```
import pandas as pd
```

```
news_data=pd.read_csv('FakeNewsNet.csv')
news_data_new=pd.read_csv('FakeNewsNet.csv')
```

```
news_data
```

	title	news_url	source_domain	tweet_num	real
0	Kandi Burruss Explodes Over Rape Accusation on...	http://toofab.com/2017/05/08/real-housewives-a...	toofab.com	42.0	1.0
1	People's Choice Awards 2018: The best red carp...	https://www.today.com/style/see-people-s-choic...	www.today.com	0.0	1.0
2	Sophia Bush Sends Sweet Birthday Message to 'O...	https://www.etonline.com/news/220806_sophia_bu...	www.etonline.com	63.0	1.0
3	Colombian singer Maluma sparks rumours of inap...	https://www.dailymail.co.uk/news/article-33655...	www.dailymail.co.uk	20.0	1.0
4	Gossip Girl 10 Years Later: How Upper East Sid...	https://www.zerchoo.com/entertainment/gossip-g...	www.zerchoo.com	38.0	1.0
...	...	...	...	...	...
5620	Queen Mathilde of Belgium	https://en.wikipedia.org/wiki/Queen_Mathilde_o...	en.wikipedia.org	57.0	1.0
5621	Megan Keeps Having Terrible Flashbacks to Her ...	https://www.longroom.com/discussion/918151/meg...	www.longroom.com	15.0	1.0
5622	Debbie Reynolds Net Worth	www.bankrate.com/lifestyle/celebrity-money/deb...	www.bankrate.com	68.0	0.0
5623	Russell Simmons denies rape claim in new $10 m...	https://www.usatoday.com/story/life/people/201...	www.usatoday.com	107.0	1.0
5624	Naya Rivera's Ex Rya	NaN	NaN	NaN	NaN

5625 rows × 5 columns

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

```
news_data.dropna(inplace=True)
news_data.reset_index(inplace=True)
'''news_data.drop(["index","news_url","source_domain","tweet_num"],axis=1,inplace=True)'''
news_data.head()
```

	index	title	news_url	source_domain	tweet_num	real
0	0	Kandi Burruss Explodes Over Rape Accusation on...	http://toofab.com/2017/05/08/real-housewives-a...	toofab.com	42.0	1.(
1	1	People's Choice Awards 2018: The best red carp...	https://www.today.com/style/see-people-s-choic...	www.today.com	0.0	1.(
2	2	Sophia Bush Sends Sweet Birthday Message to 'O...	https://www.etonline.com/news/220806_sophia_bu...	www.etonline.com	63.0	1.(
3	3	Colombian singer Maluma sparks rumours of inap...	https://www.dailymail.co.uk/news/article-33655...	www.dailymail.co.uk	20.0	1.(
4	4	Gossip Girl 10 Years Later: How Upper East Sid...	https://www.zerchoo.com/entertainment/gossip-g...	www.zerchoo.com	38.0	1.(

```
import nltk
nltk.download('wordnet')
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
True
```

```
nltk.download('stopwords')
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
True
```

```
stopwords = nltk.corpus.stopwords.words("english")
```

```
import nltk
nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
True
```

```
import re


lemmatizer = nltk.stem.WordNetLemmatizer()

def lemTitles(title):
    words = nltk.word_tokenize(title)
    words = [re.sub("[^a-zA-Z0-9]","",i).lower().strip() for i in words]
    words = [lemmatizer.lemmatize(i) for i in words if i not in stopwords]
    title = " ".join(words)
    return title

news_data["title"] = news_data["title"].apply(lemTitles)
news_data.head()
```

	index	title	news_url	source_domain	tweet_num	real
0	0	kandi burruss explodes rape accusation real ho...	http://toofab.com/2017/05/08/real-housewives-a...	toofab.com	42.0	1.0
1	1	people choice award 2018 best red carpet look	https://www.today.com/style/see-people-s-choic...	www.today.com	0.0	1.0
2	2	sophia bush sends sweet birthday message one t...	https://www.etonline.com/news/220806_sophia_bu...	www.etonline.com	63.0	1.0
3	3	colombian singer maluma spark rumour inappropr...	https://www.dailymail.co.uk/news/article-33655...	www.dailymail.co.uk	20.0	1.0
4	4	gossip girl 10 year later upper east siders s...	https://www.zerchoo.com/entertainment/gossip-g...	www.zerchoo.com	38.0	1.0

```
VOCAB_SIZE = 10000
DIMENSION = 100
MAXLEN = 20
```

```
import tensorflow as tf
from tensorflow.keras.preprocessing.text import Tokenizer
```

```
def oneHot(title):
    return tf.keras.preprocessing.text.one_hot(title,VOCAB_SIZE)

news_data["title"] = news_data["title"].apply(oneHot)
```

```
news_data.head()
```

	index	title	news_url	source_domain	tweet_num	real
0	0	[4924, 807, 11, 6132, 57, 4382, 1804, 8423, 32...	http://toofab.com/2017/05/08/real-housewives-a...	toofab.com	42.0	1.0
1	1	[2720, 829, 3916, 3166, 5464, 6439, 8177, 7910]	https://www.today.com/style/see-people-s-choic...	www.today.com	0.0	1.0
2	2	[3424, 5765, 266, 5662, 4977, 9734, 1818, 3825...	https://www.etonline.com/news/220806_sophia_bu...	www.etonline.com	63.0	1.0
3	3	[2810, 3330, 4254, 7571, 4225, 6385, 8158, 8268]	https://www.dailymail.co.uk/news/article-33655...	www.dailymail.co.uk	20.0	1.0
4	4	[7740, 5039, 1932, 9645, 2232, 7404, 8346, 218...	https://www.zerchoo.com/entertainment/gossip-g...	www.zerchoo.com	38.0	1.0

```
df = pd.DataFrame(tf.keras.utils.pad_sequences(news_data["title"],padding="pre",maxlen=MAXLEN))
df.shape
```

```
(5542, 20)
```

```
news_data = pd.concat((news_data,df),axis=1)
news_data.shape
```

```
(5542, 26)
```

```
import numpy as np
```

```
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score,roc_auc_score,precision_score, confusion_matrix

# cols = [i for i in news_data.columns if (i!="index" and i!="title" and i!="real")]
cols = [i for i in range(0,20)]
seed = np.random.seed(6)
#cols.append(news_data['source_domain'])

X = news_data[cols]
y = news_data[["real",'source_domain']]

X_train, X_test, y_train, y_test = train_test_split(X,y, test_size=0.25,random_state=seed)
```

```
X_test, y_test
```

```
(      0  1  2  3  4  5  6  7  8  9  10  11    12    13    14    15    16  \
 4771  0  0  0  0  0  0  0  0  0  0   0   0     0     0     0     0     0
 5431  0  0  0  0  0  0  0  0  0  0   0   0  7460  1962  8038  3304  5464
 4397  0  0  0  0  0  0  0  0  0  0   0   0     0  8879  8118  3245  1728
 2951  0  0  0  0  0  0  0  0  0  0   0   0     0  3069  3451  8167  7827
 3395  0  0  0  0  0  0  0  0  0  0   0   0  3156  9041  1604  3952  3297
 ...  .. .. .. .. .. .. .. .. .. ..  ..  ..   ...   ...   ...   ...   ...
 4608  0  0  0  0  0  0  0  0  0  0   0   0  8343  4991  9598  8646   649
 1071  0  0  0  0  0  0  0  0  0  0   0   0     0  9946   460  5520  2044
 5326  0  0  0  0  0  0  0  0  0  0   0   0     0  5623  9411  4184  2270
 4688  0  0  0  0  0  0  0  0  0  0   0   0     0  8636   747  3304  3795
 5292  0  0  0  0  0  0  0  0  0  0   0   0     0     0     0  1572  4991

         17    18    19
 4771   830  7932  6244
 5431  5953  4528  9229
 4397  8984  9260  9408
 2951  7947  4265  3310
 3395  9489   890  7944
 ...    ...   ...   ...
 4608  7854  4111  6332
 1071    56  7571  9296
 5326  6176  2867  1615
 4688  1637  7479  7963
 5292  3499  3785  7257

 [1386 rows x 20 columns],
       real        source_domain
 4771   1.0    www.livestrong.com
 5431   0.0     hollywoodlife.com
 4397   0.0          www.imdb.com
 2951   1.0     www.teenvogue.com
 3395   1.0      www.usatoday.com
 ...    ...                   ...
 4608   1.0      www.broadway.com
 1071   1.0   www.dailymail.co.uk
 5326   1.0           www.upi.com
 4688   1.0      www.etonline.com
 5292   0.0   www.dailymail.co.uk

 [1386 rows x 2 columns])
```

```
X_new=news_data['source_domain']
Y_new=news_data["title"]

new_x,new_xt,new_y,new_yt= train_test_split(X_new,Y_new,test_size=0.25,random_state=seed)
```

```
new_xt
```

```
2494          www.zerchoo.com
3078           www.thecut.com
4453    www.intouchweekly.com
3223        hollywoodlife.com
1222                   ew.com
                ...
4950           tvthisweek.us
173         hollywoodlife.com
836           www.cbsnews.com
4167          en.wikipedia.org
2780      www.dailymail.co.uk
Name: source_domain, Length: 1386, dtype: object
```

```
model = tf.keras.models.Sequential()
model.add(tf.keras.layers.Embedding(VOCAB_SIZE,DIMENSION,input_length=len(cols)))
model.add(tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(128,return_sequences=True)))
model.add(tf.keras.layers.Dropout(0.2))
model.add(tf.keras.layers.Bidirectional(tf.keras.layers.LSTM(128)))
model.add(tf.keras.layers.Dropout(0.2))
model.add(tf.keras.layers.Dense(1,activation="sigmoid"))
model.compile(optimizer="adam",loss="binary_crossentropy",metrics=["accuracy"])
model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 embedding (Embedding)       (None, 20, 100)           1000000

 bidirectional (Bidirection  (None, 20, 256)           234496
 al)
 dropout (Dropout)           (None, 20, 256)           0

 bidirectional_1 (Bidirecti  (None, 256)               394240
 onal)

 dropout_1 (Dropout)         (None, 256)               0

 dense (Dense)               (None, 1)                 257

=================================================================
Total params: 1628993 (6.21 MB)
Trainable params: 1628993 (6.21 MB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

```
Y_new=y_train['source_domain']
Yt_new=y_test['source_domain']
y_train.drop(['source_domain'],axis=1,inplace=True)
y_test.drop(['source_domain'],axis=1,inplace=True)
#Y_new
```

```
earlyStopping = tf.keras.callbacks.EarlyStopping(monitor="val_accuracy",patience=3,start_from_epoch=5,restore_best_weights=True,)
```

```
history = model.fit(X_train,y_train,validation_data=(X_test,y_test),epochs=20,batch_size=64)
```

```
Epoch 1/20
65/65 [==============================] - 33s 322ms/step - loss: 0.5311 - accuracy: 0.7709 - val_loss: 0.4304 - val_accuracy: 0.8153
Epoch 2/20
65/65 [==============================] - 19s 289ms/step - loss: 0.3303 - accuracy: 0.8607 - val_loss: 0.4349 - val_accuracy: 0.8139
Epoch 3/20
65/65 [==============================] - 18s 283ms/step - loss: 0.2061 - accuracy: 0.9177 - val_loss: 0.5263 - val_accuracy: 0.8074
Epoch 4/20
65/65 [==============================] - 18s 282ms/step - loss: 0.1309 - accuracy: 0.9528 - val_loss: 0.6814 - val_accuracy: 0.7843
Epoch 5/20
65/65 [==============================] - 18s 284ms/step - loss: 0.0722 - accuracy: 0.9735 - val_loss: 0.8013 - val_accuracy: 0.7893
Epoch 6/20
65/65 [==============================] - 19s 293ms/step - loss: 0.0538 - accuracy: 0.9800 - val_loss: 1.0341 - val_accuracy: 0.7698
Epoch 7/20
65/65 [==============================] - 18s 279ms/step - loss: 0.0380 - accuracy: 0.9875 - val_loss: 1.1108 - val_accuracy: 0.7734
Epoch 8/20
65/65 [==============================] - 20s 312ms/step - loss: 0.0217 - accuracy: 0.9928 - val_loss: 1.1788 - val_accuracy: 0.7734
Epoch 9/20
65/65 [==============================] - 18s 273ms/step - loss: 0.0286 - accuracy: 0.9909 - val_loss: 1.0192 - val_accuracy: 0.7749
Epoch 10/20
65/65 [==============================] - 18s 271ms/step - loss: 0.0311 - accuracy: 0.9904 - val_loss: 1.2816 - val_accuracy: 0.7706
Epoch 11/20
65/65 [==============================] - 17s 264ms/step - loss: 0.0151 - accuracy: 0.9949 - val_loss: 1.4690 - val_accuracy: 0.7583
Epoch 12/20
65/65 [==============================] - 17s 265ms/step - loss: 0.0126 - accuracy: 0.9959 - val_loss: 1.3273 - val_accuracy: 0.7756
Epoch 13/20
65/65 [==============================] - 18s 275ms/step - loss: 0.0132 - accuracy: 0.9949 - val_loss: 1.2387 - val_accuracy: 0.7799
Epoch 14/20
65/65 [==============================] - 16s 248ms/step - loss: 0.0130 - accuracy: 0.9952 - val_loss: 1.6914 - val_accuracy: 0.7662
Epoch 15/20
65/65 [==============================] - 16s 245ms/step - loss: 0.0148 - accuracy: 0.9945 - val_loss: 1.6187 - val_accuracy: 0.7677
Epoch 16/20
65/65 [==============================] - 18s 275ms/step - loss: 0.0194 - accuracy: 0.9930 - val_loss: 1.5023 - val_accuracy: 0.7511
Epoch 17/20
65/65 [==============================] - 17s 255ms/step - loss: 0.0153 - accuracy: 0.9935 - val_loss: 1.5201 - val_accuracy: 0.7691
Epoch 18/20
65/65 [==============================] - 16s 251ms/step - loss: 0.0163 - accuracy: 0.9928 - val_loss: 1.5185 - val_accuracy: 0.7439
Epoch 19/20
65/65 [==============================] - 16s 252ms/step - loss: 0.0178 - accuracy: 0.9935 - val_loss: 1.6099 - val_accuracy: 0.7525
Epoch 20/20
65/65 [==============================] - 17s 268ms/step - loss: 0.0112 - accuracy: 0.9947 - val_loss: 1.5603 - val_accuracy: 0.7590
```

```
def recommend_news(user):
    user_history = []  # Simulated user browsing history
    recommended_news = real_news[real_news['source_domain'].isin(user_history)]
    return recommended_news
```

```
test_loss, test_accuracy = model.evaluate(X_test, y_test)
print("Test Loss:", test_loss)
print("Test Accuracy:", test_accuracy)
```

```
44/44 [==============================] - 2s 44ms/step - loss: 1.5603 - accuracy: 0.7590
Test Loss: 1.5602902173995972
Test Accuracy: 0.7590187788009644
```

```
Y_pred = model.predict(X_test).argmax(axis=1)
print("y-pred")
```

```
Y_pred
```

```
44/44 [==============================] - 2s 47ms/step
y-pred
array([0, 0, 0, ..., 0, 0, 0])
```

```
Yt_new
```

```
4771     www.livestrong.com
5431       hollywoodlife.com
4397           www.imdb.com
2951      www.teenvogue.com
3395       www.usatoday.com
                ...
4608       www.broadway.com
1071    www.dailymail.co.uk
5326            www.upi.com
4688       www.etonline.com
5292    www.dailymail.co.uk
Name: source_domain, Length: 1386, dtype: object
```

Filling inboxes of fake news spreaders

```
# Create a DataFrame with 'source_domain' and the index
source_domain_df = pd.DataFrame({'source_domain': Yt_new})

y_test=y_test.flatten()
Y_pred=Y_pred.flatten()
# Create DataFrames from Series with 'source_domain' column
y_test_df = pd.DataFrame({'y_test': y_test, 'source_domain': Yt_new})
Y_pred_df = pd.DataFrame({'Y_pred': Y_pred, 'source_domain': Yt_new})
```

```
'# Combine the 'source_domain' DataFrame with 'y_test' and 'Y_pred'\ny_test_df = pd.concat([Y_test_series, source_domain_df], axis=1
domain_df], axis=1)\n\n# Rename the columns\ny_test_df.columns = ['y_test', 'source_domain']\nY_pred_df.columns = ['Y_pred', 'source
```

```
Y_pred_df
```

	Y_pred	source_domain
4771	0	www.livestrong.com
5431	0	hollywoodlife.com
4397	0	www.imdb.com
2951	0	www.teenvogue.com
3395	0	www.usatoday.com
...	...	...
4608	0	www.broadway.com
1071	0	www.dailymail.co.uk
5326	0	www.upi.com
4688	0	www.etonline.com
5292	0	www.dailymail.co.uk

1386 rows × 2 columns

```
official_news_content = {
    0: "We believe in responsible and accurate information sharing. It has come to our attention that some of the content you shared may
}
```

```
count=0
for index, row in Y_pred_df.iterrows():
    if (row['Y_pred']==0):
        count+=1
        official_message = official_news_content[0]
        # You can send the official message to the spreader's inbox using email, notifications, etc.
        print(f"Sent to {row['source_domain']}: {official_message}")
print(count)
```

```
Sent to www.whowhatwear.com: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to www.bbc.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the c
Sent to www.instyle.com: We believe in responsible and accurate information sharing. It has come to our attention that some of t
Sent to www.businessinsider.com: We believe in responsible and accurate information sharing. It has come to our attention that s
Sent to www.nytimes.com: We believe in responsible and accurate information sharing. It has come to our attention that some of t
Sent to time.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the cont
Sent to www.etonline.com: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to omgcheckitout.com: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to wstale.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the co
Sent to liveindex.co.uk: We believe in responsible and accurate information sharing. It has come to our attention that some of t
Sent to okmagazine.com: We believe in responsible and accurate information sharing. It has come to our attention that some of th
Sent to www.inquisitr.com: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to www.elledecor.com: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to www.newshub.co.nz: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to www.usmagazine.com: We believe in responsible and accurate information sharing. It has come to our attention that some o
Sent to www.mtv.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the c
Sent to variety.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the c
Sent to seuss.wikia.com: We believe in responsible and accurate information sharing. It has come to our attention that some of t
Sent to www.dailymail.co.uk: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to people.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the co
Sent to popculture.com: We believe in responsible and accurate information sharing. It has come to our attention that some of th
Sent to www.bbc.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the c
Sent to www.elle.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the
Sent to www.thecut.com: We believe in responsible and accurate information sharing. It has come to our attention that some of th
Sent to 35.196.40.165: We believe in responsible and accurate information sharing. It has come to our attention that some of the
Sent to people.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the co
Sent to www.theloop.ca: We believe in responsible and accurate information sharing. It has come to our attention that some of th
Sent to www.parkerxl.com: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to www.guardian.co.uk: We believe in responsible and accurate information sharing. It has come to our attention that some o
Sent to www.fox10phoenix.com: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to wstale.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the co
Sent to usa24info.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the
Sent to www.news9ontime.com: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to www.dailymail.co.uk: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to www.thehollywoodgossip.com: We believe in responsible and accurate information sharing. It has come to our attention tha
Sent to people.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the co
Sent to www.accessonline.com: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to mashable.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the
Sent to www.bustle.com: We believe in responsible and accurate information sharing. It has come to our attention that some of th
Sent to us.hola.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the c
Sent to theworldnews.net: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to www.washingtonpost.com: We believe in responsible and accurate information sharing. It has come to our attention that so
Sent to www.trueara.com: We believe in responsible and accurate information sharing. It has come to our attention that some of t
Sent to www.dailymail.co.uk: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to www.newslocker.com: We believe in responsible and accurate information sharing. It has come to our attention that some o
Sent to www.rollingstone.com: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to www.dailymail.co.uk: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to people.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the co
Sent to en.wikipedia.org: We believe in responsible and accurate information sharing. It has come to our attention that some of
Sent to rip.trendolizer.com: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to www.instyle.com: We believe in responsible and accurate information sharing. It has come to our attention that some of t
Sent to www.freep.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the
Sent to www.bravotv.com: We believe in responsible and accurate information sharing. It has come to our attention that some of t
Sent to ew.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the conten
Sent to people.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the co
Sent to www.dailymail.co.uk: We believe in responsible and accurate information sharing. It has come to our attention that some
Sent to nextdivas.com: We believe in responsible and accurate information sharing. It has come to our attention that some of the
```

```
y_test
```

```
array([1., 0., 0., ..., 1., 1., 0.])
```

```
news_data
```

0	0	9960, 3159, 8325, 8164, 7096, 6103, 797...	http://toofab.com/2017/05/08/real-housewives-a...	toofab
1	1	[662, 7472, 4156, 1731, 6354, 7818, 5667, 3499]	https://www.today.com/style/see-people-s-choic...	www.today
2	2	[7304, 8299, 8117, 2385, 7214, 6914, 876, 9391...	https://www.etonline.com/news/220806_sophia_bu...	www.etonline
3	3	[3500, 1388, 1341, 394, 4625, 2824, 4130, 4745]	https://www.dailymail.co.uk/news/article-33655...	www.dailymail.
4	4	[6849, 9998, 4274, 4893, 6162, 6837, 6242, 234...	https://www.zerchoo.com/entertainment/gossip-g...	www.zerchoo
...	...	...	...	
22861	23191	[4414, 3538, 3038, 7570, 4693, 4414, 1343, 867...	https://www.express.co.uk/news/royal/807049/pi...	www.express.
		[4762, 5501, 5949		

```
news_data['source_domain']
```

```
0                 toofab.com
1              www.today.com
2           www.etonline.com
3          www.dailymail.co.uk
4             www.zerchoo.com
                   ...
22861       www.express.co.uk
22862       hollywoodlife.com
22863        www.justjared.com
22864    www.intouchweekly.com
22865        www.billboard.com
Name: source_domain, Length: 22866, dtype: object
```

```
X_test
```

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
12064	0	0	0	0	0	0	0	0	0	0	1000	5702	6498	6009	9015	2792
8817	0	0	0	0	0	0	0	0	0	0	0	0	0	933	5034	4737
19431	0	0	0	0	0	0	0	0	0	9710	1666	2443	7088	7725	9618	2215
10502	0	0	0	0	0	0	0	7514	5080	1598	8501	4700	5157	8612	8589	3278
21291	0	0	0	0	0	0	0	0	0	0	0	0	816	1178	912	2687
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
22777	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
712	0	0	0	0	0	0	0	0	0	0	0	0	0	0	5199	8519
5480	0	0	0	0	0	0	0	0	0	0	0	0	3166	6704	7423	9168
4804	0	0	0	0	0	0	0	0	0	0	0	0	0	2342	3259	6388
9895	0	0	0	0	0	0	0	0	0	0	0	0	8661	2626	4588	5064

5717 rows × 20 columns

Multilingual_support

```
!pip install langdetect
from langdetect import detect

# Function to detect the language of a text
def detect_language(text):
    try:
        return detect(text)
    except:
        return "unknown"  # Handle cases where language detection fails

# Apply language detection to each news article
news_data['language'] = news_data['news_url'].apply(detect_language)

# Filter the dataset for a specific language (e.g., English)
english_data = news_data[news_data['language'] == 'en']
```

```
Collecting langdetect
  Downloading langdetect-1.0.9.tar.gz (981 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 981.5/981.5 kB 8.3 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: six in /usr/local/lib/python3.10/dist-packages (from langdetect) (1.16.0)
Building wheels for collected packages: langdetect
  Building wheel for langdetect (setup.py) ... done
  Created wheel for langdetect: filename=langdetect-1.0.9-py3-none-any.whl size=993224 sha256=d111f4c648f9cce4421183ff5175745ad031c
  Stored in directory: /root/.cache/pip/wheels/95/03/7d/59ea870c70ce4e5a370638b5462a7711ab78fba2f655d05106
Successfully built langdetect
Installing collected packages: langdetect
Successfully installed langdetect-1.0.9
```

```
import nltk
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

# Download Hindi stopwords and word tokenizer
nltk.download('stopwords')
nltk.download('punkt')

# Define a function to preprocess text
def preprocess_text(text, language='english'):
    if language == 'hindi':
        # Replace with appropriate Hindi stopwords and word tokenizer
        stopwords_list = set(stopwords.words('hindi'))
        word_tokenizer = nltk.data.load('tokenizers/punkt/indian.pickle')
    else:
        stopwords_list = set(stopwords.words('english'))
        word_tokenizer = word_tokenize

    text = text.lower()
    words = word_tokenizer(text)

    # Remove stopwords
    words = [word for word in words if word not in stopwords_list]

    return ' '.join(words)
```