Fake News Detection Project

Submitted by:

SHEFALI MEVADA

ACKNOWLEDGMENT

Problem Definition

-> **Project Overview**

Fake News Detection Project

In this blog-post, I will go through the whole process of creating a machine learning model on the Fake news detection project dataset.

To build a model to accurately classify a piece of news as REAL or FAKE.

This advance python project of detecting fake news deals with fake and real news. Using sklearn, we build a CountVectorizer on our dataset. Then, we initialize a Decisiontreeclassifier and fit the model. In the end, the accuracy score and the confusion matrix tell us how well our model fares.

In which It provides the authenticity of Information has become a longstanding issue affecting businesses and society, both for printed and digital media. On social networks, the reach and effects of information spread occur at such a fast pace and so amplified that distorted, inaccurate, or false information acquires a tremendous potential to cause real-world impacts, within minutes, for millions of users. Recently, several public concerns about this problem and some approaches to mitigate the problem were expressed.

Analysis

Data Extraction

The dataset we will use for this python project- we will call it train-news.csv. This dataset has a shape of 20800 rows × 6 columns. The first column identifies the serial number ,the second identifies the Unique id

of each news article, the third and fourth are the headline and written_by , the fifth column has news and the sixth column has labels denoting whether the news is REAL or FAKE.

1 . Make Necessary Imports:

import pandas as pd

Import numpy as np

2 . Now , let's read the data into DataFrame, and get the output.

Read the data

df=pd.read_csv(r"C:\Users\DELL\Downloads\Fake-news-project\Fake news project\train_news.csv")

df

In [2]:
```
# Read the data
df=pd.read_csv(r"C:\Users\DELL\Downloads\Fake-news-project\Fake news project\train_news.csv")
df
```

Out[2]:

	Unnamed: 0	id	headline	written_by	news	label
0	0	9653	Ethics Questions Dogged Agriculture Nominee as...	Eric Lipton and Steve Eder	WASHINGTON — In Sonny Perdue's telling, Geo...	0
1	1	10041	U.S. Must Dig Deep to Stop Argentina's Lionel ...	David Waldstein	HOUSTON — Venezuela had a plan. It was a ta...	0
2	2	19113	Cotton to House: 'Do Not Walk the Plank and Vo...	Pam Key	Sunday on ABC's "This Week," while discussing ...	0
3	3	6868	Paul LePage, Besieged Maine Governor, Sends Co...	Jess Bidgood	AUGUSTA, Me. — The beleaguered Republican g...	0
4	4	7596	A Digital 9/11 If Trump Wins	Finian Cunningham	Finian Cunningham has written extensively on...	1
...	...	...	...	...	...	...
20795	20795	5671	NaN	NeverSurrender	No, you'll be a dog licking of the vomit of yo...	1
20796	20796	14831	Albert Pike and the European Migrant Crisis	Rixon Stewart	By Rixon Stewart on November 5, 2016 Rixon Ste...	1
20797	20797	18142	Dakota Access Caught Infiltrating Protests to ...	Eddy Lavine	posted by Eddie You know the Dakota Access Pip...	1
20798	20798	12139	How to Stretch the Summer Solstice - The New Y...	Alison S. Cohn	It's officially summer, and the Society Boutiq...	0
20799	20799	15660	Emory University to Pay for '100 Percent' of U...	Tom Ciccotta	Emory University in Atlanta, Georgia, has anno...	0

20800 rows × 6 columns

3 . Dropping unnecessary columns

#dropping columns

df=df.drop(['Unnamed: 0','id'], axis=1)

df=df.dropna()

And getting the first five values

#Get head

df.head()

```
In [3]: #dropping columns
        df=df.drop(['Unnamed: 0','id'], axis=1)
        df=df.dropna()
```

```
In [4]: #Get head
        df.head()
```

Out[4]:

	headline	written_by	news	label
0	Ethics Questions Dogged Agriculture Nominee as...	Eric Lipton and Steve Eder	WASHINGTON — In Sonny Perdue's telling, Geo...	0
1	U.S. Must Dig Deep to Stop Argentina's Lionel ...	David Waldstein	HOUSTON — Venezuela had a plan. It was a ta...	0
2	Cotton to House: "Do Not Walk the Plank and Vo...	Pam Key	Sunday on ABC's "This Week," while discussing ...	0
3	Paul LePage, Besieged Maine Governor, Sends Co...	Jess Bidgood	AUGUSTA, Me. — The beleaguered Republican g...	0
4	A Digital 9/11 If Trump Wins	Finian Cunningham	Finian Cunningham has written extensively on...	1

Get the labels

X=df.iloc[:,:-1].values

y=df.iloc[:,-1].values

4 . Let's Initialize a Countvectorizer

Initialize a CountVectorizer

fit and transform train set, transform test set

from sklearn.feature_extraction.text import CountVectorizer

cv=CountVectorizer(max_features=5000)

mat_body=cv.fit_transform(X[:,1]).todense()

```
# fit and transform train set, transform test set

from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer(max_features=5000)
mat_body=cv.fit_transform(X[:,1]).todense()
```

```
In [9]: mat_body

Out[9]: matrix([[0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                ...,
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0],
                [0, 0, 0, ..., 0, 0, 0]], dtype=int64)

In [10]: cv_head=CountVectorizer(max_features=5000)
         mat_head=cv_head.fit_transform(X[:,0]).todense()

In [11]: mat_head

Out[11]: matrix([[0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0],
                 ...,
                 [0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0],
                 [0, 0, 0, ..., 0, 0, 0]], dtype=int64)

In [12]: X_mat=np.hstack((mat_head,mat_body))
```

5 . Split the dataset into training and testing sets.

split the dataset

from sklearn.model_selection import train_test_split

X_train,X_test,y_train,y_test=train_test_split(X_mat,y,test_size=0.2,random_state=0)

```
In [13]: # split the dataset

         from sklearn.model_selection import train_test_split
         X_train,X_test,y_train,y_test=train_test_split(X_mat,y,test_size=0.2,random_state=0)
```

6 . Next, we will initialize a DecisionTreeClassifier.

Then, we will predict on the test set from the CountVectorizer and calculate the accuracy.

```
In [15]: # Initialize a DecisionTreeClassifier

         # Predict on the test set and calculate the accuracy

         from sklearn.tree import DecisionTreeClassifier
         dtc=DecisionTreeClassifier(criterion='entropy')
         dtc.fit(X_train,y_train)
         y_pred=dtc.predict(X_test)
```

7 . Let's print out a confusion matrix to gain insight into the number of false and true negatives and positives.

Build confusion matrix

from sklearn.metrics import confusion_matrix

confusion_matrix(y_test,y_pred)

Output Screenshot:

```
In [16]: # Build confusion matrix

         from sklearn.metrics import confusion_matrix
         confusion_matrix(y_test,y_pred)

Out[16]: array([[2064,   14],
                [  13, 1566]], dtype=int64)

In [17]: (2064+1566)/(2064+1566+14+13)

Out[17]: 0.992616899097621
```

So with this model,we have 2064 true positives, 1566 true negatives, 13 false positives, and 14 false negatives.

Summary

We detect fake news with Python. We took a dataset, implemented a CountVectorizer, initialized a Decisiontreeclassifier and fit our model.

We ended up obtaining an accuracy of 99% in magnitude.