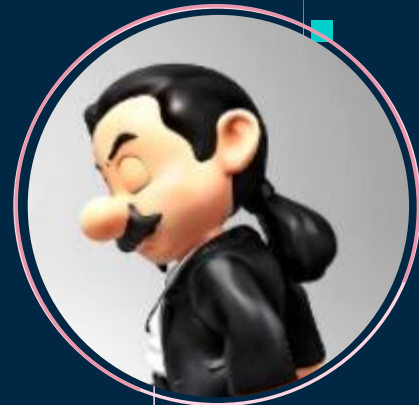


DATA**CUP** 2021

Melena Bros

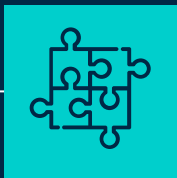


Reto 1: Vacunas

Enfocado en el análisis de datos, el proyecto trabajado consiste en filtrar según la preferencia del usuario tweets relacionados al Covid-19 según ciertos parámetros.



Puntos a cubrir



01

Procesamiento de
datos



02

Análisis de
texto



03

Presentación de
resultados

Procesamiento de datos

01

Áreas de oportunidad del proyecto

Tiempo de procesamiento

Optimización del código



Filtros

Categorías pertinentes de datos

Polarización

Ideas contrastantes de la población respecto al tema



Limpieza

Código estructurado para la fácil comprensión

Análisis de datos

02

Variables a analizar

Localización de tweets



```
import folium
from folium.plugins import FastMarkerCluster
from openpyxl import load_workbook
import requests
import urllib.parse

#Crear el mapa como un objeto
m = folium.Map(location=[37.7790262, -122.419986])

#Global tooltip
#tooltip = "Click para más información"

wb = load_workbook('vaccination_all_tweets.xlsx')
sheet = wb.worksheets[0]

coordenadas = []

i=2
for row in sheet.iter_rows(min_row=2, max_row=46060, values_only=True):
    address = row[2]
    while True:
        try:
            url = 'https://nominatim.openstreetmap.org/search/' + urllib.parse.quote(address) + '?format=json'
            response = requests.get(url).json()
            xy = [ response[0]["lat"], response[0]["lon"] ]
            print([row[2],xy])
            coordenadas.append(xy)
            #i+=1
            break
        except TypeError:
            #i+=1
            print([row[2], "Este no"])
            break
        except IndexError:
            print([row[2], "IndexError (chechar)"])
            break
```

Variables a analizar

Parámetros de tweets para un posterior análisis

```
tit = []  
for row in sheet.iter_rows(min_row=1,max_row=1,values_only=True):  
    tit += row  
print(tit)
```

Permite ver los elementos del documento para poder
clasificar y filtrar datos

Análisis de texto

Análisis de sentimientos mediante palabras clave en el texto

```
reacciones = { "positivo": ["safe", "treatment", "administration", "administered", "dose", "doses", "health", "healthy",  
| "family", "admiration", "courage", "brave", "bravery", "serious", "seriously", "merry", "merrier", "Same", "paste", "effective"],  
| "negativo": ["bad", "crime", "cheat", "cheated", "greed", "side", "effects", "corruption", "hurt", "hurts", "hate", "hating",  
| "diplomacy", "fake", "stall", "stalled", "war", "ineffective"],  
| "neutral": ["facts", "fact", "sources", "source", "information", "cases", "case", "deaths", "distribution", "specialist",  
| "programme", "inoculation", "inoculating", "needle", "medicine", "symptoms", "available", "update", "schedule", "immunity",  
| "authorization", "authorized", "information", "approving", "approved", "manufacture", "manufacturing"]}  
  
vaccines_dict = {"PfizerBioNTech": 0, "AstraZeneca": 0, "SputnikV": 0, "Moderna": 0, "johnsonandjohnson": 0, "Oxford": 0,  
| "Novavax": 0, "Sinovac": 0, "Cansino": 0, "Bharat": 0}  
vaccines = ["PfizerBioNTech", "AstraZeneca", "SputnikV", "Moderna", "johnsonandjohnson", "Oxford", "Novavax",  
| "Sinovac", "Cansino", "Bharat"]
```

Palabras detonantes más frecuentes en tweets
correspondientes a cierto sentimiento a analizar organizadas
en diccionarios

Análisis de texto

Algoritmo para determinar la frecuencia de palabras detonantes

```
# Búsqueda de tweets mediante palabras clave
Pos = 0
Neg = 0
Neu = 0
total_words = dict()
for row in sheet.iter_rows(min_row = 2, max_row = sheet.max_row, values_only = True):
    rate = 0
    words = row[10].split()
    if row[11] != None : hashtags = row[11].split("")
    for word in words:
        if word in reacciones["positivo"]:
            rate += 1
        elif word in reacciones["negativo"]:
            rate += -1
        elif word not in total_words:
            total_words[word] = 1
        elif word in total_words:
            total_words[word] += 1
        elif word[0] == "#":
            hashtags.append(word)

    if rate >= 1 : Pos += 1
    elif rate <= -1 : Neg += 1
    else : Neu += 1

    vaccines_index = list()
    for hashtag in hashtags :
        if hashtag not in vaccines_index and hashtag in vaccines :
            vaccines_index.append(hashtag)
            vaccines_dict[hashtag] += 1

end = time.time()
print("Positive: ", Pos, "\nNegative: ", Neg, "\nNeutral: ", Neu, "\nvaccines: ", vaccines_dict, "\nTime: ", end - start)
```

Frecuencia de palabras detonantes de algún sentimiento, así como de las vacunas más aplicadas

Positive: 6620

Negative: 1139

Neutral: 38300

vaccines: {'PfizerBioNTech': 4405, 'AstraZeneca': 1387, 'SputnikV': 5146, 'Cansino': 1, 'Bharat': 20}

'Moderna': 9238, 'johnsonandjohnson': 164, 'Oxford': 77, 'Novavax': 31, 'Sinovac': 2040, '

Time: 9.396013259887695



Presentación de resultados

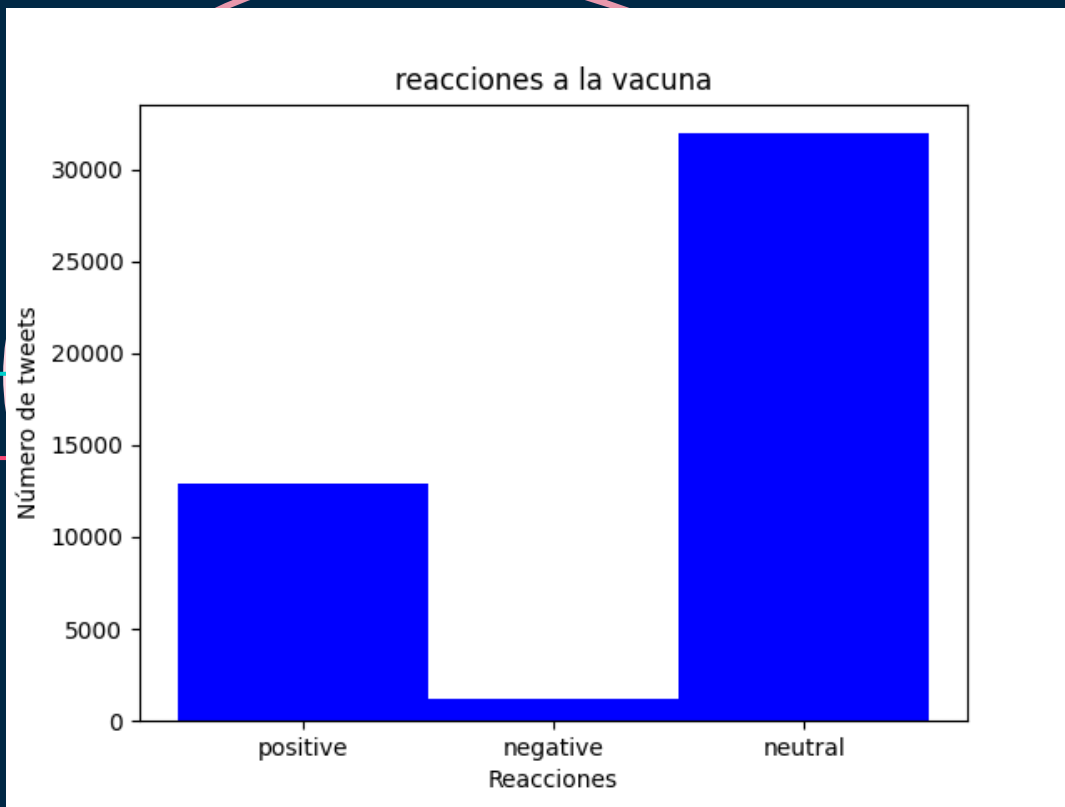
03

Mapa de tweets

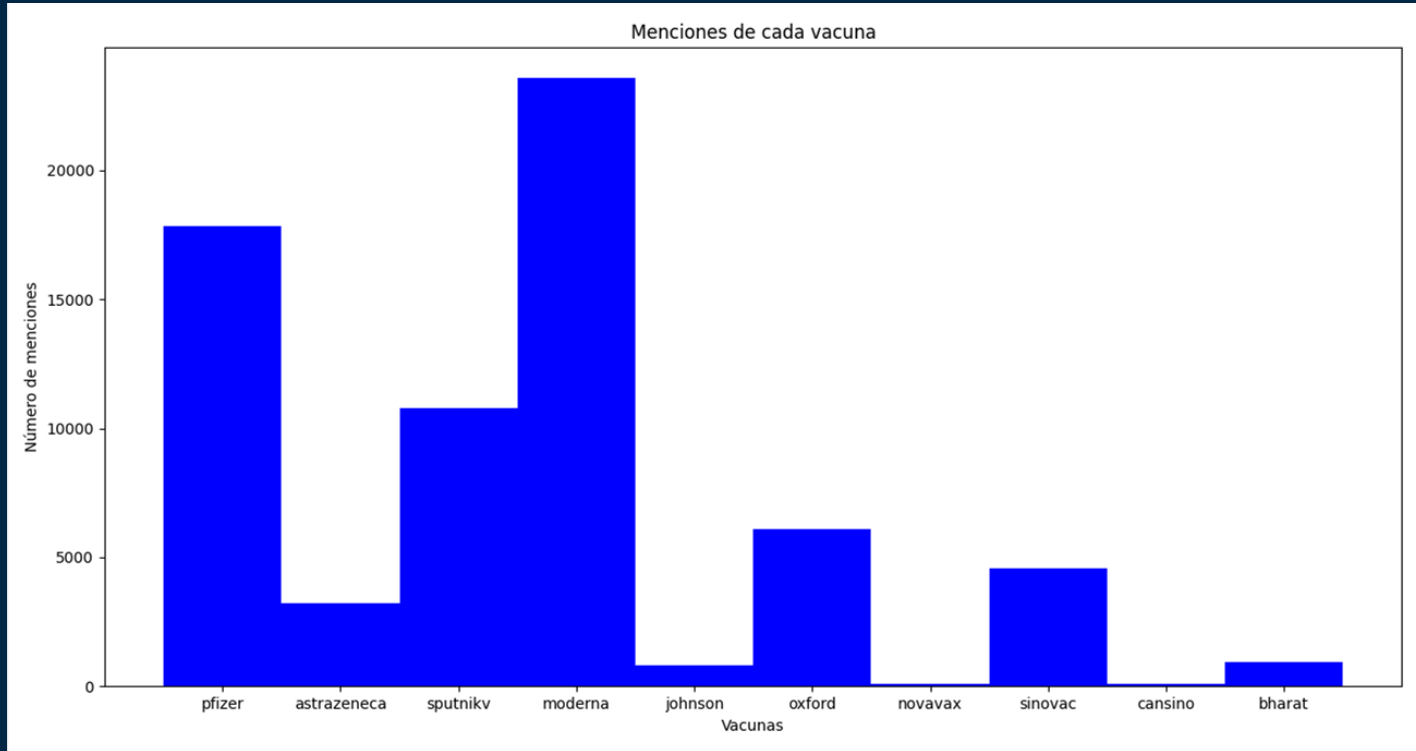
<file:///D:/descargas/vis.html>



Sentimientos de los tweets



Mención de Vacunas



Mayores interacciones

Most retweets:

Why we need Two Doses of mRNA Vaccine ðŸ’‰ #vaccines #COVID19 #Pfizer #moderna #VaccinesSaveLives #vaccinated <https://t.co/RFRmPAYubD>

Most liked:

Got my jab. For the curious, it was #Covaxin.

Felt secure, will travel safely. <https://t.co/8PL7PZMEsf>

Greatest impact:

Why we need Two Doses of mRNA Vaccine ðŸ’‰ #vaccines #COVID19 #Pfizer #moderna #VaccinesSaveLives #vaccinated <https://t.co/RFRmPAYubD>

Agradecemos
su atención



python™