

▼ Laboratorio 6 - Analisis de Redes Sociales

Introduccion

Problema 1

Extraiga los datos relacionados con los temblores en Guatemala, puede utilizar cualquier red social, a la que se tenga acceso, y pueden extraerse datos de varias. Por ejemplo si extrae datos de twitter podría utilizar el hashtag #temblorgt

- Explore los datos que extraiga y descubra conocimiento, tendencias y elementos interesantes.

Problema 2

Extraiga los datos relacionados con el tráfico en la ciudad de Guatemala, puede utilizar cualquier red social, a la que se tenga acceso, y pueden extraerse datos de varias. Por ejemplo si extrae datos de twitter podría utilizar los hashtag #TraficoGTo #TransitoGT

- Explore los datos que extraiga y descubra conocimiento, tendencias y elementos interesantes.

Problema 3

Extraiga los datos de las redes sociales de una empresa, una campaña de marketing de una empresa o el lanzamiento de un producto.

Agregue la descripción de la empresa, campaña o producto y analice las interacciones de los clientes con ellos, determine el éxito de la campaña, producto o empresa en las interacciones de los clientes

Empresa

Se estudiara el desempeño para transmitir y el impacto que tiene Leag Of Leagends internacionalmente con el #Worlds2018, ya que estan a punto de tener la final.

ya que estan a punto de entrar a la final del campeonato 2018


▼ Configuraciones Iniciales

```
!pip install twitter
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import re
import io
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize
from nltk import ngrams
# Load library
```

```

from nltk.corpus import stopwords
import os
# You will have to download the set of stop words the first time
import nltk
import operator
nltk.download('stopwords')
stop_words = set(stopwords.words('spanish'))

```

 Requirement already satisfied: twitter in /usr/local/lib/python3.6/dist-packages (1.18)

 [nltk_data] Downloading package stopwords to /root/nltk_data...

 [nltk_data] Package stopwords is already up-to-date!

▼ Solucion problema 1

▼ Descarga de Informacion de Twitter

```

from twitter import Twitter, OAuth

ACCESS_TOKEN = '339324943-grMlvtueGS0gmP4L1ycn0HylyfAtfYkq01bNmpsz'
ACCESS_SECRET = 'i2LEct5Gf18oTyGxgwjYyVdxLMbHk1Qqj6wrSulK28KQf'
CONSUMER_KEY = 'xmdPRdhFjTEmYGvL8nCNe18bv'
CONSUMER_SECRET = 'BW2o4ndwdPE9KecS8YpQe5kSSJmIie77ZqifYcDW7VLxrfjYtV'

oauth = OAuth(ACCESS_TOKEN, ACCESS_SECRET, CONSUMER_KEY, CONSUMER_SECRET)
t = Twitter(auth=oauth)

```

```

query = t.search.tweets(q='%23temblorgt')

for s in query['statuses']:
    print(s['created_at'], s['text'], '\n')

```



Sat Oct 27 14:30:23 +0000 2018 📍 Efemérides #TemblorGT 39 Años
27/10/1979 – Magnitud: 6.8, Profundidad: 58km; #Guatemala
🔍 <https://t.co/rmrwx3Nwxt>

Sat Oct 27 13:00:26 +0000 2018 📍 Efemérides #TemblorGT 39 Años
27/10/1979 – Magnitud: 6.6, Profundidad: 65km; #Guatemala
🔍 <https://t.co/f1UxyyNAL2>

Sat Oct 27 06:14:24 +0000 2018 #Temblorgt ?

Fri Oct 26 23:43:06 +0000 2018 RT @prensa_libre: #TemblorGT | Actualización. Sismo fue

Fri Oct 26 22:56:49 +0000 2018 GT→#Temblor 4.1 ⚡ 🕒 15:57 Pm H/GuatemalaGT
📍 #Champerico a 136 Km al SurOeste
D. de #Retalhuleu
#GuatemalaGT... <https://t.co/garXf3eseo>

▼ Limpieza de datos

▼ Volviendo Todo en Minuscula

```
Fri Oct 26 19:03:48 +0000 2018 RT @prensa_libre: #TemblorGT | Un sismo con epicentro a
for s in query['statuses']:
    s['text'] = s['text'].lower()

Fri Oct 26 19:03:48 +0000 2018 RT @prensa_libre: #TemblorGT | Un sismo con epicentro a
```

▼ Quitar los caracteres especiales que aparecen como "#","@" o los apóstrofes.

```
for s in query['statuses']:
    s['text'] = re.sub(r"@", "", s['text'])
    s['text'] = re.sub(r"#", "", s['text'])
    s['text'] = re.sub(r'\', "", s['text'])
    s['text'] = re.sub(r"-", "", s['text'])
    s['text'] = re.sub(r'📍', "", s['text'])
    s['text'] = re.sub(r'🔍', "", s['text'])
```

▼ Quitar las url

```
for s in query['statuses']:
    s['text'] = re.sub(r"(http|ftp|https)[:]A-Za-z-//.0-9_]*$", "", s['text'])
```

▼ Quitando Stopwords

```
for s in query['statuses']:
    # s['text'] = s['text'].split()
    s['text'] = ' '.join([word for word in s['text'].split() if word not in stop_words])

# query['statuses'] = query['statuses'].split(' ').apply(lambda x: ' '.join(k for k in x if k
# dataset['reviewtext'] = dataset['reviewtext'].str.split(' ').apply(lambda x: ' '.join(k for
```

```
for s in query['statuses']:
    print(s['text'])
```

```
efemérides temblor 39 años 27/10/1979 magnitud: 6.8, profundidad: 58km; guatemala
efemérides temblor 39 años 27/10/1979 magnitud: 6.6, profundidad: 65km; guatemala
temblor ?
rt prensa_libre: temblor | actualización. sismo magnitud 4.2 epicentro 250.4 km dire
gr→temblor 4.1 ⚡ ⌚15:57 pm h/guatemalagr ⬠champerico 136 km suroeste d. retalhuleu
rt prensa_libre: temblor | actualización. sismo magnitud 4.2 epicentro 250.4 km dire
temblor | actualización. sismo magnitud 4.2 epicentro 250.4 km dirección este-norest
12:24 horas registró temblor magnitud 4.2 epicentro 250.4 kilómetros dirección este-
rt prensa_libre: temblor | sismo epicentro 11 kilómetros suroeste puerto barrios mag
rt prensa_libre: temblor | sismo epicentro 11 kilómetros suroeste puerto barrios mag
rt prensa_libre: temblor | sismo epicentro 11 kilómetros suroeste puerto barrios mag
rt prensa_libre: temblor | sismo epicentro 11 kilómetros suroeste puerto barrios mag
temblor | sismo epicentro 11 kilómetros suroeste puerto barrios magnitud 4.1 registr
ve! nadie dice temblor 3:55.
rt carolitacar1491: temblor señor sigue acosando guatemala ayuda
```

▼ Descubriendo Informacion

Se crea un diccionario con las palabras utilizadas y se van contando cuantas veces se repiten

Diccionario

```
palabrasMasComunes = {}
```

```
def word_count(str, dictionary):
    for word in str.split():
        if word in dictionary:
            dictionary[word] += 1
        else:
            dictionary[word] = 1
    return dictionary
```

```
for s in query['statuses']:
    palabrasMasComunes = word_count(s['text'], palabrasMasComunes)
```

```
# for palabra in palabrasMasComunes:
```

▼ Palabras mas Comunes

```
for key in palabrasMasComunes:
    print ("%s: %s" % (key, palabrasMasComunes[key]))
```

```
↳
```

```

efemérides: 2
temblorgt: 14
39: 2
años: 2
27/10/1979: 2
magnitud:: 2
6.8,: 1
profundidad:: 2
58km;: 1
guatemala: 3
6.6,: 1
65km;: 1
?: 1
rt: 7
prensa_libre:: 6
|: 8
actualización.: 3
sismo: 8
magnitud: 9
4.2: 4
epicentro: 9
250.4: 4
km: 4
dirección: 4
este-noreste: 3
izabal.: 3
https://...: 2
GT→temblor: 1
4.1⚡: 1
🕒15:57: 1
pm: 1
h/guatemalagt: 1
🎲champerico: 1
136: 1
suroeste: 6
d.: 1
retalhuleu: 1
guatemalagt...: 1
12:24: 1
horas: 1
registró: 6
kilómetros: 6
este-nore...: 1
11: 5
puerto: 5
barrios: 5
4.1: 5
12:...: 4
l...: 1
ve!: 1
nadie: 1
dice: 1
2.55 . 1

```

▼ Ordenamos la lista de mayor a menor

```
sigue: 1
```

```
def keyInOrder(listName, number):
```

```

keys = list(sorted(listName, key=listName.__getitem__, reverse=True))
keys = keys[:number]
firstFew = {x:listName[x] for x in keys}
return firstFew

```

```

palabrasOrdenadas = {}

```

```

for key, value in sorted(palabrasMasComunes.items(), key=operator.itemgetter(1), reverse=True):
    palabrasOrdenadas[key] = int(value)

```

```

# sorted(palabrasMasComunes.items(), key=operator.itemgetter(1), reverse=True)
print(palabrasOrdenadas)

```

```

↳ {'temblorgt': 14, 'magnitud': 9, 'epicentro': 9, '|': 8, 'sismo': 8, 'rt': 7, 'prensa_

```

```

keys = keyInOrder(palabrasOrdenadas, 10)

```

```

print (keys)

```

```

↳ {'temblorgt': 14, 'magnitud': 9, 'epicentro': 9, '|': 8, 'sismo': 8, 'rt': 7, 'prensa_

```

▼ Representacion en Graficas

```

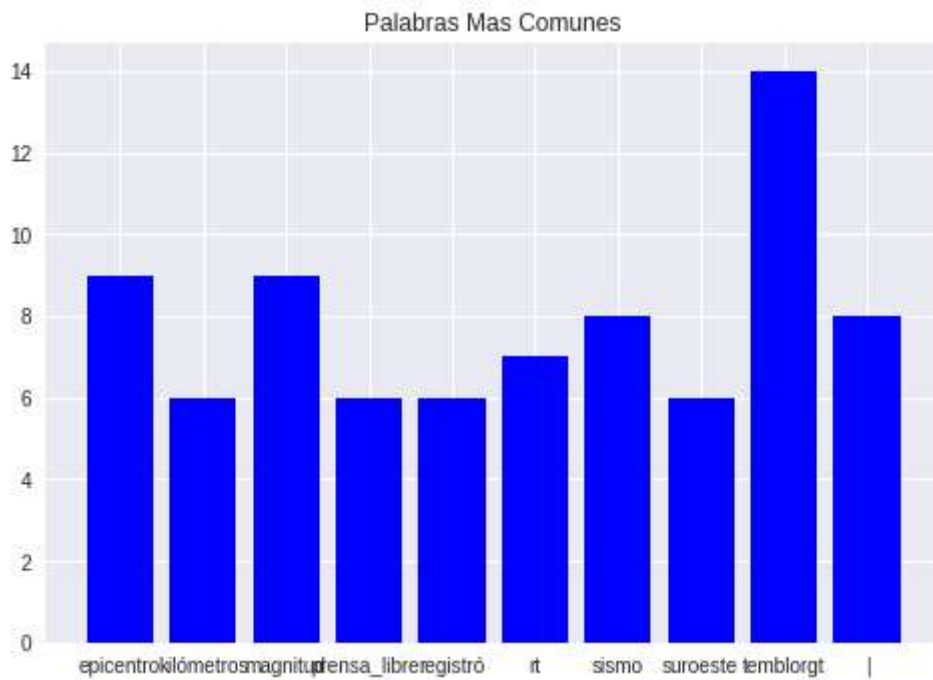
plt.bar(keys.keys(), keys.values(), color='b')
plt.title("Palabras Mas Comunes")

```

```

↳ Text(0.5,1,'Palabras Mas Comunes')

```



Conclusiones Problema 1

Viendo las palabras mas comunes: **temblorgt, guatemala, efemérides, años, magnitud**

Podemos decir que las emociones de los que estan escribiendo son negativas y preocupantes, informatica

▼ Solucion problema 2

▼ Descarga de Informacion de Twitter

Se va a buscar twitts que tengan los hashtags #TraficoGTo y #TransitoGT

```
##TraficoGTo #TransitoGT
query1 = t.search.tweets(q='%23PrecaucionGT')
query2 = t.search.tweets(q='%23TransitoGT')
```

```
for s1 in query1['statuses']:
    print(s1['created_at'], s1['text'], '\n')
```



Sun Oct 28 00:22:26 +0000 2018 RT @amilcarmontejo: #PrecaucionGT por carro averiado en Informa Eduardo Mó que...

Sat Oct 27 23:22:02 +0000 2018 RT @amilcarmontejo: #PrecaucionGT por carro averiado en Informa Eduardo Mó que...

Sat Oct 27 23:14:49 +0000 2018 #PrecaucionGT por carro averiado en Viaducto Villa Lind Informa Eduardo Mó... <https://t.co/dXYLjjeCFi>

Wed Oct 24 14:51:07 +0000 2018 RT @amilcarmontejo: #PrecaucionGT por diesel derramado Douglas...

Wed Oct 24 13:42:58 +0000 2018 RT @amilcarmontejo: #PrecaucionGT por diesel derramado Douglas...

Wed Oct 24 13:22:44 +0000 2018 RT @amilcarmontejo: #PrecaucionGT por diesel derramado Douglas...

Wed Oct 24 13:16:50 +0000 2018 #PrecaucionGT por diesel derramado en tramo de 10 avni

```
for s2 in query2['statuses']:
    print(s2['created_at'], s2['text'], '\n')
```



Sun Oct 28 01:34:00 +0000 2018 Accidente #transitogt en 6Calle Barrio El Jardín #Coate

Sun Oct 28 01:19:07 +0000 2018 #SurOccidenteGT

Accidente #transitogt en 6Calle Barrio El Jardín #Coatepeque, #Quetzaltenango, motoris

Sun Oct 28 00:58:11 +0000 2018 RT @amilcarmontejo: MARIO ROBERTO ECUTE MONTENEGRO se l

Sun Oct 28 00:42:50 +0000 2018 RT @amilcarmontejo: MARIO ROBERTO ECUTE MONTENEGRO se l

Sun Oct 28 00:22:26 +0000 2018 RT @amilcarmontejo: #PrecaucionGT por carro averiado en

▼ Limpieza de datos

▼ Volviendo Todo en Minuscula

```
Sat Oct 27 23:22:02 +0000 2018 RT @amilcarmontejo: #PrecaucionGT por carro averiado en
for s1 in query1['statuses']:
    s1['text'] = s1['text'].lower()

for s2 in query2['statuses']:
    s2['text'] = s2['text'].lower()

Sat Oct 27 23:14:49 +0000 2018 #PrecaucionGT por carro averiado en Viaducto Villa Lind
```

▼ Quitar los caracteres especiales que aparecen como "#", "@" o los apóstrofes.

```
for s1 in query1['statuses']:
    s1['text'] = re.sub(r"@", "", s1['text'])
    s1['text'] = re.sub(r"#", "", s1['text'])
    s1['text'] = re.sub(r"\'", "", s1['text'])
    s1['text'] = re.sub(r"rt", "", s1['text'])

for s2 in query2['statuses']:
    s2['text'] = re.sub(r"@", "", s2['text'])
    s2['text'] = re.sub(r"#", "", s2['text'])
    s2['text'] = re.sub(r"\'", "", s2['text'])
    s2['text'] = re.sub(r"rt", "", s2['text'])
```

▼ Quitar las url

```
for s1 in query1['statuses']:
    s1['text'] = re.sub(r"(http|ftp|https)[:A-Za-z//.0-9_]*$", "", s1['text'])

for s2 in query2['statuses']:
    s2['text'] = re.sub(r"(http|ftp|https)[:A-Za-z//.0-9_]*$", "", s2['text'])
```

▼ Quitando Stopwords

```
for s1 in query1['statuses']:
    s1['text'] = ' '.join([word for word in s1['text'].split() if word not in stopwords.words('es')])

for s2 in query2['statuses']:
    s2['text'] = ' '.join([word for word in s2['text'].split() if word not in stopwords.words('es')])
```

```
s2['text'] = ' '.join([word for word in s2['text'].split() if word not in stopwords.words('!)]
```

▼ Descubriendo Informacion

Se crea un diccionario con las palabras utilizadas y se van contando cuantas veces se repiten

Diccionario

```
palabrasMasComunes1 = {}  
palabrasMasComunes2 = {}
```

```
for s1 in query1['statuses']:  
    palabrasMasComunes1 = word_count(s1['text'], palabrasMasComunes1)  
  
for s2 in query2['statuses']:  
    palabrasMasComunes2 = word_count(s2['text'], palabrasMasComunes2)
```

▼ Palabras mas Comunes

```
for key in palabrasMasComunes1:  
    print ("%s: %s" % (key, palabrasMasComunes1[key]))
```



```
amilcarmontejo:: 12
precauciongt: 14
carro: 3
averiado: 3
viaducto: 3
villa: 3
linda,: 3
periférico: 3
sur: 3
17: 3
calle: 24
zona: 14
7.: 3
informa: 3
eduardo: 3
mó: 2
que...: 2
mó...: 1
diesel: 10
derramado: 10
tramo: 10
10: 10
avenida,: 10
maí: 10
4: 10
```

```
for key in palabrasMasComunes2:
    print ("%s: %s" % (key, palabrasMasComunes2[key]))
```



```

accidente: 2
transitogt: 4
6calle: 2
barrio: 2
jardín: 2
coatepeque,: 2
quetzaltenango,: 2
motoristas: 2
atropellados,: 1
socorristas...: 1
suroccidentegt: 1
atropell...: 1
amilcarmontejo:: 12
mario: 8
roboe: 8
ecute: 8
montenegro: 8
informa: 11
varios: 8
documentos: 8
nombre: 8
dinero: 8
efectivo,: 8
encon...: 8

```

▼ Ordenamos la lista de mayor a menor

```

palabrasOrdenadas1 = {}
palabrasOrdenadas2 = {}

```

```
periferico: 3
```

```

for key, value in sorted(palabrasMasComunes1.items(), key=operator.itemgetter(1), reverse=True):
    palabrasOrdenadas1[key] = int(value)

```

```

for key, value in sorted(palabrasMasComunes2.items(), key=operator.itemgetter(1), reverse=True):
    palabrasOrdenadas2[key] = int(value)

```

```
equaruo: 3
```

```

print(palabrasOrdenadas1)
print(palabrasOrdenadas2)

```

```

➞ {'calle': 24, 'precauciongt': 14, 'zona': 14, 'amilcarmontejo.': 12, 'diesel': 10, 'de
   {'amilcarmontejo.': 12, 'informa': 11, 'mario': 8, 'roboe': 8, 'ecute': 8, 'montenegro

```

```
km: 2
```

```
keys1 = keyInOrder(palabrasOrdenadas1, 50)
```

```
keys2 = keyInOrder(palabrasOrdenadas2, 50)
```

```
print (keys1)
```

```
print (keys2)
```

```

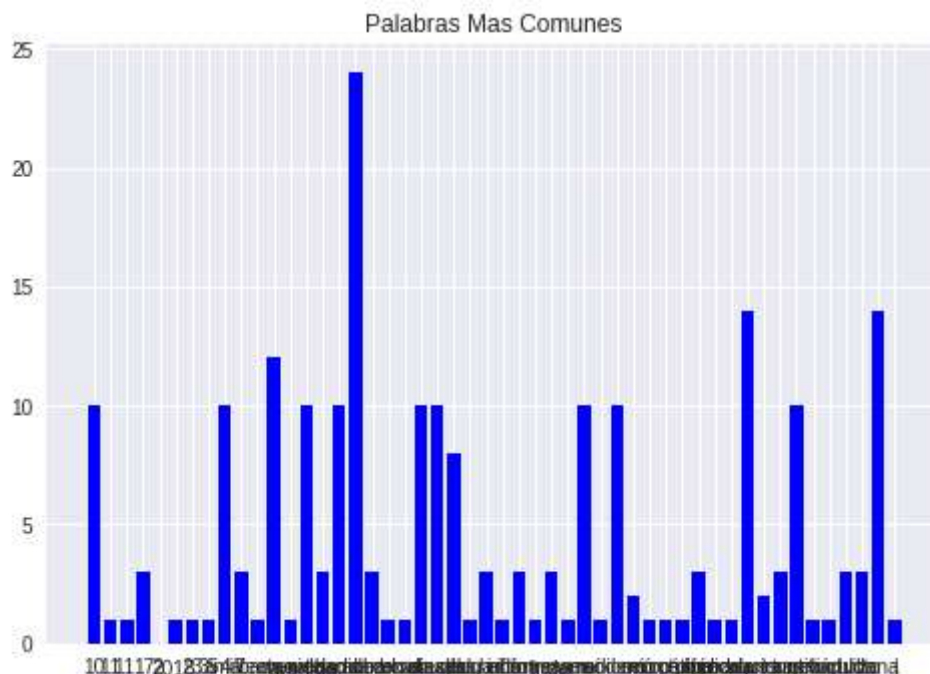
➞ {'calle': 24, 'precauciongt': 14, 'zona': 14, 'amilcarmontejo.': 12, 'diesel': 10, 'de
   {'amilcarmontejo.': 12, 'informa': 11, 'mario': 8, 'roboe': 8, 'ecute': 8, 'montenegro

```

▼ Representacion en Graficas

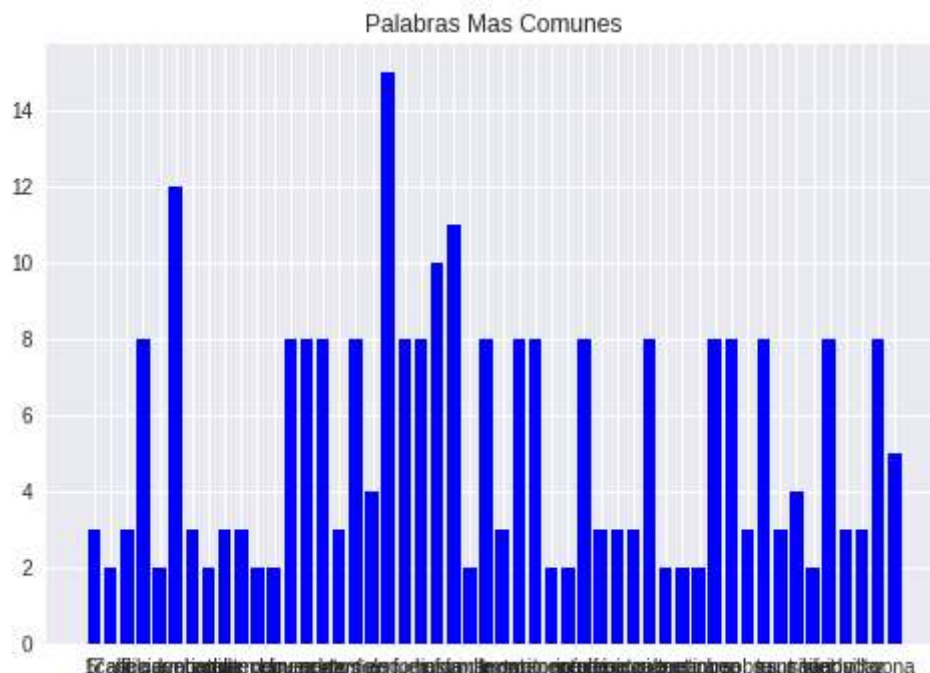
```
plt.bar(keys1.keys(), keys1.values(), color='b')
plt.title("Palabras Mas Comunes")
```

```
Text(0.5,1,'Palabras Mas Comunes')
```



```
plt.bar(keys2.keys(), keys2.values(), color='b')
plt.title("Palabras Mas Comunes")
```

```
Text(0.5,1,'Palabras Mas Comunes')
```



Conclusiones Problema 2

Primero se puede observar que para #TraficoGTo, no se encuentran tweets y es porque ha estado inactivo desde 2016

por lo cual se cambiará a #PrecaucionGT

con lo que podemos observar que al ver las palabras más comunes: Dissel, Derramado, Carro averiado, villa nueva, zona "4 y 2"

por lo cual se puede decir que las zonas 4 y 2 son las zonas con más alertas denunciadas, el aceite derramado es el mayor causante de las alertas así como carros averiados.

Ahora viendo las palabras más comunes de TransitoGT

Se observa que hay varios RT sobre Almicar Montejo, por lo cual es muy difícil hacer el análisis

por lo cual se decide hacer un estudio de una lista de 50 palabras en la cual se encuentran las palabras: Villa linda, villa nueva, Carro averiado, Fallas, tráiler, periférico.

por lo cual podemos discutir que la mayor influencia sobre accidentes o avisos es por carros averiados, fallas con tráiler, con villa nueva, villa linda y el periférico siendo las áreas más afectadas.

▼ Solucion problema 3

▼ Descarga de Informacion de Twitter

Se va a buscar twitts que tengan los hashtags #Worlds2018

```
query3 = t.search.tweets(q='%23Worlds2018')
for s3 in query3['statuses']:
    print(s3['created_at'], s3['text'], '\n')
```



Sun Oct 28 01:46:56 +0000 2018 RT @Cloud9: There are less than 8 hours left before the

Sun Oct 28 01:46:46 +0000 2018 RT @redbullesports: Dear @Cloud9.

We don't ask for much, but today, we have to ask for a 3-0 #C9WIN tomorrow.

You know what? 3-2 #C9WIN i...

Sun Oct 28 01:46:46 +0000 2018 RT @Cloud9: There are less than 8 hours left before the

Sun Oct 28 01:46:39 +0000 2018 RT @Cloud9: Don't you ever just get the feeling that...

Tomorrow is gonna be exciting 😊? #Worlds2018

Sun Oct 28 01:46:39 +0000 2018 RT @idleguide: We suspect the girls [MIGHT] be cosplayi

Sun Oct 28 01:46:35 +0000 2018 RT @lolesports: DUNKED!

Take a closer look at how TheShy's Aatrox destroyed G2's health bars in game 3 of thei

Sun Oct 28 01:46:19 +0000 2018 I'm rooting for #FNCWIN! Support your team at #worlds20

Sun Oct 28 01:46:09 +0000 2018 RT @Cloud9: 🐾 SPAM 🐾 THIS 🐾 SHOE 🐾 TO 🐾 HELP 🐾

GET THIS BREAD, @G2esports! #WESTWIN #G2WIN #Worlds2018

Sun Oct 28 01:45:24 +0000 2018 I'm rooting for #C9WIN! Support your team at #worlds201

Sun Oct 28 01:45:14 +0000 2018 RT @FionnOnFire: iG is the first non-Uzi Chinese team t

This is a big accomplishment #Worlds2018

Sun Oct 28 01:44:59 +0000 2018 RT @Official_LJL: 【#Worlds2018 SEMI FINALS特集記事】
「勝利まであと一步」

<https://t.co/4wEXu0G6v>

▼ Limpieza de datos

▼ Volviendo Todo en Minuscula

Take a closer look at how TheShy's Aatrox destroyed G2's health bars in game 3 of thei

```
for s3 in query3['statuses']:
    s3['text'] = s3['text'].lower()
```

▼ Quitar los caracteres especiales que aparecen como "#","@" o los apóstrofes.

```
for s3 in query3['statuses']:
    s3['text'] = re.sub(r"@"," ", s3['text'])
    s3['text'] = re.sub(r"#"," ", s3['text'])
    s3['text'] = re.sub(r"\'", " ", s3['text'])
    s3['text'] = re.sub(r"rt", " ", s3['text'])
    s3['text'] = re.sub(r"worlds2018", " ", s3['text'])
```

▼ Quitar las url

```
for s3 in query3['statuses']:
    s3['text'] = re.sub(r"(http|ftp|https)[:A-Za-z//.0-9_]*$", "", s3['text'])
```

▼ Quitando Stopwords

```
for s3 in query3['statuses']:
    s3['text'] = ' '.join([word for word in s3['text'].split() if word not in stopwords.words('english')])
```

▼ Descubriendo Informacion

Se crea un diccionario con las palabras utilizadas y se van contando cuantas veces se repiten

Diccionario

```
palabrasMasComunes3 = {}

for s3 in query3['statuses']:
    palabrasMasComunes3 = word_count(s3['text'], palabrasMasComunes3)
```

▼ Palabras mas Comunes

```
for key in palabrasMasComunes3:
    print ("%s: %s" % (key, palabrasMasComunes3[key]))
```




```

ever: 1
get: 2
feeling: 1
that...: 1
tomorrow: 1
gonna: 1
exciting: 1
☺?: 1
idleguide:: 1
suspect: 1
girls: 1
[might]: 1
cosplaying: 1
performance,: 1
world: 2
championship: 1
finals.: 1
here's: 1
sho: 1
clip: 1
one: 1
o...: 1
lolespos:: 3
dunked!: 2
take: 2
closer: 2
look: 2
theshys: 2
aatrox: 2
destroyed: 2
g2s: 2
health: 2
bars: 2

```

▼ Ordenamos la lista de mayor a menor

```

...
palabrasOrdenadas3 = {}

for key, value in sorted(palabrasMasComunes3.items(), key=operator.itemgetter(1), reverse=True):
    palabrasOrdenadas3[key] = int(value)

print(palabrasOrdenadas3)

keys3 = keyInOrder(palabrasOrdenadas3, 10)

print (keys3)

☞ {'👉': 7, 'cloud9': 4, 'help': 3, 'c9win': 3, 'lolespos': 3, 'suppo': 3, 'team': 3,
  '👉': 7, 'cloud9': 4, 'help': 3, 'c9win': 3, 'lolespos': 3, 'suppo': 3, 'team': 3,

...

```

▼ Representacion en Graficas

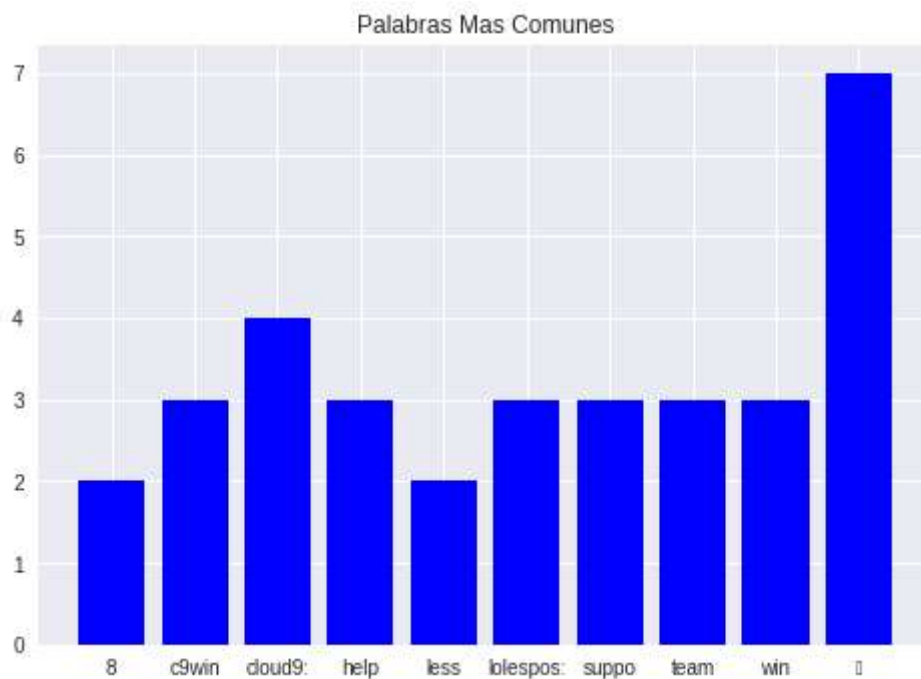
```

g2esnos!: 1

plt.bar(keys3.keys(), keys3.values(), color='b')
plt.title("Palabras Mas Comunes")

```

```
Text(0.5,1,'Palabras Mas Comunes')
```



Conclusiones Problema 3

Se puede observar que la mayoría de fans están a favor de cloud9 que es un equipo que proviene de USA