#**CLASSIFICATION** : ENTITES NOMMEES-TRUE_FALSE vs OTHER :

**Membres:** Hadjoudja Bachir (21811363), Zeggar Rym (21909615), Bendahmane Rania (21811387), Labiad Youcef (21710780).

```python
#les imports utilisés dans ce notebook
import sys
from numpy import vstack
import pandas as pd
from pandas import read_csv
from sklearn.preprocessing import LabelEncoder
from sklearn.metrics import accuracy_score
from torch.utils.data import Dataset
from torch.utils.data import DataLoader
from torch.utils.data import random_split
from torch import Tensor
from torch.nn import Linear
from torch.nn import ReLU
from torch.nn import Sigmoid
from torch.nn import Module
from torch.optim import SGD
from torch.nn import BCELoss
from torch.nn.init import kaiming_uniform_
from torch.nn.init import xavier_uniform_
import re
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
from pandas import read_csv
from sklearn.feature_extraction.text import TfidfVectorizer
from  sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
import pickle
import string

import nltk
from nltk.stem import WordNetLemmatizer
from nltk.stem import PorterStemmer
from nltk.corpus import stopwords
from nltk import word_tokenize
from sklearn.pipeline import Pipeline
import spacy

import sklearn
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
```

```python
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics import precision_recall_fscore_support as score
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
from sklearn.ensemble import RandomForestClassifier
```

```python
# Importation des différentes librairies utiles pour le notebook
#Sickit learn met régulièrement à jour des versions et
#indique des futurs warnings.
#ces deux lignes permettent de ne pas les afficher.
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)

import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import sys
import pandas as pd
import numpy as np
import sklearn
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score
from sklearn.model_selection import train_test_split
from sklearn.model_selection import KFold
from sklearn.model_selection import cross_val_score
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
import seaborn as sns
import matplotlib.pyplot as plt
from sklearn.metrics import precision_recall_fscore_support as score
from sklearn.linear_model import LogisticRegression
from sklearn.tree import DecisionTreeClassifier
from sklearn.neighbors import KNeighborsClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.svm import SVC
from sklearn.model_selection import GridSearchCV
from sklearn.naive_bayes import MultinomialNB
from sklearn.ensemble import RandomForestClassifier
from sklearn.pipeline import Pipeline
from sklearn.feature_extraction.text import TfidfVectorizer
#Sickit learn met régulièrement à jour des versions et indique des
futurs warnings.
#ces deux lignes permettent de ne pas les afficher.
```

```python
import warnings
warnings.filterwarnings("ignore", category=FutureWarning)
from sklearn.metrics._plot.confusion_matrix import
ConfusionMatrixDisplay
from sklearn import metrics
from sklearn.metrics import confusion_matrix , ConfusionMatrixDisplay
from sklearn.metrics import classification_report
```

autorisation

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

```
Mounted at /content/gdrive/
```

chemin spécifique Google Drive

```python
my_local_drive='/content/gdrive/My Drive/Colab Notebooks'
# Ajout du path pour les librairies, fonctions et données
sys.path.append(my_local_drive)
# Se positionner sur le répertoire associé
%cd $my_local_drive
%ls

%pwd
```

```
/content/gdrive/My Drive/Colab Notebooks
 bonFakeNEWS4.ipynb
'BON_TRUE FALSE_vs_OTHER_entités_nommées.ipynb'
'Copie de FakeNEWS.ipynb'
'Copie de True_False_Other_Mixture.ipynb'
'Copie de TRUE FALSE_vs_OTHER_entités_nommées.ipynb'
'Copie de  Vrai_Faux._entites_marche_bien.ipynb'
 FakeNewsLastVersion.ipynb
 ml_entiteesNommeesTest.ipynb
'Traitement sémantique'/
 True_False_Other_Mixture_final.ipynb
 Untitled0.ipynb
 version2ml_entiteesNommeesTest.ipynb
'VRAI FAUX OTHER MIXTURE_avec_entites_nommees.ipynb'
```

{"type":"string"}

La fonction qui sera utilisée pour les prétraitements: MyCleanText

- Mettre le texte en minuscule
- Se débarasser des stopwords
- Se débarasser des nombres
- Stemmatisation
- Lemmatisation ..

La fonction MyshowAllScores prend le y_test et le y_predict, affiche l'accuracy et le classification report avec la matrice de confusion.

```
#...................................................Fonction
MyCleanText ..........................................................
...................
# mettre en minuscule
#enlever les stopwords
#se debarasser des nombres
#stemmatisation
#lemmatisation
#.....................................................................
.....................................................................
...........


nltk.download('wordnet')
nltk.download('stopwords')
nltk.download('punkt')
#liste des stopwords en anglais
stop_words = set(stopwords.words('english'))

def MyCleanText(X,
                lowercase=False, #mettre en minuscule
                removestopwords=False, #supprimer les stopwords
                removedigit=False, #supprimer les nombres
                getstemmer=False, #conserver la racine des termes
                getlemmatisation=False #lemmatisation des termes
                ):
  #conversion du texte d'entrée en chaîne de caractères
    sentence=str(X)
    #suppression des caractères spéciaux
    sentence = re.sub(r'[^\w\s]',' ', sentence)
    # suppression de tous les caractères uniques
    sentence = re.sub(r'\s+[a-zA-Z]\s+', ' ', sentence)
    # substitution des espaces multiples par un seul espace
    sentence = re.sub(r'\s+', ' ', sentence, flags=re.I)

    # decoupage en mots
    tokens = word_tokenize(sentence)
    if lowercase:
          tokens = [token.lower() for token in tokens]

    # suppression ponctuation
    table = str.maketrans('', '', string.punctuation)
    words = [token.translate(table) for token in tokens]

    # suppression des tokens non alphabetique ou numerique
    words = [word for word in words if word.isalnum()]
```

```
    # suppression des tokens numerique
    if removedigit:
        words = [word for word in words if not word.isdigit()]

    # suppression des stopwords
    if removestopwords:
        words = [word for word in words if not word in stop_words]

    # lemmatisation
    if getlemmatisation:
        lemmatizer=WordNetLemmatizer()
        words = [lemmatizer.lemmatize(word)for word in words]


    # racinisation
    if getstemmer:
        ps = PorterStemmer()
        words=[ps.stem(word) for word in words]

    sentence= ' '.join(words)

    return sentence

# fonction qui affiche le classification report et la matrice de
confusion
def MyshowAllScores(y_test,y_pred):
  classes= np.unique(y_test)
  print("Accuracy : %0.3f"%(accuracy_score(y_test,y_pred)))
  print("Classification Report")
  print(classification_report(y_test,y_pred,digits=5))
  cnf_matrix = confusion_matrix(y_test,y_pred)
  disp=ConfusionMatrixDisplay(cnf_matrix,display_labels=classes)
  disp.plot()
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Unzipping tokenizers/punkt.zip.
```

La classe TextNormalizer qui contiendra la fonction MyCleanText.

Fit_transform de mon corpus propre.

```
#..................................................Etape 1 :
prétraitement du
texte ..............................................................
..................................
#...................................................Fonction
TextNormalizer  ....................................................
......................
```

```python
#fit_transform de mon corpus propre
#..............................................................
.....................................................................
...........

from sklearn.base import BaseEstimator, TransformerMixin

class TextNormalizer(BaseEstimator, TransformerMixin):
    def __init__(self,
                 removestopwords=False, # suppression des stopwords
                 lowercase=False,# passage en minuscule
                 removedigit=False, # supprimer les nombres
                 getstemmer=False,# racinisation des termes
                 getlemmatisation=False # lemmatisation des termes
                ):

        self.lowercase=lowercase
        self.getstemmer=getstemmer
        self.removestopwords=removestopwords
        self.getlemmatisation=getlemmatisation
        self.removedigit=removedigit

    def transform(self, X, **transform_params):
        # Nettoyage du texte
        X=X.copy() # pour conserver le fichier d'origine
        return [MyCleanText(text,lowercase=self.lowercase,
                            getstemmer=self.getstemmer,
                            removestopwords=self.removestopwords,
                            getlemmatisation=self.getlemmatisation,
                            removedigit=self.removedigit) for text in
X]

    def fit(self, X, y=None, **fit_params):
        return self

    def fit_transform(self, X, y=None, **fit_params):
        return self.fit(X).transform(X)

    def get_params(self, deep=True):
        return {
            'lowercase':self.lowercase,
            'getstemmer':self.getstemmer,
            'removestopwords':self.removestopwords,
            'getlemmatisation':self.getlemmatisation,
            'removedigit':self.removedigit
        }

    def set_params (self, **parameters):
        for parameter, value in parameters.items():
```

```
            setattr(self,parameter,value)
        return self
```

##**1ere classification : True/False VS Mixture/OTHER**

###**Etape 1 : Préparer les données**

- Charger et préparer les données à partir des 2 fichiers csv
- Affichages pour tester si cela a bien été fait
- Récupérer que les lignes où on a TRUE FALSE ou bien OTHER
- Créer une colonne regrouped qui à partir de rating regarde si on a True ou False on met TRUE/FALSE sinon OTHER

```
#Ici je cherche à séléctionner que les labels TRUE/FALSE et OTHER,
donc les LIGNES qui contiennent au rating TRUE,FALSE et OTHER
uniquement, le reste on enlève

dftrain =
pd.read_csv("/content/gdrive/MyDrive/projet_ML/newsTrain2.csv",
names=['id','text','title','rating'], header=0,sep=',',
encoding='utf8')
dftrain.reset_index(drop = True, inplace = True)

dftrain2 = pd.read_csv("/content/gdrive/MyDrive/projet_ML/newsTrain -
newsTrain.csv", names=['id','text','title','rating'],
header=0,sep=',', encoding='utf8')
dftrain2.reset_index(drop = True, inplace = True)


# concaténer les deux dataframes en ajoutant les lignes du deuxième à
la fin du premier
dftrain = pd.concat([dftrain, dftrain2], ignore_index=True)
dftrain = dftrain.loc[dftrain['rating'].isin(['TRUE','FALSE',
'other'])]

#On crée une colonne regroupe qui va mettre dans les lignes là où a
true ou bien false la valeur TRUE/FALSE et OTHER ça laisse
dftrain['regrouped'] = dftrain['rating'].apply(lambda x:'TRUE/FALSE'
if x in ['TRUE', 'FALSE'] else 'OTHER')

#Quelques affichages pour aider à mieux visualiser nos données
print("Echantillon de mon dataset \n")
print(dftrain.sample(n=10))
print("\n")
print("Quelques informations importantes \n")
dftrain.info()
```

```
Echantillon de mon dataset

            id                                                  text  \
```

```
2277  ca53fa81  SHARE By of the Sun Prairie - With the economy...
727   b9ccbcb4  Though health officials have warned Americans ...
212   4ce1af1d  Martin Gugino is a 75-year-old professional ag...
565   8a9e86f3  Regulation promotes self-sufficiency and immig...
536   9d3ac1f0  Three in four labour wards have no consultants...
242   26898e5b  Joe Biden’s Inauguration has been cancelled, P...
951   347530a3  On Tuesday, radio show host John Fredricks sta...
138   19e13d4f  Recount observers check ballots during a Milwa...
2056  8fba8857  In a scenario straight out of “The Twilight Zo...
1078  1c068671  Pressure on the government to help struggling ...

                                                   title rating  
regrouped  
2277  Pastor Dies After 30 Days of Fasting To Beat J...  FALSE  
TRUE/FALSE  
727   New Eavesdropping Equipment Sucks All Data Off...  FALSE  
TRUE/FALSE  
212   Buffalo Officials Duped By Professional Antifa...  FALSE  
TRUE/FALSE  
565                                                 NaN  FALSE  
TRUE/FALSE  
536   ‘It won’t simply be sanctions’: Biden adviser ...   TRUE  
TRUE/FALSE  
242   Inauguration Cancelled, Trump Remains in Offic...  FALSE  
TRUE/FALSE  
951                           Warren Statement on Boeing  other  
OTHER  
138   Trump objects to counting thousands of Wiscons...  FALSE  
TRUE/FALSE  
2056  Christian Pastor In Vermont Sentenced To One Y...  FALSE  
TRUE/FALSE  
1078  UK faces return to inequality of Thatcher year...  other  
OTHER  
```

Quelques informations importantes

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1812 entries, 0 to 2527
Data columns (total 5 columns):
 #   Column     Non-Null Count  Dtype 
---  ------     --------------  ----- 
 0   id         1812 non-null   object
 1   text       1812 non-null   object
 2   title      1784 non-null   object
 3   rating     1812 non-null   object
 4   regrouped  1812 non-null   object
dtypes: object(5)
memory usage: 84.9+ KB
```

Le jeu de données étant déséquilibré, on a pensé à appliquer le downsampling pour équilibrer nos données. on séléctionne des lignes aléatoirement de TRUE/FALSE de telle sorte que le nombre de lignes de TRUE/FALSE soit = au nbr de lignes de OTHER. et on mélange le DataFrame

```
#On applique du sous-échantillonnage (downsampling) : car on a plus de
FALSE (578) que des TRUE (211)

# Séparer les classes en deux dataframes
df_false_true = dftrain[dftrain['regrouped'] == 'TRUE/FALSE']
df_other = dftrain [dftrain['regrouped'] == 'OTHER']

# Sous-échantillonner la classe majoritaire (FALSE) pour obtenir un
nombre égal d'échantillons pour chaque classe
df_subsampled = df_false_true.sample(n=len(df_other), random_state=42)

# Concaténer les deux dataframes
dftrain = pd.concat([df_subsampled, df_other])

# Mélanger aléatoirement les données
dftrain = dftrain.sample(frac=1, random_state=42)

X_text=dftrain.iloc[0:,1:2]
X_title=dftrain.iloc[0:,2:3]

print("le texte est")
display(X_text)
print("le titre est")
display(X_title)

y=dftrain.iloc[0:,-1]
print("le y est")
display(y)
print("la taille de X_text est",X_text.shape)
print("la taille de y_train est " ,y.shape)
print("les valeurs de TRUE et FALSE maintenant sont
" ,y.value_counts())
```

```
le texte est

                                                    text
947   War-torn eastern regions of Ukraine have no la...
2224  TIJUANA, Mexico — It’s the image from the unfo...
1307  Today, Congresswoman Maxine Waters D-CA, Chair...
798   Meghan Markle will use the furore over her int...
320   Further proof that Democrats are the greatest ...
...                                                  ...
1160  The scale of Antarctica is startling. Miles of...
570   Coronavirus may be sexually transmitted and ca...
1200  Like what?  Helen Harwatt is a researcher trai...
```

```
2190                   Tumeric kills cancer not patient
391   WASHINGTON, DC – The Pentagon has issued an in...

[468 rows x 1 columns]
```

```
le titre est
```

```
                                                  title
947   Look No Further, The Best Doctor Strange in th...
2224  A discussion of ‘smokers’ black lungs’ started...
1307  Democratic Lawmaker introduces bill to rename ...
798   Newton Emerson: Swiss model offers food for th...
320   Democrats Introduce Bill To ‘Euthanize Seniors...
...                                                 ...
1160               Miles of Ice Collapsing Into the Sea
570   Universal Credit leaves working families worse...
1200              If Everyone Ate Beans Instead of Beef
2190  Vermont state trooper revived with Narcan afte...
391   Pentagon Confirms Coronavirus Accidently Got I...

[468 rows x 1 columns]
```

```
le y est
```

```
947     TRUE/FALSE
2224    TRUE/FALSE
1307    TRUE/FALSE
798          OTHER
320     TRUE/FALSE
           ...
1160    TRUE/FALSE
570          OTHER
1200         OTHER
2190         OTHER
391     TRUE/FALSE
Name: regrouped, Length: 468, dtype: object
```

```
la taille de X_text est (468, 1)
la taille de y_train est  (468,)
les valeurs de TRUE et FALSE maintenant sont  TRUE/FALSE    234
OTHER         234
Name: regrouped, dtype: int64
```

**ENTITY_RECOGNITION**

Avant de classifier on applique les entités nommées, Cette fonction utilise Spacy pour extraire les entités nommées à partir d'une liste de textes et les stocke dans une liste pour être affichées avec leurs labels correspondants.

```
from spacy import displacy
```

```python
nlp = spacy.load("en_core_web_sm")
texte=dftrain['text']
#Créer une liste pour stocker les entités nommées
entities = []
# Traiter chaque texte individuellement et ajouter les entités à la
liste
for phrase in texte:
    phrase_str = str(phrase)  # Convertir l'élément en chaîne de
caractères
    doc = nlp(phrase_str)
    for ent in doc.ents:
        entities.append((ent.text, ent.label_))


# Afficher les entités nommées
for entity in entities:
    print(entity[0], entity[1])
```

```
Le flux de sortie a été tronqué et ne contient que les 5000 dernières
lignes.
2007 DATE
Italy GPE
Britain GPE
the European Arrest Warrant ORG
EU ORG
over 5,000 CARDINAL
EU ORG
2010 DATE
European NORP
Britain GPE
EU ORG
the United Nations ORG
Nato ORG
EU ORG
Russia GPE
Ukraine GPE
the Horn of Africa LOC
British NORP
1bn ORDINAL
West Africa GPE
Europe LOC
first ORDINAL
EU ORG
Europe LOC
Norway GPE
Switzerland GPE
Europe LOC
Russia GPE
Europe LOC
EU ORG
```

EU ORG
thousands CARDINAL
Europe LOC
Europe LOC
Britain GPE
British NORP
Outers PRODUCT
Europe LOC
EU ORG
790,000 CARDINAL
UK GPE
2030 DATE
58bn MONEY
Government ORG
Britain GPE
Europe LOC
Europe LOC
Europe LOC
Britain GPE
Europe LOC
Europe LOC
British NORP
Straw ORG
Britain Stronger ORG
Europe LOC
Coronavirus ORG
Last week DATE
the days DATE
Sanders ORG
two CARDINAL
Wednesday DATE
Vermont GPE
Biden PERSON
Sanders ORG
Faiz Shakir PERSON
Jeff Weaver PERSON
Biden PERSON
two CARDINAL
Anita Dunn PERSON
Ron Klain PERSON
Vermont GPE
last week DATE
Sanders ORG
Biden PERSON
Sanders ORG
Democrats NORP
2016 DATE
Sanders ORG
Hillary Clinton PERSON
2016 DATE

Democrats NORP
Trump ORG
November DATE
Clinton PERSON
Sanders ORG
this year DATE
Sanders ORG
Biden PERSON
Sanders ORG
Sanders ORG
Biden PERSON
2016 DATE
Sanders ORG
Biden PERSON
Sanders ORG
Democratic NORP
Bernie PERSON
Biden PERSON
January 2016 DATE
Hillary PERSON
Biden PERSON
Bernie PERSON
Monday DATE
Biden PERSON
recent weeks DATE
Sanders ORG
The Center for Disease Control and Prevention ORG
CDC ORG
COVID-19 ORG
Tuesday DATE
COVID-19 PERSON
COVID-19 ORG
PUNK PERSON
Dan Bongino PERSON
CoV PERSON
COVID-19 PERSON
Middle East Respiratory Syndrome LOC
Severe Acute Respiratory Syndrome ORG
nCoV ORG
CDC ORG
J.B. Neiman PERSON
Texas GPE
13 CARDINAL
New York Times ORG
Alex Berenson PERSON
COVID-19 ORG
Neiman ORG
Berenson PERSON
Covid PERSON
Covid PERSON

Covid PERSON
June 27 DATE
623 CARDINAL
COVID-19 ORG
OurWorldInData.com ORG
Two days later DATE
U.S. GPE
265 CARDINAL
4,928 CARDINAL
U.S. GPE
the peak day DATE
April 16 DATE
June 26 DATE
2,437 CARDINAL
1,270 CARDINAL
Tuesday DATE
at least one CARDINAL
yesterday DATE
more than 600 CARDINAL
NYC LOC
three weeks DATE
week over week DATE
about 25% PERCENT
Berenson PERSON
Twitter PRODUCT
South African NORP
Alfred Ndlovu PERSON
30 days DATE
Jesus Christ PERSON
40 days DATE
40 nights DATE
Buzz South Africa PERSON
44-year-old DATE
June 17 DATE
bush PERSON
Jesus PERSON
Jesus Christ PERSON
40 days DATE
Alfred Ndlovu PERSON
just a month DATE
One CARDINAL
2015 DATE
the year DATE
just the past few months DATE
Pakistan GPE
India GPE
more than 1,000 CARDINAL
Washington GPE
Olympic National Park FAC
first ORDINAL

London GPE
98 degrees QUANTITY
Fahrenheit WORK_OF_ART
July day DATE
U.K. GPE
Guardian ORG
California GPE
a millennium DATE
50-acre QUANTITY
a matter of hours TIME
I-15 FAC
a few days later DATE
summer DATE
Puerto Rico GPE
El Niño ORG
Pacific Ocean LOC
July 20th DATE
James Hansen PERSON
NASA ORG
the summer of 1988 DATE
Antarctica LOC
10 CARDINAL
10 feet QUANTITY
2065 DATE
Eric Rignot PERSON
NASA ORG
the University of California-Irvine ORG
Hansen PERSON
two CARDINAL
Hansen PERSON
Greenland GPE
Michael Mann PERSON
Atlantic LOC
the United States GPE
Europe LOC
one CARDINAL
Hansen PERSON
the years DATE
Mann ORG
winters DATE
the East Coast LOC
Earth LOC
this year DATE
this year DATE
the North Pacific LOC
Apocalypse Soon PERSON
Eighty-year-old DATE
Roger Thomas PERSON
San Francisco GPE
earlier this year DATE

Thomas PERSON
25 CARDINAL
three CARDINAL
July 4th DATE
115 CARDINAL
a single hour TIME
the Farallon Islands LOC
California GPE
Thomas PERSON
Last fall DATE
Alaska GPE
the Arctic where LOC
Shell ORG
35,000 CARDINAL
Monterey Bay GPE
California GPE
last summer DATE
Catalina Island LOC
1,000 miles QUANTITY
California GPE
Mexican NORP
one CARDINAL
the Pacific Northwest LOC
Every two weeks DATE
Bill Peterson PERSON
the National Oceanic and Atmospheric Administration’s ORG
Northwest Fisheries Science Center ORG
Oregon GPE
this year DATE
West Coast LOC
hundreds of miles QUANTITY
Earth LOC
This year DATE
California GPE
One CARDINAL
Chinook PRODUCT
the next few years DATE
Peterson PERSON
the past months DATE
the Pacific Decadal Oscillation ORG
the North Pacific LOC
PDO ORG
15 to 20 years DATE
El Niños ORG
year DATE
the next few years DATE
PDO ORG
Peterson PERSON
the last few years DATE
Peterson PERSON

July DATE
Stephanie Dutkiewicz PERSON
MIT ORG
Hansen PERSON
Dutkiewicz NORP
Dutkiewicz ORG
just decades DATE
Jacquelyn Gill PERSON
the University of Maine ORG
n’t GPE
less than a decade DATE
the past year DATE
two CARDINAL
the North Pacific LOC
Hawaii GPE
Alaska GPE
Baja California GPE
El Niño ORG
this year DATE
Arctic sea LOC
the past few years DATE
California GPE
Northeast LOC
the past two years DATE
the West Coast LOC
the past 18 months DATE
Daniel Swain PERSON
Stanford University ORG
the “Ridiculously Resilient Ridge LAW
the North Pacific LOC
years DATE
Earth LOC
Pacific LOC
the North Pacific LOC
2014 DATE
2015 DATE
the El Niño ORG
Pacific LOC
Six percent PERCENT
the end of the decade DATE
One CARDINAL
Atlantic Ocean LOC
Greenland GPE
four-inch QUANTITY
Northeast LOC
just two years DATE
2009 DATE
2010 DATE
last year DATE
the West Coast LOC

One CARDINAL
Nina Bednaršek PERSON
Science ORG
more than a dozen CARDINAL
this July DATE
this century DATE
as little as a decade DATE
nearly half CARDINAL
James Barry PERSON
the Monterey Bay Aquarium Research Institute ORG
California GPE
the Seattle Times ORG
2013 DATE
several decades DATE
every 10 years DATE
1960 DATE
the coming decades DATE
the 21st century DATE
thousands of miles QUANTITY
the past year DATE
Long Island GPE
Phoenix GPE
Detroit GPE
Baltimore GPE
Houston GPE
Pensacola GPE
Florida GPE
250 million years ago DATE
Great Dying WORK_OF_ART
more than 90 percent PERCENT
Great Dying WORK_OF_ART
hundreds of thousands CARDINAL
100 years or DATE
Sarah Moffitt PERSON
the University of California-Davis ORG
the ocean thousands of years CARDINAL
today DATE
Moffitt ORG
two dozen CARDINAL
Simone Alin PERSON
NOAA ORG
Pacific Marine Environmental Laboratory ORG
Seattle GPE
The Puget Sound EVENT
the coming decades DATE
Alin PERSON
Dutkiewicz NORP
Alin PERSON
every day DATE
four years ago DATE

Katharine Hayhoe PERSON
Christian NORP
Canada GPE
Texas GPE
Canada GPE
James Hansen PERSON
NASA ORG
2013 DATE
Hansen PERSON
tomorrow DATE
the United States GPE
China GPE
Congress ORG
Hansen PERSON
July DATE
Hansen PERSON
U.S.-China NORP
United Nations ORG
Paris GPE
One CARDINAL
Hansen PERSON
Our Children’s Trust ORG
U.S. GPE
EPA ORG
the Clean Air Act LAW
years DATE
U.S. GPE
West LOC
New Jersey GPE
The National Oceanic and Atmospheric Administration ORG
NOAA ORG
U.S. GPE
NOAA ORG
NOAA ORG
winter DATE
NOAA ORG
January 2018 DATE
Northeast U.S. LOC
New England LOC
NY ORG
PA GPE
NJ ORG
DE GPE
MD GPE
Paul Homewood PERSON
NOAA ORG
NOAA ORG
3.1 degrees QUANTITY
first ORDINAL
New York GPE

NOAA ORG
winter DATE
New York GPE
2013/14 CARDINAL
Jan 2nd DATE
Arctic LOC
March DATE
NWS ORG
the end of the winter DATE
The winter of DATE
2013 CARDINAL
New York State GPE
Western NORP
North Central New York ORG
This winter DATE
two CARDINAL
winters DATE
this winter DATE
this winter DATE
normal every month DATE
January through March DATE
at least 4 CARDINAL
two CARDINAL
Western New York ORG
Buffalo GPE
Rochester GPE
the month of January DATE
NOAA ORG
30th ORDINAL
1895 DATE
New York State GPE
16.9F CARDINAL
January 1943 and January 2014 – months DATE
NOAA ORG
NOAA ORG
2014 DATE
NOAA ORG
1943 DATE
Jan 2014 DATE
2.7F CARDINAL
1943 DATE
NOAA ORG
NOAA ORG
Homewood GPE
Big Freeze PERSON
the winter of 2017/2018 DATE
NOAA ORG
January 2018 DATE
New York GPE
January 1943 DATE

three CARDINAL
Ithaca GPE
Auburn FAC
Geneva GPE
January 2018 DATE
January 1943 DATE
1.0 CARDINAL
1.7 DATE
1.3F CARDINAL
NOAA ORG
2.1F warmer last month DATE
NOAA ORG
last month DATE
1943 DATE
NOAA ORG
Central Lakes LOC
one CARDINAL
US GPE
NOAA ORG
One CARDINAL
NOAA ORG
Syracuse GPE
1929 DATE
Homewood ORG
two CARDINAL
two million CARDINAL
the Urban Heat Island ORG
NOAA ORG
NOAA ORG
U.S. GPE
885 CARDINAL
the Association of School and College Leaders ORG
73 per cent MONEY
84 per cent MONEY
ASCL ORG
Allan Foulds PERSON
75 per cent MONEY
33 per cent MONEY
25 per cent MONEY
14 per cent MONEY
89 per cent MONEY
Last month DATE
the National Audit Office ORG
the Department for Education ORG
the past four years DATE
Michael Wilshaw PERSON
last week DATE
ASCL ORG
annual DATE
this weekend DATE

Education ORG
Nicky Morgan PERSON
Malcolm Trobe PERSON
ASCL ORG
Please ORG
Please ORG
# CARDINAL
The Independent ORG
# CARDINAL
The Independent ORG
Teachers ORG
the Department for Education ORG
ASCL ORG
Dominic Raab PERSON
Muhammad Ali PERSON
Mother Teresa PERSON
20% PERCENT
the past 25 years DATE
Britain GPE
post-Brexit NORP
The Sunday Times ORG
Dominic Raab PERSON
the Migration Advisory Committee ORG
MAC ORG
this autumn DATE
Raab PERSON
Brexiteer PERSON
Tory NORP
first ORDINAL
Tory NORP
Bill Clinton PERSON
Clinton Health Access Initiative ORG
Africa LOC
The Daily Caller News Foundation ORG
titled,“The Clinton Foundation ORG
The India Success Story ORG
Marsha Blackburn PERSON
Tennessee GPE
Republican NORP
the House Energy and Commerce Committee ORG
one CARDINAL
the Clinton Foundation’s ORG
Clinton PERSON
decade DATE
Indian NORP
Ranbaxy ORG
Dinesh Thakur ORG
Ranbaxy ORG
U.S. GPE
Indian NORP

U.S. GPE
U.S. GPE
2013 DATE
seven CARDINAL
The Department of Justice ORG
$500 million MONEY
the District of Maryland GPE
U.S. GPE
the District of Maryland GPE
Rod J. Rosenstein PERSON
FDA ORG
Stuart F. Delery PERSON
the Department of Justice ORG
Indian NORP
The Department of Justice ORG
Blackburn PERSON
Clinton PERSON
the final weeks DATE
Hillary Clinton PERSON
Blackburn PERSON
the Department of Health and Human Services ORG
the Department of State ORG
Hillary PERSON
Barack Obama PERSON
first ORDINAL
Bill PERSON
two CARDINAL
Indian-Americans NORP
the Food and Drug Administration FDA ORG
the Securities and Exchange Commission ORG
the Clinton Foundation’s ORG
Ranbaxy ORG
Indian NORP
FDA ORG
ProPublica ORG
2007 DATE
the U.S. Agency for International Development ORG
$9 million MONEY
more than $1.8 million MONEY
Roger Bate PERSON
the American Enterprise Institute ORG
Thakur PERSON
WBC ORG
Ranbaxy ORG
first ORDINAL
August 2004 DATE
one year DATE
The World Health Organization ORG
three CARDINAL
South Africa GPE

FDA ORG
2006 DATE
Ranbaxy ORG
U.S. GPE
Bill PERSON
2013 DATE
Mumbai GPE
millions CARDINAL
the Clinton Foundation ORG
Ranbaxy ORG
Thakur PERSON
Clinton Foundation ORG
the Clinton Foundation ORG
the Clinton Foundation ORG
2010 DATE
Bill Clinton PERSON
Charles Ortel PERSON
the Clinton Foundation ORG
Bill Clinton PERSON
Ortel ORG
Bill PERSON
2000 DATE
U.S. GPE
October 2003 DATE
four CARDINAL
Cipla of Mumbai ORG
India GPE
Matrix Labs PERSON
Hydrabad GPE
India GPE
Aspen GPE
Pharmacare GPE
Johannesburg GPE
South Africa GPE
U.S. GPE
$15 billion MONEY
George W. Bush PERSON
U.S. GPE
Clinton PERSON
four CARDINAL
10 CARDINAL
three CARDINAL
FDA ORG
13 CARDINAL
Foundation ORG
several years DATE
Clinton PERSON
the Clinton Foundation ORG
One CARDINAL
the American Indian Foundation ORG

Clinton PERSON
Indian-American NORP
Rajat Gupta PERSON
Vinod Gupta PERSON
2001 DATE
Rajat PERSON
2012 DATE
Vinod GPE
InfoGroup ORG
$9 million MONEY
Securities and Exchange Commission ORG
One CARDINAL
Vinod ORG
$3.3 million MONEY
Blackburn PERSON
the Clinton Foundation ORG
Follow Richard PERSON
Twitter PRODUCT
Content ORG
The Daily Caller News Foundation ORG
White House ORG
Sarah Huckabee Sanders PERSON
Monday DATE
Sanders ORG
the White House ORG
Sanders ORG
Chicago GPE
over 4,000 CARDINAL
last year DATE
the White House ORG
Democrats NORP
Las Vegas GPE
Chicago GPE
Trump ORG
last fall DATE
2013 DATE
4,368 CARDINAL
Chicago GPE
last year DATE
2,877 CARDINAL
this year DATE
Sanders ORG
Hillary Clinton’s PERSON
House ORG
one CARDINAL
Stephen Paddock ORG
Las Vegas GPE
Sanders ORG
The Washington Post’s ORG
Philip Bump ORG

Sanders ORG
last night TIME
Clinton PERSON
NBC ORG
Hallie Jackson PERSON
Orlando GPE
that day DATE
Twitter PERSON
Jackson PERSON
the White House ORG
Trump ORG
Trump ORG
today DATE
Sanders ORG
Democrats NORP
the White House ORG
Bois State University ORG
The Daily Wire ORG
Thursday DATE
Wilson ORG
Boise GPE
Idaho GPE
Abraham Lincoln PERSON
Terry Joe Wilson PERSON
37 DATE
late Tuesday night TIME
last week DATE
The Idaho Statesman ORG
Wilson ORG
Boise State University ORG
the Boise Black Lives Matter ORG
February DATE
Lincoln ORG
Julia Davis Park PERSON
Lincoln ORG
The Idaho Statesman ORG
Parks PERSON
the Boise Police Department ORG
Wilson PERSON
marijuana PERSON
Wilson ORG
Wilson ORG
marijuana PERSON
Wilson ORG
the Ada County Jail LOC
Wilson PERSON
marijuana PERSON
The Idaho Statesman ORG
Wilson ORG
Boise State University ORG

Wilson’s Instagram ORG
November DATE
Donald Tump PERSON
Joe Biden PERSON
Black Lives Matter WORK_OF_ART
decades DATE
ASBSU ORG
Matter ORG
# CARDINAL
#ASBSU#fuckwhitesupremacy#fuckbigcitycoffee
@astridwilde#blacklivesmatter # MONEY
# CARDINAL
August 2020 DATE
Council ORG
Elaine Clegg PERSON
The Idaho Statesman ORG
Wilson ORG
the Boise Black Lives Matter ORG
Boise PERSON
Council ORG
Elaine Clegg PERSON
North End GPE
Terry Wilson II PERSON
Boise PERSON
Clegg GPE
four CARDINAL
Lauren McLean PERSON
2021 DATE
2021 CARDINAL
the Boise Police Department ORG
2020 DATE
Wilson ORG
Clegg PERSON
GameStop PRODUCT
Auntie Anne’s LOC
the United States GPE
4 MONEY
last summer DATE
20 MONEY
the end of 2020 DATE
40 MONEY
two weeks ago DATE
100 MONEY
Monday DATE
Tuesday DATE
close to $300 MONEY
one CARDINAL
GameStop PRODUCT
months DATE
many millions of dollars MONEY

GameStop PRODUCT
Ryan Cohen PERSON
Chewy.com ORG
millions CARDINAL
GameStop GPE
last year DATE
A few weeks ago DATE
14 CARDINAL
GameStop PRODUCT
401k PRODUCT
Elon Musk PERSON
Jim Cramer PERSON
one CARDINAL
the United States GPE
the 1990s DATE
the last few years DATE
Sony ORG
Microsoft ORG
Nintendo ORG
GameStop PRODUCT
GameStop PRODUCT
COVID-19 ORG
a third CARDINAL
the first quarter of 2020 DATE
GameStop PRODUCT
Elon Musk PERSON
millions CARDINAL
hundreds of millions of dollars MONEY
Advertisement Sometimes PERSON
20 MONEY
one CARDINAL
20 MONEY
$1 million MONEY
999,980 MONEY
Melvin Capital Management ORG
millions of dollars MONEY
GameStop PRODUCT
Citron Research ORG
GameStop PRODUCT
the last several months DATE
GameStop PRODUCT
GameStop PRODUCT
MELVIN ORG
CITRON ORG
Citron GPE
Melvin PERSON
the last few months DATE
GameStop PRODUCT
GameStop PRODUCT
earlier this month DATE

309 percent PERCENT
Ryan Cohen PERSON
Chewy GPE
Cohen PERSON
Cohen PERSON
nearly 10 percent PERCENT
GameStop PRODUCT
September of last year DATE
nearly 13 percent PERCENT
December DATE
GameStop ORG
GME ORG
Melvin PERSON
Citron GPE
GameStop PRODUCT
billions CARDINAL
hundreds of thousands or millions CARDINAL
many hundreds or thousands CARDINAL
Advertisement Anyways PERSON
Reddit NORP
Reddit NORP
Discord GPE
Bloomberg Terminal ORG
4chan CARDINAL
401k PRODUCT
a ton QUANTITY
YOLO ORG
Ouija NORP
the last 24 months DATE
hundreds of thousands CARDINAL
millions CARDINAL
WALLSTREETBETS PERSON
September 2019 DATE
DeepFuckingValue ORG
53,000 MONEY
GameStop PRODUCT
the preceding few months DATE
between 30 cents and 75 cents MONEY
that day DATE
113,000 MONEY
86 percent PERCENT
85 cents MONEY
the months DATE
RoaringKitty ORG
YouTube ORG
YouTube ORG
GameStop PRODUCT
GameStop PRODUCT
Advertisement Other PERSON
GameStop PRODUCT

more than a million CARDINAL
DeepFuckingValue ORG
December DATE
around $4 MONEY
GameStop PRODUCT
GameStop PRODUCT
Discord GPE
Melvin Capital PERSON
Citron Research ORG
YouTube ORG
GameStop PRODUCT
GameStop PRODUCT
Mars LOC
Redditor PERSON
DeepFuckingValue ORG
50k MONEY
100k MONEY
several million CARDINAL
$10 million MONEY
$20 million MONEY
Melvin PERSON
this week DATE
Melvin Capital PERSON
Earlier this week DATE
$2.75 billion MONEY
GameStop PRODUCT
GameStop PRODUCT
Elon Musk PERSON
Tesla ORG
GameStop PRODUCT
Discord GPE
GameStop PRODUCT
over $1,000 MONEY
5,000 MONEY
GME ORG
350 MONEY
hundreds of thousands or millions CARDINAL
all day every day DATE
NOKIA ORG
BLACKBERRY ORG
AMC ORG
ETC ORG
Nokia ORG
Best Buy GPE
AMC Theaters ORG
GameStop PRODUCT
GameStop PRODUCT
BAD ORG
the Donald Trump-ification FAC
Anthony Scaramucci PERSON

the "French Revolution EVENT
decades DATE
Goldman Sachs ORG
Fidelity ORG
this morning TIME
US GPE
thousands of dollars MONEY
GameStop PRODUCT
thousands of dollars MONEY
millions CARDINAL
U.S. GPE
Robinhood PRODUCT
SEC ORG
Fidelity ORG
Blackrock GPE
tens of millions MONEY
GameStop PRODUCT
Vaccine PERSON
Merck ORG
two CARDINAL
V590 PRODUCT
V591 PRODUCT
MK-7110 and MK-4482 DATE
greater than 50 percent PERCENT
COVID-19 PERSON
MK-7110 PERSON
Merck ORG
around $356 million MONEY
US GPE
Operation Warp Speed LAW
Michael Nally PERSON
Bloomberg PERSON
Merck ORG
some 20 million CARDINAL
MK-4482 DATE
five days DATE
German NORP
UK GPE
Oxford NORP
less than 8% PERCENT
65s DATE
German NORP
German NORP
UK GPE
the European Union ORG
AstraZeneca ORG
EU ORG
Britain GPE
British NORP
EU ORG

Pfizer ORG
UK GPE
Covid PERSON
Paul Nuttall PERSON
Labour ORG
SUBSCRIBE Invalid PRODUCT
Leave GPE
EU ORG
Mr Nuttall PERSON
39 per cent MONEY
Labour's ORG
Europhile ORG
Gareth Snell PERSON
second ORDINAL
33 per cent MONEY
Tory PERSON
Jack Brereton PERSON
Remain PERSON
third ORDINAL
Zulfiqar Ali PERSON
EU ORG
Ukip GPE
Arron Banks PERSON
last month DATE
Labour ORG
Tristam Hunt PERSON
GETTY Ukip's ORG
Paul Nuttall PERSON
the Stoke Central ORG
Nutall PERSON
Nigel Farage PERSON
Ukip GPE
November DATE
Eurosceptic LOC
Brexit PERSON
last year's DATE
EU ORG
Labour ORG
Brexit PERSON
Twitter PRODUCT
Eurosceptic LOC
Nigel Farage PERSON
April 3, 2017 DATE
Nigel Farage PERSON
British NORP
the UK Independence Party ORG
October 2016 DATE
Getty 1 PERSON
48 CARDINAL
Nigel Farage PERSON

February 23 DATE
Nuttall PERSON
Stoke LOC
England GPE
Stoke Central LOC
Remain PERSON
GETTY ORG
Nuttall PERSON
Stoke LOC
Ukip  Mr Farage ORG
Earlier this week DATE
Nuttall PERSON
Donald Trump PERSON
Muslim NORP
US GPE
UK GPE
Nigel Farage PERSON
the Mental Health Taskforce ORG
Paul Farmer PERSON
2021 DATE
70,000 CARDINAL
30,000 CARDINAL
25% PERCENT
10% PERCENT
Health ORG
Jeremy Hunt PERSON
7 day DATE
first ORDINAL
One CARDINAL
4 CARDINAL
NHS ORG
£105 billion MONEY
every year DATE
2010 DATE
NHS ORG
£11.7 billion MONEY
last year DATE
first ORDINAL
£1.4 billion MONEY
Alistair Burt PERSON
William Happer PERSON
Princeton University ORG
Highlights of the video FAC
first ORDINAL
2:39 TIME
1998 DATE
Happer NORP
3:48 TIME
A thousand CARDINAL
ppm ORG

CO2 ORG
Several thousand CARDINAL
Happer NORP
several thousand CARDINAL
million CARDINAL
roughly 70 to 80 million years ago MONEY
3,000 CARDINAL
ten CARDINAL
several thousand CARDINAL
Glacier Bay LOC
CO2 Happer PERSON
Alaska GPE
Glacier Bay LOC
the 1800s DATE
John Muir PERSON
the Sierra Club ORG
Glacier Bay LOC
1879 DATE
Happer PERSON
Harrison H. Schmitt PERSON
In Defense of Carbon Dioxide WORK_OF_ART
The Wall Street Journal ORG
mid 2013 DATE
2014 DATE
Taliban ORG
US GPE
Afghanistan GPE
the Daesh Takfiri ORG
Taliban ORG
Iran GPE
Afghan NORP
Tehran GPE
Monday DATE
US GPE
Afghanistan GPE
Daesh GPE
Taliban ORG
Daesh GPE
Nangarhar GPE
Kunar PERSON
Taliban ORG
Suhail Shaheen PERSON
Taliban ORG
Taliban ORG
American NORP
Afghan NORP
Americans NORP
Taliban ORG
Afghanistan GPE
US GPE

Taliban ORG
last February DATE
US GPE
Americans NORP
Afghanistan GPE
intra-Afghan NORP
nightly DATE
Taliban ORG
Afghanistan GPE
US GPE
Afghanistan GPE
Afghanistan GPE
20 years DATE
Taliban ORG
Last week DATE
Taliban ORG
Tehran GPE
the Iranian Foreign Ministry ORG
the past months DATE
Iran GPE
Afghanistan GPE
decades DATE
intra-Afghan NORP
the United States GPE
12,000 CARDINAL
US GPE
Afghanistan GPE
Taliban ORG
two CARDINAL
February 2020 DATE
recent months DATE
Afghanistan GPE
Taliban ORG
Afghan NORP
US GPE
Kabul GPE
Taliban ORG
US GPE
Joe Biden PERSON
last year DATE
US GPE
first ORDINAL
Afghanistan GPE
2001 DATE
Taliban ORG
one CARDINAL
US GPE
Afghanistan GPE
Washington GPE
more than two trillion dollars MONEY

Over 2,400 CARDINAL
US GPE
tens of thousands CARDINAL
Afghan NORP
More than 300 CARDINAL
between June and August DATE
under 18 DATE
796 CARDINAL
312 CARDINAL
under 18 DATE
the Press Association ORG
Prevent NORP
Channel ORG
last week DATE
14-year-old DATE
Britain GPE
Australia GPE
the National Police Chiefs’ Council ORG
July DATE
349 CARDINAL
more than 10 CARDINAL
the previous month DATE
327 CARDINAL
120 CARDINAL
August DATE
the summer DATE
between June and August DATE
2012/13 CARDINAL
the first year DATE
England GPE
Wales GPE
the first three months DATE
2014/15 CARDINAL
Channel ORG
2007 DATE
one CARDINAL
five CARDINAL
Rashad Ali PERSON
the Institute for Strategic Dialogue ORG
later this year DATE
David Anderson QC PERSON
Britain GPE
Muslim NORP
last month DATE
14-year-old DATE
Muslim NORP
Isis ORG
London GPE
November 8, 2013 DATE
Natalie Betts PERSON

Global Business Recruitment & Expansion Coordinator ORG
512 CARDINAL
974 CARDINAL
Austin GPE
Austin Orientation Sessions ORG
the year DATE
International Welcome Program ORG
Vietnamese NORP
Korean NORP
Mandarin LANGUAGE
Austin GPE
Nearly 20% PERCENT
Austin GPE
US GPE
Natalie Betts PERSON
the City of Austin’s Global Business Recruitment & Expansion Coordinator GPE
Asian American Resource Center ORG
Taja Beekley PERSON
the Asian American Resource Center ORG
The City’s Economic Development Global Business Recruitment and Expansion Division ORG
the International Welcome Program ORG
January of this year DATE
InternationalAustin.org ORG
Austin Orientation Session ORG
Saturday, November 9 DATE
Asian American Resource Center ORG
TX ORG
78701 DATE
the City of Austin Economic Development Department GPE
The Economic Department ORG
Austin PERSON
Austin GPE
Austin Orientation PERSON
this Saturday DATE
November 9 DATE
10am to 4pm TIME
the Asian American Resource Center ORG
8401 DATE
Austin GPE
Austin Orientation Sessions ORG
the year DATE
International Welcome Program ORG
Vietnamese NORP
Korean NORP
Mandarin LANGUAGE
Nearly 20% PERCENT
Austin GPE
US GPE

Natalie Betts PERSON
the City of Austin’s Global Business Recruitment & Expansion Coordinator GPE
Asian American Resource Center ORG
Taja Beekley PERSON
the Asian American Resource Center ORG
The City’s Economic Development Global Business Recruitment and Expansion Division ORG
the International Welcome Program ORG
January of this year DATE
InternationalAustin.org ORG
Saturday, November 9 DATE
10am TIME
American Resource ORG
Cameron RoadAustin PERSON
TX 78701The Economic Department ORG
Austin PERSON
# # # MONEY
Pennsylvania GPE
372,000 CARDINAL
One CARDINAL
11 CARDINAL
October 17, 2020 DATE
Pennsylvania GPE
approximately 372,000 CARDINAL
ProPublica ORG
the Philadelphia Inquirer ORG
Friday DATE
372,000 CARDINAL
one CARDINAL
five CARDINAL
90% PERCENT
about 336,000 CARDINAL
November DATE
Pennsylvanians NORP
June DATE
November DATE
second ORDINAL
June DATE
Bill Turner PERSON
Chester County GPE
Marybeth Kuznik PERSON
Armstrong County GPE
Pittsburgh GPE
Pennsylvania GPE
the Pennsylvania Department of State ORG
seven days DATE
hundreds CARDINAL
three CARDINAL
One CARDINAL

11 CARDINAL
Over 2.7 million CARDINAL
Pennsylvanians NORP
this week DATE
nearly 30,000 CARDINAL
Pennsylvania GPE
About 28,879 CARDINAL
Allegheny County GPE
Pittsburgh GPE
Midwest Direct LOC
Fox News ORG
Thursday DATE
Earlier this week DATE
over 100 CARDINAL
Kentucky GPE
the U.S. Postal Service Office ORG
U.S. Postal Service ORG
England GPE
next April DATE
more than a year DATE
5bn ORDINAL
each year DATE
England GPE
more than 300 CARDINAL
2bn ORDINAL
UK GPE
EU ORG
more than 80% PERCENT
nine CARDINAL
10 CARDINAL
Michael Gove PERSON
just a few minutes TIME
hundreds of years DATE
Campaigners PERSON
Hugo Tagholm PERSON
Surfers Against Sewage ORG
Emma Priestland PERSON
Friends of the Earth ORG
three CARDINAL
centuries DATE
U.S. GPE
U.S. GPE
almost 70% PERCENT
between 1970 and 2014 DATE
about two pounds QUANTITY
the next century DATE
as much as one CARDINAL
1.8 degrees QUANTITY
two CARDINAL
more than £40,000 MONEY

Four CARDINAL
five 79% PERCENT
England GPE
the previous week DATE
58% PERCENT
the week DATE
NHS Digital England ORG
the Office for National Statistics ORG
The ONS Opinions ORG
Lifestyle Survey PERSON
Great Britain GPE
2017 DATE
the week DATE
England GPE
the previous week DATE
57.8% PERCENT
Scotland GPE
53.5% PERCENT
Wales GPE
50.0% PERCENT
English LANGUAGE
fifth ORDINAL
between 25 and 64 DATE
65 DATE
Friday DATE
Saturday DATE
nights TIME
the week DATE
previous years DATE
the Health Survey for England ORG
every day DATE
Steve Clarke PERSON
the Priory Group ORG
four CARDINAL
five, six days DATE
2016-17 DATE
England GPE
337,000 CARDINAL
17% PERCENT
fifth ORDINAL
2006-07 DATE
Tuesday DATE
Scotland GPE
ONS ORG
England GPE
Scotland GPE
Scotland GPE
MUP ORG
England GPE
Ian Gilmore PERSON

the Alcohol Health Alliance ORG
more than 50 CARDINAL
England GPE
Scotland GPE
more than 23,000 CARDINAL
England GPE
MUP PERSON
MUP ORG
England GPE
Scotland GPE
50p DATE
two CARDINAL
at least £1 MONEY
nine CARDINAL
at least £ MONEY
4.50 MONEY
Justine Greening PERSON
Monday DATE
Theresa May PERSON
the Department for Work and Pensions ORG
May DATE
Greening ORG
Damian Hinds ORG
Earlier May TIME
Jeremy Hunt PERSON
Hunt PERSON
May DATE
winter DATE
NHS ORG
David Lidington PERSON
the Cabinet Office ORG
May DATE
Hunt PERSON
May DATE
Two CARDINAL
Sajid Javid PERSON
Greg Clark PERSON
Javid PERSON
May DATE
Clark PERSON
Twitter PERSON
Chris Grayling PERSON
Brandon Lewis PERSON
Amber Rudd ORG
Philip Hammond PERSON
David Davis PERSON
Brexit PERSON
Boris Johnson PERSON
Downing Street FAC
Gavin Williamson PERSON

Karen Bradley PERSON
Northern Ireland GPE
James Brokenshire PERSON
Lidington PERSON
Commons ORG
Tory PERSON
the Cabinet Office ORG
the Duchy of Lancaster ORG
David Gauke PERSON
the justice department ORG
Lidington PERSON
just six months DATE
Gauke PERSON
sixth ORDINAL
Conservatives NORP
2010 DATE
Downing Street ORG
Lidington PERSON
first ORDINAL
state ORG
May DATE
Green PERSON
December DATE
2008 DATE
Lidington ORG
Cabinet Office ORG
Green PERSON
first ORDINAL
first ORDINAL
the day DATE
Brokenshire PERSON
Grayling PERSON
Conservative Twitter PERSON
Grayling GPE
Chris Grayling PERSON
Conservative ORG
Lewis PERSON
BBC ORG
Grayling ORG
no10 PRODUCT
Conservatives NORP
Anushka Asthana PERSON
January 8, 2018 DATE
About an hour later TIME
Downing Street FAC
Lewis PERSON
the Home Office ORG
Patrick McLoughlin PERSON
May DATE
P45 PRODUCT

Brokenshire GPE
May DATE
Northern Ireland GPE
Brokenshire GPE
May DATE
Brokenshire PERSON
James Brokenshire PERSON
Jack Taylor/Getty Images PERSON
James Cleverly PERSON
Braintree GPE
2015 DATE
Chris Skidmore PERSON
Ben Bradley PERSON
Kemi Badenoch GPE
June DATE
Bradley ORG
27 DATE
Maria Caulfield PERSON
Rehman Chishti PERSON
Helen Grant PERSON
Andrew Jones PERSON
Marcus Jones PERSON
James Morris PERSON
Skidmore GPE
Andrew Jones PERSON
Marcus Jones PERSON
the Cabinet Office ORG
Andrew Jones PERSON
Treasury ORG
Marcus Jones PERSON
Oxford University ORG
British NORP
first ORDINAL
the year DATE
Tuesday DATE
Welfare and Equal Opportunity ORG
Oxford SU Disabilities Campaign ORG
Welfare and Equal Opportunity ORG
Oxford SU Disabilities Campaign ORG
2015 DATE
One CARDINAL
Oxford University Student Union ORG
British NORP
the Sheldonian Theatre ORG
Oxford GPE
the University of Manchester Students Union ORG
September last year DATE
Ms McCallion PERSON
British Sign Language LOC
Last year DATE

0.04 Celsius PERSON
2016 DATE
the hottest year DATE
The National Oceanic and Atmospheric Administration ORG
NOAA ORG
this week DATE
last year DATE
137-year DATE
2016 DATE
the third consecutive year DATE
2016 DATE
2015 DATE
0.04 degrees Celsius QUANTITY
previous years DATE
last year DATE
2015 DATE
2016 DATE
years DATE
El Niño ORG
Pacific LOC
The Cato Institute’s ORG
Patrick Michaels PERSON
1998 DATE
Fahrenheit WORK_OF_ART
the following years DATE
2015 DATE
the end of 2016 DATE
2014 DATE
NOAA ORG
El Niño’s ORG
Judith Curry PERSON
Congress ORG
2014 DATE
U.N. Intergovernmental Panel on Climate Change ORG
0.2 degrees Celsius QUANTITY
the early 21st century DATE
the first 15 years DATE
0.05 degrees Celsius QUANTITY
more than 40% PERCENT
1900 DATE
between 1910 and 1945 DATE
only 10% PERCENT
1040ET CARDINAL
Feigl-Ding PERSON
January 31, 2020 DATE
the past few days DATE
as many as 70,000 CARDINAL
Wuhan GPE
China GPE
Canadian NORP

the Institute of Virology ORG
Wuhan GPE
Wuhan GPE
Peng Zhou PERSON
China GPE
Hunan GPE
2019 DATE
HIV-1 CARDINAL
Gag PERSON
Indian NORP
RNA ORG
first ORDINAL
2019 DATE
four CARDINAL
the Wuhan Coronavirus ORG
2019 DATE
nCoV ORG
Fig1 NORP
2019 DATE
2019 DATE
2019- CARDINAL
nCoV ORG
4 CARDINAL
one CARDINAL
2019-nCoV[Fig DATE
4 CARDINAL
1 CARDINAL
2 DATE
3 CARDINAL
4 CARDINAL
2019 DATE
China GPE
three CARDINAL
China GPE
three CARDINAL
Zhou et PERSON
al. GPE
2020 DATE
Wuhan GPE
2019 DATE
Bat PERSON
Fig PERSON
S4 PRODUCT
4 CARDINAL
2019 DATE
nCoV ORG
2019 DATE
100% PERCENT
four CARDINAL
HIV-1 PRODUCT

2019 DATE
HIV-1 CARDINAL
Table 1 FAC
first ORDINAL
3 CARDINAL
1,2 CARDINAL
3 CARDINAL
HIV-1 CARDINAL
4 CARDINAL
HIV-1 Gag WORK_OF_ART
1 6 CARDINAL
2 6 CARDINAL
2019 DATE
100% PERCENT
HIV-1 CARDINAL
3 12 CARDINAL
2019- CARDINAL
nCoV ORG
HIV-1 CARDINAL
Table 1 PRODUCT
4 8 CARDINAL
HIV-1 Gag WORK_OF_ART
2019 DATE
nCoV ORG
two CARDINAL
MultiAlin GPE
four CARDINAL
2019-nCoV- DATE
GTNGTKR ORG
2 CARDINAL
Coronaviridae PERSON
2019 DATE
nCoV ORG
2019 DATE
BLASTp ORG
HIV-1 CARDINAL
HIV-1 PRODUCT
2019 DATE
gp120 ORG
404-409 CARDINAL
462-467 CARDINAL
136 CARDINAL
Gag PERSON
366 CARDINAL
Table 1 PRODUCT
GP120 GPE
CXCR4 and/or CCR5 ORG
2019 DATE
2019-nCoV. DATE
2019 DATE

2019 DATE
Feigl-Ding PERSON
China GPE
China GPE
Indian NORP
Prashant Pradhan PERSON
the Indian Institute of Technology ORG
16 CARDINAL
2019 DATE
HIV-1 CARDINAL
Gag PERSON
January 31 DATE
2020 DATE
17 CARDINAL
4 CARDINAL
16 MONEY
# CARDINAL
Gp120- FAC
January 31, 2020  18 DATE
Eric Feigl-Ding PERSON
January 31, 2020  19 DATE
2 CARDINAL
# CARDINAL
January 31 DATE
2020 DATE
20 CARDINAL
2019 DATE
HIV-1 CARDINAL
Eric Feigl-Ding PERSON
January 31, 2020  21 DATE
2019-nCoV. DATE
2019 DATE
Eric Feigl-Ding PERSON
January 31, 2020   DATE
22 DATE
2019- CARDINAL
nCoV ORG
HIV-1 CARDINAL
Gag PERSON
January 31 DATE
2020 DATE
daily DATE
Feigl-Ding PERSON
Chinese NORP
Indian NORP
2019 DATE
nCov ORG
Chinese NORP
January 31 DATE
2020 DATE

NHS ORG
Europe LOC
14 Jun 2016 DATE
Frances O'Grady PERSON
Today DATE
NHS ORG
Congress House ORG
NHS ORG
NHS ORG
NHS ORG
EU ORG
NHS ORG
NHS ORG
NHS ORG
50,000 CARDINAL
the European Economic Area ORG
9,000 CARDINAL
18,000 CARDINAL
over 2,000 CARDINAL
British NORP
NHS ORG
NHS ORG
NHS ORG
UK GPE
UK GPE
NHS ORG
five-year DATE
Tories NORP
EU ORG
NHS ORG
EU ORG
Working Time Directive ORG
long hours TIME
Brexit PERSON
NHS ORG
Britain GPE
EU ORG
NHS ORG
EU ORG
each year DATE
NHS England LOC
3 months DATE
NHS ORG
as much as £10 billion MONEY
the end of the decade DATE
England GPE
1,000 CARDINAL
155 CARDINAL
NHS ORG
Brexit PERSON

NHS ORG
Boris Johnson PERSON
NHS ORG
Boris Johnson PERSON
NHS ORG
Michael Gove PERSON
NHS ORG
the 21st century DATE
Duncan Smith PERSON
Nigel Farage PERSON
the next ten years DATE
NHS ORG
NHS ORG
EU GPE
US GPE
EU ORG
Leave PERSON
EU ORG
NHS ORG
first ORDINAL
Leave PERSON
GP ORG
Sarah Wollaston PERSON
last week DATE
UK GPE
Remain PERSON
NHS ORG
EU ORG
NHS ORG
EU ORG
British NORP
Just last year DATE
UK GPE
£232 million MONEY
EU ORG
EU ORG
Great Ormond Street children’s FAC
University Hospital Birmingham ORG
NHS Blood FAC
NHS ORG
NHS ORG
NHS ORG
NHS ORG
Leave GPE
NHS ORG
NHS ORG
Tories NORP
Brexit PERSON
UK GPE
NHS ORG

the State Department ORG
Donald Trump PERSON
Mike Pence PERSON
Monday DATE
nine days DATE
Joe Biden PERSON
two CARDINAL
BuzzFeed News ORG
the State Department’s ORG
days DATE
Trump ORG
US GPE
Capitol FAC
Democrats NORP
second ORDINAL
Pence ORG
the 25th Amendment LAW
Trump ORG
Donald J. Trump's PERSON
2021-01-11 DATE
2021-01-11 DATE
Trump ORG
around 3:50 p.m. TIME
404 CARDINAL
One CARDINAL
State ORG
Mike Pompeo PERSON
the State Department ORG
this week DATE
Biden PERSON
Nir Shaviv GPE
Israeli NORP
Jerusalem GPE
Hebrew University’s ORG
earth LOC
Al Gore’s PERSON
Shaviv PERSON
97% PERCENT
97% PERCENT
Shaviv PERSON
100% PERCENT
one CARDINAL
Shaviv PERSON
Israel GPE
Technion University ORG
MIT ORG
the age of 13 DATE
MA ORG
the Israel Defense Force’s ORG
Intelligence ORG

Technion ORG
California Institute of Technology ORG
the Canadian Institute for Theoretical Astrophysics ORG
The Institute for Advanced Study ORG
Princeton GPE
Shaviv GPE
American NORP
Shaviv GPE
Trump ORG
the United Nations Intergovernmental Panel on Climate Change (IPCC ORG
2003 DATE
billions CARDINAL
Shaviv GPE
earth LOC
twentieth-century DATE
roughly eleven years or DATE
2008 DATE
as clear as day DATE
the twentieth century DATE
between half to two-thirds CARDINAL
about 1.0 CARDINAL
Earth LOC
Galactic ORG
Earth LOC
earth LOC
Earth LOC
Roy W. Spencer PERSON
Ph.D. WORK_OF_ART
Today DATE
hundreds of millions of MONEY
the past few decades DATE
Shaviv GPE
A thousand years ago DATE
today DATE
three hundred years ago DATE
first ORDINAL
second ORDINAL
2001 DATE
the last millennium DATE
the twentieth century DATE
Orwellian NORP
Shaviv GPE
the National Science Foundation ORG
Shaviv GPE
Shaviv GPE
third ORDINAL
the World Bank ORG
NASA ORG
the past 200 years DATE
Earth LOC

the World Meteorological Organisation ORG
Cedro ORG
Quixadá GPE
Brazil GPE
2016 DATE
the hottest year DATE
2017 DATE
the World Meteorological Organisation ORG
WMO ORG
2016 DATE
Tuesday DATE
El Niño ORG
2016 DATE
US GPE
February DATE
El Niño ORG
2017 DATE
David Carlson PERSON
WMO ORG
Earth LOC
Jeffrey Kargel PERSON
the University of Arizona ORG
US GPE
Prof David Reay PERSON
the University of Edinburgh ORG
Donald Trump PERSON
Trump ORG
Republicans NORP
Congress ORG
Robert Watson PERSON
UK GPE
University of East Anglia ORG
UN ORG
Watson PERSON
Trump ORG
WMO ORG
Petteri Taalas PERSON
1880 DATE
about 115,000 years ago DATE
4m years DATE
2017 DATE
US GPE
February DATE
Australia GPE
Arctic ice LOC
October DATE
six consecutive months DATE
four-decade DATE
Prof Julienne Stroeve PERSON
University College London ORG

UK GPE
Eel PERSON
Lowell’s Cove, FAC
Maine GPE
US GPE
Emily Shuckburgh PERSON
the British Antarctic Survey ORG
Arctic LOC
Greenland GPE
Arctic sea LOC
Europe LOC
Asia LOC
North America LOC
between November 2014 and February 2016 DATE
El Niño ORG
15mm QUANTITY
five years DATE
recent decades DATE
2016 DATE
Arctic LOC
tens CARDINAL
Australia GPE
February DATE
Taalas GPE
Robin Williams PERSON
Last week DATE
Home Office ORG
a decade DATE
Nearly 9 per cent MONEY
16 to CARDINAL
24-year-olds DATE
the past 12 months DATE
Britain GPE
Europe LOC
Last week DATE
the days DATE
Sanders ORG
two CARDINAL
Wednesday DATE
Vermont GPE
Biden PERSON
Sanders ORG
Faiz Shakir PERSON
Jeff Weaver PERSON
Biden PERSON
two CARDINAL
Anita Dunn PERSON
Ron Klain PERSON
Vermont GPE
last week DATE

Sanders ORG
Biden PERSON
Sanders ORG
Democrats NORP
2016 DATE
Sanders ORG
Hillary Clinton PERSON
2016 DATE
Democrats NORP
Trump ORG
November DATE
Clinton PERSON
Sanders ORG
this year DATE
Sanders ORG
Biden PERSON
Sanders ORG
Sanders ORG
Biden PERSON
2016 DATE
Sanders ORG
Biden PERSON
Sanders ORG
Democratic NORP
Bernie PERSON
Biden PERSON
January 2016 DATE
Hillary PERSON
Biden PERSON
Bernie PERSON
Monday DATE
Biden PERSON
recent weeks DATE
Sanders ORG
2020 DATE
Mark PERSON
Trump ORG
America GPE
the early hours of Wednesday November 4 and the days TIME
Americans NORP
First ORDINAL
Trump PERSON
11 million CARDINAL
2016 DATE
third ORDINAL
Obama PERSON
2012 DATE
3.5 million CARDINAL
2008 DATE
Trump ORG

Ninety-five percent PERCENT
Republicans NORP
Trump ORG
50 percent PERCENT
2016 DATE
Joe Biden’s PERSON
90 percent PERCENT
Democratic NORP
Trump ORG
Hispanic NORP
35 percent PERCENT
60 percent PERCENT
Hispanic NORP
Democratic NORP
Florida GPE
Arizona GPE
Nevada GPE
New Mexico GPE
Trump ORG
2016 DATE
Florida GPE
Ohio GPE
Iowa GPE
America GPE
Trump ORG
1852 DATE
Richard Nixon PERSON
the Electoral College ORG
1960 DATE
John F. Kennedy PERSON
Midwestern NORP
Michigan GPE
Pennsylvania GPE
Wisconsin GPE
Ohio GPE
Iowa GPE
Ohio GPE
Florida GPE
Trump ORG
Biden PERSON
Michigan GPE
Pennsylvania GPE
Wisconsin GPE
Detroit GPE
Philadelphia GPE
Milwaukee GPE
Biden PERSON
Biden PERSON
17 percent PERCENT
524 CARDINAL

873 CARDINAL
Obama GPE
2008 DATE
Biden PERSON
Obama LOC
Biden PERSON
Republicans NORP
Senate ORG
House ORG
27 CARDINAL
Trump ORG
80 percent PERCENT
100 percent PERCENT
Trump ORG
Trump ORG
Trump ORG
one CARDINAL
first ORDINAL
1 CARDINAL
election night TIME
Trump ORG
2 CARDINAL
hundreds of thousands CARDINAL
90 percent PERCENT
Biden PERSON
3 CARDINAL
Pennsylvania GPE
23,000 CARDINAL
another 86,000 CARDINAL
4 CARDINAL
5 CARDINAL
Biden PERSON
Robert Barnes PERSON
Trump ORG
Delaware County GPE
Pennsylvania GPE
50,000 CARDINAL
47 CARDINAL
7 CARDINAL
Matt Braynard’s PERSON
Voter Integrity Project ORG
20,312 CARDINAL
Georgia GPE
Biden PERSON
12,670 CARDINAL
Bernie Sanders PERSON
Vermont GPE
Senate ORG
Democratic NORP
Americans NORP

Fannie ORG
Lou Hamer PERSON
Medgar Evers PERSON
Daisy Bates PERSON
10 to 1 CARDINAL
Americans NORP
Americans NORP
second ORDINAL
Americans NORP
Sixty-five years DATE
Brown v. Board of Education ORG
Topeka GPE
Americans NORP
Princeton PERSON
Keeanga Yamahtta-Taylor PERSON
Black Americans NORP
40 percent PERCENT
2009 DATE
Black Americans NORP
Americans NORP
Americans NORP
15 MONEY
African Americans NORP
42 percent PERCENT
Walton PERSON
40 percent PERCENT
Americans NORP
Joe Biden PERSON
500 CARDINAL
the United States GPE
Rashad Robinson PERSON
Color of Change ORG
America GPE
first ORDINAL
Americans NORP
Walmart ORG
African Americans NORP
one CARDINAL
Britain GPE
seven years DATE
one CARDINAL
50 CARDINAL
Britain GPE
King's College London ORG
almost 200,000 CARDINAL
every day DATE
about £700,000 MONEY
one CARDINAL
50 CARDINAL
Londoners ORG

daily DATE
1,000 CARDINAL
392 CARDINAL
1,000 CARDINAL
2011 DATE
40 CARDINAL
nine million CARDINAL
one CARDINAL
50 DATE
daily DATE
London GPE
weekdays DATE
the weekend DATE
London GPE
the week DATE
the weekend   DATE
Last year DATE
the Global Drug Survey ORG
British NORP
one CARDINAL
London GPE
Bristol GPE
Barcelona GPE
Antwerp GPE
Zurich GPE
Amsterdam GPE
Europe LOC
the European Monitoring Centre for Drugs and Drug Addiction ORG
2015 DATE
London GPE
Europe LOC
Cocaine PRODUCT
Latin America  Traces LOC
Suffolk GPE
last summer DATE
Britain GPE
the University of Naples Federico II ORG
Leon Barron PERSON
King's College London ORG
Latin America LOC
the National Crime Agency ORG
Cressida Dick PERSON
Pure PRODUCT
about £100 per gram MONEY
Cocaine ORG
five CARDINAL
42.8 tons QUANTITY
2013-14 DATE
122.9 tons QUANTITY
2017-18 DATE

Telegraph ORG
Dr Barron PERSON
two CARDINAL
more than £40,000 MONEY
Four CARDINAL
five 79% PERCENT
England GPE
the previous week DATE
58% PERCENT
the week DATE
NHS Digital England ORG
the Office for National Statistics ORG
The ONS Opinions ORG
Lifestyle Survey PERSON
Great Britain GPE
2017 DATE
the week DATE
England GPE
the previous week DATE
57.8% PERCENT
Scotland GPE
53.5% PERCENT
Wales GPE
50.0% PERCENT
English LANGUAGE
fifth ORDINAL
between 25 and 64 DATE
65 DATE
Friday DATE
Saturday DATE
nights TIME
the week DATE
previous years DATE
the Health Survey for England ORG
every day DATE
Steve Clarke PERSON
the Priory Group ORG
four CARDINAL
five, six days DATE
2016-17 DATE
England GPE
337,000 CARDINAL
17% PERCENT
fifth ORDINAL
2006-07 DATE
Tuesday DATE
Scotland GPE
ONS ORG
England GPE
Scotland GPE

Scotland GPE
MUP ORG
England GPE
Ian Gilmore PERSON
the Alcohol Health Alliance ORG
more than 50 CARDINAL
England GPE
Scotland GPE
more than 23,000 CARDINAL
England GPE
MUP PERSON
MUP ORG
England GPE
Scotland GPE
50p DATE
two CARDINAL
at least £1 MONEY
nine CARDINAL
at least £ MONEY
4.50 MONEY
Fifa ORG
England GPE
Scotland GPE
DavidsonTESS DE LA ORG
31 CARDINAL
1 Nov 2017 DATE
FIFA ORG
England GPE
Scotland GPE
the World Cup EVENT
next week DATE
Armistice Day EVENT
Fifa ORG
England GPE
Scotland World Cup ORG
next week DATE
Fifa ORG
England GPE
Scotland World Cup ORG
Getty Images FA ORG
Fifa ORG
England GPE
Scotland GPE
next week DATE
World Cup EVENT
Sun ORG
the Royal British Legion ORG
more than a century DATE
5 CARDINAL
Press Association FA ORG

5 CARDINAL
banCredit ORG
Wembley ORG
November 11 DATE
just two days DATE
Sunday DATE
England GPE
Remembrance Sunday PERSON
Spain GPE
2011 DATE
Fifa ORG
Prince William PERSON
Veterans PERSON
Fifa ORG
Fifa PERSON
the World Cup EVENT
English LANGUAGE
last night TIME
England GPE
last night TIME
Fifa ORG
Simon Weston PERSON
second ORDINAL
England GPE
RAF ORG
John Nichol PERSON
Sun ORG
5 CARDINAL
RAF ORG
John Nichol PERSON
Sun ORG
Jordan Henderson PERSON
England GPE
Slovenia GPE
5 CARDINAL
Jordan Henderson PERSON
England GPE
SloveniaCredit ORG
World Wars EVENT
Fifa ORG
Fifa PERSON
Three CARDINAL
Scotland GPE
Fifa ORG
the day DATE
Britain GPE
two-minute TIME
England GPE
England GPE
Fifa ORG

last night TIME
Sun ORG
RAF ORG
John Nichol PERSON
Fifa ORG
England LANGUAGE
Scottish NORP
Fifa PERSON
The Royal British Legion ORG
Sun ORG
the Royal British Legion ORG
this year DATE
the Armed Forces ORG
Fifa ORG
just five CARDINAL
four years DATE
at least one CARDINAL
only four CARDINAL
just four years 3 DATE
four CARDINAL
Getty Images PERSON
2011-12 DATE
719 PRODUCT
2014-15 DATE
1,955 CARDINAL
ten CARDINAL
some 4,643 CARDINAL
the four years DATE
UK GPE
Lucy Russell     Children’s PERSON
Plan International ORG
UK GPE
Government ORG
Lucy Russell PERSON
Ms Russell PERSON
Schools PERSON
Sixty-six CARDINAL
94 per cent MONEY
England GPE
five-year-old DATE
five CARDINAL
five-year-old DATE
five CARDINAL
Thirty-four CARDINAL
England GPE
Wales GPE
Northern Ireland GPE
Brexit PERSON
Liam Fox’s PERSON
Department for International Trade DIT ORG

the United Kingdom GPE
the European Union ORG
Pro-Brexit NORP
Britain GPE
EU ORG
Boris Johnson PERSON
the British Chambers of Commerce ORG
Tuesday DATE
EU ORG
Last December DATE
Deloitte PERSON
Downing Street’s ORG
article 50 LAW
six months DATE
Democrat NORP
Tim Farron PERSON
Nineteen Eighty-Four DATE
Brexit PERSON
Tories NORP
Farron PERSON
two CARDINAL
two CARDINAL
One CARDINAL
EU ORG
UK GPE
Theresa PERSON
article 50 LAW
two CARDINAL
DIT ORG
the Department for Exiting the European Union ORG
Britain GPE
one CARDINAL
two CARDINAL
DIT NORP
15% PERCENT
Britain GPE
EU ORG
one CARDINAL
UK GPE
Changing Europe LOC
Brexit PERSON
Fox ORG
UK GPE
EU ORG
Brexit PERSON
Article 50 LAW
two-year DATE
27 CARDINAL
EU ORG
May DATE

the end of March DATE
DIT NORP
Biden Has Been PERSON
Jerome Corsi PERSON
Joe Biden PERSON
first ORDINAL
Jerome Corsi PERSON
KXEL ORG
1540 DATE
KXEL ORG
Fox News Radio ORG
Joe Biden PERSON
Fox ORG
Chris Wallace PERSON
Jeff Stein PERSON
EDITOR ORG
Conservative NORP
only one CARDINAL
FREE ORG
Greenland GPE
annual DATE
two CARDINAL
Johan Petersen PERSON
Greenland GPE
six metres QUANTITY
annual DATE
Greenland GPE
Greenland GPE
six metres QUANTITY
Greenland GPE
20,000 years ago DATE
GPS ORG
Greenland GPE
12mm QUANTITY
19 cubic kilometres QUANTITY
each year DATE
about 8% PERCENT
Greenland GPE
Greenland GPE
40m years ago DATE
Iceland GPE
Greenland GPE
millennia GPE
Greenland GPE
the past and today DATE
the last 20 years DATE
Greenland GPE
decades DATE
Prof Jonathan Bamber PERSON
the University of Bristol ORG

UK GPE
one CARDINAL
Science Advances ORG
One CARDINAL
Dr Christopher Harig PERSON
the University of Arizona ORG
the end of the century DATE
years ago DATE
Pippa Whitehouse PERSON
the University of Durham ORG
GPS ORG
Greenland GPE
Melt PERSON
Greenland GPE
10 June, 2014 DATE
15 June, 2016 DATE
Every spring DATE
early summer DATE
one CARDINAL
2016 DATE
Antarctica LOC
April DATE
Greenland GPE
August DATE
OA GPE
Netflix GPE
Netflix Streaming TV Renewal Scorecard ORG
OA GPE
Netflix GPE
Zal Batmanglij PERSON
Instagram GPE
Monday DATE
OA GPE
Netflix GPE
earlier in the day DATE
two-season DATE
Zal PERSON
first ORDINAL
Netflix ORG
the early days DATE
Hap GPE
Queens GPE
Prairie Johnson PERSON
seven years earlier DATE
Emory Johnson PERSON
Scott Wilson PERSON
Phyllis Smith PERSON
Alice Krige PERSON
Patrick Gibson PERSON
Jason Isaacs PERSON

16 CARDINAL
OA GPE
Superbowl GPE
the last minute TIME
American NORP
Superbowl GPE
Sylvester Stallone PERSON
Sammy Hagar PERSON
Over the Top WORK_OF_ART
NFL ORG
This year DATE
Superbowl GPE
Arabic LANGUAGE
55 CARDINAL
the Kansas City Chiefs ORG
Tampa Bay GPE
The Tampa Bay Buccaneers ORG
first ORDINAL
3 CARDINAL
Google ORG
one CARDINAL
Orlando GPE
Peter Brady PERSON
at least three CARDINAL
Viagra PRODUCT
You-Know PERSON
MVP ORG
the White House ORG
all night TIME
Coronavirus ORG
US GPE
NASA ORG
Solar PRODUCT
Heliospheric Observatory PERSON
Sun LOC
Sun ORG
Scott Waring PERSON
ten CARDINAL
Earth LOC
Earth LOC
3,958 miles QUANTITY
6,37km QUANTITY
more than 39,580 miles QUANTITY
63,370km km QUANTITY
Waring PERSON
Sun ORG
Michigan GPE
the Great Lakes Justice Center FAC
two CARDINAL
Detroit GPE

Wayne County GPE
Michigan GPE
David A. Kallman PERSON
Court ORG
Wayne County GPE
Michigan GPE
Pennsylvania GPE
two CARDINAL
Donald Trump PERSON
Naomi Wolf PERSON
Cheryl A. Costantino PERSON
Edward P. McCall Jr. PERSON
the Qualified Voter File QVF PERSON
November 3 DATE
2020 CARDINAL
Michigan GPE
The Western Journal ORG
Privacy Policy and Terms of Use You ORG
100% PERCENT
12 CARDINAL
Votes ORG
0% PERCENT
the TCF Center FAC
tens of thousands CARDINAL
Democratic NORP
the TCF Center FAC
Joe Biden PERSON
Federal Investigators Arrive ORG
Michigan GPE
the TCF Center FAC
tens of thousands CARDINAL
thousands CARDINAL
Detroit GPE
two CARDINAL
David Fink PERSON
The Detroit News ORG
last week DATE
The Western Journal ORG
35 days DATE
Emmett Till GPE
the Emmett Till Interpretive Center ORG
Glendale GPE
Mississippi GPE
TIME Magazine ORG
third ORDINAL
Patrick Weems PERSON
the Emmett Till Interpretive Center ORG
Sumner GPE
Miss. GPE
14-year-old DATE

two CARDINAL
Tallahatchie FAC
Carolyn Bryant PERSON
decades later DATE
Till PERSON
2007 DATE
52 years DATE
Till PERSON
first ORDINAL
one year DATE
New York Times ORG
first ORDINAL
June DATE
Weems PERSON
one CARDINAL
Weems PERSON
daily DATE
Invalid Email Something PERSON
under ten DATE
the Royal College of Paediatricians and Child Health RCPCH ORG
Switzerland GPE
today DATE
Bubbles GPE
China GPE
REUTERS ORG
nine-year-old DATE
British NORP
French NORP
Alps LOC
more than 170 CARDINAL
three CARDINAL
Dr Alasdair Munro ORG
Covid-19 PERSON
Russell Viner PERSON
RCPCH ORG
Prof Viner PERSON
Anthony Fauci PERSON
4 CARDINAL
U.S. GPE
COVID-19 PERSON
Justia.com ORG
Anthony S. Fauci PERSON
HIV-1 LAW
the Acquired Immune Deficiency Syndrome ORG
1990 DATE
Fauci PERSON
NIH ORG
today DATE
Glycoprotein PERSON
120 CARDINAL

GP120 ORG
COVID-19 ORG
HIV-1 PRODUCT
Severe Acute Respiratory System ORG
2002-2003 DATE
COVID-19 ORG
CoV-2 ORG
four CARDINAL
Fauci ORG
9896509 CARDINAL
August 3, 2016 DATE
4?7 CARDINAL
Publication Number: 20160333309 WORK_OF_ART
August 3, 2016 DATE
Integrin NORP
9441041 CARDINAL
September 13, 2016 DATE
GP120 PRODUCT
4?7 CARDINAL
2016007586 DATE
September 21, 2015 DATE
the Interaction Between HIV GP120 ORG
Integrin NORP
India GPE
January 2020 DATE
2019 DATE
nCov ORG
HIV-1 PRODUCT
Gag PERSON
India GPE
4 CARDINAL
2019 DATE
4 CARDINAL
HIV-1 PRODUCT
India GPE
COVID-19 ORG
HIV-1 PRODUCT
Fauci ORG
4 CARDINAL
COVID-19 PERSON
GP120 PRODUCT
the HIV-1 1990s era DATE
Fauci NORP
Britons PERSON
Margaret Thatcher PERSON
The Resolution Foundation ORG
Theresa May PERSON
Brexit PERSON
more than two years DATE
December DATE

The Resolution Foundation’s ORG
half CARDINAL
the mid-1960s DATE
the early years DATE
Thatcher PERSON
1979-90 DATE
the early 1980s DATE
83% to 60% PERCENT
the Resolution Foundation ORG
between 2015 DATE
2020 CARDINAL
half CARDINAL
2% PERCENT
4% PERCENT
1% PERCENT
between 2005 DATE
2010 DATE
five-year DATE
the 1930s DATE
Torsten Bell ORG
the Resolution Foundation ORG
Britain GPE
recent years DATE
Britain GPE
Margaret Thatcher PERSON
Downing Street FAC
the next few years DATE
the 1980s DATE
half CARDINAL
about 5% PERCENT
fifth ORDINAL
the next four years DATE
more than £12bn MONEY
Brexit PERSON
the European Union ORG
the next few years DATE
Britain GPE
2020 DATE
three CARDINAL
90:10 TIME
90% PERCENT
only 10% PERCENT
80:20 TIME
those 80% PERCENT
20% PERCENT
Palma PERSON
10% PERCENT
40% PERCENT
the next four years DATE
2020-21 DATE

the past 20 years DATE
May DATE
UK GPE
the Resolution Foundation ORG
south-east LOC
England GPE
Incomes PERSON
more than 10% PERCENT
south-east LOC
England GPE
North-east England GPE
the West Midlands LOC
20% PERCENT
London GPE
roughly 10% PERCENT
Treasury ORG
2015 DATE
Millions CARDINAL
John McDonnell PERSON
the Resolution Foundation ORG
autumn DATE
Philip Hammond PERSON
the day DATE
a Tory Brexit PERSON
Margaret Thatcher PERSON
Medical Doctors ORG
PhD Scientists Speak Out Against Vaccinations WORK_OF_ART
26 CARDINAL
0 CARDINAL
5 2 CARDINAL
January 26, 2020 DATE
C.  Physicians PERSON
decades DATE
Vitamin C ORG
C.” Robert F. Cathcart PERSON
MD GPE
the Orthomolecular Medicine News Service ORG
the International Society for Orthomolecular Medicine ORG
Vitamin C ORG
3,000 CARDINAL
daily DATE
Vitamin D3 FAC
2,000 CARDINAL
International Units ORG
5,000 CARDINAL
IU GPE
two weeks DATE
2,000 CARDINAL
400 CARDINAL
daily DATE

Zinc PERSON
20 mg QUANTITY
daily DATE
100 mcg QUANTITY
daily DATE
Vitamin C ORG
1 CARDINAL
Vitamin D PERSON
2 CARDINAL
3 CARDINAL
4 CARDINAL
5 CARDINAL
first ORDINAL
the late 1940s DATE
6 CARDINAL
the decades DATE
1980 DATE
Orthomolecular.com  References ORG
1 CARDINAL
Orthomolecular Medicine News Service ORG
2018 DATE
Vitamin PERSON
J Orthomol Med ORG
June, 2018 DATE
333 CARDINAL
Gonzalez MJ PERSON
Berdiel MJ PERSON
2018 DATE
June, 2018 DATE
333 CARDINAL
Physiol Ther PERSON
22:8 CARDINAL
530 CARDINAL
Gorton HC PERSON
Jarvis K 1999 PERSON
22:8 CARDINAL
530 CARDINAL
Nutrients ORG
94 CARDINAL
E339 CARDINAL
2017 DATE
Vitamin C ORG
94 CARDINAL
E339 CARDINAL
https://www.ncbi.nlm.nih.gov/pubmed/28353648  Hickey S PERSON
2015 DATE
Basic Health Pub ORG
978-1591202233 CARDINAL
Orthomolecular Medicine News Service ORG
Levy TE PERSON

2014 DATE
C.  OMNS PERSON
2007 DATE
2007 DATE
Vitamin C LAW
2009 DATE
Vitamin C ORG
2009 DATE
Vitamin C ORG
OMNS ORG
Taylor PERSON
2017 DATE
Vitamin C ORG
Immune Netw ORG
13:70-74 DATE
Yejin Kim PERSON
Hyemin Kim PERSON
Seyeon Bae et al PERSON
2013 DATE
Vitamin C ORG
2 CARDINAL
Vitamin ORG
134:1129-1140 DATE
Cannell JJ PERSON
Vieth R PERSON
Umhau JC et al PERSON
D.134:1129-1140 DATE
Virol J. 5:29 PERSON
Zasloff M ORG
Garland CF GPE
influenza.5:29 ORG
Arch Intern Med ORG
169:384-390 CARDINAL
Mansbach NORP
25 CARDINAL
the Third National Health and Nutrition Examination ORG
BMJ ORG
356 CARDINAL
Martineau AR PERSON
Hooper RL PERSON
2017 DATE
Vitamin PRODUCT
data.356 ORG
Urashima M ORG
Segawa ORG
Okazaki M PERSON
https://www.ncbi.nlm.nih.gov/pubmed/20219962 PERSON
11:344-349 CARDINAL
von Essen MR PERSON
Kongsbak M ORG

2010 DATE
Vitamin PERSON
OMNS ORG
Dean C 2017 Magnesium ORG
The Magnesium Miracle LOC
2nd Ed PERSON
Ballantine Books ORG
978-0399594441 CARDINAL
Levy TE 2019 PERSON
Medfox Pub ORG
978-0998312408 CARDINAL
J Nutr PRODUCT
Laakko PERSON
Vollmer TL ORG
3:386-400 CARDINAL
Liu MJ PERSON
Bao S PERSON
Gálvez-Peralta M ORG
2013 DATE
NF-?B.3:386-400 PRODUCT
J Nutr WORK_OF_ART
Mocchegiani E PERSON
Shankar AH PERSON
Zinc PERSON
J Nutr PRODUCT
Beck MA PERSON
OA PERSON
Handy J. 2003 PERSON
52:1273-1280 DATE
Berry MJ PERSON
2008 DATE
Adv Nutr WORK_OF_ART
6:73-82 DATE
Steinbrenner H PERSON
Al-Quraishy S PERSON
Dkhil MA PERSON
al. GPE
2015 DATE
J South Med Surg 1949 PRODUCT
111:210 CARDINAL
6 CARDINAL
Klenner FR ORG
C.1949 ORG
111:210 CARDINAL
Australian NORP
1980 DATE
94:9 CARDINAL
7 CARDINAL
94:9 CARDINAL
January 27, 2020 DATE

COVID-19 ORG
Centers for Disease Control and Prevention ORG
CDC ORG
a few days DATE
American Thinker ORG
New York City GPE
COVID-19 ORG
the Big Apple ORG
CDC ORG
Americans NORP
CDC ORG
COVID-19 ORG
Kalispell ORG
Montana GPE
Annie Bukacek PERSON
MD GPE
Montana GPE
Annie Bukacek PERSON
COVID-19 PERSON
YouTube ORG
April 6 DATE
today DATE
Bukacek ORG
over 30 years DATE
Bukacek ORG
COVID-19 ORG
COVID-19 PERSON
COVID-19 ORG
The Centers for Disease Control ORG
yesterday DATE
April 4th DATE
COVID-19 ORG
CDC ORG
COVID-19 ORG
COVID-19 ORG
COVID-19 PERSON
COVID-19 ORG
60 percent PERCENT
March 24 DATE
CDC ORG
Steven Schwartz PERSON
the Division of Vital Statistics ORG
the National Center for Health Statistics ORG
COVID-19 Alert No. PERSON
COVID-19 ORG
COVID-19 ORG
One CARDINAL
COVID-19 ORG
COVID-19 PERSON
Bukacek ORG

CDC ORG
COVID-19 ORG
COVID-19 ORG
Bukacek PERSON
Chinese NORP
America GPE
Americans NORP
Americans NORP
CDC ORG
Americans NORP
First ORDINAL
2.2 million CARDINAL
Americans NORP
around 200,000 CARDINAL
April 8 DATE
August 4 DATE
60,000 CARDINAL
the University of Washington ORG
the White House ORG
at least 60 percent PERCENT
20 seconds TIME
Americans NORP
Epidemiology ORG
Matthew Vadum PERSON
Washington GPE
D.C. GPE
Team Jihad ORG
Leftists NORP
the United States GPE
Ripping Off American Taxpayers ORG
May 28, 2020 DATE
New Research Study Clarifies Health Outcomes ORG
Print GPE
May 28, 2020   DATE
Redding CA WORK_OF_ART
three CARDINAL
the United States GPE
3 years DATE
2000 DATE
three CARDINAL
November 2005 and June 2015 DATE
one year of age DATE
30.9% PERCENT
one year of age DATE
one year of age DATE
six months of age DATE
Brian Hooker PERSON
Neil Miller PERSON
Hooker PERSON
their first year DATE

### MONEY
SAN FRANCISCO GPE
CA WORK_OF_ART
Joe Biden PERSON
Twitter PERSON
Thursday DATE
one CARDINAL
Jack Dorsey PERSON
Dorsey PERSON
Break In Case Of Bad Publicity For Democrats WORK_OF_ART
Twitter PERSON
Shut PERSON
Dorsey PERSON
Twitter PERSON
Babylon Bee PERSON
Jackson Baker PERSON
The Babylon Bee WORK_OF_ART
Rastafarian Association of Ghana ORG
Jah Eddy Bongo PERSON
Ghana GPE
marijuana PERSON
Ghana GPE
Coronavirus NORP
Italy GPE
one CARDINAL
Hubert Osei Welbeck PERSON
Bongo PERSON
Baifikrom GPE
the Mankessim Municipality LOC
25 CARDINAL
Ghanaians NORP
less than 24 hours TIME
COVID-19 WORLDWIDE PERSON
Coronavirus PRODUCT
710,987 CARDINAL
the United States GPE
135,000 CARDINAL
Americans NORP
Coronavirus FAC
33,558 CARDINAL
150,825 CARDINAL
November 24, 2013 DATE
Theodore Roosevelt PERSON
1858-1919 DATE
about 2007 DATE
Roosevelt PERSON
half CARDINAL
at least 1995 DATE
Google Groups ORG
BigBird/Barney SUCKS-PBSgate William Davenant 2/9/95 ORG

Brown Cafe UPS Forum Raw ORG
08:28 TIME
ANGER ORG
TRUTH ORG
TED PERSON
November 18, 2008 DATE
12:42 AM TIME
steve orris PERSON
@steveorris PRODUCT
3:15 PM - 3 TIME
08 CARDINAL
7/23/2009 DATE
12:26:45 PM TIME
Canada GPE
Teddy Roosevelt PERSON
Franc Pohole ORG
Conservative GPE
Liberal GPE
8:21 PM - 19 TIME
Sep ORG
07:19 PM TIME
Teddy Roosevelt PERSON
Theodore Roosevelt PERSON
today DATE
LIE ORG
Teddy Roosevelt PERSON
6:00 PM - 1 Nov 13 TIME
first ORDINAL
20s and 30s DATE
30 years DATE
the Health Foundation ORG
first ORDINAL
zero-hours TIME
Femi Fani-Kayode PERSON
Abba Kyari PERSON
COVID-19 ORG
last week DATE
Thursday DATE
Fani-Kayode PERSON
Kyari PERSON
Muhammadu Buhari PERSON
Abba Kyari PERSON
@MBuhari ORG
Abba Kyari PERSON
@MBuhari ORG
Femi Fani-Kayode PERSON
April 2, 2020 Advertisement TheCable DATE
Kyari PERSON
Lagos GPE
Lagos GPE

Kyari PERSON
TheCable ORG
September 29 DATE
Democratic Party ORG
Joe Biden PERSON
Trump PERSON
Antifa PERSON
Antifa PERSON
Robby Starbuck PERSON
Joe Biden PERSON
Antifa PERSON
Joe Biden’s Antifa PERSON
Antifa PERSON
Joe PERSON
100% PERCENT
#JustSayNoToAntifaJoe pic.twitter.com/CMVdh9RpGL MONEY
Robby Starbuck PERSON
October 1, DATE
2020 DATE
WalkAway ORG
Brandon Straka PERSON
one CARDINAL
America GPE
this morning TIME
Antifa PERSON
Adam Rahuba PERSON
Pittsburg GPE
PA GPE
Trump PERSON
Joe Biden PERSON
America GPE
Sunday DATE
noon TIME
Antifa PERSON
Brandon Straka PERSON
November 24, 2020 DATE
Democrats NORP
Privacy Policy PRODUCT
Adam Rahuba PERSON
Twitter PRODUCT
MAGA ORG
Rahuba PERSON
Biden PERSON
November 1 DATE
Rahuba ORG
Pittsburg GPE
Venmo PERSON
WayBackMachine ORG
Rahuba GPE
Twitter PRODUCT

the United States GPE
Twitter PERSON
Rahuba GPE
Trump PERSON
WASHINGTON GPE
Donald Trump PERSON
2020 DATE
nearly every day DATE
2016 DATE
Trump ORG
Justice ORG
Homeland Security ORG
Trump ORG
Joe Biden's PERSON
306 CARDINAL
Congress ORG
Electoral College ORG
144 years DATE
one CARDINAL
Wednesday DATE
13 CARDINAL
more than 100 CARDINAL
Republican NORP
Biden PERSON
Congress ORG
USA TODAY ORG
62 CARDINAL
62 CARDINAL
Marc Elias PERSON
Democratic NORP
61 CARDINAL
62 CARDINAL
61 CARDINAL
Elias PERSON
Democratic NORP
Republican NORP
Trump ORG
The 24 hours TIME
Trump ORG
State Supreme Courts ORG
Arizona GPE
Nevada GPE
Arizona GPE
Trump ORG
Pennsylvania GPE
Michigan supreme ORG
60th ORDINAL
61st ORDINAL
recent days DATE
Last Friday DATE

Trump ORG
Texas GPE
Louie Gohmert PERSON
Texas GPE
Mike Pence PERSON
Electoral College ORG
supreme courts ORG
Biden PERSON
Monday DATE
Wisconsin GPE
Pennsylvania GPE
Georgia GPE
Michigan GPE
Arizona GPE
Trump ORG
Pennsylvania GPE
Trump ORG
three days DATE
Pennsylvania GPE
Biden PERSON
81,660 CARDINAL
6 CARDINAL
six CARDINAL
Biden PERSON
Arizona GPE
Georgia GPE
Michigan GPE
Nevada GPE
Pennsylvania GPE
Wisconsin GPE
Trump ORG
five CARDINAL
four years ago DATE
Hillary Clinton PERSON
Biden PERSON
Democratic NORP
House ORG
Republicans NORP
Electoral College ORG
Wednesday DATE
six CARDINAL
the US Supreme Court ORG
Trump ORG
2 CARDINAL
The U.S. Supreme Court ORG
Trump ORG
Nov. 3 DATE
one CARDINAL
the Supreme Court ORG
Dec. 8 DATE

Pennsylvania GPE
Republicans NORP
Biden PERSON
Mike Kelly PERSON
Pa. GPE
Republican NORP
Three days later DATE
the Supreme Court ORG
Texas GPE
four CARDINAL
Trump ORG
Texas GPE
3 CARDINAL
Georgia GPE
two CARDINAL
Biden PERSON
Wisconsin GPE
one CARDINAL
Biden PERSON
first ORDINAL
Georgia GPE
Biden PERSON
12,284 CARDINAL
14,196 CARDINAL
four CARDINAL
second ORDINAL
Georgia GPE
Trump ORG
Biden PERSON
11,779 CARDINAL
Votes Trump ORG
2,343 CARDINAL
Wisconsin GPE
Biden PERSON
74 CARDINAL
two CARDINAL
Democratic NORP
Milwaukee GPE
Dane NORP
Biden PERSON
20,682 CARDINAL
about 3 million CARDINAL
Trump ORG
2,343 CARDINAL
Georgia GPE
Wisconsin GPE
Trump ORG
At least 6 CARDINAL
first ORDINAL
Trump ORG

Election Day DATE
2012 DATE
Barack Obama PERSON
Republican NORP
Mitt Romney PERSON
Trump ORG
Romney PERSON
Obama GPE
Four years later DATE
Ted Cruz PERSON
Texas GPE
Trump ORG
Iowa GPE
Trump ORG
Ted Cruz PERSON
Iowa GPE
Ted Cruz PERSON
Iowa GPE
Donald J. Trump PERSON
@realDonaldTrump ORG
February 3, 2016 DATE
Trump ORG
2016 DATE
Democratic NORP
Bernie Sanders PERSON
2020 DATE
2016 CARDINAL
Trump ORG
more than 3 million CARDINAL
White House ORG
Trump ORG
2018 DATE
Arizona GPE
SIGNATURES NORP
Election PRODUCT
Donald J. Trump PERSON
@realDonaldTrump ORG
November 9, 2018 DATE
2020 DATE
Democratic NORP
Sanders ORG
Trump ORG
0 CARDINAL
the Electoral College ORG
Dec. 14 DATE
Biden PERSON
306 CARDINAL
232 CARDINAL
Biden PERSON
81,281,502 CARDINAL

Trump ORG
7 million CARDINAL
51.3% PERCENT
Biden PERSON
Franklin D. Roosevelt PERSON
1932 DATE
Trump ORG
46.8% PERCENT
Biden PERSON
Wednesday DATE
Congress ORG
13 CARDINAL
U.S. GPE
more than 100 CARDINAL
Republican NORP
House ORG
six CARDINAL
Biden PERSON
Democratic NORP
House ORG
Republican NORP
Senate ORG
hours TIME
House ORG
Senate ORG
Biden PERSON
Harris PERSON
noon TIME
Jan. 20 DATE
November 1, 2019 DATE
3:07 pm TIME
The Old Farmer’s Almanac ORG
last winter’s DATE
2020 DATE
The Old Farmer’s Almanac ORG
Americans NORP
this year DATE
no fewer than seven CARDINAL
This winter DATE
Janice Stillman PERSON
Greenwich GPE
London GPE
the Office for National Statistics ONS ORG
Lambeth PERSON
second ORDINAL
Islington ORG
third ORDINAL
Haringey fourth ORG
Croydon GPE
fifth ORDINAL

Redbridge PERSON
Richmond GPE
Newham ORG
Loading PERSON
half-decade DATE
Britain GPE
first ORDINAL
London GPE
Loading PERSON
three CARDINAL
Hammersmith PERSON
Fulham ORG
Sutton GPE
Greenwich GPE
London GPE
Loading PERSON
UK GPE
ONS ORG
10 CARDINAL
the last five years DATE
UK GPE
Chorley ORG
Lancashire GPE
seven CARDINAL
10 CARDINAL
Stirling GPE
Scotland GPE
8.1 CARDINAL
ONS ORG
Dawn Snape ORG
UK GPE
the past five years DATE
today DATE
36 percent PERCENT
1.36 CARDINAL
1.51 CARDINAL
Table 5 LAW
Influenza NORP
Department of Defense ORG
2017–2018 influenza season DATE
Flu Vaccine Interference  Paraphrasing PERSON
the Department of Veterans Affairs’ ORG
two weeks ago DATE
one CARDINAL
PubMed ORG
Informed Consent For ORG
Flu Vaccine  Now PERSON
Table 5 LOC
Table 5 LAW
Coronavirus GPE

1.36 DATE
36% PERCENT
0.57 CARDINAL
Three CARDINAL
Three CARDINAL
8 CARDINAL
36% PERCENT
1 MONEY
this year DATE
Americans NORP
VA ORG
Vaccine PERSON
January 2020 DATE
the Armed Forces Health Surveillance Branch Air Force Satellite ORG
Wright-Patterson AFB ORG
OH GPE
the Air Force Research Laboratory Institutional Review Board ORG
VA ORG
1151 DATE
the US Court of Claims – Vaccine Court ORG
US GPE
DOD ORG
VA ORG
Veterans PERSON
VA GPE
Vietnam GPE
Vietnam GPE
VA ORG
the Department of Defense Global Respiratory Pathogen Surveillance Program DoDGRS ORG
the Global Emerging Infections Surveillance ORG
2017-2018 DATE
2017–2018 CARDINAL
1 October 2017 and ended 29 September 2018 DATE
Chlamydia ORG
Mycoplasma GPE
non-influenza NORP
the season DATE
the season DATE
US GPE
YouTube ORG
60 CARDINAL
Australia GPE
Wuhan GPE
4 CARDINAL
European NORP
apps PERSON
third ORDINAL
Terms of Service ORG
Greenland GPE

annual DATE
two CARDINAL
Johan Petersen PERSON
Greenland GPE
six metres QUANTITY
annual DATE
Greenland GPE
Greenland GPE
six metres QUANTITY
Greenland GPE
20,000 years ago DATE
GPS ORG
Greenland GPE
12mm QUANTITY
19 cubic kilometres QUANTITY
each year DATE
about 8% PERCENT
Greenland GPE
Greenland GPE
40m years ago DATE
Iceland GPE
Greenland GPE
millennia GPE
Greenland GPE
the past and today DATE
the last 20 years DATE
Greenland GPE
decades DATE
Prof Jonathan Bamber PERSON
the University of Bristol ORG
UK GPE
one CARDINAL
Science Advances ORG
One CARDINAL
Dr Christopher Harig PERSON
the University of Arizona ORG
the end of the century DATE
years ago DATE
Pippa Whitehouse PERSON
the University of Durham ORG
GPS ORG
Greenland GPE
Melt PERSON
Greenland GPE
10 June, 2014 DATE
15 June, 2016 DATE
Every spring DATE
early summer DATE
one CARDINAL
2016 DATE

Antarctica LOC
April DATE
Greenland GPE
August DATE
Jonathan Ashworth PERSON
Labour’s Shadow Health ORG
today DATE
3-4 months DATE
Labour ORG
Britain GPE
3-4 months DATE
Labour ORG
an additional £25 million MONEY
Labour ORG
National Child Health Fund ORG
Britain GPE
Today DATE
Labour ORG
Jonathan Ashworth PERSON
Health Visitor Programme ORG
2015 DATE
Between 2016/17 CARDINAL
2017/18 DATE
£55.2 million MONEY
0 CARDINAL
this Tory Government ORG
8,244 CARDINAL
NHS ORG
December 2017 DATE
August 2013 DATE
10,309 CARDINAL
October 2015 DATE
more than 2,065 CARDINAL
20% PERCENT
just over two years DATE
12% PERCENT
a New Birth Visit ORG
17% PERCENT
the South West LOC
England GPE
17% PERCENT
6-8 week DATE
London GPE
1/3 CARDINAL
25% PERCENT
one year DATE
12 CARDINAL
London GPE
44% PERCENT
90% PERCENT

North East LOC
two and a half year DATE
just 64% PERCENT
London GPE
73% PERCENT
the East of England LOC
Jonathan Ashworth PERSON
today DATE
Infant Feeding PERSON
a Labour Government ORG
the Coalition Government ORG
UNICEF ORG
Scotland GPE
Northern Ireland GPE
70% PERCENT
the year 2030 DATE
Jonathan Ashworth PERSON
Labour’s Shadow Health ORG
London GPE
Michael Marmot PERSON
UK GPE
70th ORDINAL
NHS ORG
David Cameron PERSON
Theresa May PERSON
more than 20 percent PERCENT
just over two years DATE
England GPE
UK GPE
Health Visitors ORG
3-4 months DATE
£25 million MONEY
National Child Health Fund ORG
20 per cent MONEY
100 per cent MONEY
Powell River LOC
B.C. GPE
ALS ORG
last Tuesday after DATE
years DATE
24-hour TIME
Sean Tagert PERSON
41 DATE
11-year-old DATE
Tagert PERSON
ALS ORG
Lou Gehrig's PERSON
March 2013 DATE
years DATE
late October 2017 DATE

Tagert PERSON
last year DATE
Tagert PERSON
24-hour TIME
Vancouver Coastal Health ORG
15.5 hours TIME
Independent Living ORG
24-hour TIME
as much as 20 hours QUANTITY
Tagert PERSON
CBC News ORG
last September DATE
Sean Tagert PERSON
Aidan PERSON
Tagert PERSON
Tagert PERSON
263.50 MONEY
Sean NORP
Sean NORP
Sean Tagert PERSON
Trish Mennitti PERSON
Sean NORP
ALS ORG
Tagert PERSON
Texas GPE
Canada GPE
Mackenzie LOC
B.C. GPE
Powell River FAC
Sean NORP
Aidan PERSON
Sean NORP
Aidan PERSON
Tagert PERSON
the ALS Society ORG
British Columbia GPE
Caya PERSON
day DATE
Invalid ORG
Yesterday DATE
UK GPE
Fathers 4 Justice ORG
Hallmark GPE
ComRes ORG
more than 1000 CARDINAL
37 per cent MONEY
UK GPE
2.3 million CARDINAL
18 DATE
one CARDINAL

five CARDINAL
20 per cent MONEY
22 per cent MONEY
One third 32 CARDINAL
Matt O'Connor PERSON
Fathers 4 Justice ORG
Telegraph ORG
daily DATE
millions CARDINAL
Women ORG
ComRes ORG
nearly one CARDINAL
six CARDINAL
17 per cent MONEY
quarter 27 DATE
2001 DATE
750,000 CARDINAL
roughly one CARDINAL
four CARDINAL
10 CARDINAL
the Community Services Facility ORG
The Ulster Hall FAC
Belfast GPE
0800 CARDINAL
4695 CARDINAL
• PERSON
25% PERCENT
Feeding Britain PERSON
Northumbria University’s ORG
NI GPE
one CARDINAL
four CARDINAL
daily DATE
Sorry PERSON
half CARDINAL
Nearly one CARDINAL
three CARDINAL
Kevin Higgins PERSON
Northern Ireland GPE
Covid PERSON
Community Helpline ORG
0808-802-0020 DATE
18,000 CARDINAL
March 27 DATE
69% PERCENT
Higgins PERSON
some 150,000 CARDINAL
NI ORG
Universal Credit ORG
NI GPE

70,000 CARDINAL
134,000 CARDINAL
Universal Credit ORG
Covid PERSON
Prof Greta Defeyter PERSON
Northumbria University’s Healthy Living Lab ORG
UK GPE
Andrew Forsey PERSON
Feeding Britain GPE
millions CARDINAL
Universal Credit ORG
two CARDINAL
year-round DATE
one CARDINAL
four CARDINAL
Northern Ireland GPE
newsletter.co.uk ORG
Northern Ireland GPE
UK GPE
more than 5 CARDINAL
Alistair Bushe PERSON
United Nations ORG
last year DATE
Paris GPE
Obama PERSON
America GPE
Paris GPE
2016 DATE
Trump ORG
this year DATE
Paris GPE
UN ORG
Paris GPE
only a third CARDINAL
between now and 2030 DATE
3 degrees Celsius QUANTITY
2100 DATE
Paris GPE
2 degrees QUANTITY
Eric Solheim PERSON
the U.N. Environment Program ORG
annual DATE
this week DATE
One year DATE
the Paris Agreement ORG
hundreds of millions MONEY
2020 DATE
2030 DATE
2 degrees QUANTITY
UN ORG

2020 DATE
a little over two years from now DATE
30% PERCENT
15% PERCENT
less than 1% PERCENT
today DATE
20% PERCENT
$1 trillion MONEY
273 CARDINAL
570 CARDINAL
UN ORG
42% PERCENT
Paris GPE
Trump ORG
Paris GPE
UN ORG
UN ORG
Tenths DATE
The California Heat Wave Sparked ORG
Boba PERSON
14-year-old DATE
China GPE
Zhejiang GPE
around a hundred CARDINAL
AsiaOne ORG
Chinese NORP
EBC Dongsen News ORG
five days DATE
CT ORG
AKA ORG
EBC Dongsen News ORG
five days earlier DATE
Zhuji People’s Hospital’s ORG
boba PERSON
Chinese NORP
5 days DATE
AsiaOne ORG
@asiaonecom ORG
June 6, 2019 DATE
91 CARDINAL
FDA ORG
250 CARDINAL
Texas GPE
House ORG
Senate ORG
last month DATE
Save Chick-fil-A WORK_OF_ART
Greg Abbott PERSON
LGBTQ ORG
CNN ORG

A year DATE
the Supreme Court ORG
Colorado GPE
Masterpiece Cakeshop ORG
New York Post] Sweetgreen ORG
weekday DATE
NRN ORG
Katy Perry PERSON
Taylor Swift PERSON
Aperol ORG
Negronis NORP
McDonald’s ORG
American NORP
Washington Post ORG
Detroit GPE
@SauceGod_Sosa PERSON
June 11, 2019 DATE
daily DATE
Invalid Email Something PERSON
under ten DATE
the Royal College of Paediatricians and Child Health RCPCH ORG
Switzerland GPE
today DATE
Bubbles GPE
China GPE
REUTERS ORG
nine-year-old DATE
British NORP
French NORP
Alps LOC
more than 170 CARDINAL
three CARDINAL
Dr Alasdair Munro ORG
Covid-19 PERSON
Russell Viner PERSON
RCPCH ORG
Prof Viner PERSON
GETTY - STOCK IMAGE Official ORG
EU ORG
nine CARDINAL
Britain GPE
SUBSCRIBE Invalid PRODUCT
last night TIME
six years DATE
2010 DATE
some 2.2 million CARDINAL
Britain GPE
28.2 million CARDINAL
just 1.1 million CARDINAL
Europe LOC

last year DATE
3.4 million CARDINAL
30.3 million CARDINAL
2.2 million CARDINAL
EU ORG
1.2 million CARDINAL
Anti-EU ORG
years DATE
Britain GPE
last year’s DATE
Steven Woolfe PERSON
EU ORG
Theresa May PERSON
British NORP
six years DATE
EU ORG
Brexit PERSON
28 CARDINAL
the European Union ORG
Getty 1 PERSON
29 CARDINAL
the European Union A ORG
the Office for National Statistics ORG
11.2 per cent MONEY
UK GPE
2016 DATE
EU ORG
Steven Woolfe PERSON
761,000 CARDINAL
508,000 CARDINAL
EU ORG
669,000 CARDINAL
510,000 CARDINAL
EU ORG
Eight per cent MONEY
eight CARDINAL
European NORP
EU8 FAC
EU ORG
2004 DATE
Steven Woolfe PERSON
Czech Republic GPE
Estonia GPE
Hungary GPE
Latvia GPE
Lithuania GPE
Poland GPE
Slovakia GPE
Slovenia GPE
EU ORG

11 per cent MONEY
Seven per cent MONEY
eight CARDINAL
EU ORG
One CARDINAL
eight CARDINAL
382,000 CARDINAL
EU ORG
Ukip GPE
Jane Collins PERSON
EU ORG
Brussels GPE
Some 701,000 CARDINAL
UK GPE
more than a quarter DATE
EU ORG
Yesterday DATE
tens of thousands CARDINAL
European NORP
Britain GPE
Theresa May PERSON
UK GPE
NHS ORG
England GPE
more than £1 billion MONEY
NHS Digital ORG
more than £422 million MONEY
the last 10 years DATE
Almost one CARDINAL
20 CARDINAL
2 CARDINAL
around 90% PERCENT
Robin Hewings PERSON
UK GPE
the last 20 years DATE
26,000 CARDINAL
1bn ORDINAL
NHS ORG
Hewings PERSON
2 CARDINAL
years earlier' DATE
Postcode PERSON
Presentational NORP
over three million CARDINAL
England GPE
the last two decades DATE
nearly 100,000 CARDINAL
92% PERCENT
2 CARDINAL
2 CARDINAL

1 CARDINAL
Almost seven CARDINAL
10 CARDINAL
66.8% PERCENT
almost six CARDINAL
10 CARDINAL
57.8% PERCENT
England GPE
2 CARDINAL
more than 80 percent PERCENT
obese NORP
2 CARDINAL
NHS England LOC
Nearly £477 million MONEY
2017-18 DATE
the same year DATE
around £350 million MONEY
£181 million MONEY
Jonathan Valabhji PERSON
NHS England FAC
NHS ORG
2 CARDINAL
first ORDINAL
The NHS Diabetes Prevention Programme ORG
million CARDINAL
2 CARDINAL
Democrats NORP
Democrat NORP
Ginny Talia PERSON
Monday DATE
Americans NORP
70 years old DATE
Senate Sub-committee ORG
Social Security ORG
one CARDINAL
Talia PERSON
Talia GPE
Talia PERSON
Swedish NORP
Swedish NORP
Swedes NORP
Earlier this month DATE
U.K. GPE
Swedes NORP
Swedish NORP
Swedish Microchip Procedures Pictures Flood Instagram  Sandra
Wuerthner ORG
Wuerthner ORG
Wuerthner ORG
Swedish NORP

Microchipping Is Voluntary  Corporations WORK_OF_ART
Jim Satney PrepForThat PERSON
Amazon ORG
CDC ORG
COVID-19 ORG
Amazon ORG
more than 15 CARDINAL
Nigerian House of Reps ORG
Coronavirus GPE
Bala Mohammed PERSON
Coronavirus LOC
Atiku Abubakar PERSON
Mohammed PERSON
Coronavirus GPE
NCDC ORG
Pearlsnews ORG
House ORG
Mohammed Atiku PERSON
Trump PERSON
over 100,000 CARDINAL
the morning TIME
2020 DATE
Tonight TIME
23,000 CARDINAL
Biden PERSON
Georgia GPE
Trump PERSON
23,000 CARDINAL
Biden PERSON
Georgia GPE
Biden PERSON
10,000 CARDINAL
Michigan GPE
Gives Behind PERSON
the Scenes Update ORG
one CARDINAL
Georgia GPE
GEORGIA GPE
BOMBSHELL Edison Analysis ORG
BIDEN ORG
98% PERCENT
23,487 CARDINAL
12:18AM CARDINAL
Georgia GPE
98% PERCENT
BIDEN ORG
Kanekoa ORG
November 22 DATE
2020 DATE
Two CARDINAL

Biden PERSON
Democrats NORP
Privacy Policy and Terms of Use You ORG
97% PERCENT
1385 DATE
3% PERCENT
39 CARDINAL
Georgia GPE
Democrat NORP
Hundreds CARDINAL
Biden PERSON
Kanekoa ORG
November 22 DATE
2020 DATE
Biden PERSON
Georgia GPE
20 years DATE
Ballots PERSON
approximately 98% PERCENT
Joe PERSON
Kanekoa ORG
November 22 DATE
2020 DATE
Republicans NORP
Fulton County GPE
the State Farm Arena ORG
10:30PM DATE
election night TIME
1:00AM TIME
This 98% PERCENT
23,487 CARDINAL
Biden PERSON
12:18AM EST TIME
Nov 4 DATE
pic.twitter.com/YdHtLm7tOM ORG
Kanekoa ORG
November 22 DATE
2020 DATE
Georgia GPE
Trump ORG
Georgians PERSON
Biden PERSON
Alabama GPE
Walmart ORG
decades DATE
Robert Garner PERSON
Decatur WalMart ORG
hundreds CARDINAL
thousands CARDINAL
crystal meth PRODUCT

Chief Garner PERSON
Walmart ORG
CNBC/Change Research ORG
only three percent PERCENT
three percent PERCENT
Donald Trump PERSON
Joe Biden PERSON
2024 DATE
CNBC ORG
73% PERCENT
Trump ORG
Another 24% PERCENT
Only three percent PERCENT
Biden PERSON
Another 31% PERCENT
Two-thirds CARDINAL
66% PERCENT
Trump PERSON
LOL ORG
CNBC ORG
Trump ORG
Trump ORG
CNBC ORG
CNBC ORG
Trump ORG
only 21 days DATE
Al Gore PERSON
38 days DATE
2000 DATE
Trump ORG
November 3 DATE
four CARDINAL
four CARDINAL
Pennsylvania GPE
Wisconsin GPE
Georgia GPE
Michigan GPE
the night TIME
Trump PERSON
Biden PERSON
1984 DATE
Trump ORG
81% PERCENT
Biden PERSON
Only 19% PERCENT
Biden PERSON
LOL ORG
Trump ORG
Democrats NORP
Trump ORG

Obama/Biden ORG
Trump ORG
Biden PERSON
CNBC ORG
1,203 CARDINAL
Trump ORG
2020 DATE
Wednesday to Saturday DATE
minus 2.83 CARDINAL
CNBC ORG
NBC ORG
CNBC ORG
LOL ORG
Trump ORG
# CARDINAL
Jonathan Ashworth PERSON
Labour’s Shadow Health ORG
today DATE
3-4 months DATE
Labour ORG
Britain GPE
3-4 months DATE
Labour ORG
an additional £25 million MONEY
Labour ORG
National Child Health Fund ORG
Britain GPE
Today DATE
Labour ORG
Jonathan Ashworth PERSON
Health Visitor Programme ORG
2015 DATE
Between 2016/17 CARDINAL
2017/18 DATE
£55.2 million MONEY
0 CARDINAL
this Tory Government ORG
8,244 CARDINAL
NHS ORG
December 2017 DATE
August 2013 DATE
10,309 CARDINAL
October 2015 DATE
more than 2,065 CARDINAL
20% PERCENT
just over two years DATE
12% PERCENT
a New Birth Visit ORG
17% PERCENT
the South West LOC

England GPE
17% PERCENT
6-8 week DATE
London GPE
1/3 CARDINAL
25% PERCENT
one year DATE
12 CARDINAL
London GPE
44% PERCENT
90% PERCENT
North East LOC
two and a half year DATE
just 64% PERCENT
London GPE
73% PERCENT
the East of England LOC
Jonathan Ashworth PERSON
today DATE
Infant Feeding PERSON
a Labour Government ORG
the Coalition Government ORG
UNICEF ORG
Scotland GPE
Northern Ireland GPE
70% PERCENT
the year 2030 DATE
Jonathan Ashworth PERSON
Labour’s Shadow Health ORG
London GPE
Michael Marmot PERSON
UK GPE
70th ORDINAL
NHS ORG
David Cameron PERSON
Theresa May PERSON
more than 20 percent PERCENT
just over two years DATE
England GPE
UK GPE
Health Visitors ORG
3-4 months DATE
£25 million MONEY
National Child Health Fund ORG
20 per cent MONEY
100 per cent MONEY
South African NORP
Alfred Ndlovu PERSON
30 days DATE
Jesus Christ PERSON

40 days DATE
40 nights DATE
Buzz South Africa PERSON
44-year-old DATE
June 17 DATE
bush PERSON
Jesus PERSON
Jesus Christ PERSON
40 days DATE
Alfred Ndlovu PERSON
just a month DATE
One CARDINAL
2018 DATE
an election year DATE
Tammy Baldwin PERSON
Wisconsin GPE
the last 25 years DATE
Baldwin PERSON
Wisconsin GPE
Almost overnight TIME
months and months DATE
the past two decades DATE
Washington GPE
California GPE
New York GPE
the Badger State LOC
Baldwin PERSON
Wisconsin GPE
Republican NORP
Baldwin PERSON
Democrat NORP
millions CARDINAL
Wisconsinites NORP
Americans NORP
Baldwin PERSON
Washington GPE
America GPE
Washington D.C. GPE
Buy America LOC
Washington GPE
Baldwin PERSON
almost 30 years DATE
Wisconsin GPE
Wisconsin GPE
four CARDINAL
more than $2,500 MONEY
Baldwin PERSON
$200 plus MONEY
New York GPE
California GPE

Beloit ORG
Wisconsinites NORP
California GPE
New York GPE
New Jersey’s GPE
Congressional NORP
Wisconsin GPE
Baldwin PERSON
one CARDINAL
semiannual DATE
election-year DATE
Wisconsin GPE
Baldwin PERSON
Wisconsin GPE
Wisconsin GPE
U.S. NORP
Leah Vukmir PERSON
Brookfield GPE
Wisconsin GPE
The Daily Caller ORG
ROME ORG
Reuters ORG
Catholic NORP
Italy GPE
these days DATE
St. Mark's Square GPE
Italy GPE
Venice GPE
Italy GPE
March 10, 2020 DATE
REUTERS ORG
Manuel Silvestri PERSON
Italy GPE
Catholic Italy ORG
Europe LOC
Last Saturday DATE
Italian NORP
first ORDINAL
Sunday DATE
Riccardo Lamba PERSON
San Ponziano GPE
Rome GPE
About 70 CARDINAL
2 CARDINAL
Lamba PERSON
this Sunday DATE
the day DATE
Tuesday DATE
Basilica LOC
Tuesday DATE

Vatican ORG
at least April 3 DATE
a matter of hours TIME
Lamba ORG
at least April 3 DATE
Sabrina Bucci PERSON
Esclusivevent GPE
Rome GPE
St. Peter’s Basilica GPE
five CARDINAL
Catholics NORP
morning TIME
Stefano Chiericoni PERSON
one CARDINAL
Rome GPE
the Mental Health Taskforce ORG
Paul Farmer PERSON
2021 DATE
70,000 CARDINAL
30,000 CARDINAL
25% PERCENT
10% PERCENT
Health ORG
Jeremy Hunt PERSON
7 day DATE
first ORDINAL
One CARDINAL
4 CARDINAL
NHS ORG
£105 billion MONEY
every year DATE
2010 DATE
NHS ORG
£11.7 billion MONEY
last year DATE
first ORDINAL
£1.4 billion MONEY
Alistair Burt PERSON
Sea LOC
Elizabeth Keatinge PERSON
Earth LOC
Arctic LOC
Antarctic LOC
January DATE
this week DATE
the summer DATE
winter DATE
Arctic sea LOC
this January DATE
5.17 million square miles QUANTITY

the month DATE
38-year DATE
the National Snow and Ice Data Center ORG
100,000 square miles QUANTITY
the previous January DATE
just last year DATE
Arctic sea LOC
January 2017 DATE
January DATE
38 CARDINAL
NSIDC News (@NSIDC ORG
February 7, 2017  January DATE
the Arctic Ocean LOC
NASA ORG
Arctic LOC
9 degrees QUANTITY
the month DATE
Antarctica LOC
Antarctic LOC
summer DATE
the Amundsen Sea LOC
Arctic LOC
U.S. GPE
Sea LOC
the Southern Hemisphere LOC
January 2017 DATE
#Antarctica https://t.co/3pPss4vRVJ pic.twitter.com/IKvLna3Ull MONEY
February 7, 2017 DATE
summer DATE
Arctic LOC
the past few decades DATE
the National Oceanic and Atmospheric Administration ORG
the latter half of the 20th century DATE
Antarctic LOC
wildly year to year DATE
NASA ORG
Walt Meier PERSON
thousands of years DATE
Advertisement Sani Aliyu PERSON
PTF ORG
COVID-19 ORG
PTF ORG
Thursday DATE
Aliyu PERSON
tertiary ORDINAL
March DATE
COVID-19 ORG
July DATE
the West African Senior School Certificate Examination ORG
WASSCE ORG

NHS ORG
last year DATE
The British Social Attitudes Survey ORG
1983 DATE
2015 DATE
nearly 2,200 CARDINAL
NHS ORG
60% PERCENT
70% PERCENT
2010 DATE
Some 23% PERCENT
eight CARDINAL
the year before DATE
a year DATE
NHS ORG
over half CARDINAL
A&E ORG
GP ORG
NatCen Social Research - covered ORG
Scotland GPE
Wales GPE
England GPE
three CARDINAL
NHS ORG
NHS ORG
Chris Ham PERSON
the King's Fund ORG
NHS ORG
NHS ORG
Rob Webster PERSON
the NHS Confederation ORG
NHS ORG
the Department of Health in England ORG
NHS ORG
Scottish NORP
Shona Robison PERSON
Antarctica LOC
Miles PERSON
the South Pole LOC
20 miles QUANTITY
ROSS ORG
Lewis Bay PERSON
Earth LOC
Aurora Glacier Scientists PERSON
McMurdo Station ORG
the Ross Ice Shelf ORG
Texas GPE
Bight Hut Point Peninsula PERSON
Williams Field PERSON
Mount Heine McMURDO ORG

BLACK ISLAND LOC
Heald Island Eady LOC
Koettlitz Glacier PERSON
the New York City Marathon ORG
about four and a half hours CARDINAL
the Ross Ice Shelf GPE
Lankester Hoffman Point Bertoglio Glacier Brosnahan Island Mount
Keltie PERSON
RANGE Cape Murray ROSS PERSON
The Nozzle” Brown Hills Tentacle WORK_OF_ART
Ridge Cranfield PERSON
Spur BYRD PERSON
Darwin Mountains PERSON
Yancey Glacier PERSON
Mount Ash GPE
Lieske Glacier PERSON
Derrick Peak PERSON
Glacier MOUNT LOC
Peckham Glacier PERSON
Henderson Mount PERSON
Nunataks Vantage Hill ORG
Peaks ORG
Sefton Glacier Bates PERSON
2,300 more miles QUANTITY
East Antarctic LOC
Antarctica LOC
Miles PERSON
the South Pole LOC
Cape Tennyson ROSS PERSON
McMurdo Station ORG
the Ross Ice Shelf ORG
Texas GPE
McMurdo PERSON
the New York City Marathon ORG
about four and a half hours CARDINAL
Judith Glacier PERSON
Mountain Vantage Hill FAC
2,300 more miles QUANTITY
East Antarctic LOC
Antarctica LOC
Miles PERSON
the South Pole LOC
Lewis Bay PERSON
McMurdo Station ORG
the Ross Ice Shelf ORG
Texas GPE
Hut Point Peninsula McMURDO ORG
BLACK ISLAND LOC
Heald Island Eady LOC
Koettlitz Glacier PERSON

the New York City Marathon ORG
about four and a half hours CARDINAL
Ross Ice Shelf ORG
Lankester Hoffman Point Bertoglio Glacier Brosnahan Island Mount
Keltie PERSON
RANGE Cape Murray ROSS PERSON
The Nozzle” Brown Hills Tentacle WORK_OF_ART
Ridge Cranfield PERSON
Spur BYRD PERSON
Ragotzkie Glacier PERSON
Darwin Mountains PERSON
Yancey Glacier PERSON
Forbes Ridge HATHERTON PERSON
Lieske Glacier PERSON
Derrick Peak PERSON
Judith Glacier PERSON
Peckham Glacier PERSON
Henderson Haven PERSON
Peaks ORG
Sefton Glacier Bates PERSON
2,300 more miles QUANTITY
East Antarctic LOC
three CARDINAL
New York Times ORG
Antarctica LOC
the Ross Ice Shelf ORG
100 feet QUANTITY
900 feet QUANTITY
thousand-year DATE
the Ross Ice Shelf ORG
Antarctica LOC
West Antarctic LOC
the Ross Ice Shelf ORG
the Ross Ice Shelf ORG
the middle of this century DATE
Antarctica LOC
West Antarctic LOC
10 to 15 feet QUANTITY
well over a century DATE
as much as six feet QUANTITY
the end of this century DATE
West Antarctic LOC
hundreds of years DATE
Robin E. Bell PERSON
Columbia University ORG
the Ross Ice Shelf ORG
late last year DATE
December DATE
Columbia ORG
Kirsty J. Tinto PERSON

a thousand years DATE
Kirsty J. Tinto PERSON
Hercules GPE
the Ross Ice Shelf ORG
The Ross Ice Shelf ORG
Columbia ORG
West Antarctica LOC
Robert A. Bindschadler PERSON
NASA ORG
centuries DATE
Eric J. Steig PERSON
the University of Washington ORG
Antarctica LOC
West Antarctica LOC
West Antarctica LOC
the Amundsen Sea LOC
Ross Ice Shelf Pine Island Glacier Thwaites ORG
2010 DATE
The Amundsen Sea LOC
American NORP
British NORP
two CARDINAL
tens of millions of dollars MONEY
Ted A. Scambos PERSON
University of Colorado ORG
Antarctica LOC
About 120,000 years ago DATE
the last ice age DATE
coming decades DATE
20 to 30 feet QUANTITY
today DATE
Greenland GPE
Antarctica LOC
last year DATE
Robert M. DeConto PERSON
the University of Massachusetts ORG
Amherst GPE
David Pollard PERSON
Pennsylvania State University ORG
the next few decades DATE
Antarctica LOC
DeConto PERSON
Antarctica LOC
Coronavirus ORG
Helen Harwatt PERSON
Harwatt PERSON
Oregon State University ORG
Bard College ORG
Loma Linda University ORG
American NORP

```
one CARDINAL
U.S. GPE
2020 DATE
Barack Obama PERSON
2009 DATE
one CARDINAL
between 46 and 74 percent DATE
Harwatt PERSON
Brazilian NORP
Amazon ORG
38,000 CARDINAL
900 metric tons QUANTITY
Brazil GPE
around 212 million CARDINAL
June DATE
U.S. GPE
Brazil GPE
the United Nations ORG
33 percent PERCENT
Earth LOC
26 percent PERCENT
Earth LOC
almost a third CARDINAL
Earth LOC
WASHINGTON GPE
Pentagon ORG
Pentagon ORG
FOX News ORG
Trump PERSON
```

Exemple d'affichage graphique des entités nommées d'un texte

```
#displacy.render(nlp(textenlp),style="ent", jupyter=True)
nlp = spacy.load("en_core_web_sm")
text = dftrain['text'].iloc[3]  # Choisir le premier texte comme exemple
doc = nlp(text)

displacy.render(doc, style="ent", jupyter=True)
```

```
<IPython.core.display.HTML object>
```

Cette fonction a pour but d'ajouter le type d'entité de chaque token détecté à coté de chaque entité détéctée dans le texte qu'elle prend en entrée, et retourne le texte modifié avec les informations ajoutées sur les entités nommées.

```
def add_entity_name(text):
    if text is None or pd.isna(text):
        return ""

    doc = nlp(text)
```

```
    # Créer une liste pour stocker les nouveaux tokens
    nouveaux_tokens = []

    # Parcourir les tokens et ajouter des informations selon le type
d'entité
    for token in doc:
        if token.ent_type_:
            # Ajouter le nom de l'entité (type d'entité) à côté de
l'entité
            nouveaux_tokens.append(f"{token.text}
({token.ent_type_})")
        else:
            nouveaux_tokens.append(token.text)

    # Reconstruire le texte avec les informations ajoutées
    nouveau_texte = " ".join(nouveaux_tokens)
    return nouveau_texte
```

Application de la fonction précédente à la colonne "title" ainsi que la colonne "texte avec la méthode apply

```
# Assurez-vous que les colonnes "text" et "titre" contiennent des
chaînes de caractères
dftrain['text'] = dftrain['text'].astype(str)
dftrain['title'] = dftrain['title'].astype(str)

# Appliquer la fonction à la colonne "title"
dftrain['text'] = dftrain['text'].apply(add_entity_name)
dftrain['title'] = dftrain['title'].apply(add_entity_name)

print(dftrain['text'])
print(dftrain['title'])
```

```
947     War - torn eastern regions of Ukraine (GPE) ha...
2224    TIJUANA , Mexico (GPE) — It ’s the image from ...
1307    Today , Congresswoman Maxine Waters D - CA , C...
798     Meghan (PERSON) Markle (PERSON) will use the f...
320     Further proof that Democrats (NORP) are the gr...
                              ...
1160    The scale of Antarctica (LOC) is startling . M...
570     Coronavirus (ORG) may be sexually transmitted ...
1200    Like what ?   Helen (PERSON) Harwatt (PERSON) ...
2190                         Tumeric kills cancer not patient
391     WASHINGTON (GPE) , DC — The Pentagon (ORG) has...
Name: text, Length: 468, dtype: object
947     Look No Further , The Best Doctor Strange in t...
2224    A discussion of ‘ smokers ’ black lungs ’ star...
1307    Democratic (NORP) Lawmaker introduces bill to ...
798     Newton (GPE) Emerson : Swiss model offers food...
320     Democrats (NORP) Introduce Bill (PERSON) To (P...
```

```
                              ...
1160        Miles (PERSON) of Ice Collapsing Into the Sea
570     Universal (ORG) Credit (ORG) leaves working fa...
1200    If Everyone Ate Beans Instead of Beef (WORK_OF...
2190    Vermont (GPE) state trooper revived with Narca...
391     Pentagon (ORG) Confirms Coronavirus Accidently...
Name: title, Length: 468, dtype: object
```

On scinde les données de la colonne "title" en jeu d'apprentissage et jeu de test

```
X=dftrain.iloc[0:, 1:4]
print(X)

X_train,X_test,y_train,y_test=train_test_split(X,y,test_size = 0.2,
random_state=10)
print("X_train is",X_train)
print("y_train is",y_train)
print("X_test is",X_test)
print("y_test is",y_test)
```

```
                                                   text  \
947   War - torn eastern regions of Ukraine (GPE) ha...
2224  TIJUANA , Mexico (GPE) — It ’s the image from ...
1307  Today , Congresswoman Maxine Waters D - CA , C...
798   Meghan (PERSON) Markle (PERSON) will use the f...
320   Further proof that Democrats (NORP) are the gr...
...                                                 ...
1160  The scale of Antarctica (LOC) is startling . M...
570   Coronavirus (ORG) may be sexually transmitted ...
1200  Like what ?   Helen (PERSON) Harwatt (PERSON) ...
2190                    Tumeric kills cancer not patient
391   WASHINGTON (GPE) , DC — The Pentagon (ORG) has...

                                                  title rating
947   Look No Further , The Best Doctor Strange in t...  FALSE
2224  A discussion of ‘ smokers ’ black lungs ’ star...   TRUE
1307  Democratic (NORP) Lawmaker introduces bill to ...  FALSE
798   Newton (GPE) Emerson : Swiss model offers food...  other
320   Democrats (NORP) Introduce Bill (PERSON) To (P...  FALSE
...                                                 ...    ...
1160      Miles (PERSON) of Ice Collapsing Into the Sea   TRUE
570   Universal (ORG) Credit (ORG) leaves working fa...  other
1200  If Everyone Ate Beans Instead of Beef (WORK_OF...  other
2190  Vermont (GPE) state trooper revived with Narca...  other
391   Pentagon (ORG) Confirms Coronavirus Accidently...  FALSE

[468 rows x 3 columns]
X_train is                                                    text  \
2006  Historians may look to 2015 (DATE) as the (DAT...
```

```
1834  Coronavirus (ORG) may be sexually transmitted ...
530   Contractors bidding for work with the governme...
564   More CO2 would actually help the planet , says...
340   To say out - loud that you find the results of...
...                                                  ...
1775  Last (DATE) week (DATE) , in the (DATE) days (...
208   This is one in a series of articles taken from...
562   Food parcels arriving at the (ORG) Community (...
676   On Tuesday (DATE) , radio show host John (PERS...
1925  A South (NORP) African (NORP) pastor , Alfred ...

                                                  title rating
2006                                                nan  other
1834  Universal (ORG) Credit (ORG) leaves working fa...  other
530   Firms bidding for government contracts asked i...  other
564   Mitt (PERSON) Romney (PERSON) transfers $ (MON...  other
340   Reasons why the 2020 (DATE) presidential elect...  other
...                                                 ...    ...
1775  Bernie (PERSON) makes it official : It ’s Bide...  other
208   European (NORP) royals killing naked children ...  other
562   Adam (PERSON) Castillejo (PERSON) is still fre...  other
676   Warren (PERSON) Statement (PERSON) on Boeing (...  other
1925  Joy (PERSON) Covey (PERSON) : Amazon (ORG) pio...  other

[374 rows x 3 columns]
y_train is 2006    OTHER
1834    OTHER
530     OTHER
564     OTHER
340     OTHER
        ...
1775    OTHER
208     OTHER
562     OTHER
676     OTHER
1925    OTHER
Name: regrouped, Length: 374, dtype: object
X_test is                                                     text  \
459   Thank you to Universities UK (GPE) UUK for hos...
1519  Can any government statistics on COVID-19 (ORG...
925   MOSCOW (GPE) — Russian (NORP) President Vladim...
1857  Please enable cookies on your web browser in o...
2386  PUPILS aged just (CARDINAL) five (CARDINAL) ha...
...                                                  ...
1778  With a smile on her face , City Clerk Susana (...
2205  It was an accurate and judicious answer , so n...
1568  Barack (PERSON) Obama (PERSON) , a former Pres...
66    Pennsylvania (GPE) rejects 372,000 (CARDINAL) ...
2355  Rises in National (ORG) Insurance (ORG) Contri...
```

```
                                                  title rating
459   USCIS (ORG) Announces Final Rule Enforcing Lon...   TRUE
1519  The CDC (ORG) Confesses to Lying About COVID-1...  FALSE
925         Short breaks damage young people ’s futures  other
1857  Denying 2000 (DATE) years (DATE) of the (FAC) ...  FALSE
2386  Pervs (ORG) ' aged five (CARDINAL) School (ORG...  other
...                                                 ...    ...
1778  Trump (ORG) administration asks Supreme (ORG) ...   TRUE
2205          A 62 (PERCENT) % (PERCENT) Top Tax Rate ?  other
1568  Former President Barack (PERSON) Obama (PERSON...  FALSE
66    Pennsylvania (GPE) rejects 372,000 (CARDINAL) ...  FALSE
2355  Budget 2017 (DATE) : National (ORG) Insurance ...  other

[94 rows x 3 columns]
y_test is 459     TRUE/FALSE
1519    TRUE/FALSE
925          OTHER
1857    TRUE/FALSE
2386         OTHER
           ...    
1778    TRUE/FALSE
2205         OTHER
1568    TRUE/FALSE
66      TRUE/FALSE
2355         OTHER
Name: regrouped, Length: 94, dtype: object
```

### Etape 2 : Classification selon la colonne TEXTE

On met la colonne "texte" dans les variables X_train et X_test

```
X_train_text=X_train['text']
X_train_text.reset_index(drop = True, inplace = True)
X_test_text=X_test['text']
X_test_text.reset_index(drop = True, inplace = True)
```

Ici, c'est une étape importante, on va tester différents classifieurs, pour chacun des classifieurs, on va appliquer le prétraitement + Vectorisation TfIdf, et on applique une cross_val_score avec un Kfold de 10 fois, par la suite on stocke dans une liste all_results la moyenne des accuracy + l'écart type et on la trie par ordre décroissant de moyenne d'accuracy et d'écart type. on remarque que les 2 meilleurs sont SVM et RF qu'on va séléctionner pour leur appliquer le GridSearch sur les paramètres des prétraitements + leurs hyperparamètres pour pouvoir choisir le meilleur.

```
from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.pipeline import Pipeline
from sklearn.metrics import accuracy_score
```

```python
from sklearn.naive_bayes import MultinomialNB
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.naive_bayes import MultinomialNB
import time

np.random.seed(42)  # Set the random seed for NumPy

score = 'accuracy'
seed = 7
allresults = []
results = []
names = []


# Liste des modèles à tester
models = [
    ('MultinomialNB', MultinomialNB()),
    ('LogisticRegression', LogisticRegression(random_state=42)),
    ('KNN', KNeighborsClassifier()),
    ('CART', DecisionTreeClassifier(random_state=42)),
    ('RF', RandomForestClassifier(random_state=42)),
    ('SVM', SVC(random_state=42))
]


# Création d'un pipeline pour chaque modèle
pipelines = []
for name,model in models:
    pipeline = Pipeline([
        ('normalize', TextNormalizer()),
        ('tfidf', TfidfVectorizer()),
        (name,model)
    ])
    pipelines.append((name,pipeline))

all_results=[]
scores=[]
names=[]
for p in pipelines:
    print(p[1])
    # cross validation en 10 fois
    kfold = KFold(n_splits=10,random_state=seed,shuffle=True)
    start_time = time.time()
    # application de la classification
    cv_results = cross_val_score(p[1],X_train_text,y_train, cv=kfold,
scoring=score)
    scores.append(cv_results)
    names.append(p[0])
```

```
    all_results.append((p[0],cv_results.mean(),cv_results.std()))
    end_time = time.time()


print("all resultats", all_results)

all_results = sorted(all_results, key=lambda x: (-x[1], -x[2]))
print("all resultats", all_results)
```

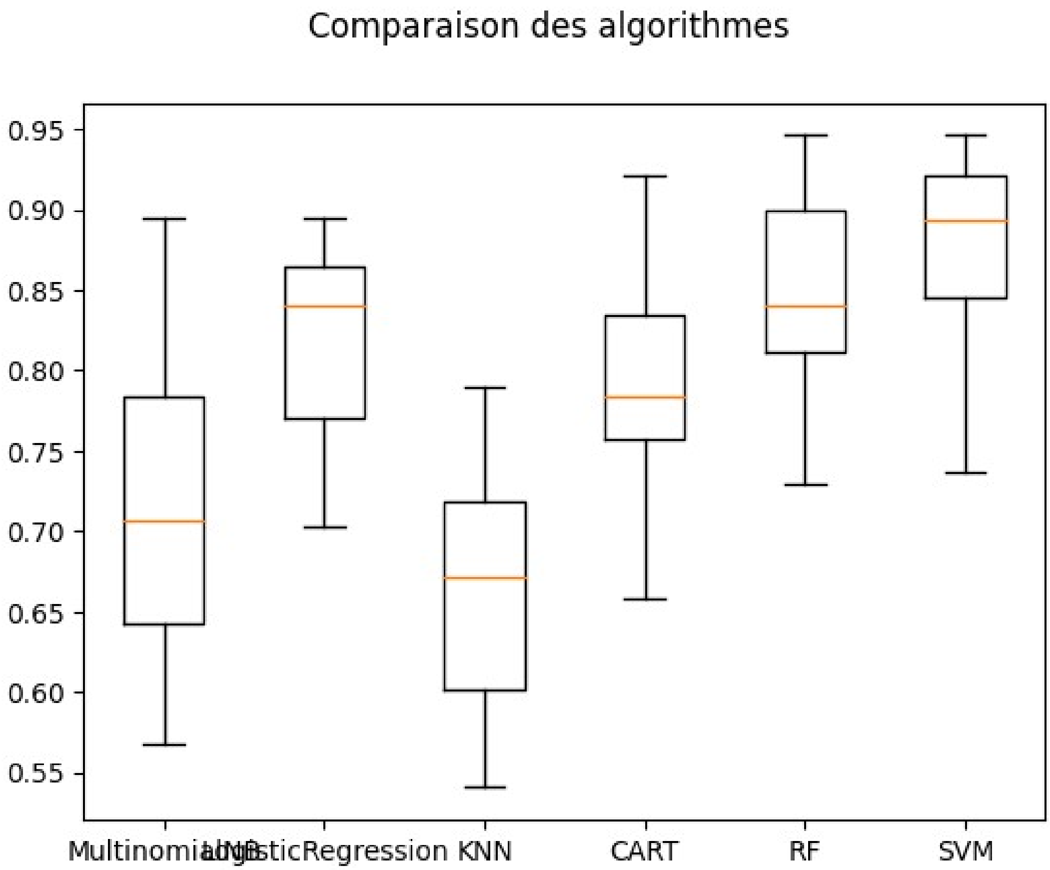
```
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('MultinomialNB', MultinomialNB())])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('LogisticRegression',
LogisticRegression(random_state=42))])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('KNN', KNeighborsClassifier())])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('CART', DecisionTreeClassifier(random_state=42))])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('RF', RandomForestClassifier(random_state=42))])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('SVM', SVC(random_state=42))])
all resultats [('MultinomialNB', 0.7166429587482218,
0.0966310908000528), ('LogisticRegression', 0.8179943100995732,
0.06122136042831569), ('KNN', 0.6626600284495022,
0.07788362978229361), ('CART', 0.796443812233286, 0.0719178992869139),
('RF', 0.8422475106685633, 0.0661464690353706), ('SVM',
0.8770981507823613, 0.06295300149444748)]
all resultats [('SVM', 0.8770981507823613, 0.06295300149444748),
('RF', 0.8422475106685633, 0.0661464690353706), ('LogisticRegression',
0.8179943100995732, 0.06122136042831569), ('CART', 0.796443812233286,
0.0719178992869139), ('MultinomialNB', 0.7166429587482218,
0.0966310908000528), ('KNN', 0.6626600284495022, 0.07788362978229361)]
```

On affiche les boites à moustache pour mieux visualiser les résultats

```
import matplotlib.pyplot as plt
fig = plt.figure()
fig.suptitle('Comparaison des algorithmes')
ax = fig.add_subplot(111)
plt.boxplot(scores)
ax.set_xticklabels(names)
```

```
[Text(1, 0, 'MultinomialNB'),
 Text(2, 0, 'LogisticRegression'),
```

```
 Text(3, 0, 'KNN'),
 Text(4, 0, 'CART'),
 Text(5, 0, 'RF'),
 Text(6, 0, 'SVM')]
```




*Choisir les meilleurs paramètres pour SVM, RF et LR :*

On a un pipeline pour chaque prétraitement différent, on essaye pas mal (miniscule, lemmatisation, miniscule + lemmatisation..) et on stocke le fit_transorm de nos X_train, X_test sur les pipelines dans des listes qui vont contenir tous les fit_transform des pipelines pour chaque classifieur, par la suite on parcourt ces listes là, on itère dessus, et chaque élement de la liste (train) va passer par le GridSearch et puis on predict sur son corresapondant dans liste (test).

```python
from sklearn.model_selection import GridSearchCV

np.random.seed(42)  # Set the random seed for NumPy

from sklearn.datasets import fetch_20newsgroups
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfVectorizer
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.pipeline import Pipeline
```

```python
from sklearn.metrics import accuracy_score
from sklearn.naive_bayes import MultinomialNB
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.naive_bayes import MultinomialNB
from tabulate import tabulate


# pipeline de l'utilisation de TfidfVectorizer avec differents pre-
traitements
TFIDF_brut = Pipeline ([('cleaner', TextNormalizer()),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])

TFIDF_lowcase = Pipeline([('cleaner',
TextNormalizer(removestopwords=False,lowercase=True,

getstemmer=False,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])
TFIDF_lowStop = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=False,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])

TFIDF_lowStopstem = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=True,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])


# Liste de tous les modeles à tester
all_models = [
    ("TFIDF_lowcase", TFIDF_lowcase),
    ("TFIDF_lowStop", TFIDF_lowStop),
    ("TFIDF_lowStopstem",TFIDF_lowStopstem),
    ("TFIDF_brut", TFIDF_brut)
]

X_train_text_SVC = []
X_test_text_SVC = []

X_train_text_RandomForestClassifier = []
X_test_text_RandomForestClassifier = []
```

```
X_train_text_LogisticRegression = []
X_test_text_LogisticRegression = []

for name, pipeline in all_models :

X_train_text_SVC.append(pipeline.fit_transform(X_train_text).toarray()
)
    X_test_text_SVC.append(pipeline.transform(X_test_text).toarray())

X_train_text_RandomForestClassifier.append(pipeline.fit_transform(X_tr
ain_text).toarray())

X_test_text_RandomForestClassifier.append(pipeline.transform(X_test_te
xt).toarray())

X_train_text_LogisticRegression.append(pipeline.fit_transform(X_train_
text).toarray())

X_test_text_LogisticRegression.append(pipeline.transform(X_test_text).
toarray())


models = {
    'SVC': SVC(random_state=42),
    'LogisticRegression' : LogisticRegression(random_state=42),
    'RandomForestClassifier': RandomForestClassifier(random_state=42)
}


params = {'SVC': [{'C': [0.001, 0.01, 0.1, 1,2,5,7,10]},
             {'gamma': [0.001, 0.01, 0.1,0.2,0.3,0.5,0.7,1]},
             {'kernel': ['linear', 'rbf']}],
    'RandomForestClassifier': [{'n_estimators': [10, 50, 100, 200,
300]},
                              {'max_features': ['auto', 'sqrt',
'log2']}],
    'LogisticRegression': [{'penalty': ['l1', 'l2','elasticnet',
'none']},
                              {'C': [0.001, 0.01, 0.1, 1, 10, 100]},
                              {'fit_intercept': [True,False]},
                              {'solver': ['newton-cg', 'lbfgs',
'liblinear', 'sag', 'saga']},
                              {'max_iter': [100, 1000, 10000]}]
}

for model_name, model in models.items():
    score='accuracy'
    X_train_text = eval('X_train_text_' + model_name)
    X_test_text = eval('X_test_text_' + model_name)
```

```
    for i in range (len(X_train_text)):
      grid_search = GridSearchCV(model, params[model_name], n_jobs=-1,
verbose=1,scoring=score)
      print("grid search fait")
      grid_search.fit(X_train_text[i],y_train)
      print ('meilleur score %0.3f'%(grid_search.best_score_),'\n')
      print ('meilleur estimateur',grid_search.best_estimator_,'\n')
      y_pred = grid_search.predict(X_test_text[i])
      MyshowAllScores(y_test,y_pred)

      print("Ensemble des meilleurs paramètres :")
      best_parameters = grid_search.best_estimator_.get_params()
      for param_dict in params[model_name]:
        for param_name, param_value in param_dict.items():
            print("\t%s: %r" % (param_name,
best_parameters[param_name]))
```

```
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.872

meilleur estimateur SVC(C=2, random_state=42)

Accuracy : 0.915
Classification Report
              precision    recall  f1-score   support

       OTHER    0.86667   0.95122   0.90698        41
  TRUE/FALSE    0.95918   0.88679   0.92157        53

    accuracy                        0.91489        94
   macro avg    0.91293   0.91901   0.91427        94
weighted avg    0.91883   0.91489   0.91520        94

Ensemble des meilleurs paramètres :
	C: 2
	gamma: 'scale'
	kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.874

meilleur estimateur SVC(C=2, random_state=42)

Accuracy : 0.936
Classification Report
              precision    recall  f1-score   support

       OTHER    0.90698   0.95122   0.92857        41
  TRUE/FALSE    0.96078   0.92453   0.94231        53
```

```
    accuracy                        0.93617        94
   macro avg    0.93388   0.93787   0.93544        94
weighted avg    0.93732   0.93617   0.93632        94

Ensemble des meilleurs paramètres :
     C: 2
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.880

meilleur estimateur SVC(C=2, random_state=42)

Accuracy : 0.904
Classification Report
              precision    recall  f1-score   support

       OTHER    0.84783   0.95122   0.89655        41
  TRUE/FALSE    0.95833   0.86792   0.91089        53

    accuracy                        0.90426        94
   macro avg    0.90308   0.90957   0.90372        94
weighted avg    0.91013   0.90426   0.90464        94

Ensemble des meilleurs paramètres :
     C: 2
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.869

meilleur estimateur SVC(C=1, random_state=42)

Accuracy : 0.926
Classification Report
              precision    recall  f1-score   support

       OTHER    0.88636   0.95122   0.91765        41
  TRUE/FALSE    0.96000   0.90566   0.93204        53

    accuracy                        0.92553        94
   macro avg    0.92318   0.92844   0.92484        94
weighted avg    0.92788   0.92553   0.92576        94

Ensemble des meilleurs paramètres :
     C: 1
```

```
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 20 candidates, totalling 100 fits

/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_validation.py:378: FitFailedWarning:
10 fits failed out of a total of 100.
The score on these train-test partitions for these parameters will be
set to nan.
If these failures are not expected, you can try to debug them by
setting error_score='raise'.

Below are more details about the failures:
----------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
l1 penalty.

----------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
```

```
elasticnet penalty.

  warnings.warn(some_fits_failed_message, FitFailedWarning)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_searc
h.py:952: UserWarning: One or more of the test scores are non-finite:
[       nan 0.81816216        nan 0.84756757 0.51603604 0.51603604
 0.58288288 0.81816216 0.84756757 0.83956757 0.81816216 0.8261982
 0.81816216 0.81816216 0.82082883 0.81816216 0.81816216 0.81816216
 0.81816216 0.81816216]
  warnings.warn(

meilleur score 0.848

meilleur estimateur LogisticRegression(penalty='none',
random_state=42)

Accuracy : 0.862
Classification Report
              precision    recall  f1-score   support

       OTHER    0.78000   0.95122   0.85714        41
  TRUE/FALSE    0.95455   0.79245   0.86598        53

    accuracy                        0.86170        94
   macro avg    0.86727   0.87184   0.86156        94
weighted avg    0.87841   0.86170   0.86213        94

Ensemble des meilleurs paramètres :
     penalty: 'none'
     C: 1.0
     fit_intercept: True
     solver: 'lbfgs'
     max_iter: 100
grid search fait
Fitting 5 folds for each of 20 candidates, totalling 100 fits

/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_validation.py:378: FitFailedWarning:
10 fits failed out of a total of 100.
The score on these train-test partitions for these parameters will be
set to nan.
If these failures are not expected, you can try to debug them by
setting error_score='raise'.

Below are more details about the failures:
----------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
```

```
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
l1 penalty.

----------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
elasticnet penalty.

  warnings.warn(some_fits_failed_message, FitFailedWarning)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_searc
h.py:952: UserWarning: One or more of the test scores are non-finite:
[       nan 0.83145946        nan 0.85823423 0.51603604 0.51603604
 0.61225225 0.83145946 0.8609009  0.85556757 0.83145946 0.81272072
 0.83145946 0.83145946 0.82875676 0.83145946 0.83145946 0.83145946
 0.83145946 0.83145946]
  warnings.warn(

meilleur score 0.861

meilleur estimateur LogisticRegression(C=10, random_state=42)

Accuracy : 0.840
Classification Report
              precision    recall  f1-score   support
```

```
       OTHER    0.75000   0.95122   0.83871        41
  TRUE/FALSE    0.95238   0.75472   0.84211        53

    accuracy                        0.84043        94
   macro avg    0.85119   0.85297   0.84041        94
weighted avg    0.86411   0.84043   0.84062        94

Ensemble des meilleurs paramètres :
     penalty: 'l2'
     C: 10
     fit_intercept: True
     solver: 'lbfgs'
     max_iter: 100
grid search fait
Fitting 5 folds for each of 20 candidates, totalling 100 fits

/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_validation.py:378: FitFailedWarning:
10 fits failed out of a total of 100.
The score on these train-test partitions for these parameters will be
set to nan.
If these failures are not expected, you can try to debug them by
setting error_score='raise'.

Below are more details about the failures:
----------------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
l1 penalty.

----------------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
```

```
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
elasticnet penalty.

  warnings.warn(some_fits_failed_message, FitFailedWarning)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_searc
h.py:952: UserWarning: One or more of the test scores are non-finite:
[       nan 0.84756757        nan 0.87963964 0.51603604 0.51603604
 0.62828829 0.84756757 0.87697297 0.86363964 0.84756757 0.81812613
 0.84756757 0.84756757 0.8421982  0.84756757 0.84756757 0.84756757
 0.84756757 0.84756757]
  warnings.warn(

meilleur score 0.880

meilleur estimateur LogisticRegression(penalty='none',
random_state=42)

Accuracy : 0.851
Classification Report
              precision    recall  f1-score   support

       OTHER    0.76471   0.95122   0.84783        41
  TRUE/FALSE    0.95349   0.77358   0.85417        53

    accuracy                        0.85106        94
   macro avg    0.85910   0.86240   0.85100        94
weighted avg    0.87115   0.85106   0.85140        94

Ensemble des meilleurs paramètres :
     penalty: 'none'
     C: 1.0
     fit_intercept: True
     solver: 'lbfgs'
     max_iter: 100
grid search fait
Fitting 5 folds for each of 20 candidates, totalling 100 fits

/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/
_validation.py:378: FitFailedWarning:
10 fits failed out of a total of 100.
```

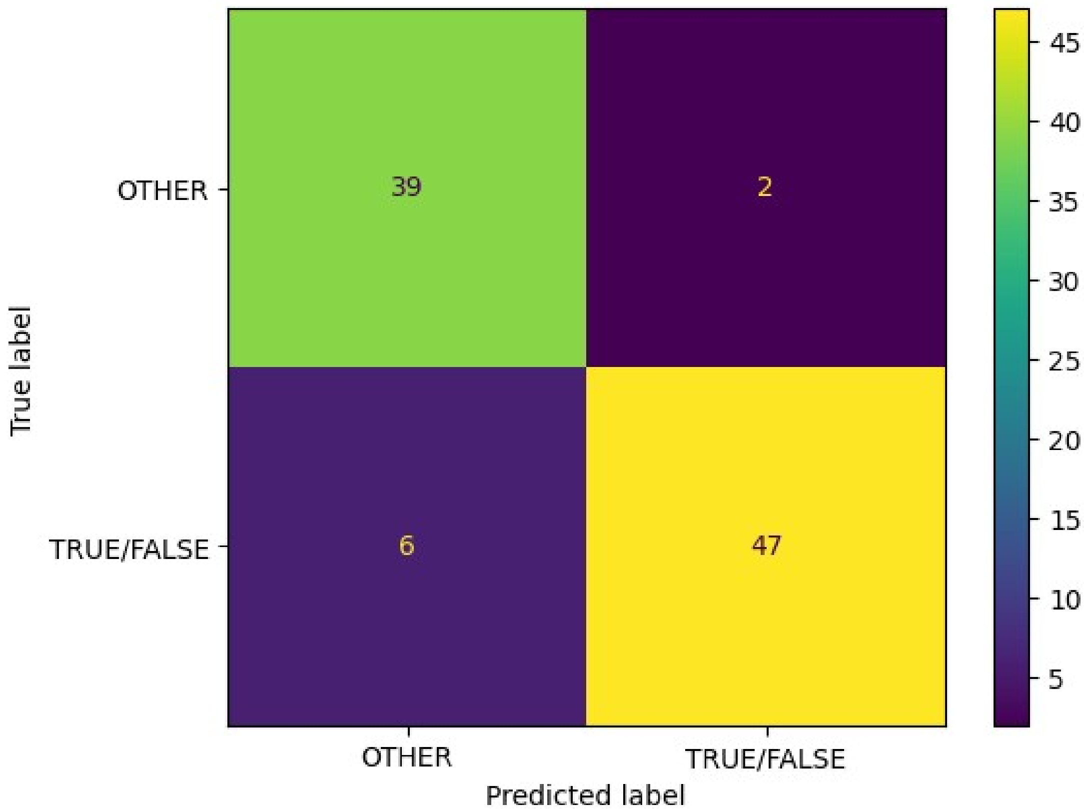
```
The score on these train-test partitions for these parameters will be
set to nan.
If these failures are not expected, you can try to debug them by
setting error_score='raise'.

Below are more details about the failures:
--------------------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
l1 penalty.

--------------------------------------------------------------------------------
----------
5 fits failed with the following error:
Traceback (most recent call last):
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_vali
dation.py", line 686, in _fit_and_score
    estimator.fit(X_train, y_train, **fit_params)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 1162, in fit
    solver = _check_solver(self.solver, self.penalty, self.dual)
  File
"/usr/local/lib/python3.10/dist-packages/sklearn/linear_model/_logisti
c.py", line 54, in _check_solver
    raise ValueError(
ValueError: Solver lbfgs supports only 'l2' or 'none' penalties, got
elasticnet penalty.

  warnings.warn(some_fits_failed_message, FitFailedWarning)
/usr/local/lib/python3.10/dist-packages/sklearn/model_selection/_searc
h.py:952: UserWarning: One or more of the test scores are non-finite:
[       nan 0.82353153        nan 0.85823423 0.51603604 0.51603604
 0.56947748 0.82353153 0.85293694 0.8529009  0.82353153 0.82082883
 0.82353153 0.82353153 0.82082883 0.82082883 0.82353153 0.82353153
```

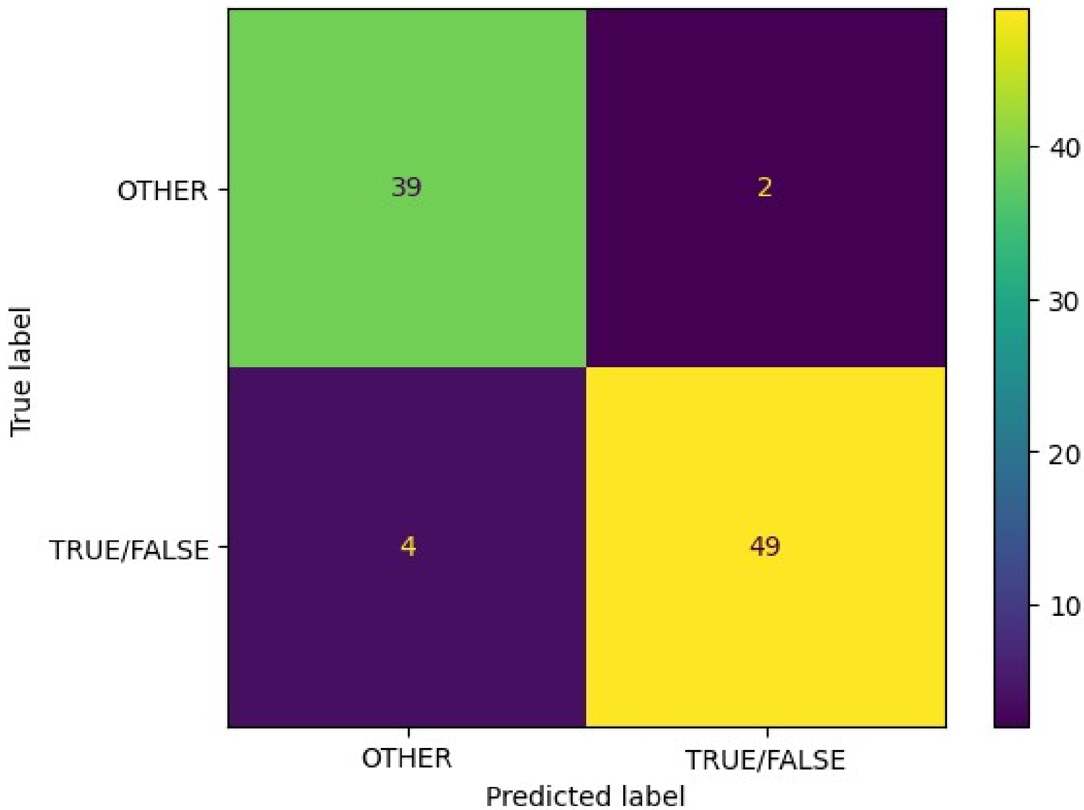
```
 0.82353153 0.82353153]
  warnings.warn(

meilleur score 0.858

meilleur estimateur LogisticRegression(penalty='none',
random_state=42)

Accuracy : 0.872
Classification Report
              precision    recall  f1-score   support

       OTHER    0.79592   0.95122   0.86667        41
  TRUE/FALSE    0.95556   0.81132   0.87755        53

    accuracy                        0.87234        94
   macro avg    0.87574   0.88127   0.87211        94
weighted avg    0.88593   0.87234   0.87280        94

Ensemble des meilleurs paramètres :
     penalty: 'none'
     C: 1.0
     fit_intercept: True
     solver: 'lbfgs'
     max_iter: 100
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.845

meilleur estimateur RandomForestClassifier(n_estimators=200,
random_state=42)

Accuracy : 0.936
Classification Report
              precision    recall  f1-score   support

       OTHER    0.90698   0.95122   0.92857        41
  TRUE/FALSE    0.96078   0.92453   0.94231        53

    accuracy                        0.93617        94
   macro avg    0.93388   0.93787   0.93544        94
weighted avg    0.93732   0.93617   0.93632        94

Ensemble des meilleurs paramètres :
     n_estimators: 200
     max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.850
```

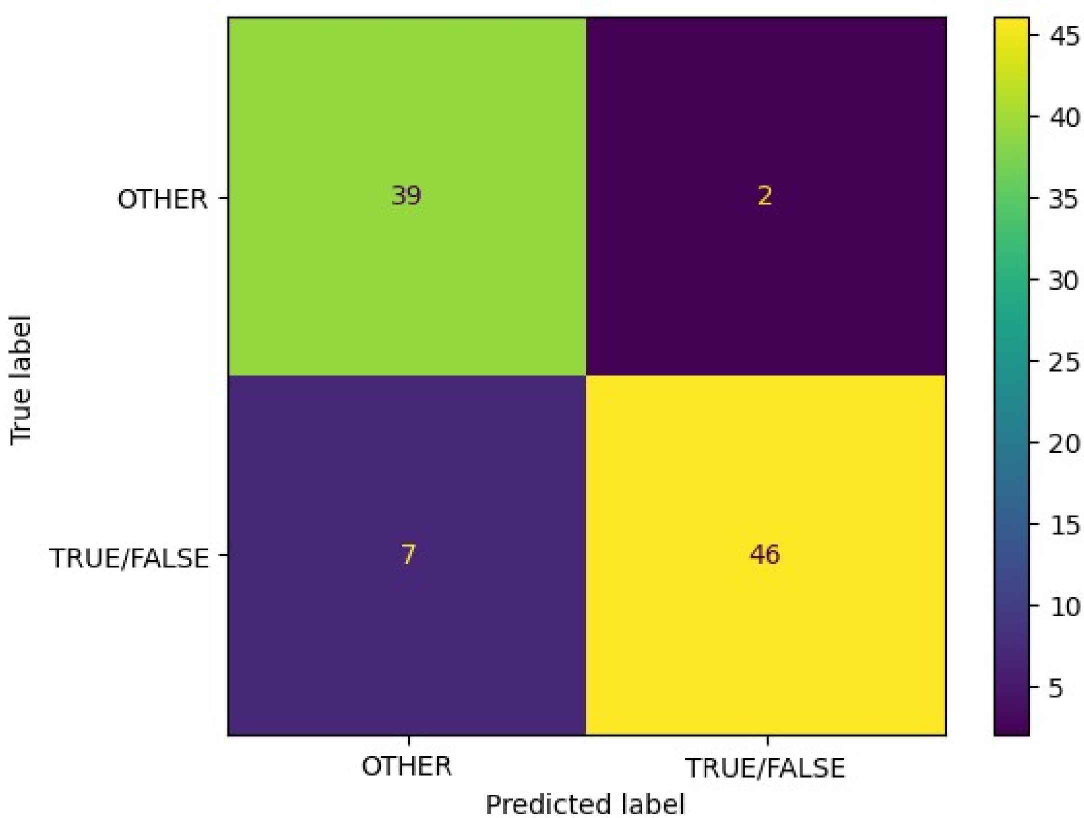
```
meilleur estimateur RandomForestClassifier(max_features='log2',
random_state=42)

Accuracy : 0.926
Classification Report
              precision    recall  f1-score   support

       OTHER    0.88636   0.95122   0.91765        41
  TRUE/FALSE    0.96000   0.90566   0.93204        53

    accuracy                        0.92553        94
   macro avg    0.92318   0.92844   0.92484        94
weighted avg    0.92788   0.92553   0.92576        94

Ensemble des meilleurs paramètres :
     n_estimators: 100
     max_features: 'log2'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.850

meilleur estimateur RandomForestClassifier(n_estimators=200,
random_state=42)

Accuracy : 0.926
Classification Report
              precision    recall  f1-score   support

       OTHER    0.88636   0.95122   0.91765        41
  TRUE/FALSE    0.96000   0.90566   0.93204        53

    accuracy                        0.92553        94
   macro avg    0.92318   0.92844   0.92484        94
weighted avg    0.92788   0.92553   0.92576        94

Ensemble des meilleurs paramètres :
     n_estimators: 200
     max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.850

meilleur estimateur RandomForestClassifier(n_estimators=300,
random_state=42)

Accuracy : 0.926
Classification Report
              precision    recall  f1-score   support
```

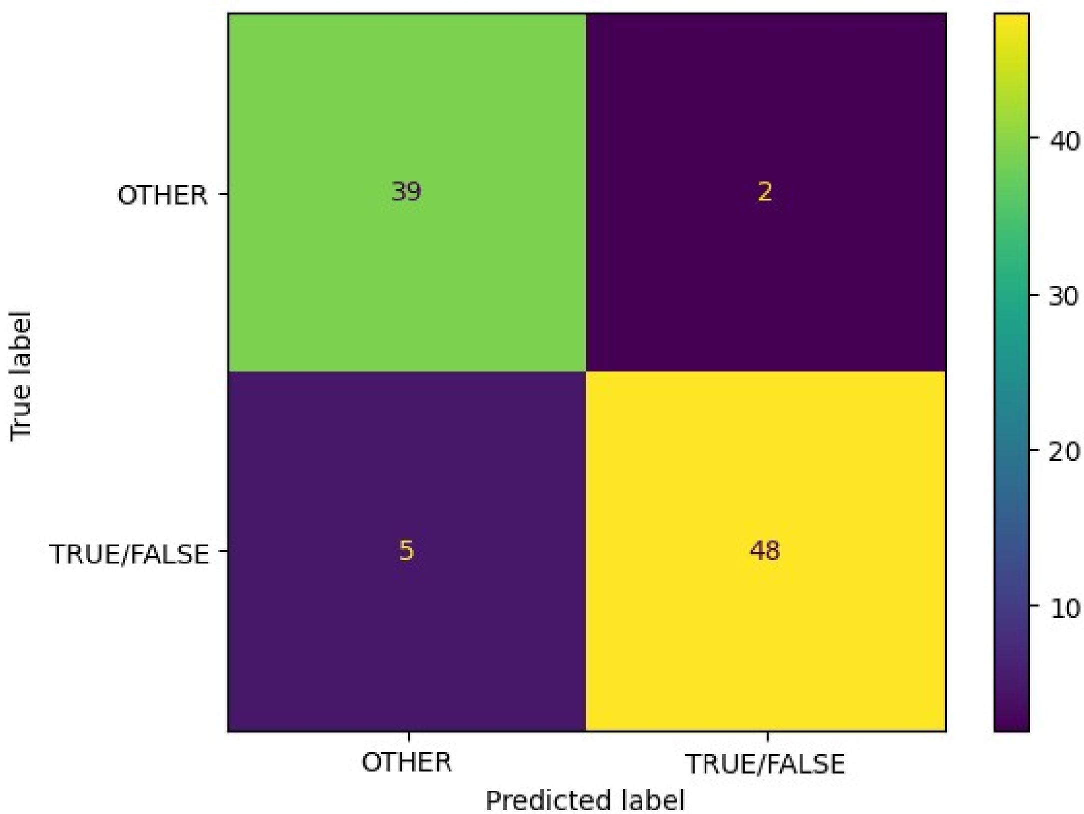
| | | | | |
|---|---|---|---|---|
| OTHER | 0.88636 | 0.95122 | 0.91765 | 41 |
| TRUE/FALSE | 0.96000 | 0.90566 | 0.93204 | 53 |
| | | | | |
| accuracy | | | 0.92553 | 94 |
| macro avg | 0.92318 | 0.92844 | 0.92484 | 94 |
| weighted avg | 0.92788 | 0.92553 | 0.92576 | 94 |

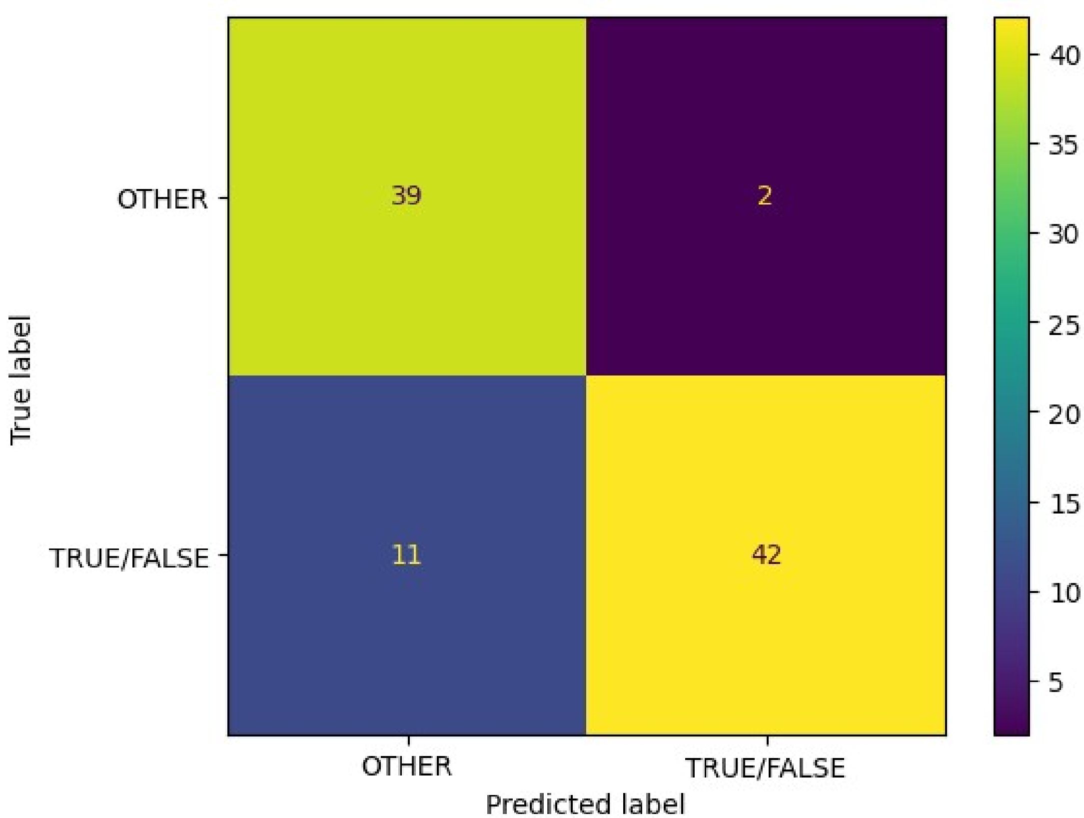
```
Ensemble des meilleurs paramètres :
      n_estimators: 300
      max_features: 'sqrt'
```

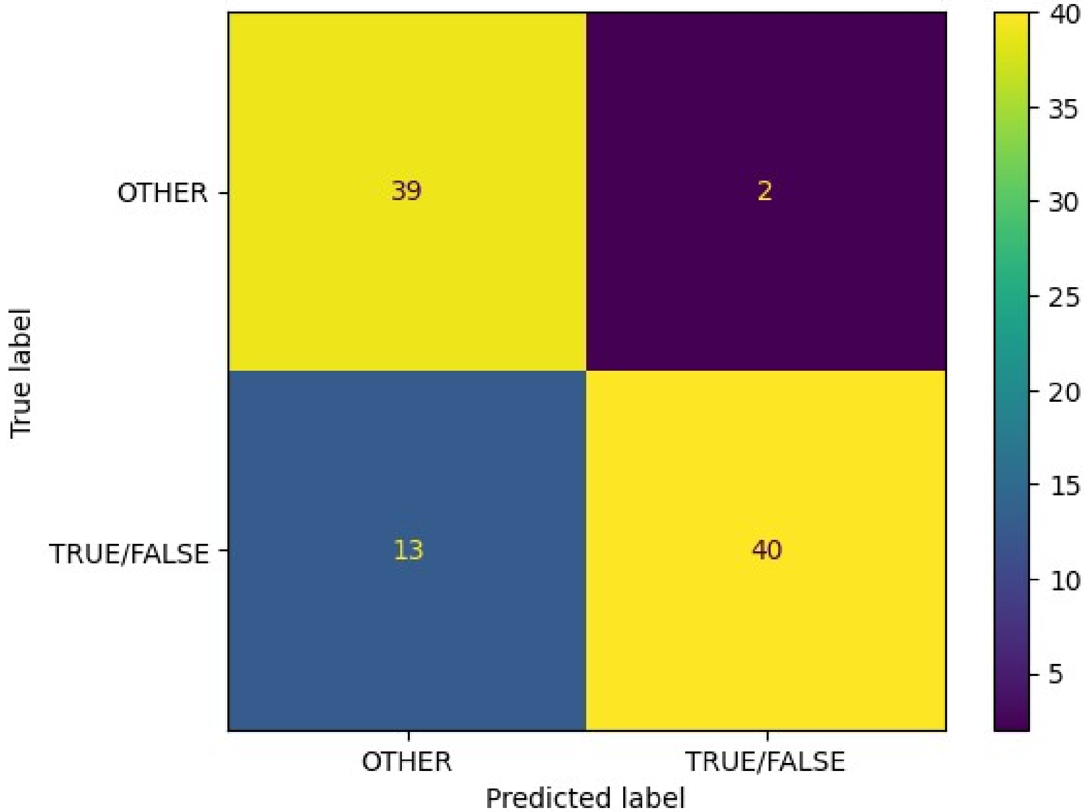


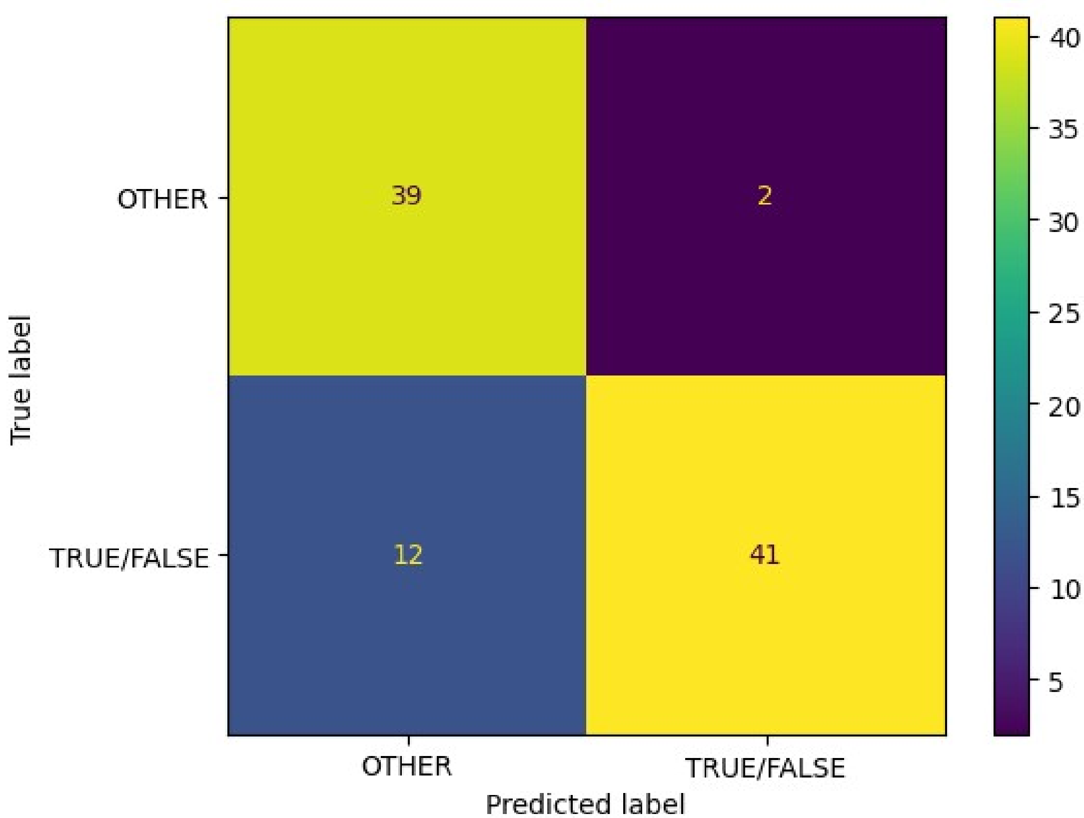

OTHER
TRUE/FALSE
True label
39
2
4
49
OTHER
TRUE/FALSE
Predicted label
40
30
20
10

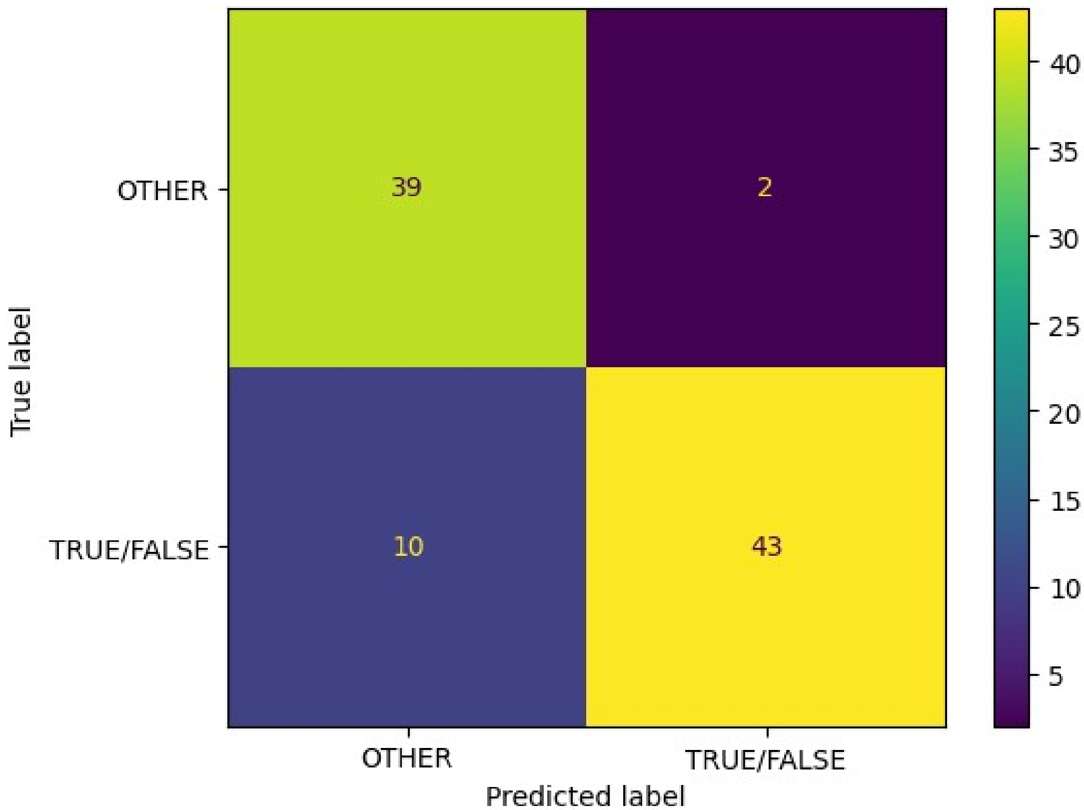

OTHER
TRUE/FALSE
True label
39
2
7
46
OTHER
TRUE/FALSE
Predicted label
45
40
35
30
25
20
15
10
5

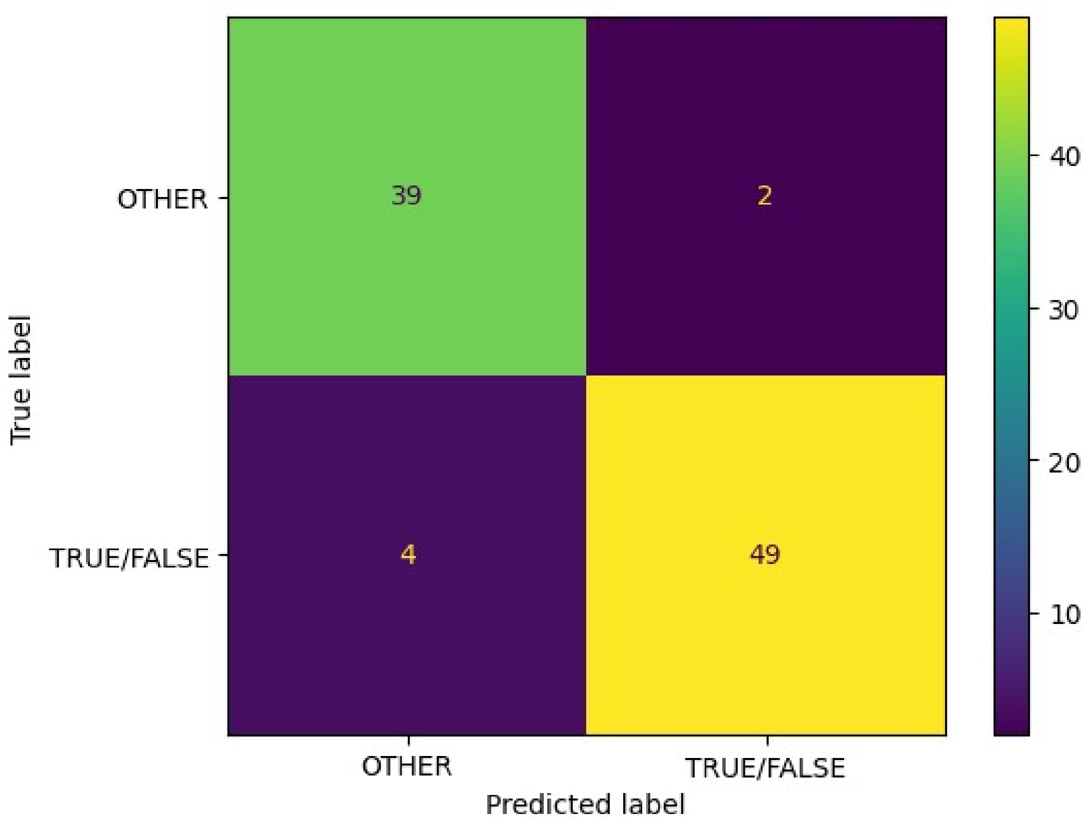

OTHER
TRUE/FALSE
True label
39
2
5
48
OTHER
TRUE/FALSE
Predicted label
40
30
20
10

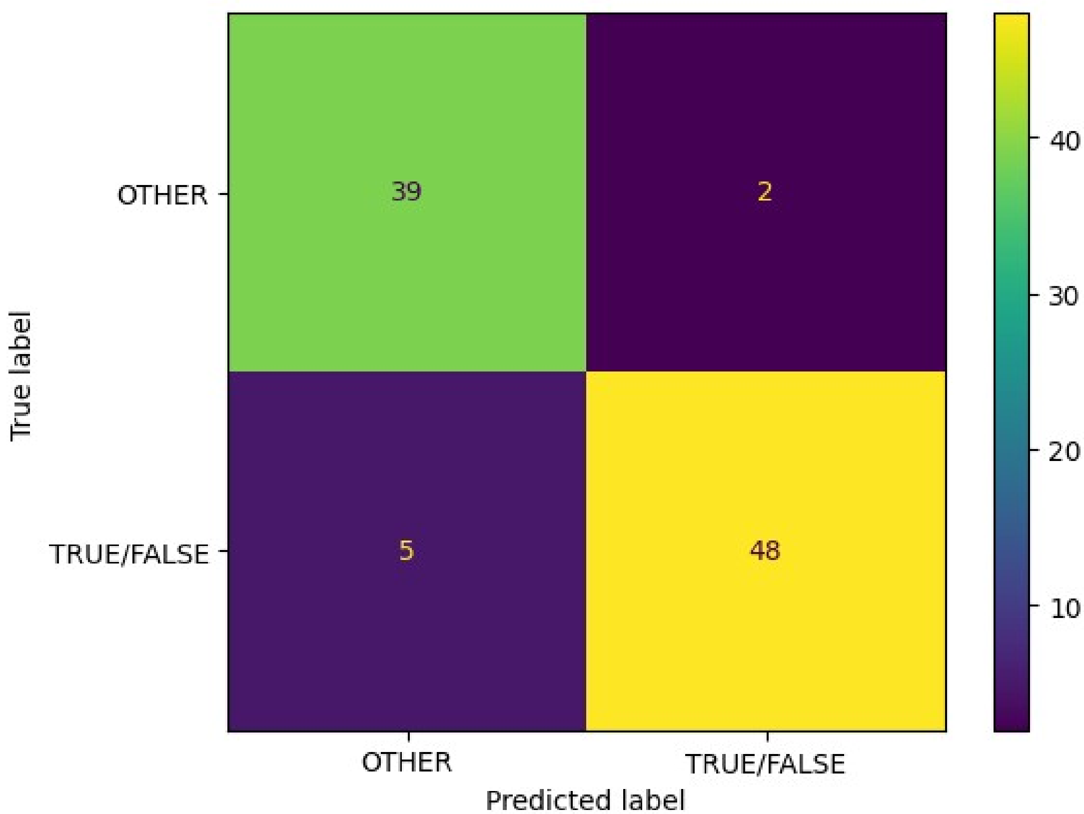

OTHER
TRUE/FALSE
True label
39
2
11
42
OTHER
TRUE/FALSE
Predicted label
40
35
30
25
20
15
10
5

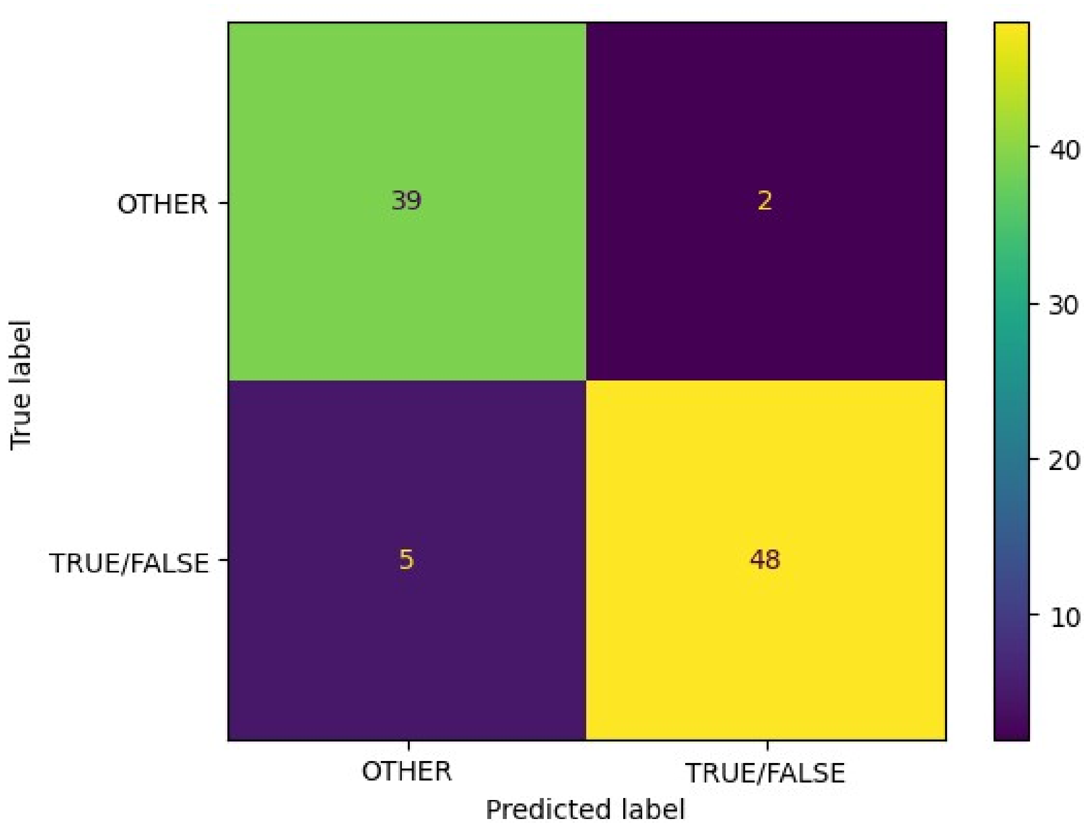

OTHER
TRUE/FALSE
True label
Predicted label
39
2
13
40
40
35
30
25
20
15
10
5

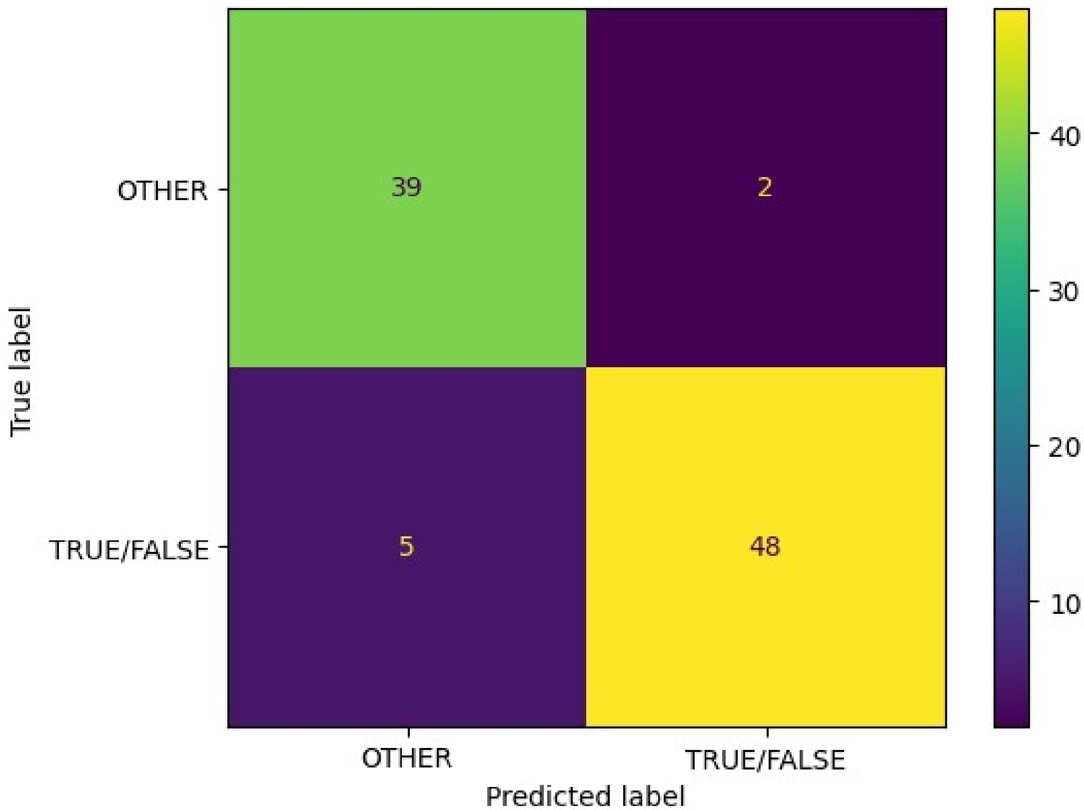

OTHER
TRUE/FALSE
True label
Predicted label
39
2
12
41
40
35
30
25
20
15
10
5


OTHER
TRUE/FALSE
True label
39
2
10
43
OTHER
TRUE/FALSE
Predicted label
40
35
30
25
20
15
10
5


OTHER
TRUE/FALSE
True label
39
2
4
49
OTHER
TRUE/FALSE
Predicted label
40
30
20
10


OTHER
TRUE/FALSE
True label
39
2
5
48
Predicted label
40
30
20
10


OTHER
TRUE/FALSE
True label
39
2
5
48
Predicted label
40
30
20
10




###**Etape 3 : Classification selon la colonne TITRE**

X_train_title prend la valeur de la colonne title dans le jeu d'entrainement et X_test_title prend la valeur de la colonne title dans le jeu de test

```
X_train_title = X_train['title']
X_train_title.reset_index(drop = True, inplace = True)
X_test_title = X_test['title']
X_test_title.reset_index(drop = True, inplace = True)
```

Ici, c'est une étape importante, on va tester différents classifieurs, pour chacun des classifieurs, on va appliquer le prétraitement + Vectorisation TfIdf, et on applique une cross_val_score avec un Kfold de 10 fois, par la suite on stocke dans une liste all_results la moyenne des accuracy + l'écart type et on la trie par ordre décroissant de moyenne d'accuracy et d'écart type. on remarque que les 2 meilleurs sont SVM et RF qu'on va séléctionner pour leur appliquer le GridSearch sur les paramètres des prétraitements + leurs hyperparamètres pour pouvoir choisir le meilleur.

```
score = 'accuracy'
seed = 7
allresults = []
results = []
names = []
```

```python
# Liste des modèles à tester
models = [
    ('MultinomialNB', MultinomialNB()),
    ('LogisticRegression', LogisticRegression(random_state=42)),
    ('KNN', KNeighborsClassifier()),
    ('CART', DecisionTreeClassifier(random_state=42)),
    ('RF', RandomForestClassifier(random_state=42)),
    ('SVM', SVC(random_state=42))
]


# Création d'un pipeline pour chaque modèle
pipelines = []
for name,model in models:
    pipeline = Pipeline([
        ('normalize', TextNormalizer()),
        ('tfidf', TfidfVectorizer()),
        (name,model)
    ])
    pipelines.append((name,pipeline))

all_results=[]
scores=[]
names=[]
for p in pipelines:
    print(p[1])
    # cross validation en 10 fois
    kfold = KFold(n_splits=10,random_state=seed,shuffle=True)
    # application de la classification
    cv_results = cross_val_score(p[1],X_train_title,y_train, cv=kfold,
scoring=score)
    scores.append(cv_results)
    names.append(p[0])
    all_results.append((p[0],cv_results.mean(),cv_results.std()))



all_results = sorted(all_results, key=lambda x: (-x[1], -x[2]))
print("all resultats", all_results)
```

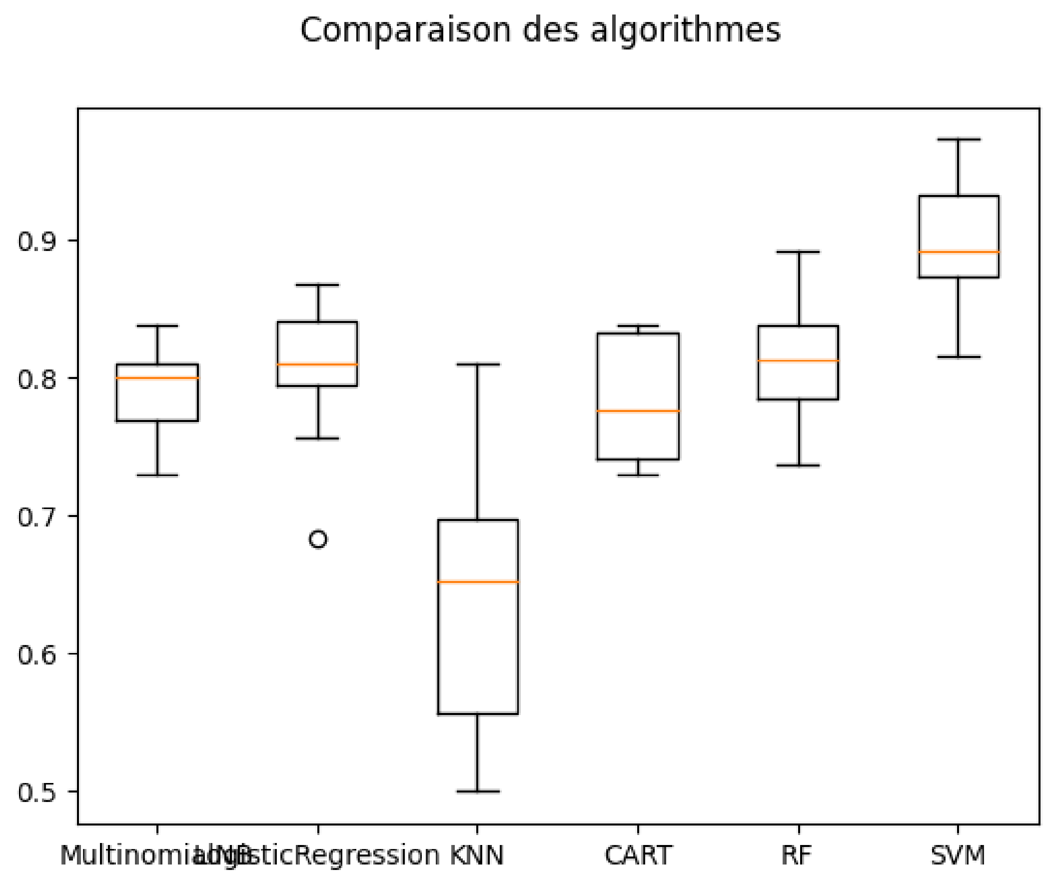
```
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('MultinomialNB', MultinomialNB())])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('LogisticRegression',
LogisticRegression(random_state=42))])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
```

```
TfidfVectorizer()),
                ('KNN', KNeighborsClassifier())])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('CART', DecisionTreeClassifier(random_state=42))])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('RF', RandomForestClassifier(random_state=42))])
Pipeline(steps=[('normalize', TextNormalizer()), ('tfidf',
TfidfVectorizer()),
                ('SVM', SVC(random_state=42))])
all resultats [('SVM', 0.8958748221906117, 0.04656971470072443),
('RF', 0.8105263157894738, 0.042092045520615184),
('LogisticRegression', 0.8076102418207682, 0.052204303468971605),
('MultinomialNB', 0.7914651493598861, 0.031128812901570786), ('CART',
0.7834992887624467, 0.04361713223988946), ('KNN', 0.6392603129445236,
0.09048317080558796)]
```

On affiche les boites à moustache pour mieux visualiser les résultats

```
fig = plt.figure()
fig.suptitle('Comparaison des algorithmes')
ax = fig.add_subplot(111)
plt.boxplot(scores)
ax.set_xticklabels(names)
```

```
[Text(1, 0, 'MultinomialNB'),
 Text(2, 0, 'LogisticRegression'),
 Text(3, 0, 'KNN'),
 Text(4, 0, 'CART'),
 Text(5, 0, 'RF'),
 Text(6, 0, 'SVM')]
```




*Choisir les meilleurs paramètres pour SVM et RF :*

```
# le plus simple est de faire un test sur differents pipelines.


# pipeline de l'utilisation de TfidfVectorizer avec differents pre-
traitements
TFIDF_brut = Pipeline ([('cleaner', TextNormalizer()),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])

TFIDF_lowcase = Pipeline([('cleaner',
TextNormalizer(removestopwords=False,lowercase=True,

getstemmer=False,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])
TFIDF_lowStop = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=False,removedigit=False)),
```

```
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])

TFIDF_lowStopstem = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=True,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])


# Liste de tous les modeles à tester
all_models = [
    ("TFIDF_lowcase", TFIDF_lowcase),
    ("TFIDF_lowStop", TFIDF_lowStop),
    ("TFIDF_lowStopstem",TFIDF_lowStopstem),
    ("TFIDF_brut", TFIDF_brut)
]

X_train_title_SVC = []
X_test_title_SVC = []

X_train_title_RandomForestClassifier = []
X_test_title_RandomForestClassifier = []


for name, pipeline in all_models :

X_train_title_SVC.append(pipeline.fit_transform(X_train_title).toarray
())

X_test_title_SVC.append(pipeline.transform(X_test_title).toarray())

X_train_title_RandomForestClassifier.append(pipeline.fit_transform(X_t
rain_title).toarray())

X_test_title_RandomForestClassifier.append(pipeline.transform(X_test_t
itle).toarray())


models = {
    'SVC': SVC(random_state=42),
    'RandomForestClassifier': RandomForestClassifier(random_state=42)
}


params = {'SVC': [{'C': [0.001, 0.01, 0.1, 1,2,5,7,10]},
             {'gamma': [0.001, 0.01, 0.1,0.2,0.3,0.5,0.7,1]},
             {'kernel': ['linear', 'rbf']}],
```

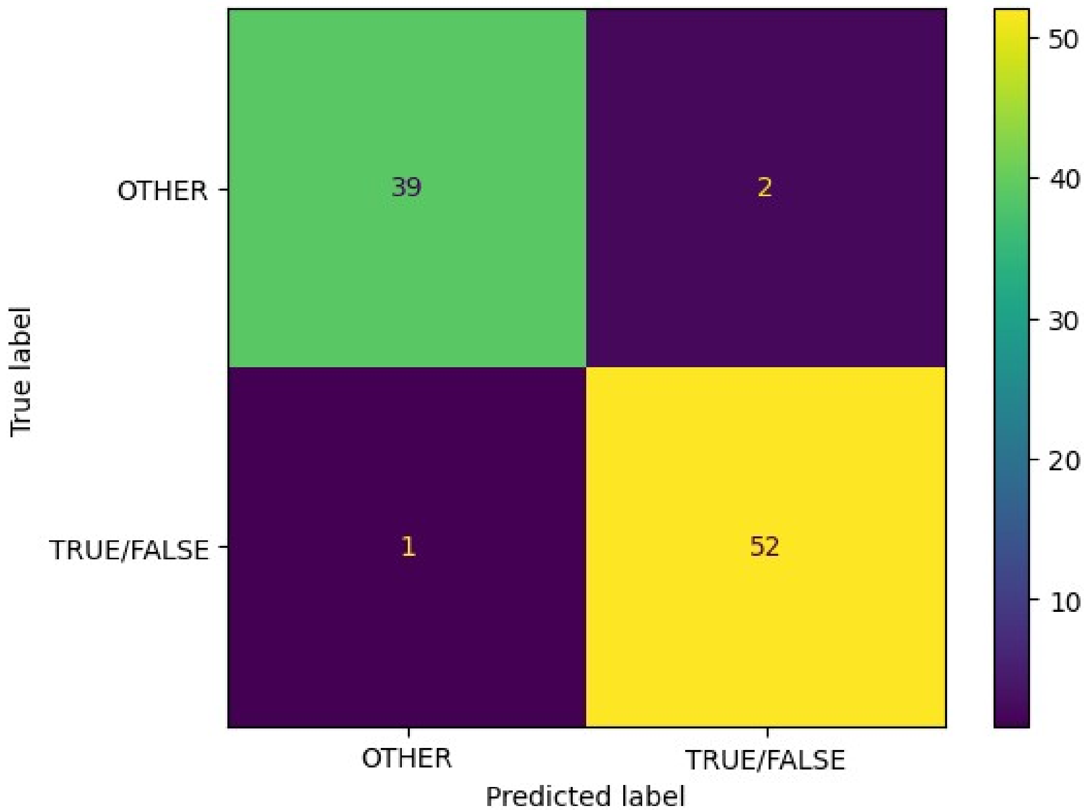
```
    'RandomForestClassifier': [{'n_estimators': [10, 50, 100, 200,
300]},
                                {'max_features': ['auto', 'sqrt',
'log2']}]
}

for model_name, model in models.items():
    score='accuracy'
    X_train_title = eval('X_train_title_' + model_name)
    X_test_title = eval('X_test_title_' + model_name)
    for i in range (len(X_train_title)):
      grid_search = GridSearchCV(model, params[model_name], n_jobs=-1,
verbose=1,scoring=score)
      print("grid search fait")
      grid_search.fit(X_train_title[i],y_train)
      print ('meilleur score %0.3f'%(grid_search.best_score_),'\n')
      print ('meilleur estimateur',grid_search.best_estimator_,'\n')
      y_pred = grid_search.predict(X_test_title[i])
      MyshowAllScores(y_test,y_pred)

      print("Ensemble des meilleurs paramètres :")
      best_parameters = grid_search.best_estimator_.get_params()
      for param_dict in params[model_name]:
        for param_name, param_value in param_dict.items():
            print("\t%s: %r" % (param_name,
best_parameters[param_name]))
```

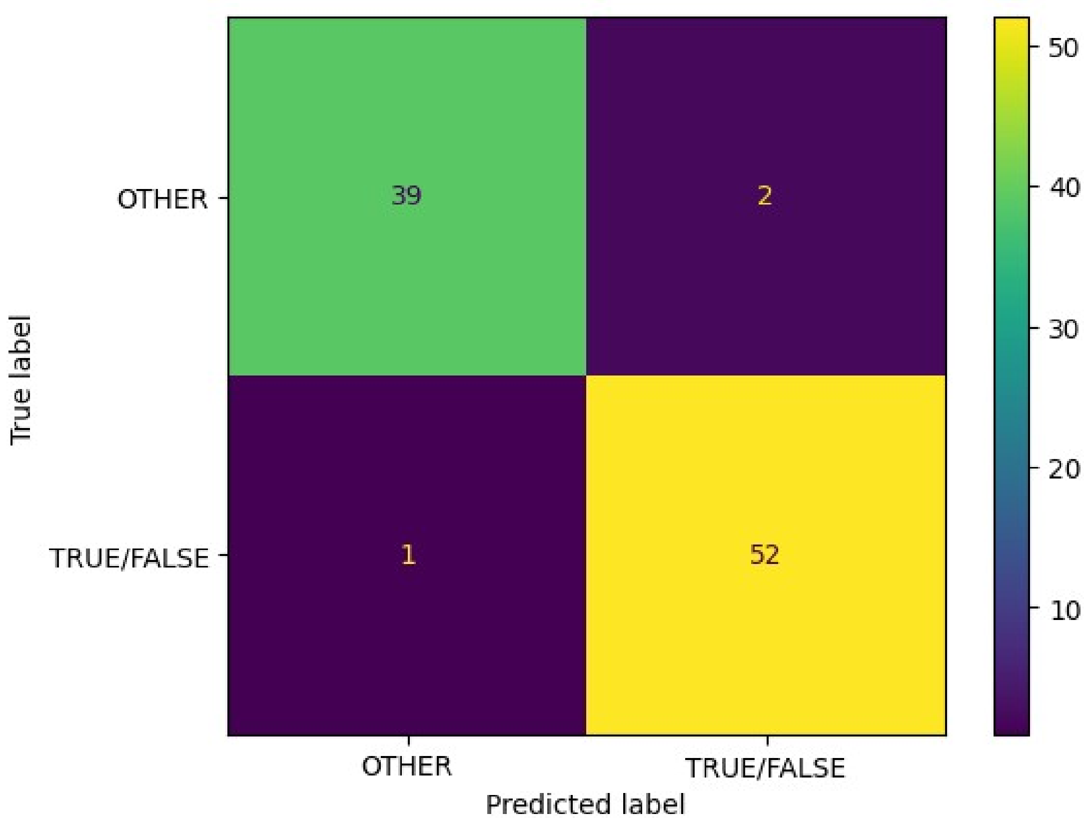
```
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.880

meilleur estimateur SVC(C=1, random_state=42)

Accuracy : 0.968
Classification Report
              precision    recall  f1-score   support

       OTHER    0.97500   0.95122   0.96296        41
  TRUE/FALSE    0.96296   0.98113   0.97196        53

    accuracy                        0.96809        94
   macro avg    0.96898   0.96618   0.96746        94
weighted avg    0.96821   0.96809   0.96804        94

Ensemble des meilleurs paramètres :
	C: 1
	gamma: 'scale'
	kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
```

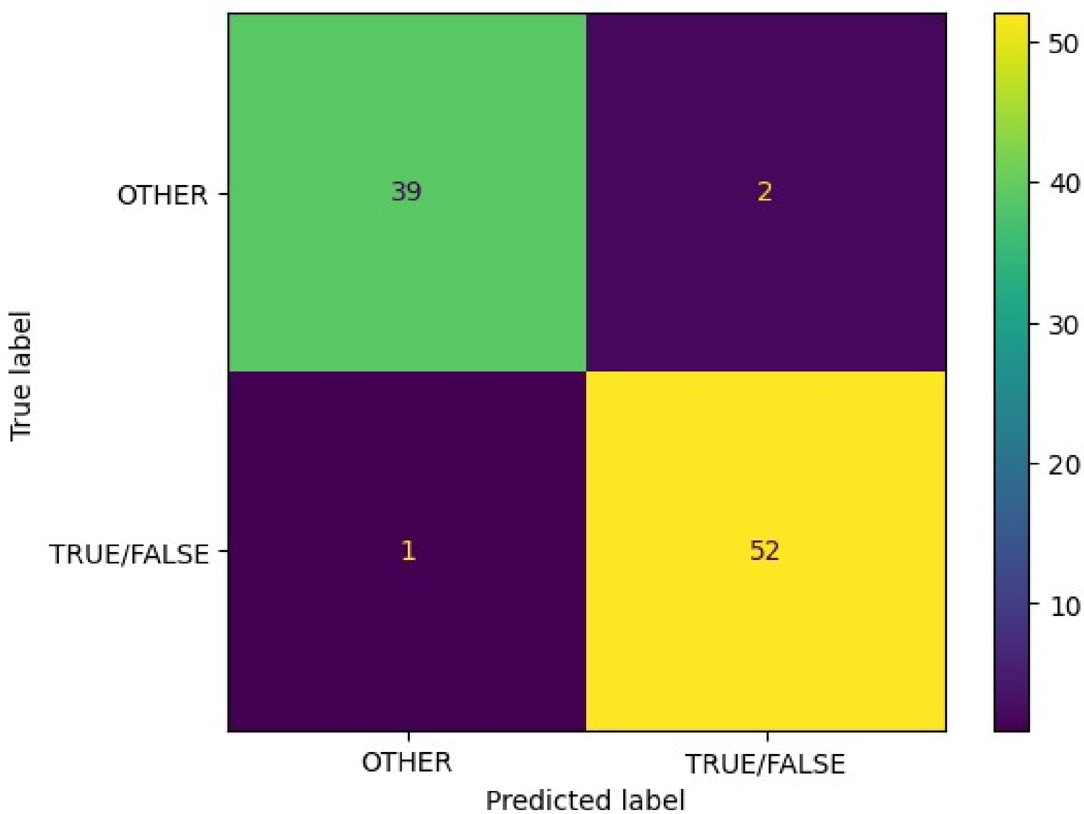
```
meilleur score 0.888

meilleur estimateur SVC(C=1, random_state=42)

Accuracy : 0.968
Classification Report
              precision    recall  f1-score   support

       OTHER    0.97500   0.95122   0.96296        41
  TRUE/FALSE    0.96296   0.98113   0.97196        53

    accuracy                        0.96809        94
   macro avg    0.96898   0.96618   0.96746        94
weighted avg    0.96821   0.96809   0.96804        94

Ensemble des meilleurs paramètres :
     C: 1
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.880

meilleur estimateur SVC(C=1, random_state=42)

Accuracy : 0.968
Classification Report
              precision    recall  f1-score   support

       OTHER    0.97500   0.95122   0.96296        41
  TRUE/FALSE    0.96296   0.98113   0.97196        53

    accuracy                        0.96809        94
   macro avg    0.96898   0.96618   0.96746        94
weighted avg    0.96821   0.96809   0.96804        94

Ensemble des meilleurs paramètres :
     C: 1
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.888

meilleur estimateur SVC(C=5, random_state=42)

Accuracy : 0.968
Classification Report
              precision    recall  f1-score   support
```

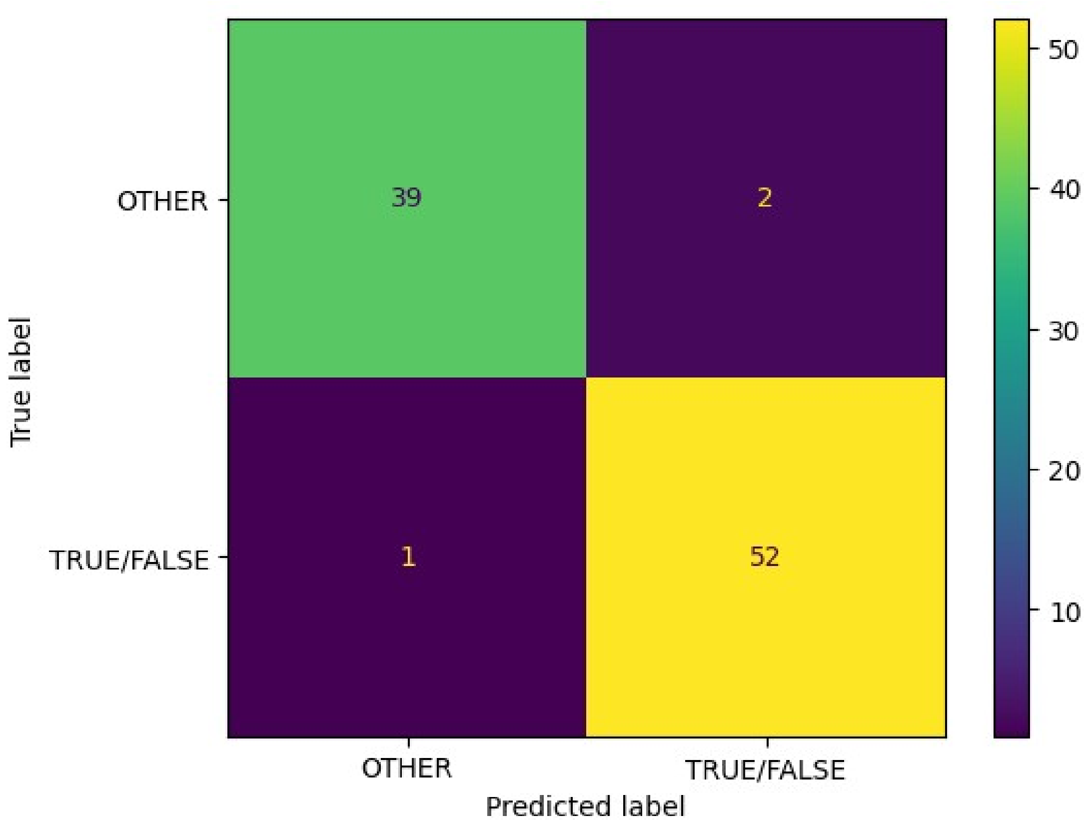
```
       OTHER    0.97500   0.95122   0.96296        41
  TRUE/FALSE    0.96296   0.98113   0.97196        53

    accuracy                        0.96809        94
   macro avg    0.96898   0.96618   0.96746        94
weighted avg    0.96821   0.96809   0.96804        94

Ensemble des meilleurs paramètres :
     C: 5
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.805

meilleur estimateur RandomForestClassifier(n_estimators=300,
random_state=42)

Accuracy : 0.872
Classification Report
              precision    recall  f1-score   support

       OTHER    0.77358   1.00000   0.87234        41
  TRUE/FALSE    1.00000   0.77358   0.87234        53

    accuracy                        0.87234        94
   macro avg    0.88679   0.88679   0.87234        94
weighted avg    0.90124   0.87234   0.87234        94

Ensemble des meilleurs paramètres :
     n_estimators: 300
     max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.816

meilleur estimateur RandomForestClassifier(n_estimators=200,
random_state=42)

Accuracy : 0.926
Classification Report
              precision    recall  f1-score   support

       OTHER    0.88636   0.95122   0.91765        41
  TRUE/FALSE    0.96000   0.90566   0.93204        53

    accuracy                        0.92553        94
   macro avg    0.92318   0.92844   0.92484        94
```

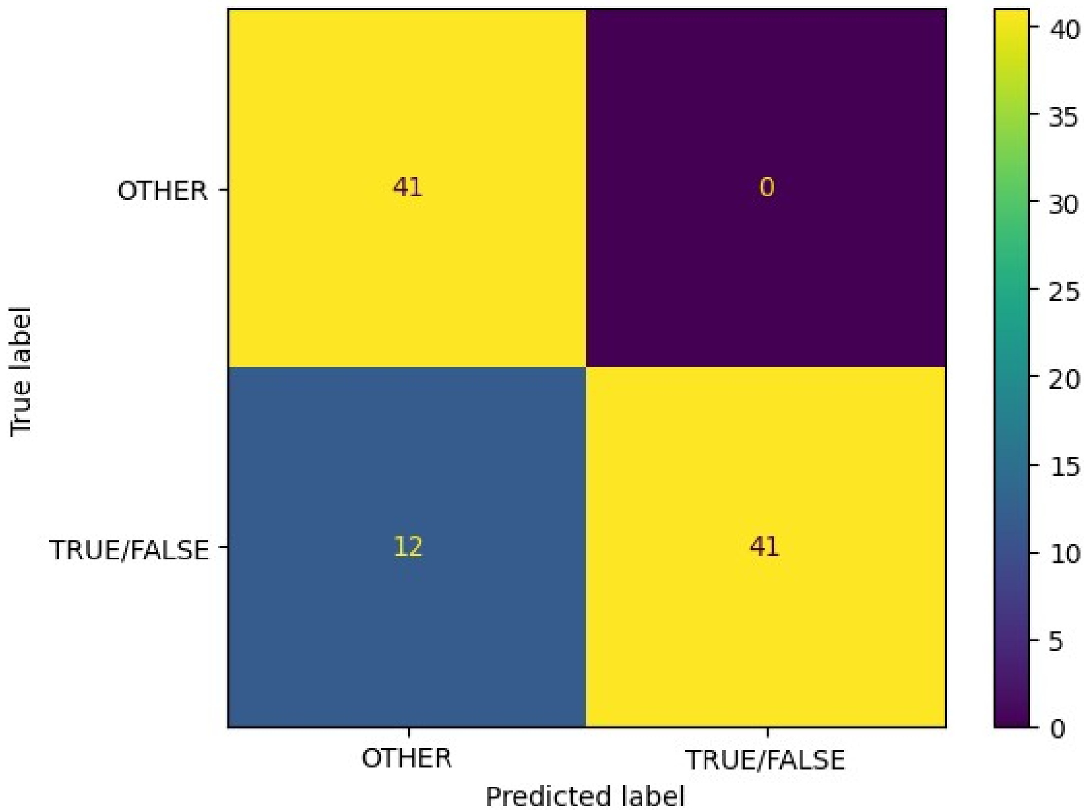
```
weighted avg    0.92788   0.92553   0.92576        94

Ensemble des meilleurs paramètres :
	n_estimators: 200
	max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.813

meilleur estimateur RandomForestClassifier(n_estimators=50,
random_state=42)

Accuracy : 0.830
Classification Report
              precision    recall  f1-score   support

       OTHER    0.73585   0.95122   0.82979        41
  TRUE/FALSE    0.95122   0.73585   0.82979        53

    accuracy                        0.82979        94
   macro avg    0.84353   0.84353   0.82979        94
weighted avg    0.85728   0.82979   0.82979        94

Ensemble des meilleurs paramètres :
	n_estimators: 50
	max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.840

meilleur estimateur RandomForestClassifier(random_state=42)

Accuracy : 0.862
Classification Report
              precision    recall  f1-score   support

       OTHER    0.78000   0.95122   0.85714        41
  TRUE/FALSE    0.95455   0.79245   0.86598        53

    accuracy                        0.86170        94
   macro avg    0.86727   0.87184   0.86156        94
weighted avg    0.87841   0.86170   0.86213        94

Ensemble des meilleurs paramètres :
	n_estimators: 100
	max_features: 'sqrt'
```

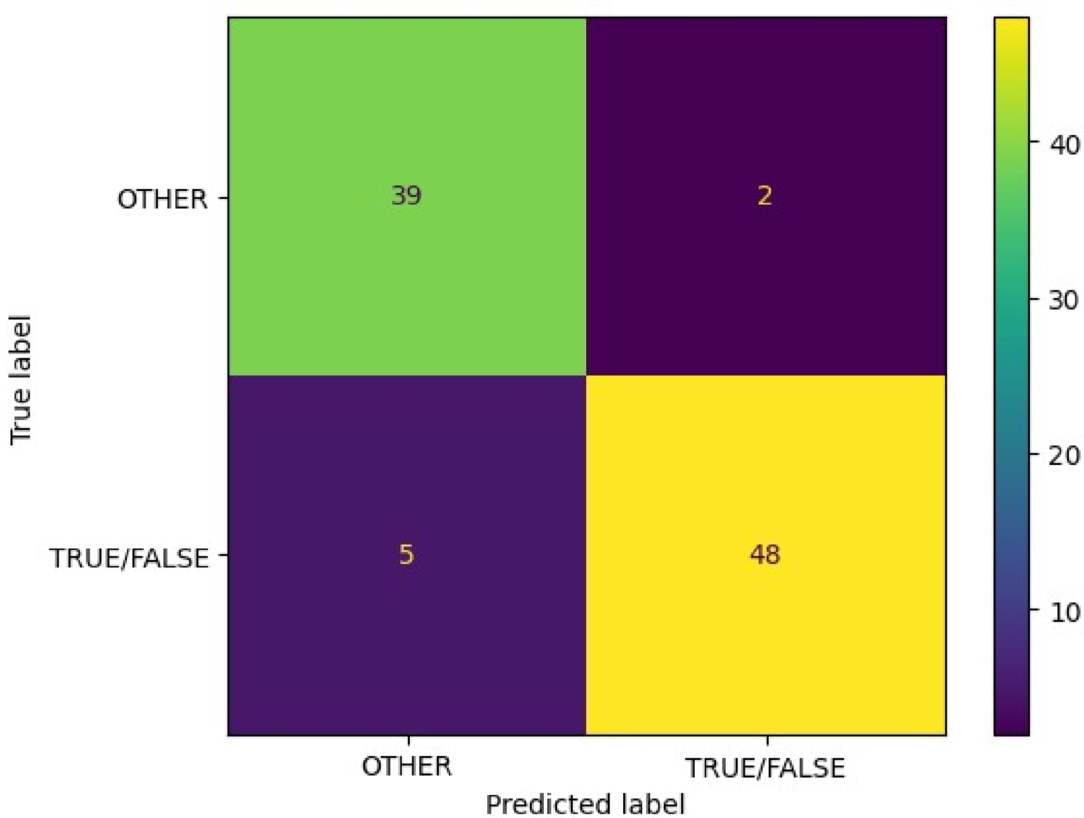

OTHER
TRUE/FALSE
True label
39
2
1
52
OTHER
TRUE/FALSE
Predicted label
50
40
30
20
10

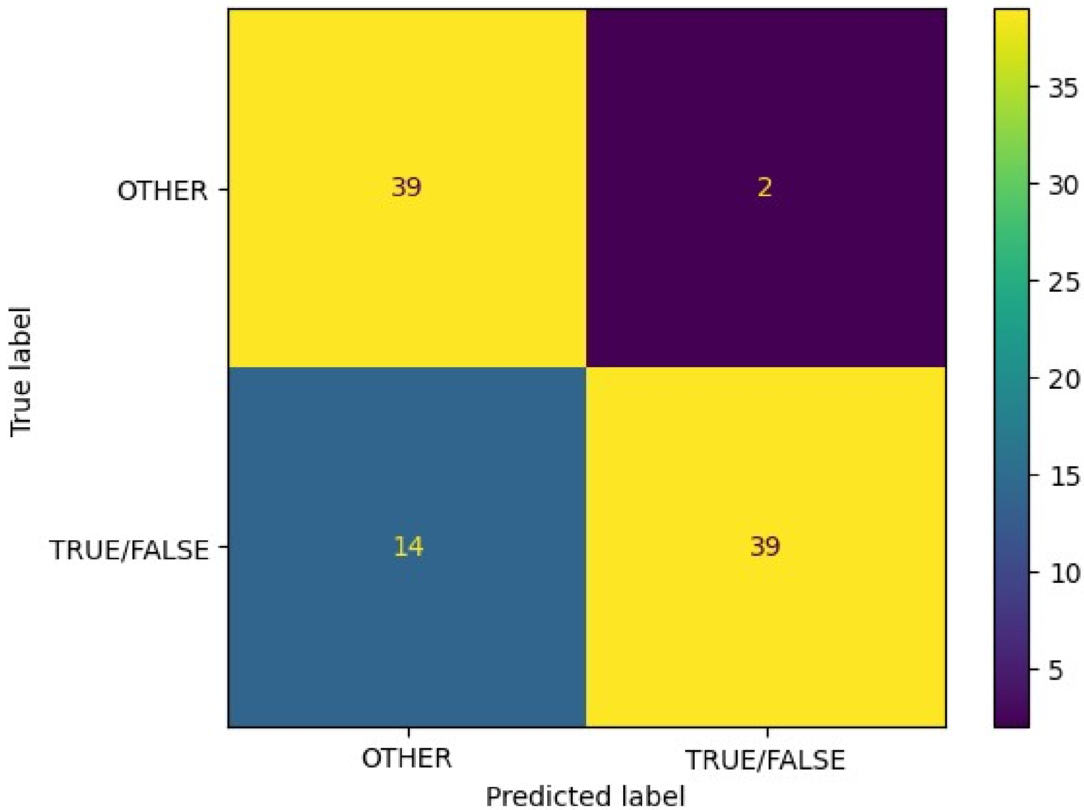

OTHER
TRUE/FALSE
True label
39
2
1
52
OTHER
TRUE/FALSE
Predicted label
50
40
30
20
10


OTHER
TRUE/FALSE
True label
39
2
1
52
OTHER
TRUE/FALSE
Predicted label
50
40
30
20
10

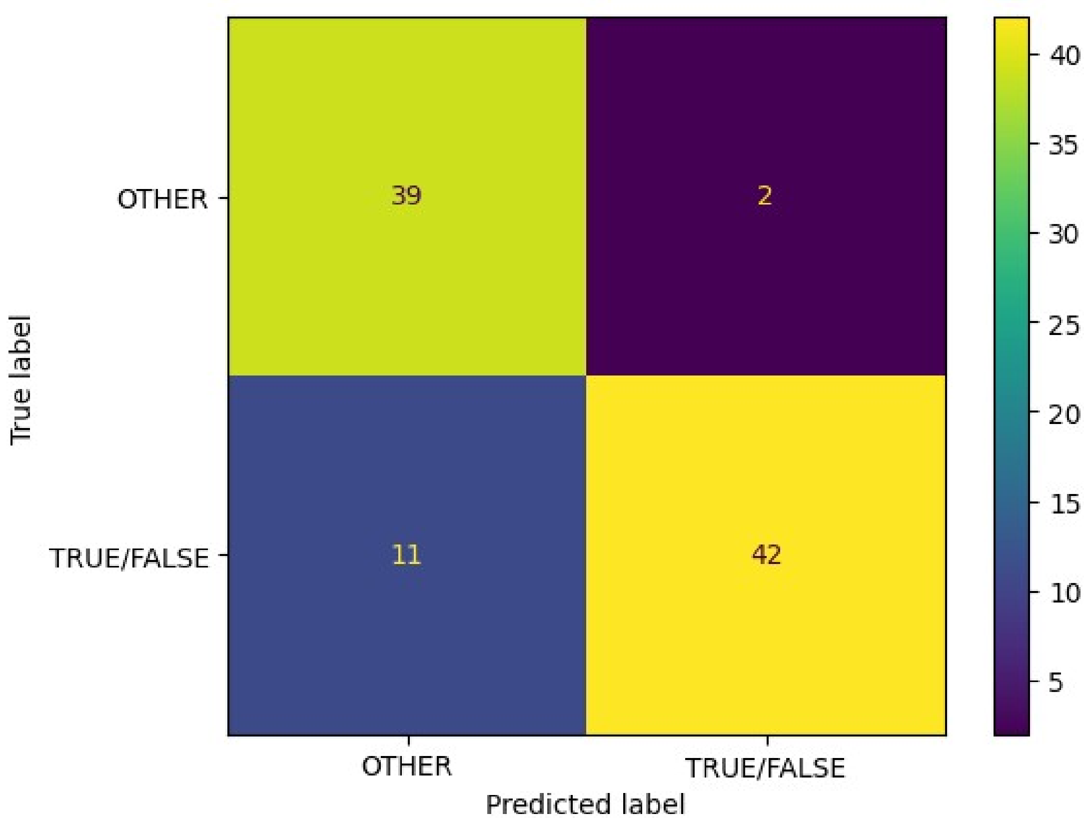

OTHER
TRUE/FALSE
True label
39
2
1
52
OTHER
TRUE/FALSE
Predicted label
50
40
30
20
10


41
0
12
41
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
0
5
10
15
20
25
30
35
40


39
2
5
48
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
10
20
30
40



39
2
14
39
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
35
30
25
20
15
10
5

39
2
11
42
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
40
35
30
25
20
15
10
5

### Etape 4 : Classification selon la colonne Text_title (concaténation de la colonne text et la colonne title) :

On concatène les deux colonnes text et titre de note DataFrame dftrain

```
train_text_title = X_train.apply(lambda x : '{}
{}'.format(x['text'],x['title']),axis=1)
test_text_title = X_test.apply(lambda x : '{}
{}'.format(x['text'],x['title']),axis=1)

X_train['text_title'] = train_text_title
X_train_text_title = X_train['text_title']
X_train_text_title.reset_index(drop = True, inplace = True)

X_test['text_title'] = test_text_title
X_test_text_title = X_test['text_title']
X_test_text_title.reset_index(drop = True, inplace = True)

print("le texte et titre du train sont")
display(X_train_text_title)

print("le texte et titre du test sont")
display(X_test_text_title)
```

Ici, c'est une étape importante, on va tester différents classifieurs, pour chacun des classifieurs, on va appliquer le prétraitement + Vectorisation TfIdf, et on applique une cross_val_score avec un Kfold de 10 fois, par la suite on stocke dans une liste all_results la moyenne des accuracy + l'écart type et on la trie par ordre décroissant de moyenne d'accuracy et d'écart type. on remarque que les 2 meilleurs sont SVM et RF qu'on va séléctionner pour leur appliquer le GridSearch sur les paramètres des prétraitements + leurs hyperparamètres pour pouvoir choisir le meilleur

```
np.random.seed(42)  # Set the random seed for NumPy

score = 'accuracy'
seed = 7
allresults = []
results = []
names = []


# Liste des modèles à tester
models = [
    ('MultinomialNB', MultinomialNB()),
    ('LogisticRegression', LogisticRegression(random_state=42)),
    ('KNN', KNeighborsClassifier()),
    ('CART', DecisionTreeClassifier(random_state=42)),
    ('RF', RandomForestClassifier(random_state=42)),
    ('SVM', SVC(random_state=42))
```

```
]


# Création d'un pipeline pour chaque modèle
pipelines = []
for name,model in models:
    pipeline = Pipeline([
        ('normalize', TextNormalizer()),
        ('tfidf', TfidfVectorizer()),
        (name,model)
    ])
    pipelines.append((name,pipeline))

all_results=[]
scores=[]
names=[]
for p in pipelines:
    print(p[1])
    # cross validation en 10 fois
    kfold = KFold(n_splits=10,random_state=seed,shuffle=True)
    start_time = time.time()
    # application de la classification
    cv_results = cross_val_score(p[1],X_train_text_title,y_train,
cv=kfold, scoring=score)
    scores.append(cv_results)
    names.append(p[0])
    all_results.append((p[0],cv_results.mean(),cv_results.std()))
    end_time = time.time()


all_results = sorted(all_results, key=lambda x: (-x[1], -x[2]))
print("all resultats", all_results)
```

On affiche les boites à moustache pour mieux visualiser les résultats

```
fig = plt.figure()
fig.suptitle('Comparaison des algorithmes')
ax = fig.add_subplot(111)
plt.boxplot(scores)
ax.set_xticklabels(names)
```

*Choisir les meilleurs paramètres pour SVM et RF :*

```
np.random.seed(42)  # Set the random seed for NumPy

# le plus simple est de faire un test sur differents pipelines.
# pipeline de l'utilisation de CountVectorizer sur le texte avec
differents pre-traitements
CV_brut = Pipeline([('cleaner', TextNormalizer()),
                    ('count_vectorizer',
CountVectorizer(lowercase=False))])
```

```
CV_lowcase = Pipeline([('cleaner',
TextNormalizer(removestopwords=False,lowercase=True,

getstemmer=False,removedigit=False)),
                    ('count_vectorizer',
CountVectorizer(lowercase=False))])
CV_lowStop = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=False,removedigit=False)),
                    ('count_vectorizer',
CountVectorizer(lowercase=False))])

CV_lowStopstem = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=True,removedigit=False)),
                    ('count_vectorizer',
CountVectorizer(lowercase=False))])

# pipeline de l'utilisation de TfidfVectorizer avec differents pre-
traitements
TFIDF_brut = Pipeline ([('cleaner', TextNormalizer()),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])

TFIDF_lowcase = Pipeline([('cleaner',
TextNormalizer(removestopwords=False,lowercase=True,

getstemmer=False,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])
TFIDF_lowStop = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=False,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])

TFIDF_lowStopstem = Pipeline([('cleaner',
TextNormalizer(removestopwords=True,lowercase=True,

getstemmer=True,removedigit=False)),
                    ('tfidf_vectorizer',
TfidfVectorizer(lowercase=False))])


# Liste de tous les modeles à tester
all_models = [
```

```
    ("CV_brut", CV_brut),
    ("CV_lowcase", CV_lowcase),
    ("CV_lowStop", CV_lowStop),
    ("CV_lowStopstem",CV_lowStopstem),
    ("TFIDF_lowcase", TFIDF_lowcase),
    ("TFIDF_lowStop", TFIDF_lowStop),
    ("TFIDF_lowStopstem",TFIDF_lowStopstem),
    ("TFIDF_brut", TFIDF_brut)
]

X_train_text_title_SVC = []
X_test_text_title_SVC = []

X_train_text_title_RandomForestClassifier = []
X_test_text_title_RandomForestClassifier = []


for name, pipeline in all_models :

X_train_text_title_SVC.append(pipeline.fit_transform(X_train_text_titl
e).toarray())

X_test_text_title_SVC.append(pipeline.transform(X_test_text_title).toa
rray())

X_train_text_title_RandomForestClassifier.append(pipeline.fit_transfor
m(X_train_text_title).toarray())

X_test_text_title_RandomForestClassifier.append(pipeline.transform(X_t
est_text_title).toarray())


models = {
    'SVC': SVC(random_state=42),
    'RandomForestClassifier': RandomForestClassifier(random_state=42)
}


params = {'SVC': [{'C': [0.001, 0.01, 0.1, 1,2,5,7,10]},
             {'gamma': [0.001, 0.01, 0.1,0.2,0.3,0.5,0.7,1]},
             {'kernel': ['linear', 'rbf']}],
    'RandomForestClassifier': [{'n_estimators': [10, 50, 100, 200,
300]},
                              {'max_features': ['auto', 'sqrt',
'log2']}]
}

for model_name, model in models.items():
    score='accuracy'
```

```python
    X_train_text_title = eval('X_train_text_title_' + model_name)
    X_test_text_title = eval('X_test_text_title_' + model_name)
    for i in range (len(X_train_text_title)):
      grid_search = GridSearchCV(model, params[model_name], n_jobs=-1,
verbose=1,scoring=score)
      print("grid search fait")
      grid_search.fit(X_train_text_title[i],y_train)
      print ('meilleur score %0.3f'%(grid_search.best_score_),'\n')
      print ('meilleur estimateur',grid_search.best_estimator_,'\n')
      y_pred = grid_search.predict(X_test_text_title[i])
      MyshowAllScores(y_test,y_pred)

      print("Ensemble des meilleurs paramètres :")
      best_parameters = grid_search.best_estimator_.get_params()
      for param_dict in params[model_name]:
        for param_name, param_value in param_dict.items():
            print("\t%s: %r" % (param_name,
best_parameters[param_name]))
```

```
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.901

meilleur estimateur SVC(gamma=0.1, random_state=42)

Accuracy : 0.979
Classification Report
              precision    recall  f1-score   support

       OTHER    1.00000   0.95122   0.97500        41
  TRUE/FALSE    0.96364   1.00000   0.98148        53

    accuracy                        0.97872        94
   macro avg    0.98182   0.97561   0.97824        94
weighted avg    0.97950   0.97872   0.97865        94

Ensemble des meilleurs paramètres :
	C: 1.0
	gamma: 0.1
	kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.901

meilleur estimateur SVC(gamma=0.1, random_state=42)

Accuracy : 0.979
Classification Report
              precision    recall  f1-score   support
```

```
       OTHER    1.00000   0.95122   0.97500        41
  TRUE/FALSE    0.96364   1.00000   0.98148        53

    accuracy                        0.97872        94
   macro avg    0.98182   0.97561   0.97824        94
weighted avg    0.97950   0.97872   0.97865        94

Ensemble des meilleurs paramètres :
      C: 1.0
      gamma: 0.1
      kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.901

meilleur estimateur SVC(gamma=0.1, random_state=42)

Accuracy : 0.979
Classification Report
              precision    recall  f1-score   support

       OTHER    1.00000   0.95122   0.97500        41
  TRUE/FALSE    0.96364   1.00000   0.98148        53

    accuracy                        0.97872        94
   macro avg    0.98182   0.97561   0.97824        94
weighted avg    0.97950   0.97872   0.97865        94

Ensemble des meilleurs paramètres :
      C: 1.0
      gamma: 0.1
      kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.901

meilleur estimateur SVC(gamma=0.1, random_state=42)

Accuracy : 0.979
Classification Report
              precision    recall  f1-score   support

       OTHER    1.00000   0.95122   0.97500        41
  TRUE/FALSE    0.96364   1.00000   0.98148        53

    accuracy                        0.97872        94
   macro avg    0.98182   0.97561   0.97824        94
weighted avg    0.97950   0.97872   0.97865        94
```

```
Ensemble des meilleurs paramètres :
      C: 1.0
      gamma: 0.1
      kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.866

meilleur estimateur SVC(C=1, random_state=42)

Accuracy : 0.915
Classification Report
              precision    recall  f1-score   support

       OTHER    0.88372   0.92683   0.90476        41
  TRUE/FALSE    0.94118   0.90566   0.92308        53

    accuracy                        0.91489        94
   macro avg    0.91245   0.91624   0.91392        94
weighted avg    0.91612   0.91489   0.91509        94

Ensemble des meilleurs paramètres :
      C: 1
      gamma: 'scale'
      kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.869

meilleur estimateur SVC(C=1, random_state=42)

Accuracy : 0.926
Classification Report
              precision    recall  f1-score   support

       OTHER    0.88636   0.95122   0.91765        41
  TRUE/FALSE    0.96000   0.90566   0.93204        53

    accuracy                        0.92553        94
   macro avg    0.92318   0.92844   0.92484        94
weighted avg    0.92788   0.92553   0.92576        94

Ensemble des meilleurs paramètres :
      C: 1
      gamma: 'scale'
      kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.869
```

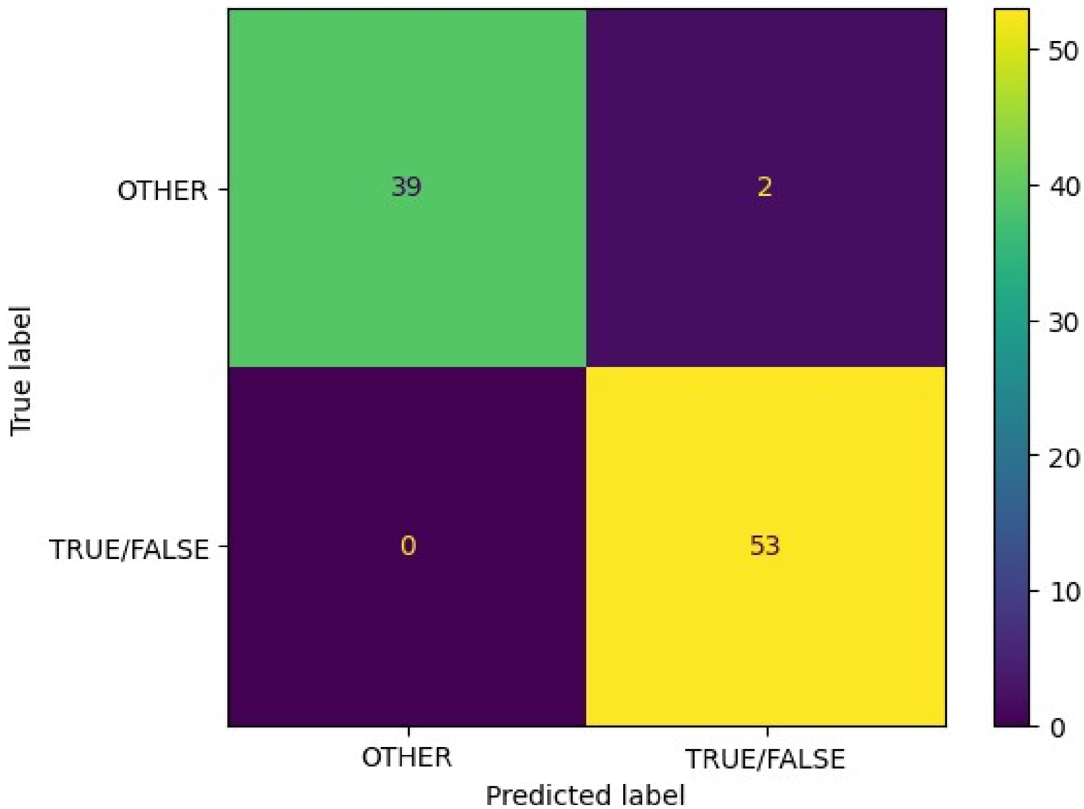
```
meilleur estimateur SVC(C=2, random_state=42)

Accuracy : 0.894
Classification Report
              precision    recall  f1-score   support

       OTHER    0.82979   0.95122   0.88636        41
  TRUE/FALSE    0.95745   0.84906   0.90000        53

    accuracy                        0.89362        94
   macro avg    0.89362   0.90014   0.89318        94
weighted avg    0.90177   0.89362   0.89405        94

Ensemble des meilleurs paramètres :
     C: 2
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 18 candidates, totalling 90 fits
meilleur score 0.874

meilleur estimateur SVC(C=1, random_state=42)

Accuracy : 0.904
Classification Report
              precision    recall  f1-score   support

       OTHER    0.86364   0.92683   0.89412        41
  TRUE/FALSE    0.94000   0.88679   0.91262        53

    accuracy                        0.90426        94
   macro avg    0.90182   0.90681   0.90337        94
weighted avg    0.90669   0.90426   0.90455        94

Ensemble des meilleurs paramètres :
     C: 1
     gamma: 'scale'
     kernel: 'rbf'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.853

meilleur estimateur RandomForestClassifier(n_estimators=300,
random_state=42)

Accuracy : 0.915
Classification Report
              precision    recall  f1-score   support
```

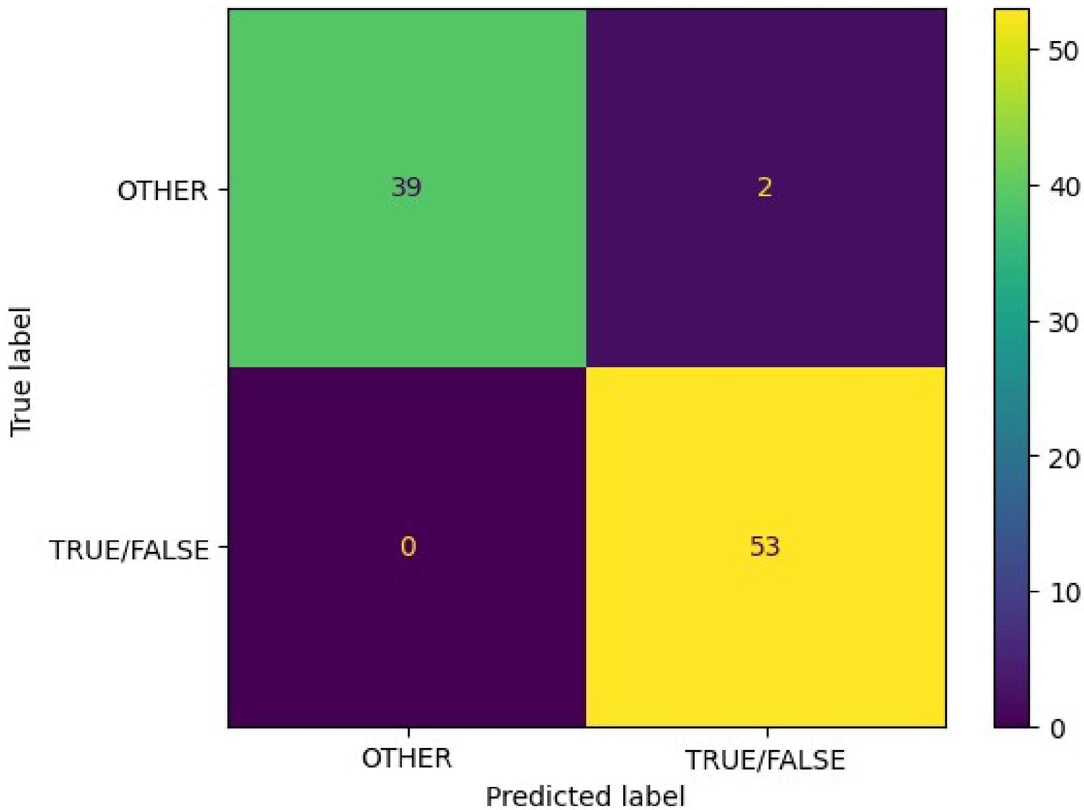
```
       OTHER    0.86667   0.95122   0.90698        41
  TRUE/FALSE    0.95918   0.88679   0.92157        53

    accuracy                        0.91489        94
   macro avg    0.91293   0.91901   0.91427        94
weighted avg    0.91883   0.91489   0.91520        94

Ensemble des meilleurs paramètres :
     n_estimators: 300
     max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.864

meilleur estimateur RandomForestClassifier(max_features='log2',
random_state=42)

Accuracy : 0.926
Classification Report
              precision    recall  f1-score   support

       OTHER    0.88636   0.95122   0.91765        41
  TRUE/FALSE    0.96000   0.90566   0.93204        53

    accuracy                        0.92553        94
   macro avg    0.92318   0.92844   0.92484        94
weighted avg    0.92788   0.92553   0.92576        94

Ensemble des meilleurs paramètres :
     n_estimators: 100
     max_features: 'log2'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.821

meilleur estimateur RandomForestClassifier(n_estimators=200,
random_state=42)

Accuracy : 0.904
Classification Report
              precision    recall  f1-score   support

       OTHER    0.84783   0.95122   0.89655        41
  TRUE/FALSE    0.95833   0.86792   0.91089        53

    accuracy                        0.90426        94
   macro avg    0.90308   0.90957   0.90372        94
weighted avg    0.91013   0.90426   0.90464        94
```

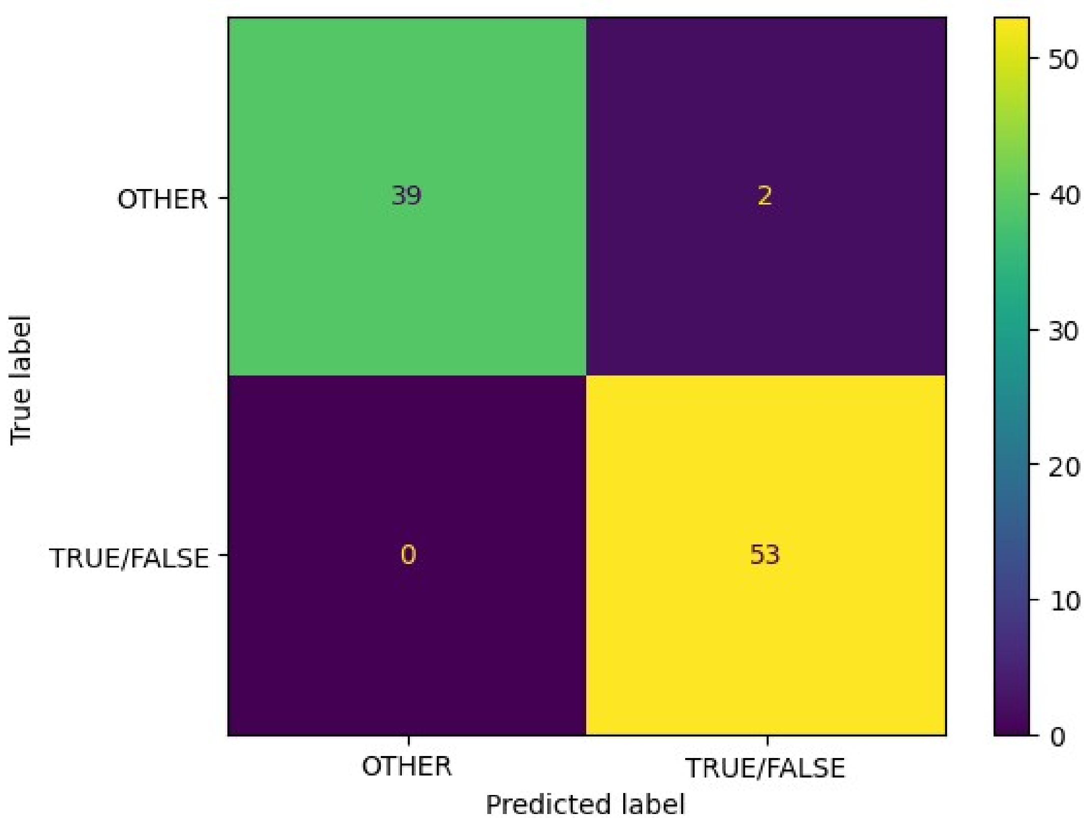
```
Ensemble des meilleurs paramètres :
      n_estimators: 200
      max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.842

meilleur estimateur RandomForestClassifier(n_estimators=300,
random_state=42)

Accuracy : 0.904
Classification Report
              precision    recall  f1-score   support

       OTHER    0.84783   0.95122   0.89655        41
  TRUE/FALSE    0.95833   0.86792   0.91089        53

    accuracy                        0.90426        94
   macro avg    0.90308   0.90957   0.90372        94
weighted avg    0.91013   0.90426   0.90464        94

Ensemble des meilleurs paramètres :
      n_estimators: 300
      max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.845

meilleur estimateur RandomForestClassifier(max_features='log2',
random_state=42)

Accuracy : 0.936
Classification Report
              precision    recall  f1-score   support

       OTHER    0.90698   0.95122   0.92857        41
  TRUE/FALSE    0.96078   0.92453   0.94231        53

    accuracy                        0.93617        94
   macro avg    0.93388   0.93787   0.93544        94
weighted avg    0.93732   0.93617   0.93632        94

Ensemble des meilleurs paramètres :
      n_estimators: 100
      max_features: 'log2'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.842
```

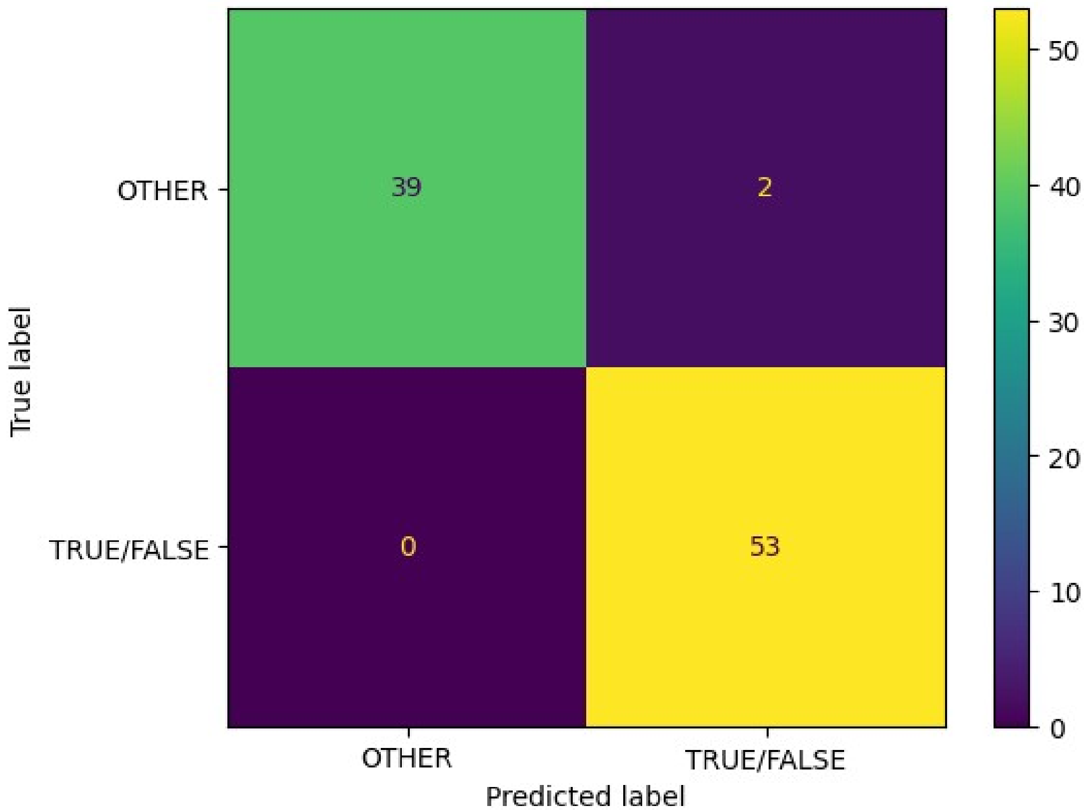
```
meilleur estimateur RandomForestClassifier(n_estimators=300,
random_state=42)

Accuracy : 0.915
Classification Report
              precision    recall  f1-score   support

       OTHER    0.86667   0.95122   0.90698        41
  TRUE/FALSE    0.95918   0.88679   0.92157        53

    accuracy                        0.91489        94
   macro avg    0.91293   0.91901   0.91427        94
weighted avg    0.91883   0.91489   0.91520        94

Ensemble des meilleurs paramètres :
     n_estimators: 300
     max_features: 'sqrt'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.842

meilleur estimateur RandomForestClassifier(max_features='log2',
random_state=42)

Accuracy : 0.872
Classification Report
              precision    recall  f1-score   support

       OTHER    0.79592   0.95122   0.86667        41
  TRUE/FALSE    0.95556   0.81132   0.87755        53

    accuracy                        0.87234        94
   macro avg    0.87574   0.88127   0.87211        94
weighted avg    0.88593   0.87234   0.87280        94

Ensemble des meilleurs paramètres :
     n_estimators: 100
     max_features: 'log2'
grid search fait
Fitting 5 folds for each of 8 candidates, totalling 40 fits
meilleur score 0.853

meilleur estimateur RandomForestClassifier(n_estimators=200,
random_state=42)

Accuracy : 0.904
Classification Report
              precision    recall  f1-score   support
```

```
       OTHER    0.84783   0.95122   0.89655        41
  TRUE/FALSE    0.95833   0.86792   0.91089        53

    accuracy                        0.90426        94
   macro avg    0.90308   0.90957   0.90372        94
weighted avg    0.91013   0.90426   0.90464        94

Ensemble des meilleurs paramètres :
      n_estimators: 200
      max_features: 'sqrt'
```

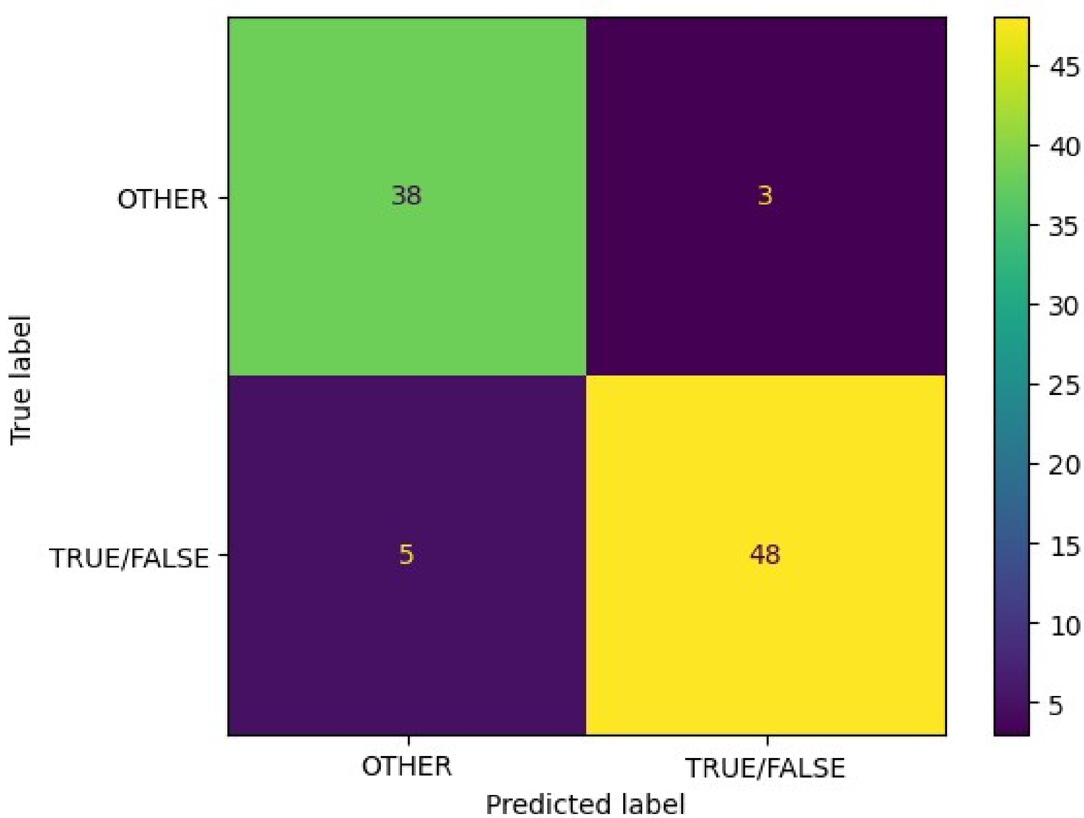



39
2
0
53
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
50
40
30
20
10
0

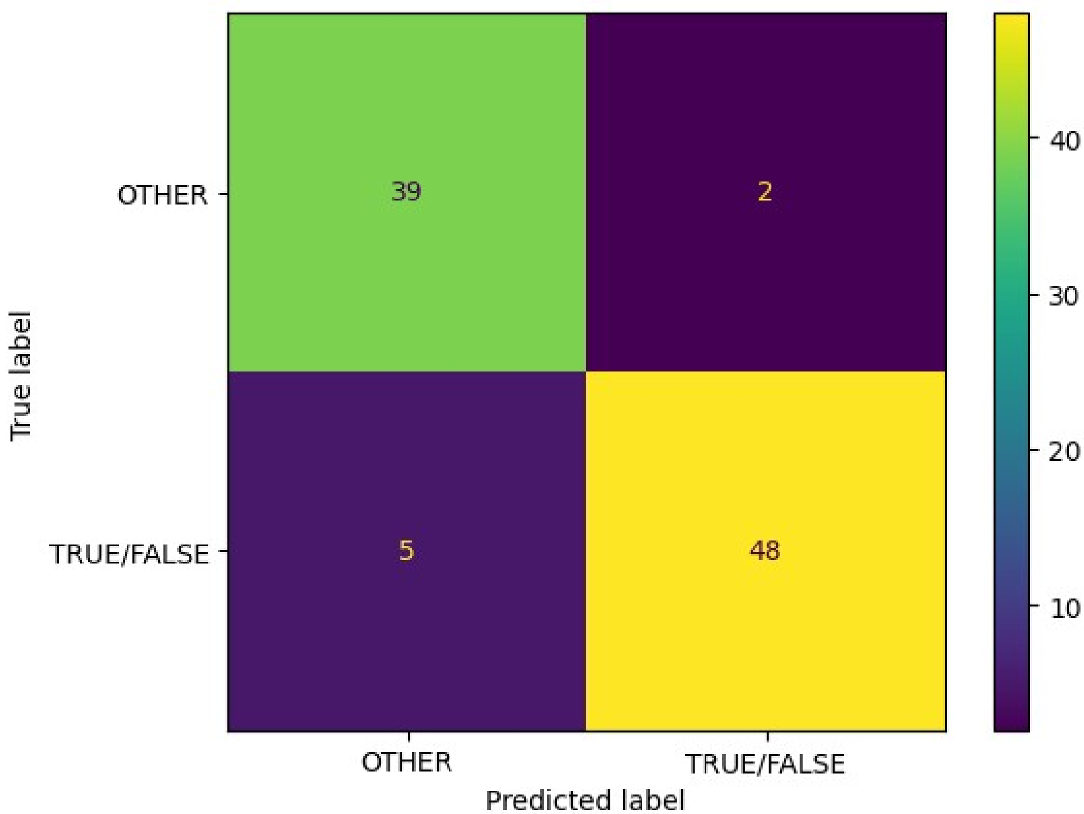

39
2
0
53
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
50
40
30
20
10
0



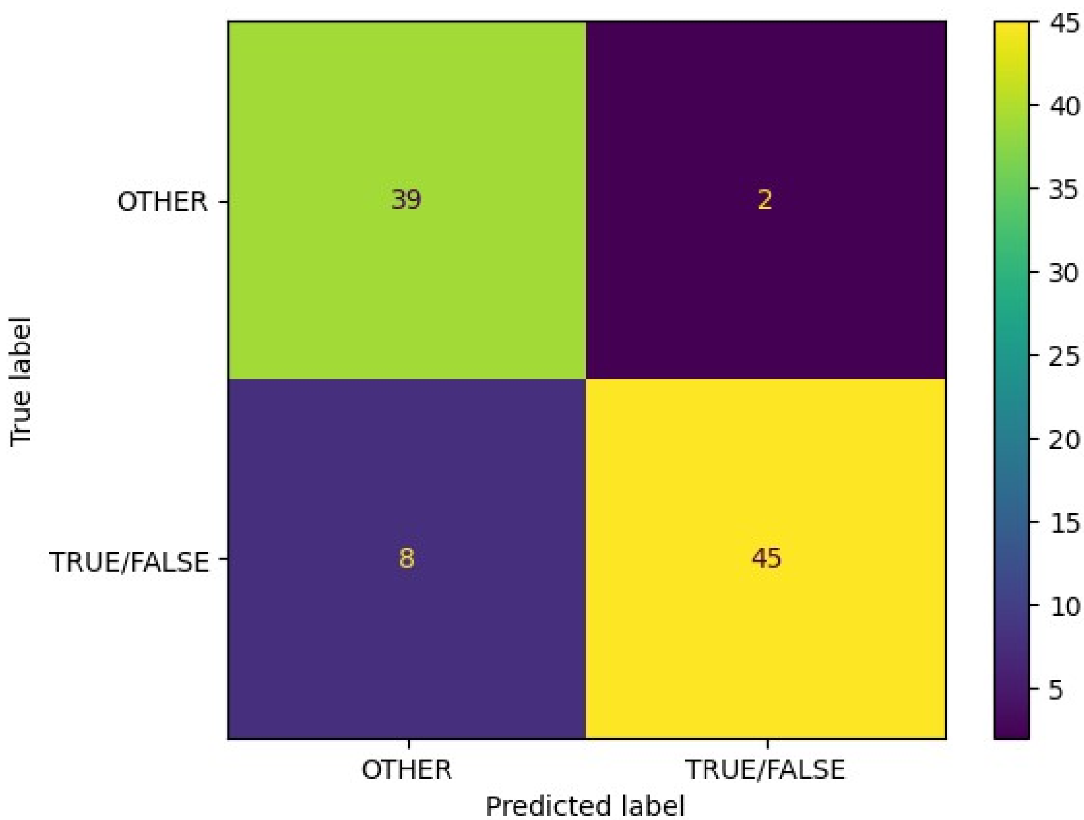
True label
OTHER
TRUE/FALSE
39
2
0
53
OTHER
TRUE/FALSE
Predicted label
50
40
30
20
10
0

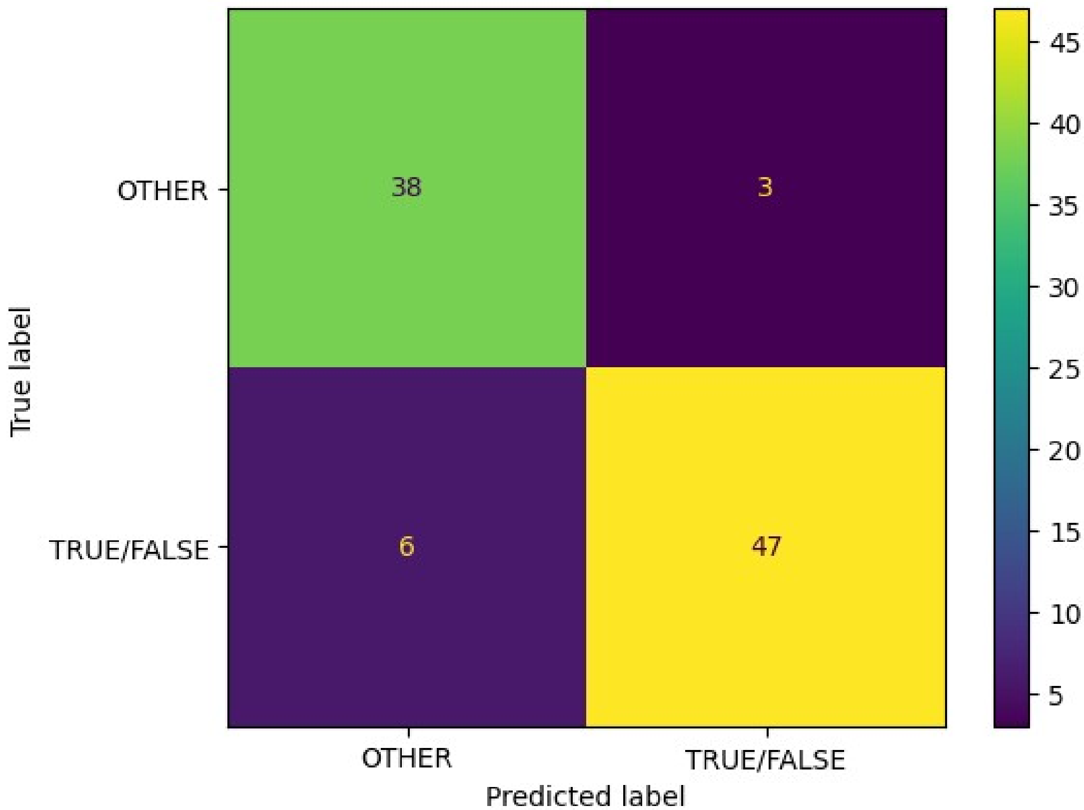
True label
OTHER
TRUE/FALSE
38
3
5
48
OTHER
TRUE/FALSE
Predicted label
45
40
35
30
25
20
15
10
5

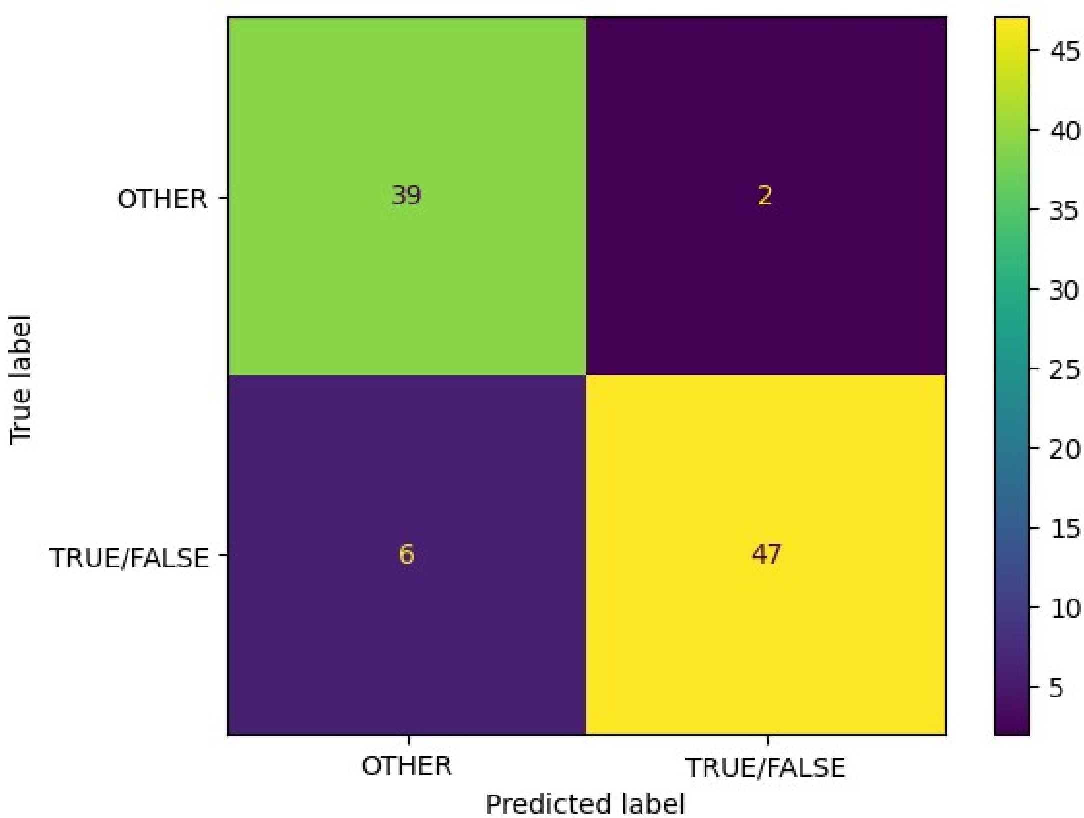


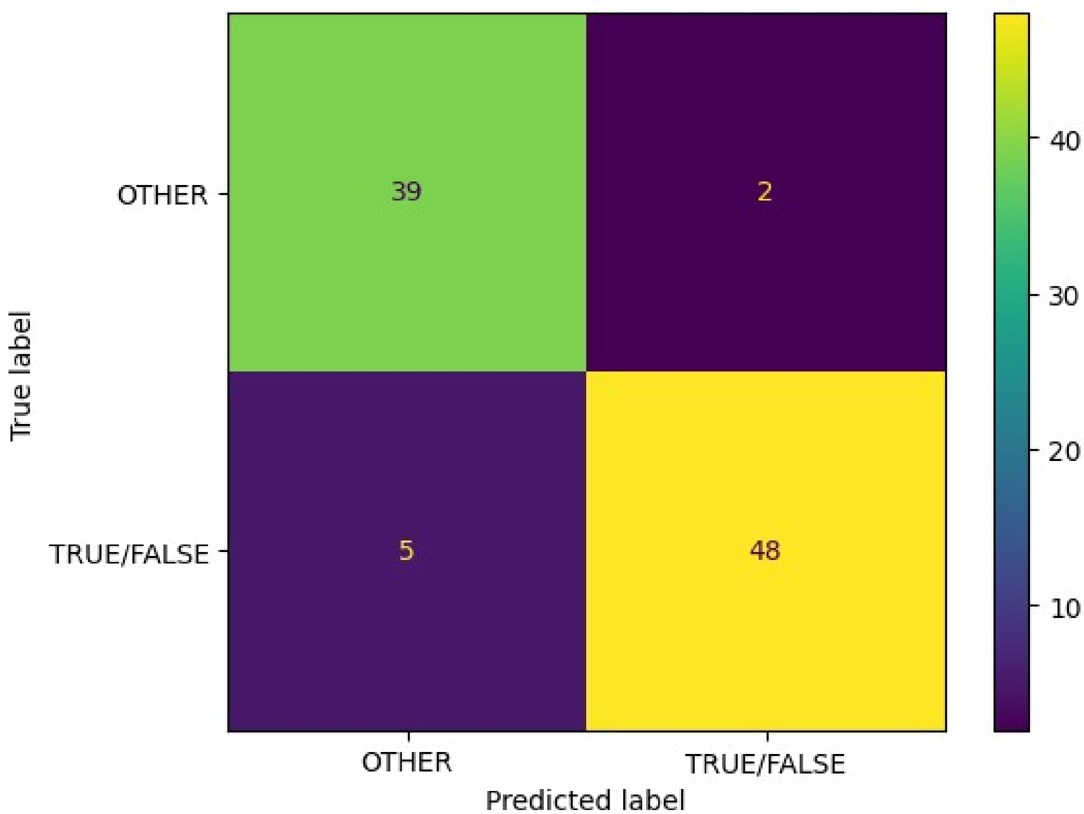
OTHER
TRUE/FALSE
39
2
5
48
True label
Predicted label
10
20
30
40

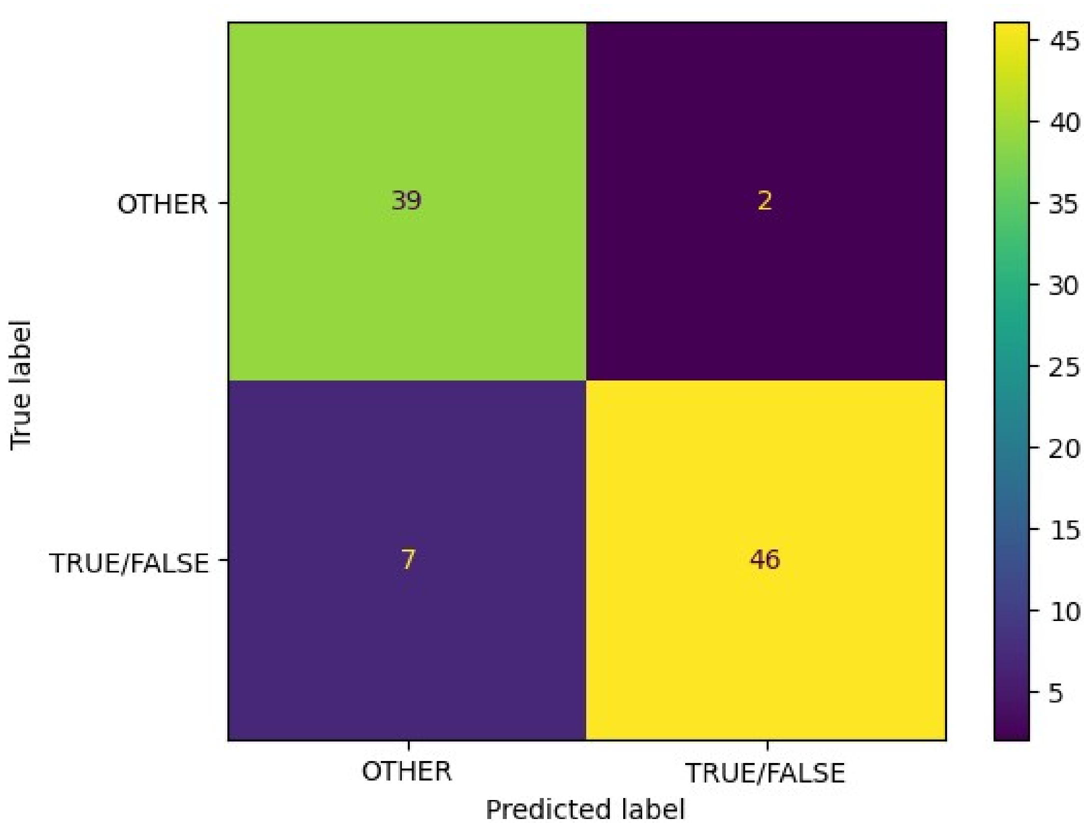
OTHER
TRUE/FALSE
39
2
8
45
True label
Predicted label
5
10
15
20
25
30
35
40
45


38
3
6
47
OTHER
TRUE/FALSE
True label
Predicted label
45
40
35
30
25
20
15
10
5

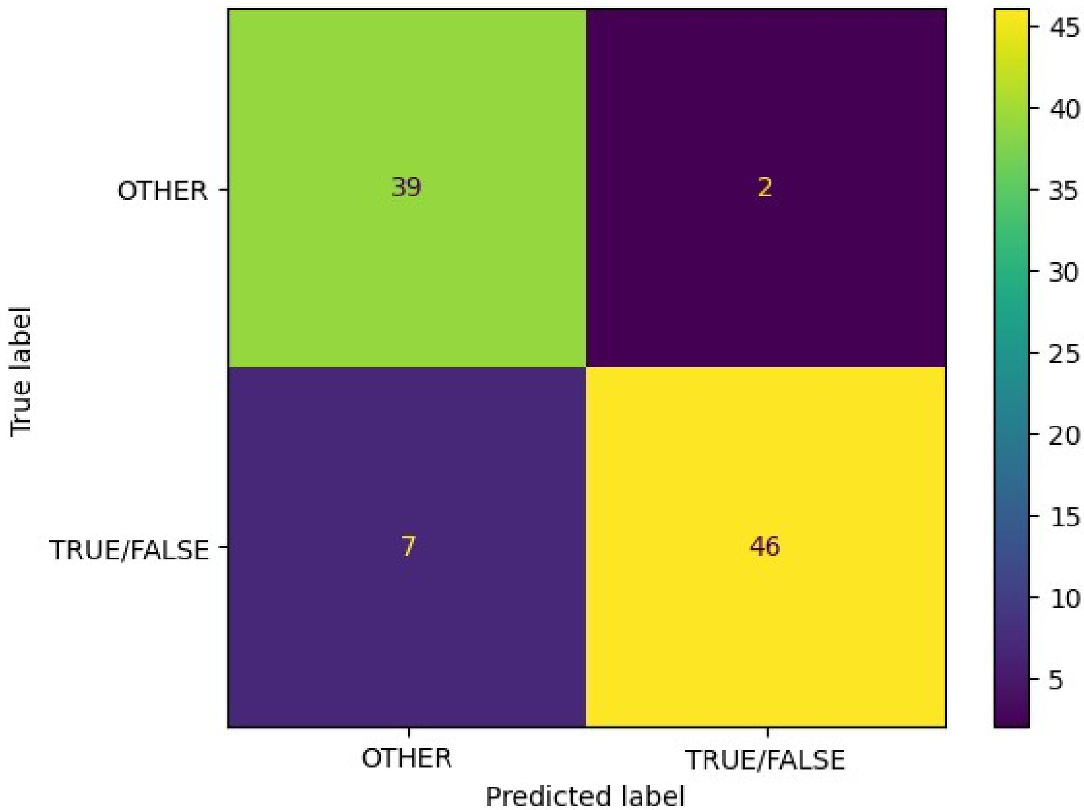

39
2
6
47
OTHER
TRUE/FALSE
True label
Predicted label
45
40
35
30
25
20
15
10
5



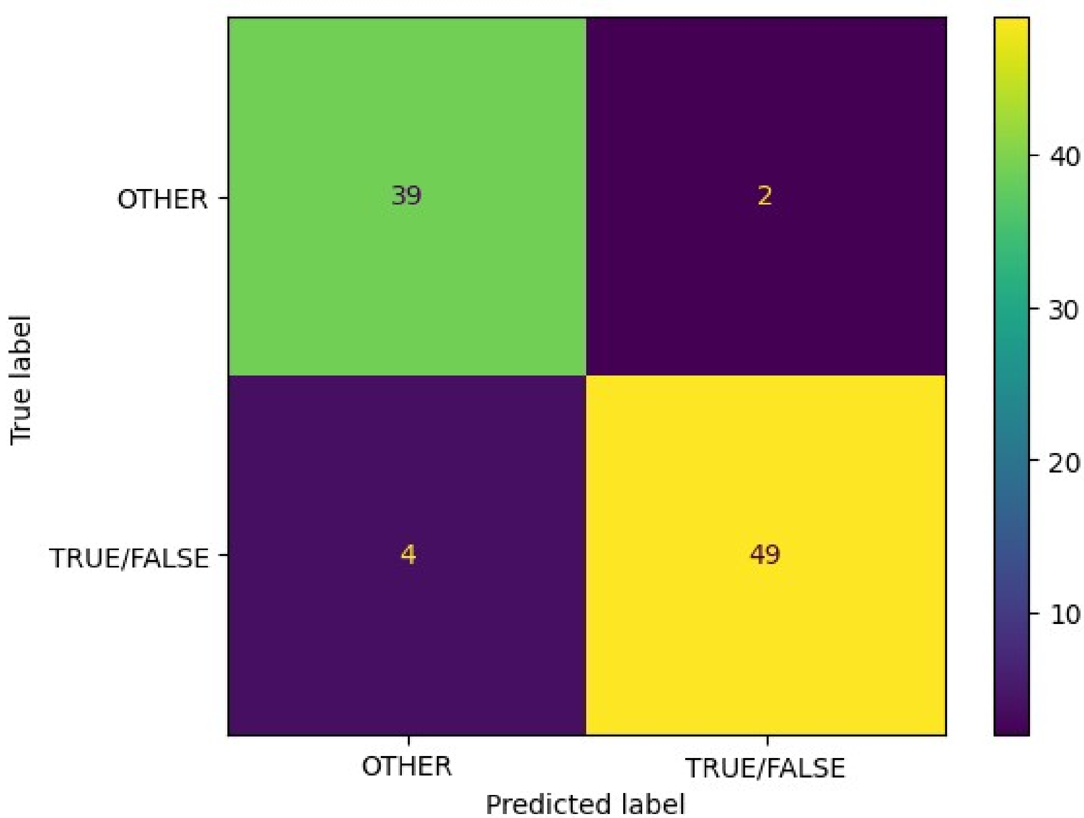
39
2
5
48
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
40
30
20
10

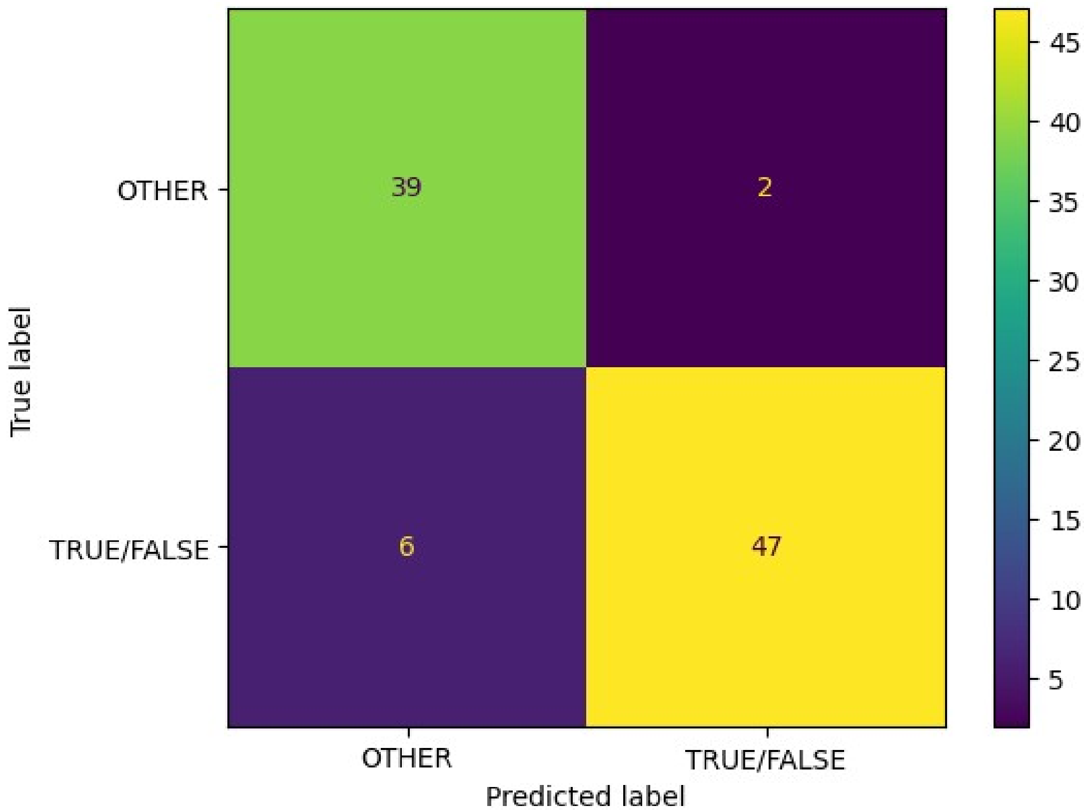
39
2
7
46
OTHER
TRUE/FALSE
True label
OTHER
TRUE/FALSE
Predicted label
45
40
35
30
25
20
15
10
5



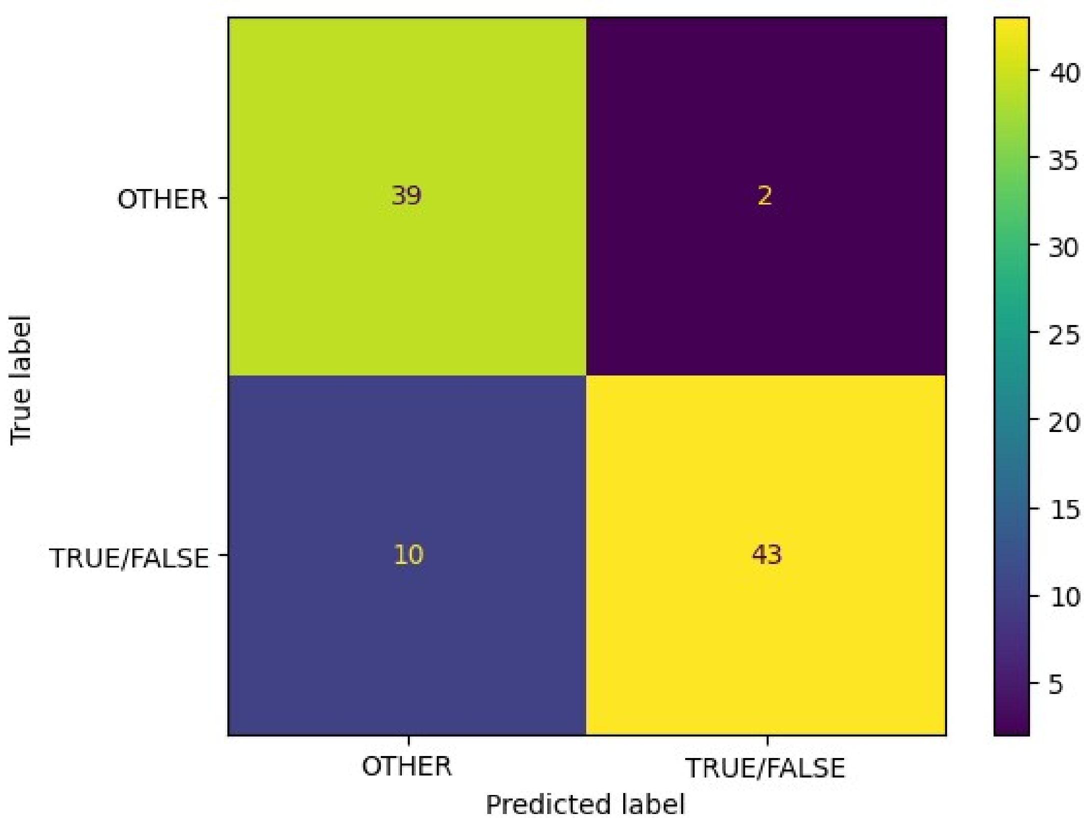
OTHER
TRUE/FALSE
39
2
7
46
True label
Predicted label
45
40
35
30
25
20
15
10
5

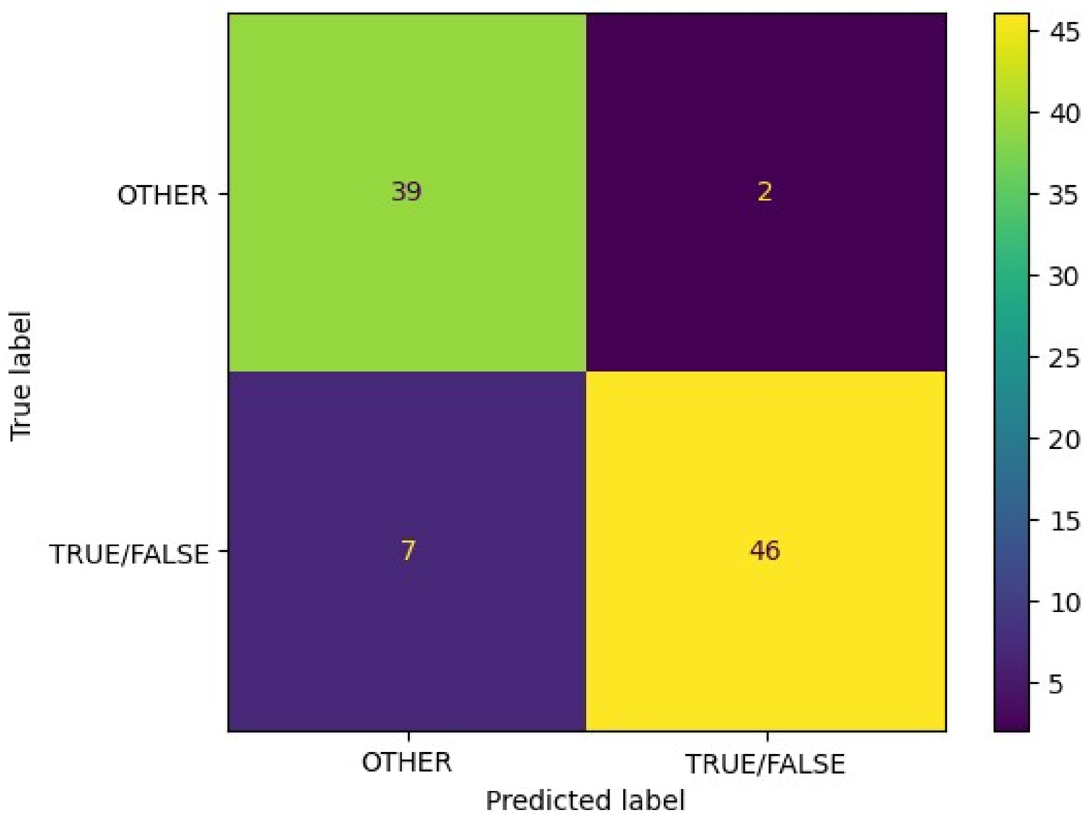
OTHER
TRUE/FALSE
39
2
4
49
True label
Predicted label
40
30
20
10



OTHER
TRUE/FALSE
True label
39
2
6
47
OTHER
TRUE/FALSE
Predicted label
45
40
35
30
25
20
15
10
5

OTHER
TRUE/FALSE
True label
39
2
10
43
OTHER
TRUE/FALSE
Predicted label
40
35
30
25
20
15
10
5



OTHER
TRUE/FALSE
True label
39
2
7
46
OTHER
TRUE/FALSE
Predicted label
45
40
35
30
25
20
15
10
5