Project 3: Fake News Detection

Bảng phân công công việc:

MSSV	Họ và tên	Công việc
19120186	Đỗ Lê Khánh Đăng	Tiền xử lí văn bản tiếng việt
19120412	Nguyễn Minh Tú	Khám phá dữ liệu
19120462	Lục Minh Bửu	Deploy mô hình
19120467	Ngô Hữu Đang	Mô hình hóa

Link web deploy

Cài đặt thêm thư viện liên quan

```
# !pip install underthesea
```

Thêm các thư viện liên quan

```
import pandas as pd
import re
import matplotlib.pyplot as plt
import joblib
from sklearn.model_selection import train_test_split
from underthesea import word_tokenize
from sklearn.naive_bayes import MultinomialNB
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.tree import DecisionTreeClassifier
```

Cài đặt hiện thị DataFrame

```
pd.set_option('display.max_rows', None)
pd.set_option('display.max_columns', None)
pd.set_option('display.width', None)
pd.set_option('display.max_colwidth', None)
```

Đọc dữ liệu

- vn_news_223_tdlfr.csv là file chứa news cần xử lý
- vietnamese-stopwords.txt chứa các stop word cần xử lý

```
data_df = pd.read_csv('vn_news_223_tdlfr.csv', encoding = 'utf-8')
file = open('vietnamese-stopwords.txt','r', encoding = 'utf-8')
stopWords = file.read().split('\n')
file.close()
```

Khám phá dữ liệu

Xem các mẫu dữ liệu

```
data_df.sample(5)
```

	text	domain	label
46	Cậu bé kinh hãi khi thấy chân lúc nhúc hàng triệu sinh vật bấu chặt, cúi xuống phát hiện sự thật bàng hoàng\nKinh ngạc cảnh tượng hàng triệu sinh vật bò nhung nhúc bấu chặt lấy chân.\nChristmas là một đảo nhỏ thuộc Úc, cách 2.600 km về Phía tây bắc thành phố Perth. Hòn đảo là quê hương của vô số các loài động – thực vật vô cùng phong phú và kỳ lạ. Nếu không chú ý, những du khách lần đầu tiên đến tham quan hòn đảo này thường gặp nhiều "phe" sởn gai ốc, khi ra vừa chỉ đứng im vài phút thôi, đã thấy dưới bàn chân mình rờn rợn, nhung nhúc vô cùng nhiều sinh vật kỳ lạ đang bò lên và bấu chặt lấy. Thậm chí, cả những con đường hay cả bờ biển nơi đây đều bị nhuộm một màu đỏ rực.\nHàng trăm triệu con cua bò lên bấu chặt lấy chân cậu bé.\nBờ biển nhuộm màu đỏ rực.\nCác sinh vật lạ nhung nhúc bò trên mỏm đá.\nSau quá trình tìm hiểu thông qua người dân địa phương, các du khách mới ngã ngửa khi biết rằng nguyên nhân của hiện tượng kỳ quái này chính là cuộc di cư thường năm của loài cua đỏ.\nCuộc di cư đặc biệt này thường diễn ra với quy mô lớn.\nVào mỗi mùa giao phối, loài cua đỏ đặc biệt này di cư với quy mô lớn, nó được ví như một cơn thủy triều đỏ khủng khiếp. Theo các nhà nghiên cứu, do điều kiện khí hậu mùa mưa giúp cho việc di chuyển thuận tiện và dễ dàng hơn nên khoảng thời gian di cư của loài cua đỏ này thường bắt đầu vào khoảng tháng 10 – 12.\nVào mùa mưa, hàng triệu con cua đỏ này lại di cư ra biển để bắt đầu mùa sinh sản mới.\nSố lượng đàn cua đỏ di chuyển trong mỗi lần di cư rất đông, nên chính quyền địa phương phải đặt các tấm biểu báo cấm đường dành cho các phương tiện và người đi bộ nhằm tránh làm tổn thương đến chúng.\nChính quyền địa phương phải đặt các tấm biểu báo cấm đường dành cho các phương tiện và người đi bộ nhằm tránh làm tổn thương đến chúng.\nXem thêm: Muốn hờn cả thế giới khi chứng kiến sự thật trắng trợn đằng sau những bức ảnh sống ảo trăm nghìn like như thế này\nTrong mỗi đợt di cư, những con cua này sẽ phải vượt qua một đoạn đường dài 8 km, bắt đầu từ các khu rừng ở đảo Christmas, Australia đến bờ biển Ấn Độ Dương, trong vòng 9 đến 18 ngày. Nhiều du khách thắc mắc tại sao loài vật bé nhỏ này lại có thể di chuyển xa như vậy? Đó là vì đến mùa sinh sản, cua tiết ra nhiều nội tiết tố hyperglycemic, giúp tăng lượng đường glucoza trong máu, đảm bảo cho việc cung cấp năng lượng trong thời gian dài.\nĐến mùa sinh sản, cua tiết ra nhiều nội tiết tố hyperglycemic, giúp tăng lượng đường glucoza trong máu, đảm bảo cho việc cung cấp năng lượng trong thời gian dài.\nĐược biết cua đỏ là loài giáp xác duy nhất có con đực cùng di cư với con cái, những con cua đực sẽ dẫn đầu làn sóng di cư đến tìm hang để ẩn nấp, sau đó là các con cua cái sẽ đến những hang này để giao phối với cua đực rồi bò ra biển để đẻ trứng.\nCua đỏ là loài giáp xác duy nhất có con đực cùng di cư với con cái.\nCác cuộc di cư thường niên của cua đỏ đã trở nên quen thuộc với người dân ở đây, nhưng lại là điều vô cùng lý thú và thu hút hàng nghìn khách du lịch tò mò ghé thăm. 'Xã hội đen' Hải Phòng xây hàng trăm nhà trái phép trên đất quốc phòng\nNhững ngôi nhà trái phép vẫn đang được xây dựng tại khu A mảnh đất 14,2 ha. Ảnh: Giang Chinh\nNgày 17/10, thượng tướng Lê Chiêm, Thứ trưởng Quốc phòng ký biên bản bàn giao khu đất rộng 14,2 ha tại phường Thành Tô và Tràng Cát (quận Hải An) cho UBND TP Hải Phòng quản lý, sử dụng. Khu đất này trước đây do Tổng công ty 319 quản lý, nhưng nay không còn sử dụng vào mục đích quốc phòng.\nNgay sau khi tiếp nhận, UBND TP Hải Phòng đã ban hành quyết định thu hồi khu đất, giao lại cho UBND quận	tinvn.info	1

Xem thông tin

Bộ Quốc phòng giao đất cho Sư đoàn 363. Đơn vị này sau đó cấu kết với

```
data_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 223 entries, 0 to 222
Data columns (total 3 columns):
 #   Column  Non-Null Count  Dtype 
---  ------  --------------  ----- 
 0   text    223 non-null    object
 1   domain  223 non-null    object
 2   label   223 non-null    int64 
dtypes: int64(1), object(2)
memory usage: 5.4+ KB
```

lãnh đạo các sở, ngành và quận Hải An khi chậm trễ trong tham mưu, đề

Xem mô tả

đạn phải nghiêm túc phối hợp với chính quyền bằng cách kê khai và đưa ra

```
data_df.describe().round(1)
```

	label
count	223.0
mean	0.4
std	0.5
min	0.0
25%	0.0
50%	0.0
75%	1.0
max	1.0

Dữ liệu gồm có bao nhiêu dòng và bao nhiêu cột?

```
num_rows=data_df.shape[0]
num_cols=data_df.shape[1]
print(num_rows)
print(num_cols)
```

```
223
3
```

Dữ liệu có 223 dòng và 3 cột

Mỗi dòng có ý nghĩa gì? Có vấn đề các dòng có ý nghĩa khác nhau không?

Mỗi dòng là thông tin của 1 tin tức và đường dẫn của trang web chứa tin tức đó. Có vẻ tất cả các dòng đều có ý nghĩa giống nhau.

Dữ liệu có các dòng bị lặp không?

```
dup=data_df.index.duplicated().sum()
dup
```

```
0
```

Mỗi cột có ý nghĩa gì?

- text: nội dung của tin tức
- domain: đường dẫn đến trang web chứa tin tức
- label: nhãn phân biệt tin giả hay tin thật

Dữ liệu có bị thiếu không?

```
data_df['text'].isnull().sum()
```

```
0
```

```
data_df['domain'].isnull().sum()
```

```
0
```

```
data_df['label'].isnull().sum()
```

```
0
```

Vậy là không có cột nào bị thiếu dữ liệu cả.

Mỗi cột hiện đang có kiểu dữ liệu gì? Có cột nào có kiểu dữ liệu chưa phù hợp để có thể xử lý tiếp không?

```
data_df.dtypes
```

```
text      object
domain    object
label      int64
dtype: object
```

Cột có dtype là object nghĩa là sao?

- Trong Pandas, kiểu dữ liệu object thường ám chỉ chuỗi, nhưng thật ra kiểu dữ liệu object có thể chứa một đối tượng bất kỳ trong Python (vì thật ra ở bên dưới kiểu dữ liệu object chứa địa chỉ).
- Nếu một cột trong dataframe có dtype là object thì có thể các phần tử trong cột này sẽ có kiểu dữ liệu khác nhau
- Để biết được kiểu dữ liệu thật sự của các phần tử trong cột này thì ta phải truy xuất vào từng phần tử. Ta muốn xem thử trong nội bộ mỗi cột này có các kiểu dữ liệu nào.

```
def open_object_dtype(s):
    dtypes = set()
    s=s.apply(type)
    dtypes.update(s.unique().tolist())
    return dtypes
```

```
open_object_dtype(data_df['text'])
```

```
{str}
```

```
open_object_dtype(data_df['domain'])
```

```
{str}
```

```
open_object_dtype(data_df['label'])
```

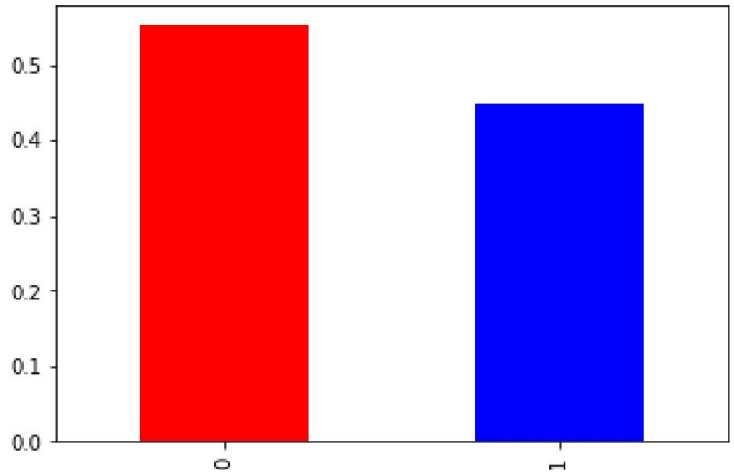
```
{int}
```

Nhìn chung kiểu dữ liệu của các cột có vẻ đúng, không cần phải chỉnh sửa gì thêm.

Kiểm tra phân bố các class có chênh lệch không?

```
data_df.label.value_counts(normalize=True).plot(kind="bar",
                                                color=["red", "blue"])
data_df['label'].value_counts(normalize=True) * 100
```

```
0    55.156951
1    44.843049
Name: label, dtype: float64
```



Tỉ lệ giữa các lớp cũng khá cân bằng, không chênh lệch gì lắm.

Các thông tin thống kê

Chiều dài trung bình mỗi record là bao nhiêu?

```
len_sum=0
for i in data_df['text']:
    len_sum+=len(i)
len_avg=len_sum/data_df['text'].count()
len_avg
```

```
2539.7713004484303
```

Chiều dài trung bình của mỗi record là 2548 ký tự.

Record dài nhất là bao nhiêu?

```
max_len=0
for i in data_df['text']:
    if len(i)>max_len:
        max_len=len(i)
max_len
```

```
10019
```

Record dài nhất chứa 10084 ký tự.

Record ngắn nhất là bao nhiêu?

```
min_len=len(data_df['text'][0])
for i in data_df['text']:
    if len(i)<min_len:
        min_len=len(i)
min_len
```

```
309
```

Record ngắn nhất chứa 311 ký tự.

Tin tức được thu thập từ trang web nào nhiều nhất?

```
data_df['domain'].value_counts()
```

```
vnexpress.net              73
tinvn.info                 68
dantri.com.vn              14
thethao.tuoitre.vn         10
news.zing.vn                6
thoibao.today               6
tuoitre.vn                  5
tintucqpvn.net              5
thanhnien.vn                5
phapluat.news               3
kinhdoanh.vnexpress.net     3
doisong.vnexpress.net       2
giadinhtiepthi.com          2
thoibao.de                  2
giaitri.vnexpress.net       2
thegioitre.vn               2
www.ipick.vn                2
baonuocmy.com               1
suckhoe.vnexpress.net       1
haiduong.tintuc.vn          1
sorry.vn                    1
dulich.vnexpress.net        1
www.gioitreviet.net         1
sohoa.vnexpress.net         1
baoangiang.com.vn           1
autoxe.net                  1
binhluan.biz                1
laodong.vn                  1
www.vietgiaitri.com/        1
https://news.zing.vn        1
Name: domain, dtype: int64
```

- Hai trang web được thu thập tin tức nhiều nhất là: vnexpress.net và tinvn.info.
- Những tin tức còn lại được thu thập rải rác ở nhiều trang web khác nhau

Tiền xử lí văn bản tiếng việt

Loại bỏ các đường link và các dấu câu, lowercase

```
def wordopt(text):
    text = text.lower()
    text = re.sub('https?:\/\/.*[\r\n]*', ' ', text)
    text = re.sub('[^\w\s]',' ',text)
    text = re.sub('\n', ' ', text)
    return text


data_df["text"]=data_df["text"].apply(wordopt)


data_df.head()
```

	text	domain	label
0	thủ tướng abe cúi đầu xin lỗi vì hành động phi thể thao của tuyển nhật theo sankei sports sáng nay thủ tướng nhật bản shinzo abe công khai gửi lời xin lỗi tới nhật hoàng và toàn bộ người dân vì tinh thần thi đấu phi thể thao của đội tuyển nhật tại world cup 2018 tối qua sau lượt trận cuối vòng bảng world cup 2018 nhật bản có cùng chỉ số phụ như senegal đội bị loại sau khi thua colombia nhưng nhật bản vào vòng sau nhờ chỉ số fair play vì nhận ít thẻ phạt hơn thủ tướng nhật bản shinzo abe cúi đầu xin lỗi với tinh thần của những võ sĩ đạo samurai nhưng đội tuyển nhật bản đã có những hành động thiếu tinh thần thượng võ trong thi đấu tại world cup để lại nhiều chỉ trích và bất bình cho toàn dân làm mất hình ảnh kiên cường của người dân nhật bản trên đấu trường quốc tế là người đứng đầu tôi xin thành thật nhận trách nhiệm và gửi lời xin lỗi sâu sắc tới nhân dân ông abe cúi đầu nhận trách nhiệm về mình người nhật bản nổi tiếng về tinh thần trách nhiệm và chất võ sĩ đạo đó là lý do đội tuyển áo xanh được mệnh danh samurai xanh tuy nhiên nhật bản sau đó nhận chỉ trích dữ dội của người hâm mộ bóng đá vì lối chơi bóng tiêu cực cố tình câu giờ để dành tấm vé đi tiếp cụ thể những phút cuối trận cuối vòng bảng gặp ba lan do biết ở trận cùng giờ colombia cũng đang thắng senegal với tỷ số 1 0 nên dù có đang bị dẫn trước với tỉ số tương tự các cầu thủ nhật bản cũng không hề muốn gỡ hòa các cầu thủ nhật vui vẻ sau trận thua ba lan 0 1 có vé vào vòng 16 đội tại world cup 2018 fifa sẽ tính điểm fair play theo quy định 1 thẻ vàng 1 điểm 2 thẻ vàng thành thẻ đỏ 3 điểm thẻ đỏ trực tiếp 4 điểm nhật có 4 thẻ vàng còn senegal có đến 6	binhluan.biz	1

Tokenizer

sáo của các cđv trên sân điều này khiến thầy trò hlv akira nishino

```
def tokenize(sentence):
    return word_tokenize(sentence, format = 'word')
```

nhật là colombia se gặp đt ann

Mô hình hóa

thể thao của đội tuyển nhật tại world cup 2018 với tinh thần của

- Chuyển đoạn văn tiếng Việt về vector, sử dụng CountVectorizer của sklearn
- với các tham số là danh sách stopwords tiếng Việt
- và tokenizer tách từ tiếng Việt

lỗi sâu sắc tới nhân dân ông abe cúi đầu nhân trách nhiệm về

```
vectorizer = CountVectorizer(
    stop_words = stopWords,
    tokenizer = tokenize
)
```

choáng có trường đeo khẩu quang quẩy bánh nóc trên sân khấu

Chia tập dữ liệu thành 2 tập

- Tập train: 75%
- Tập test: 25%

```
X = data_df["text"]
y = data_df["label"]
X_train, X_test, y_train, y_test = train_test_split(X, y,
                                                    test_size=0.25,
                                                    random_state=30)
Xv_train = vectorizer.fit_transform(X_train)
Xv_test = vectorizer.transform(X_test)
```

```
C:\ProgramData\Anaconda3\envs\min_ds-env\lib\site-packages\sklearn\feature_extraction\te
  warnings.warn('Your stop_words may be inconsistent with '
```

```
#save model vectorizer
joblib.dump(vectorizer, 'vectorizer.joblib')
```

```
['vectorizer.joblib']
```

Decision Tree

```
DT = DecisionTreeClassifier()
DT.fit(Xv_train,y_train)
```

```
DecisionTreeClassifier()
```

```
pred_dt = DT.predict(Xv_test)
print(DT.score(Xv_test, y_test))
#save model Decision Tree
joblib.dump(DT, 'DT_model.joblib')
```

```
0.7321428571428571
['DT_model.joblib']
```

Naive Bayes

```
NB = MultinomialNB()
NB.fit(Xv_train,y_train)
```

```
MultinomialNB()
```

```
pred_nb = DT.predict(Xv_test)
print(NB.score(Xv_test, y_test))
#save model Naive Bayes
joblib.dump(NB, 'NB_model.joblib')
```

```
0.9464285714285714
['NB_model.joblib']
```

Đánh giá mô hình

- Mô hình với Decesion Tree có độ chính xác khá thấp khoảng 70%
- Mô hình với Naive Bayes có độ chính xác cao khoảng 95%