:
```
print(data_merge.groupby(['subject'])['text'].count())
data_merge.groupby(['subject'])['text'].count().plot(kind="bar")
plt.title("Article per subject",size=20)
plt.xlabel("Category",size=20)
plt.ylabel(" Articles count",size=20)
plt.show()
```

```
subject
```

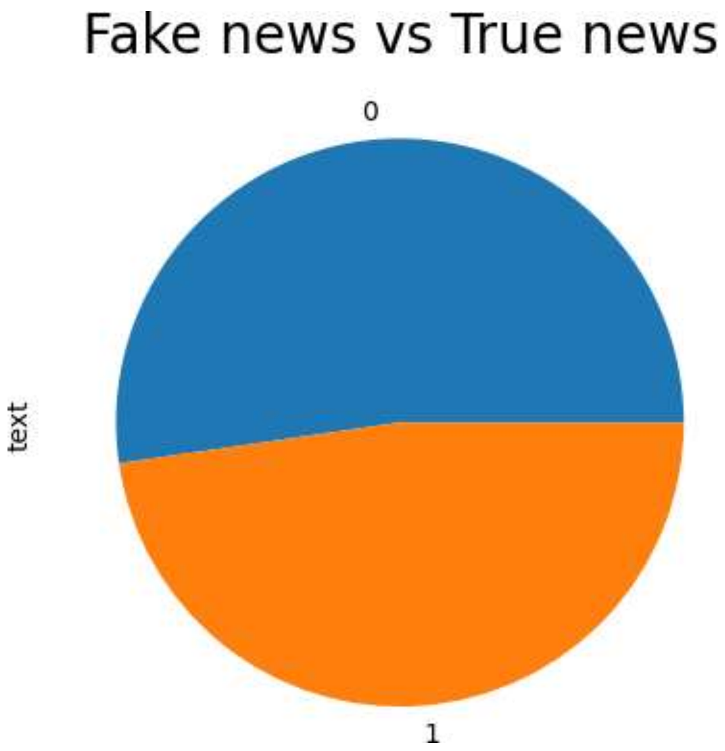
```
Government News     1570
Middle-east          778
News                9050
US_News              783
left-news           4459
politics            6841
politicsNews       11272
worldnews          10145
Name: text, dtype: int64
```



```
[ ]: print(data_merge.groupby(['class'])['text'].count())
     print("0 = Fake news\n1 = True news")
     data_merge.groupby(['class'])['text'].count().plot(kind="pie")
     plt.title("Fake news vs True news",size=20)
```

```python
plt.show()
```

```
class
0    23481
1    21417
Name: text, dtype: int64
0 = Fake news
1 = True news
```



[ ]:

```python
data = data_merge.drop(["title","subject","date"], axis=1)
data.head(10)
```

[ ]:

```
                                                text  class
0  WASHINGTON (Reuters) - The head of a conservat...      1
1  WASHINGTON (Reuters) - Transgender people will...      1
2  WASHINGTON (Reuters) - The special counsel inv...      1
3  WASHINGTON (Reuters) - Trump campaign adviser ...      1
4  SEATTLE/WASHINGTON (Reuters) - President Donal...      1
5  WEST PALM BEACH, Fla./WASHINGTON (Reuters) - T...      1
6  WEST PALM BEACH, Fla (Reuters) - President Don...      1
7  The following statements were posted to the ve...      1
8  The following statements were posted to the ve...      1
```

```
9   WASHINGTON (Reuters) - Alabama Secretary of St...       1
```

```
[ ]: data=data.sample(frac=1)
     data.head(10)
```

```
[ ]:                                                      text  class
     7865    You d think that in the year 2016, companies w...      0
     20022   OFFUTT AIR FORCE BASE, Neb. (Reuters) - The U...       1
     13480   Donald Trump, Jr was such a natural in his del...      0
     19739   A group of  deplorables   who are clearly sick ...     0
     17525   After Roy Moore s ugly loss in the Alabama Sen...      0
     18182   Speaking at a Rotary Club gathering in Kentuck...      0
     6325    Donald Trump might be desperately trying to wi...      0
     17916    (This Oct. 9 story has been refiled to add a ...      1
     17960   PBS host Judy Woodroof asked Hillary if she be...      0
     21262   SEOUL (Reuters) - North Korean leader Kim Jong...      1
```

```
[ ]: data.isnull().sum()
```

```
[ ]: text     0
     class    0
     dtype: int64
```

```
[ ]: def filtering(data):
       text=data.lower()
       text=re.sub('\[.*?\]', '', text)
       text=re.sub("\\W"," ",text)
       text=re.sub('https?://\S+|www\.\S+', '', text)
       text=re.sub('<.*?>+', '', text)
       text=re.sub('[%s]' % re.escape(string.punctuation), '', text)
       text=re.sub('\n', '', text)
       text=re.sub('\w*\d\w*', '', text)
       return text
```

```
[ ]: data["text"]=data['text'].apply(filtering)
     data.head(10)
```

```
[ ]:                                                      text  class
     7865     you d think that in the year    companies would...    0
     20022   offutt air force base   neb    reuters     the u ...    1
     13480   donald trump   jr was such a natural in his del...     0
     19739   a group of  deplorables   who are clearly sick ...     0
     17525   after roy moore s ugly loss in the alabama sen...      0
     18182   speaking at a rotary club gathering in kentuck...      0
     6325    donald trump might be desperately trying to wi...      0
     17916     this  oct     story has been refiled to add a d...     1
     17960   pbs host judy woodroof asked hillary if she be...      0
```

```
21262  seoul   reuters      north korean leader kim jong...       1
```

[ ]:
```
x=data["text"]
y=data["class"]
```

[ ]:
```
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

# Assuming 'x' and 'y' are defined as in your previous cells
x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2,␣
  ↪random_state=42)  # Split the data into training and testing sets

# Create a TfidfVectorizer object
vectorizer = TfidfVectorizer()

# Fit the vectorizer to the training data and transform both training and␣
  ↪testing data
x_train = vectorizer.fit_transform(x_train)
x_test = vectorizer.transform(x_test)

# Create and train a Logistic Regression model
model = LogisticRegression()  # Initialize the model
model.fit(x_train, y_train)  # Train the model

# Make predictions on the test set
y_pred = model.predict(x_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy:", accuracy)

# Print classification report for more detailed evaluation
print(classification_report(y_test, y_pred))
```

```
Accuracy: 0.9863028953229399
              precision    recall  f1-score   support

           0       0.99      0.98      0.99      4725
           1       0.98      0.99      0.99      4255

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

[19]:

```
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

# ... (rest of your code)

def predict_news(text):
    text_vectorized = vectorizer.transform([filtering(text)])   # Use␣
 ↪'vectorizer' instead of 'vectorization'
    prediction = model.predict(text_vectorized)  # Use 'model' instead of 'LR'
    if prediction == 1:
        return "This news is likely true."
    else:
        return "This news is likely fake."

user_input = input("Enter news text: ")
result = predict_news(user_input)
print(result)
```

```
Enter news text: modhi died
This news is likely fake.
```

[20]:

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, classification_report

# Load the dataset
data = pd.read_csv("/content/drive/MyDrive/fake_detection/manual_testing.csv")

# Preprocess the data
x = data['text']
y = data['class']

# Vectorize the text data
vectorizer = TfidfVectorizer(max_features=1000)
x_vectorized = vectorizer.fit_transform(x)

# Split the data into training and testing sets
x_train, x_test, y_train, y_test = train_test_split(x_vectorized, y,␣
 ↪test_size=0.2, random_state=42)

# Train Decision Tree model
model = DecisionTreeClassifier()
```

```
model.fit(x_train, y_train)
y_pred = model.predict(x_test)

# Evaluate the model
accuracy = accuracy_score(y_test, y_pred)
report = classification_report(y_test, y_pred)

print("Accuracy:", accuracy)
print("Classification Report:\n", report)

# Function to get user input and predict output
def get_user_input():
    user_input = input("Enter news text: ")
    user_input_vectorized = vectorizer.transform([user_input])
    return user_input_vectorized

# Get user input and predict
user_input_vectorized = get_user_input()
user_prediction = model.predict(user_input_vectorized)

print("Prediction:", "Fake news" if user_prediction[0] == 0 else "True news")
```

```
Accuracy: 1.0
Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00         2
           1       1.00      1.00      1.00         4

    accuracy                           1.00         6
   macro avg       1.00      1.00      1.00         6
weighted avg       1.00      1.00      1.00         6

Enter news text: india is a country
Prediction: True news
```

[2

```
import pandas as pd
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier  # Import␣
  ↪RandomForestClassifier
from sklearn import metrics
data = pd.read_csv("/content/drive/MyDrive/fake_detection/manual_testing.csv")

x=data[‘text’]

y=data[‘class’]
```

```python
# Vectorize the text data
vectorizer = TfidfVectorizer(max_features=1000)
x_vectorized = vectorizer.fit_transform(x)

# Split the data into training and testing sets
x_train, x_test, y_train, y_test = train_test_split(x_vectorized, y, test_size=0.2, random_state=42)

# Train Random Forest model
model = RandomForestClassifier()  #  Initialize  RandomForestClassifier
model.fit(x_train, y_train)
y_pred = model.predict(x_test)

# Evaluate the model
accuracy = metrics.accuracy_score(y_test, y_pred)
report = metrics.classification_report(y_test, y_pred)

print("Accuracy:", accuracy)
print("Classification Report:\n", report)

# Function to get user input and predict output
def get_user_input():
    user_input = input("Enter news text: ")
    user_input_vectorized = vectorizer.transform([user_input])
    return user_input_vectorized

# Get user input and predict
user_input_vectorized = get_user_input()
user_prediction = model.predict(user_input_vectorized)

print("Prediction:", "Fake news" if user_prediction[0] == 0 else "True news")
```

```
Accuracy: 1.0
Classification Report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00         2
           1       1.00      1.00      1.00         4

    accuracy                           1.00         6
   macro avg       1.00      1.00      1.00         6
weighted avg       1.00      1.00      1.00         6

Enter news text: india is country
```

```
Prediction: True news
```