**FAKE NEWS DETECTION USING NLP**

**DATA CLEANING:**

buzzfeed\_real**$**source <- **with**(buzzfeed\_real, **reorder**(source, source, **function**(x) **length**(x)))

**ggplot**(data = buzzfeed\_real) **+**

**ggtitle**("source count of real news in Buzzfeed") **+**

**geom\_bar**(**aes**(x= source),fill = "green") **+** **coord\_flip**()

**EDT :**

df.info()

df = df.drop\_duplicates()

df = df.reset\_index(drop=True)

*# Correcting some data*

df['date'] = df['date'].replace(['19-Feb-18'],'February 19, 2018')

df['date'] = df['date'].replace(['18-Feb-18'],'February 18, 2018')

df['date'] = df['date'].replace(['17-Feb-18'],'February 17, 2018')

df['date'] = df['date'].replace(['16-Feb-18'],'February 16, 2018')

df['date'] = df['date'].replace(['15-Feb-18'],'February 15, 2018')

df['date'] = df['date'].replace(['14-Feb-18'],'February 14, 2018')

df['date'] = df['date'].replace(['13-Feb-18'],'February 13, 2018')

df['date'] = df['date'].str.replace('Dec ', 'December ')

df['date'] = df['date'].str.replace('Nov ', 'November ')

df['date'] = df['date'].str.replace('Oct ', 'October ')

df['date'] = df['date'].str.replace('Sep ', 'September ')

df['date'] = df['date'].str.replace('Aug ', 'August ')

df['date'] = df['date'].str.replace('Jul ', 'July ')

df['date'] = df['date'].str.replace('Jun ', 'June ')

df['date'] = df['date'].str.replace('Apr ', 'April ')

df['date'] = df['date'].str.replace('Mar ', 'March ')

df['date'] = df['date'].str.replace('Feb ', 'February ')

df['date'] = df['date'].str.replace('Jan ', 'January ')

In [8]:

df['date'] = df['date'].str.replace(' ', '')

In [9]:

linkcode

for i, val **in** enumerate(df['date']):

df['date'].iloc[i] = pd.to\_datetime(df['date'].iloc[i], format='%B**%d**,%Y', errors='coerce') *# by setting the parameter to "coerce", we will set unappropriate values to NaT (null)*

**PREPROCESSING :**

*#Data Pre-processing*

import numpy as np

import pandas as pd

*#Data Visualisation*

import seaborn as sns

import matplotlib.pyplot as plt

import plotly.express as px

import plotly.graph\_objects as go

*#Handling Warnings*

import warnings

warnings.filterwarnings('ignore')

*#Text pre-processing*

import string

string.punctuation

import re

import nltk

from nltk.stem.porter

import PorterStemmer

from nltk.stem import WordNetLemmatizer

*#Machine Learning*

from sklearn.model\_selection import train\_test\_split,GridSearchCV,RandomizedSearchCV

from sklearn.feature\_extraction.text import TfidfVectorizer

from sklearn.feature\_selection import SelectKBest,chi2,f\_classif

from sklearn.ensemble import RandomForestClassifier,VotingClassifier,AdaBoostClassifier,GradientBoostingClassifier,BaggingClassifier

from sklearn.naive\_bayes import GaussianNB

from sklearn.naive\_bayes import BernoulliNB

from sklearn.naive\_bayes import MultinomialNB

from sklearn.neighbors import KNeighborsClassifier

from sklearn.tree import DecisionTreeClassifier

import xgboost as xgb

from sklearn import tree

from sklearn.metrics import classification\_report , confusion\_matrix

from xgboost import XGBClassifier

import os

for dirname, \_, filenames **in** os.walk('/kaggle/input'):

for filename **in** filenames:

print(os.path.join(dirname, filename))