```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')
```

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

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

```
dff = pd.read_csv("/content/drive/MyDrive/Fake.csv")
dft = pd.read_csv("/content/drive/MyDrive/True.csv")
```

```
dff.head()
```

	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

```
dft.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

```
dff['class']=0
dft['class']=1
df=pd.concat([dff,dft],axis=0)
df
```

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

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Index: 44898 entries, 0 to 21416
Data columns (total 5 columns):
 #   Column   Non-Null Count  Dtype
---  ------   --------------  -----
 0   title    44898 non-null  object
 1   text     44898 non-null  object
 2   subject  44898 non-null  object
 3   date     44898 non-null  object
 4   class    44898 non-null  int64
dtypes: int64(1), object(4)
memory usage: 2.1+ MB
```

```
df.isnull().sum()
```

	0
title	0
text	0
subject	0
date	0
class	0

dtype: int64

```
df['content']=df['title']+df['text']
df['content'].astype('str')
df.drop(['title','text'],axis=1)
```

	subject	date	class	content
0	News	December 31, 2017	0	Donald Trump Sends Out Embarrassing New Year'...
1	News	December 31, 2017	0	Drunk Bragging Trump Staffer Started Russian ...
2	News	December 30, 2017	0	Sheriff David Clarke Becomes An Internet Joke...
3	News	December 29, 2017	0	Trump Is So Obsessed He Even Has Obama's Name...
4	News	December 25, 2017	0	Pope Francis Just Called Out Donald Trump Dur...
...	...	...	...	...
21412	worldnews	August 22, 2017	1	'Fully committed' NATO backs new U.S. approach...
21413	worldnews	August 22, 2017	1	LexisNexis withdrew two products from Chinese ...
21414	worldnews	August 22, 2017	1	Minsk cultural hub becomes haven from authorit...
21415	worldnews	August 22, 2017	1	Vatican upbeat on possibility of Pope Francis ...
21416	worldnews	August 22, 2017	1	Indonesia to buy $1.14 billion worth of Russia...

44898 rows × 4 columns

```
df.head()
```

	title	text	subject	date	class	content
0	Donald Trump Sends Out Embarrassing New Year'...	Donald Trump just couldn t wish all Americans ...	News	December 31, 2017	0	Donald Trump Sends Out Embarrassing New Year'...
1	Drunk Bragging Trump Staffer Started Russian ...	House Intelligence Committee Chairman Devin Nu...	News	December 31, 2017	0	Drunk Bragging Trump Staffer Started Russian ...
2	Sheriff David Clarke Becomes An Internet Joke...	On Friday, it was revealed that former Milwauk...	News	December 30, 2017	0	Sheriff David Clarke Becomes An Internet Joke...

```
import re
import string
```

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

```
df['content'].astype('str')
df['content'] = df['content'].apply(wordopt)
```

```
x=df['content']
y=df['class']
```

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.25,random_state=42)
```

```
from sklearn.feature_extraction.text import TfidfVectorizer
vectorization = TfidfVectorizer()
xv_train = vectorization.fit_transform(x_train)
xv_test = vectorization.transform(x_test)
```

```
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report,f1_score
from sklearn.ensemble import RandomForestClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import MultinomialNB
from sklearn.ensemble import GradientBoostingClassifier
```

```
lr=LogisticRegression()
lr.fit(xv_train,y_train)
pred=lr.predict(xv_test)
print("Logistic Regression")
print("accuracy:", accuracy_score(y_test,pred))
print("confusion_matrix:\n", confusion_matrix(y_test,pred))
print("classification_report:\n", classification_report(y_test,pred))
print('\n')
```

```
Logistic Regression
accuracy: 0.9870824053452116
confusion_matrix:
 [[5815   80]
 [  65 5265]]
classification_report:
               precision    recall  f1-score   support

           0       0.99      0.99      0.99      5895
           1       0.99      0.99      0.99      5330

    accuracy                           0.99     11225
   macro avg       0.99      0.99      0.99     11225
weighted avg       0.99      0.99      0.99     11225
```

Double-click (or enter) to edit

```
rf=RandomForestClassifier(random_state=42)
rf.fit(xv_train,y_train)
pred=rf.predict(xv_test)
print("Random Forest Classifier")
print("accuracy:", accuracy_score(y_test,pred))
print("confusion_matrix:\n", confusion_matrix(y_test,pred))
print("classification_report:\n", classification_report(y_test,pred))
print('\n')
```

```
Random Forest Classifier
accuracy: 0.9893095768374165
confusion_matrix:
 [[5851   44]
 [  76 5254]]
classification_report:
               precision    recall  f1-score   support

           0       0.99      0.99      0.99      5895
           1       0.99      0.99      0.99      5330

    accuracy                           0.99     11225
   macro avg       0.99      0.99      0.99     11225
weighted avg       0.99      0.99      0.99     11225
```

```
dt=DecisionTreeClassifier()
dt.fit(xv_train,y_train)
pred=dt.predict(xv_test)
print(" Decision Tree Classifier")
print("accuracy:", accuracy_score(y_test,pred))
print("confusion_matrix:\n", confusion_matrix(y_test,pred))
print("classification_report:\n", classification_report(y_test,pred))
print('\n')
```

```
 Decision Tree Classifier
accuracy: 0.995456570155902
confusion_matrix:
 [[5868   27]
 [  24 5306]]
classification_report:
               precision    recall  f1-score   support

           0       1.00      1.00      1.00      5895
           1       0.99      1.00      1.00      5330

    accuracy                           1.00     11225
   macro avg       1.00      1.00      1.00     11225
weighted avg       1.00      1.00      1.00     11225
```

```
nb= MultinomialNB()
nb.fit(xv_train,y_train)
pred=nb.predict(xv_test)
print("multinomial naive bayes")
print("accuracy:", accuracy_score(y_test,pred))
print("confusion_matrix:\n", confusion_matrix(y_test,pred))
print("classification_report:\n", classification_report(y_test,pred))
print('\n')
```

```
multinomial naive bayes
accuracy: 0.9355011135857461
confusion_matrix:
 [[5630  265]
 [ 459 4871]]
classification_report:
               precision    recall  f1-score   support

           0       0.92      0.96      0.94      5895
           1       0.95      0.91      0.93      5330

    accuracy                           0.94     11225
   macro avg       0.94      0.93      0.94     11225
weighted avg       0.94      0.94      0.94     11225
```

```
gb=GradientBoostingClassifier(
    random_state=0
)
gb.fit(xv_train,y_train)
pred=gb.predict(xv_test)
print("Gradient boosting")
print("accuracy:", accuracy_score(y_test,pred))
print("confusion_matrix:\n", confusion_matrix(y_test,pred))
print("classification_report:\n", classification_report(y_test,pred))
print('\n')
```

```
Gradient boosting
accuracy: 0.9952783964365256
confusion_matrix:
 [[5857   38]
 [  15 5315]]
classification_report:
               precision    recall  f1-score   support

           0       1.00      0.99      1.00      5895
           1       0.99      1.00      1.00      5330

    accuracy                           1.00     11225
   macro avg       1.00      1.00      1.00     11225
weighted avg       1.00      1.00      1.00     11225
```

```
def output_lable(n):
    if n==0:
        return "Fake"
    elif n==1:
        return "Real"

def manual_testing(news, models):
    testing_news = {"content": [news]}
    new_def_test = pd.DataFrame(testing_news)
    new_def_test["content"] = new_def_test["content"].apply(wordopt)
    new_x_test = new_def_test["content"]
    new_xv_test = vectorization.transform(new_x_test)

    for model_name, model in models.items():
        pred = model.predict(new_xv_test)
        pred_label = output_lable(pred[0])
        print(f"{model_name}: {pred_label}")
```

```
news = str(input())
models = {
    "Logistic Regression": lr,
    "Random Forest Classifier": rf,
    "Decision Tree Classifier": dt,
    "Multinomial Naive Bayes": nb,
    "Gradient Boosting": gb
}
manual_testing(news, models)
```

```
pradeep
Logistic Regression: Fake
Random Forest Classifier: Fake
Decision Tree Classifier: Fake
Multinomial Naive Bayes: Real
Gradient Boosting: Fake
```

```
a
```

	Text	label
0	Top Trump Surrogate BRUTALLY Stabs Him In The...	Fake
1	U.S. conservative leader optimistic of common ...	Real
2	Trump proposes U.S. tax overhaul, stirs concer...	Real
3	Court Forces Ohio To Allow Millions Of Illega...	Fake
4	Democrats say Trump agrees to work on immigrat...	Real
...	...	...
9895	Wikileaks Admits To Screwing Up IMMENSELY Wit...	Fake
9896	Trump consults Republican senators on Fed chie...	Real
9897	Trump lawyers say judge lacks jurisdiction for...	Real
9898	WATCH: Right-Wing Pastor Falsely Credits Trum...	Fake
9899	Sean Spicer HILARIOUSLY Branded As Chickensh*...	Fake

9900 rows × 2 columns

```
x=a['Text']
y=a['label']
```

```
x.apply(wordopt)
```

Text

```
Start coding or generate with AI.
```

1 u s conservative leader optimistic of common ...

```
from sklearn.feature_extraction.text import TfidfVectorizer

vectorization = TfidfVectorizer(max_features=5000)
x_test = vectorization.fit_transform(x)
```

```
pred=lr.predict(x_test)
print("Logistic Regression")
print("accuracy:", accuracy_score(y,pred))
```

9896 trump consults republican senators on fed chie...

9897 trump lawyers say judge lacks jurisdiction for...

9898 watch right wing pastor falsely credits trum...

9899 sean spicer hilariously branded as chickensh ...

9900 rows × 1 columns

dtype: object

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
/tmp/ipython-input-2802528584.py in <cell line: 0>()
----> 1 pred=lr.predict(x_test)
      2 print("Logistic Regression")
      3 print("accuracy:", accuracy_score(y,pred))

3 frames

/usr/local/lib/python3.12/dist-packages/sklearn/utils/validation.py in _check_n_features(estimator, X, reset)
   2827
   2828     if n_features != estimator.n_features_in_:
-> 2829         raise ValueError(
   2830             f"X has {n_features} features, but {estimator.__class__.__name__} "
   2831             f"is expecting {estimator.n_features_in_} features as input."

ValueError: X has 5000 features, but LogisticRegression is expecting 109505 features as input.
```

```
pred=rf.predict(xv_test)
print("Random Forest Classifier")
print("accuracy:", accuracy_score(y_test,pred))
```

```
Random Forest Classifier
accuracy: 0.9951515151515151
```

```
pred=dt.predict(xv_test)
print(" Decision Tree Classifier")
print("accuracy:", accuracy_score(y_test,pred))
```

```
 Decision Tree Classifier
accuracy: 0.9987878787878788
```