

TFM - Máster Data Science

Análisis Sentimiento Twitter

Índice

1	Introducción.....	2
1.1.	Estado del Arte.....	2
	El Corte Inglés.....	2
	Twitter.....	2
2	Alcance TFM.....	4
2.1.	Data Engineering.....	5
	Volumetría.....	5
	Información Tweets.....	5
2.2.	Data Science.....	7
2.3.	Reporting.....	9
2.4.	Ejecución del Programa.....	10
3	Investigación.....	12
3.1.	Data Engineering.....	12
3.2.	Data Science.....	¡Error! Marcador no definido.
4	Resultados.....	13
	Estado general de enero a junio del 2020.....	13
	Opinión cuentas Following.....	14
	Opinión sobre COVID-19.....	16
5	Conclusión.....	19
6	Interfaz Gráfica.....	20
7	Anexo.....	21
7.1.	El Corte Inglés.....	21
7.2.	Twitter.....	21
7.3.	Repositorio.....	21
7.4.	Limitaciones Twitter.....	21
7.5.	MicroStrategy.....	22

1 Introducción

El objetivo del proyecto fin de máster es averiguar la opinión que tienen los usuarios de una determinada red social sobre una empresa, concepto, tendencia social, etc.

El proyecto se ha orientado sobre la empresa de retail *El Corte Inglés* [7.1] y a la plataforma de *Twitter*[7.2]. Por tanto, se va a realizar un análisis de sentimiento de los tweets generados por los usuarios en un determinado periodo de tiempo haciendo énfasis en:

- Cuentas que me siguen
- Covid-19.

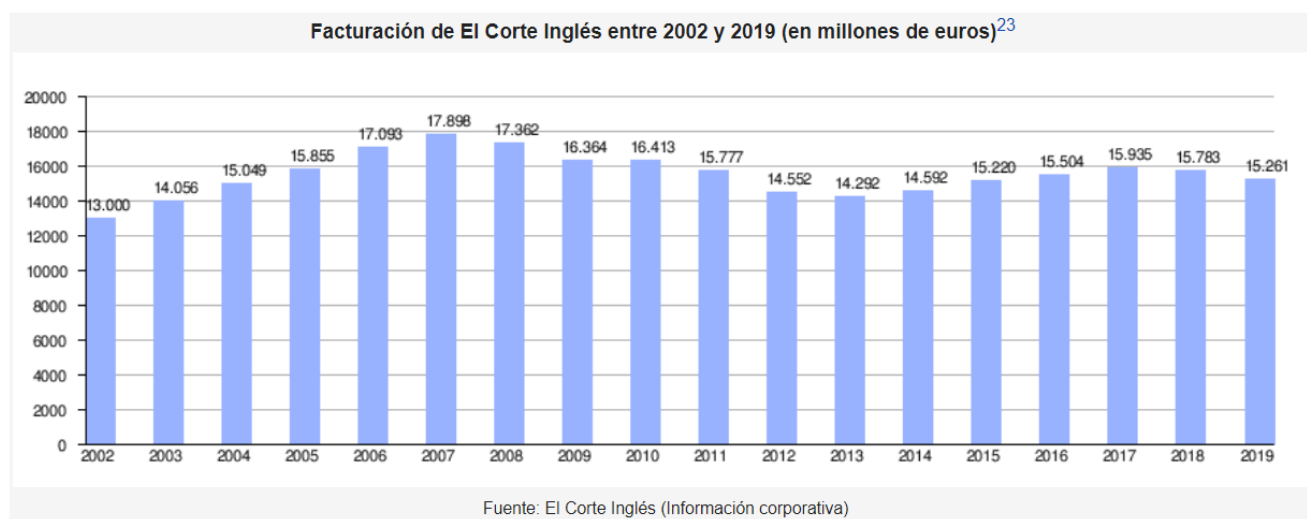


1.1. Estado del Arte

El Corte Inglés

El Corte Inglés es un grupo de distribución del sector Retail española compuesto por empresas de distintos formatos, siendo el principal el de grandes almacenes. Algunas de ellas son HiperCor, SuperCor, Sfera, etc.

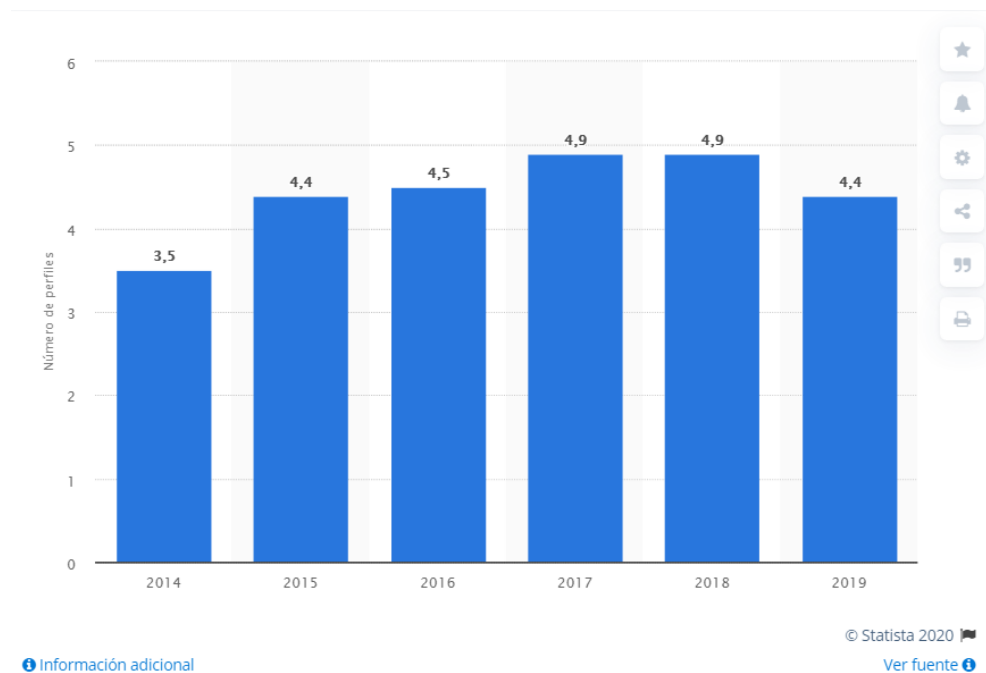
Con más de 90 centros comerciales distribuidos entre España y Portugal, es considerado la empresa de Retail más importante del país, con una facturación de más 15.000 millones de euros.



Twitter

Twitter es actualmente una de las redes sociales y plataforma de comunicación más populares del mundo, presente en todo el planeta y con más de 150 millones de usuarios activos registrados.

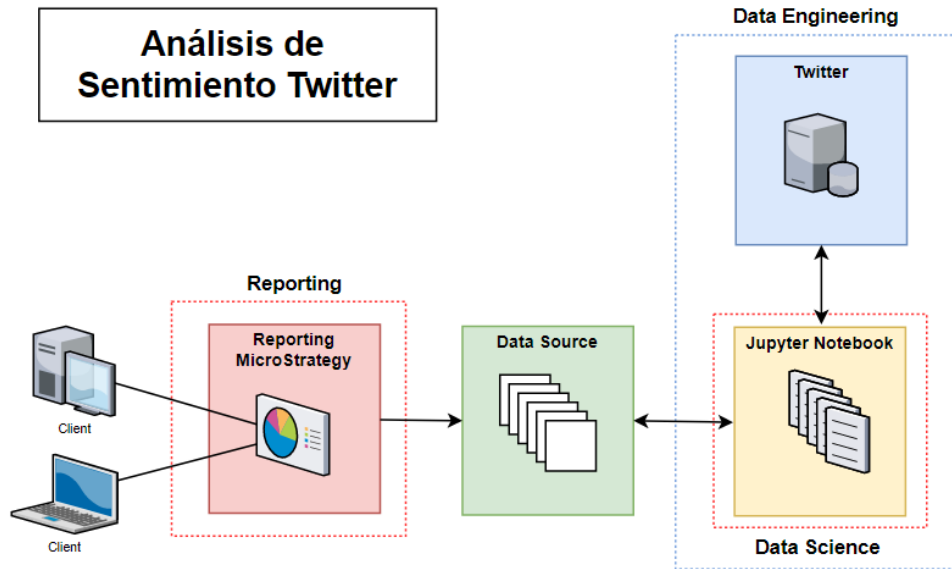
En España, son más de 4 millones de personas registradas en la plataforma.



Actualmente, existen muchos tipos de proyectos que se dediquen al análisis del sentimiento de las redes sociales en la actualidad. Sin embargo, en este TFM, se propone el extraer toda esa información y representarla en un cuadro de mando donde el usuario pueda, de manera rápida, sencilla y dinámica profundizar sobre los datos.

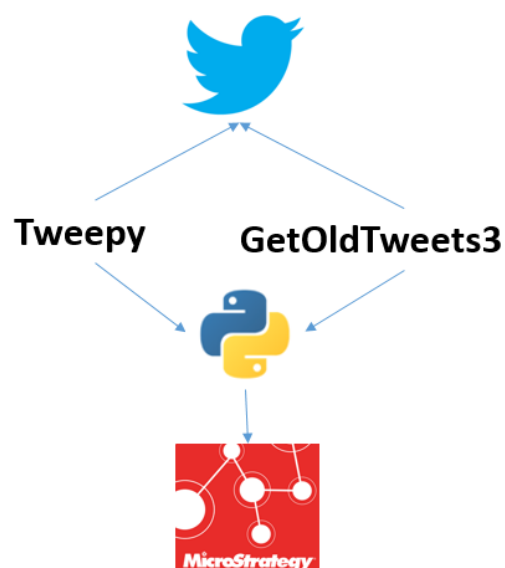
2 Alcance TFM

A continuación, se presenta un esquema de los principales componentes del proyecto:



El proyecto consta de tres partes bien diferenciadas.

- Data Engineering: Extracción de tweets de un determinado periodo de tiempo
- Data Science: Medición del sentimiento de los tweets
- Reporting: Visualización de los datos



A continuación, se detalla cada uno de los niveles:

2.1. Data Engineering

Para poder acceder a Twitter, es necesario acceder con unas credenciales que nos facilita Twitter una vez que nos hemos registrado correctamente.

#Variable de autenticación en Twitter

```
OAUTH_TOKEN='1204477616398503936-7Z5WSIa0490S7HbIoPrDzosdA8aJKb'  
OAUTH_SECRET='OCHWxZGQlrnaoNx4corTWHNZPrTgtItteYnow7nMDhtON'  
CONSUMER_KEY='IRuah0diiQDx7ayuHaFmCOvTq'  
CONSUMER_SECRET='FkYZE9vZXWMgWvovRjcqQaixPleh6ADXdD3XU05ZsIkscSFzXd'
```

Dada la limitación que tienen las cuentas gratuitas de desarrollador de Twitter, se ha utilizado la librería Python – **GetOldTweets3** [7.3] para poder recuperar los tweets con una profundidad histórica de más de una semana.

Una vez instalada y para poder extraer los datos, se ha utilizado los parámetros “desde-hasta” mediante un calendario junto con la palabra a buscar, que en este ha sido “el corte ingles”.

```
def calendar():  
    return ['2020-01-01', '2020-01-31',  
            '2020-02-01', '2020-02-29',  
            '2020-03-01', '2020-03-31',  
            '2020-04-01', '2020-04-30',  
            '2020-05-01', '2020-05-31',
```

Volumetría

El periodo a tratar en el proyecto de TFM ha sido de enero 2020 a junio 2020 con la siguiente volumetría por mes:

Mes	Volumetría
Enero	10.089
Febrero	6.931
Marzo	10.486
Abril	2.277
Mayo	8.681
Junio	9.215

Información Tweets

Los datos que se han recuperado han sido:

- ID del tweet
- Fecha de Creación del tweet
- Usuario que ha creado el tweet
- Tweet
- Geolocalización (para trabajo futuro)


```
def get_info_tweets(self, data_got, since, until):
    tweetCriteria = got.manager.TweetCriteria().setQuerySearch("\"el corte ingles\"")\
        .setSince(since)\
        .setUntil(until)

    tweets = got.manager.TweetManager.getTweets(tweetCriteria)
    for tweet in tweets:
        info_tweet = {}
        info_tweet["id"] = tweet.id
        info_tweet["date"] = tweet.date
        info_tweet["username"] = tweet.username
        info_tweet["text"] = tweet.text
        info_tweet["geo"] = tweet.geo
        data_got.append(info_tweet)

    df_tweets_aux = pd.DataFrame(data_got)
    print(since, until)
    time.sleep(60*5)
    return df_tweets_aux
```

Dado que posee una limitación de conexión abierta en GetOldTweets3, por cada iteración, se deja al procedimiento un tiempo de 5min en reposo. El resultado se almacena en un Dataframe donde se irá añadiendo los resultados de las iteraciones posteriores.

Para poder recuperar la información relativa al *El Corte Inglés*, en la actualidad, tiene 292K followers y 1.628 following.



Para extraer la información de las cuentas y realizar un análisis más exhaustivo de las cuentas followings, se utilizó la librería Tweepy [7.4] de Python.

A continuación, se expone el código para la extracción de la información las cuentas que sigue la cuenta El Corte Inglés. Para ello, primero se extrae la información de los IDs de las cuentas y posteriormente, se agrupa por grupos y se extrae la información de la cada una de las cuentas.

```
def get_following_ids(api, twitter_account='@elcorteingles'):
    while True:
        try:
            for page in tweepy.Cursor(api.friends_ids, id=twitter_account, count=5000).pages():
                following_ids.extend(page)
                #print(len(following_ids))
            break
        except tweepy.TweepError as e:
            time.sleep(60*15)
            continue
    return following_ids
```

```
def get_info_following(api, following_limit):
    data = []
    for followings in following_limit:
        following_obj=api.lookup_users(user_ids=followings)
        for user in following_obj:
            following={}
            following["id"]=user.id
            following["name"]=user.name
            following["screen_name"]=user.screen_name
            following["created_at"]=user.created_at
            following["location"]=user.location
            following["followers_count"]=user.followers_count
            following["friends_count"]=user.friends_count
            following["description"]=user.description
            data.append(following)
    return data
```

Los Dataframes generados con la información de los Tweets y de los followings son:

- *dt_tweets*
- *Info_following_el_corte_ingles*

2.2. Data Science

Una vez extraída la información de todos los tweets generados en el periodo de tiempo fijado y la información de las cuentas que sigue *El Corte Inglés*, se ha utilizado para medir el sentimiento de los tweets la librería TextBlob y NLTK.

El procedimiento que mide la subjetividad y objetividad de cada uno de ellos es el siguiente:


```
def analisis(self,df_tweets):
    polarity=0
    total = 0
    tweetPositivo = []
    tweetNeutro = []
    tweetNegativo = []
    for index,tweet in df_tweets.iterrows():

        analysis = TextBlob(tweet['Tweet_novacias'])
        polarity += analysis.sentiment.polarity
        if (analysis.sentiment.polarity ==0):
            tweetNeutro.append(tweet)
            tweet['valor'] = 0
        elif (analysis.sentiment.polarity < 0):
            tweetNegativo.append(tweet)
            tweet['valor'] = -1
        elif (analysis.sentiment.polarity > 0):
            tweetPositivo.append(tweet)
            tweet['valor'] = 1

        df_tweets.loc[index,'valor'] = tweet['valor']
        df_tweets.loc[index,'subjectividad'] = analysis.sentiment.subjectivity
        df_tweets.loc[index,'objetividad'] = analysis.sentiment.polarity

        #df_tweets_final=df_tweets_final.append(tweet)
        total += 1

    return df_tweets
```

A dicho método, se le pasa como parámetro el Dataframe obtenido de la fase anterior que contiene toda la información de los tweets. Pero antes de pasar el dato “puro” se ha procedido a realizar una limpieza del dato. Para ello, se han creado las siguientes funciones:

- Conversión de mayúsculas a minúsculas.
- Traducción de los tweets al español.
 - Se procede a traducir todos los tweets a una única lengua debido a que se aprecia información de diferentes lenguas (inglés, catalán, etc)
- Eliminación de signo de puntuación.
 - Se eliminan todos los puntos de puntuación, pero además se procede a eliminar:
 - Números
 - Palabras claves como RT y #
 - URLs
- Tokenización de los tweets.
 - Se almacena en un objeto de tipo lista las palabras resultantes de la primera fase de limpieza de los datos.
- Eliminación de palabras vacías.
 - Se eliminan aquellas palabras vacías que no aportan valor al análisis del sentimiento de los tweets. Para ello, se ha utilizado la librería NLTK.

```
def eliminar_signos(self, text):
    text = "".join([char for char in text if char not in string.punctuation])
    text = re.sub('[0-9]+', '', text)
    text = re.sub('@[A-Za-z0-9]+', '', text) # borrado menciones
    text = re.sub('#', '', text) # borrado '#'
    text = re.sub('RT[\s]+', '', text) # borrado RT
    text = re.sub('https?:\/\/\S+', '', text) # borrado url
    return text

def tokenization(self, text):
    text = re.split('\W+', text)
    return text

def palabras_vacias(self):
    return nltk.corpus.stopwords.words('spanish')

def quitar_palabras_vacias(self, text):
    text = [palabra for palabra in text if palabra not in self.palabras_vacias()]
    return text
```

Para poder extraer la valoración que representa el sentimiento de un tweet, se ha utilizado la función *sentiment.polarity* incorporada en la librería TextBlob. Para ello, para cada una de las filas del dataset se ha calculado la objetividad de cada uno de ellos, clasificando el resultado en:

- -1: Valoración Negativa
- 0: Valoración Neutra
- 1: Valoración Positiva.

Estos valores, luego serán ponderados en el cuadro de mando de MicroStrategy representando el valor real.

2.3. Reporting

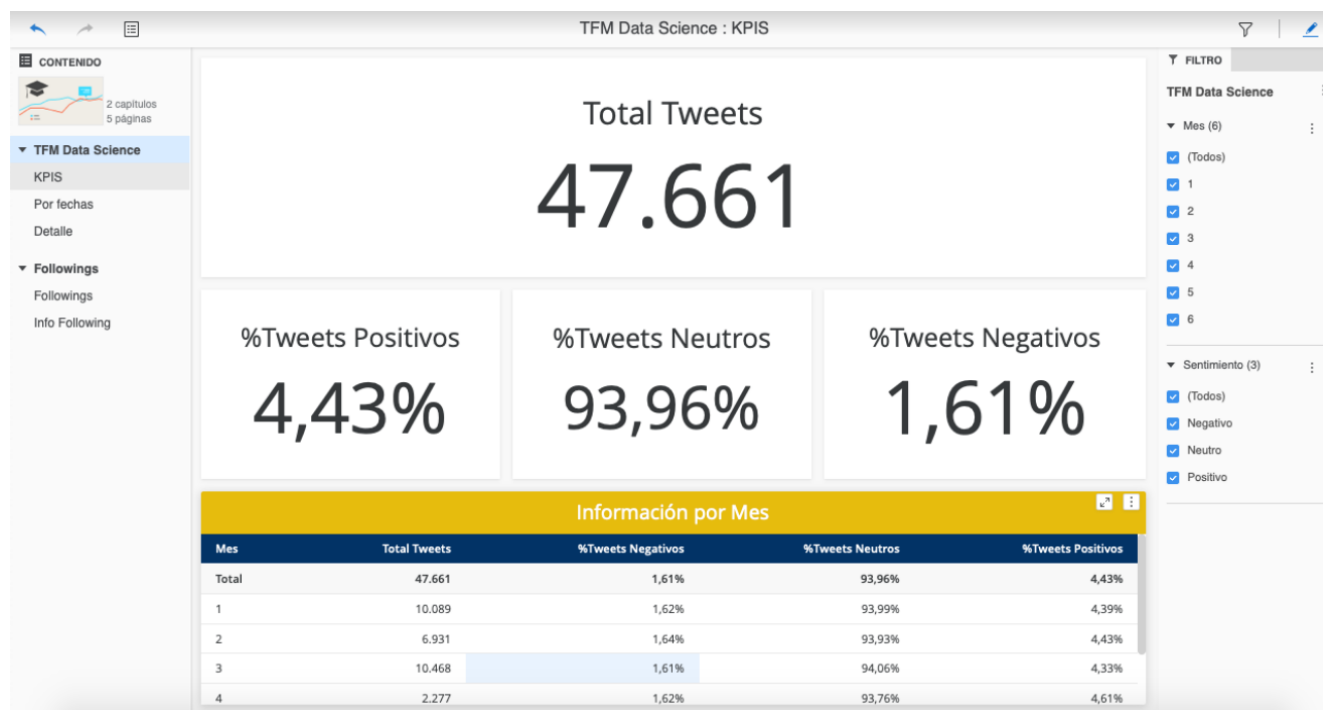
La herramienta que se ha utilizado para presentar los datos ha sido MicroStrategy ^[7.7]. Este software empresarial te permite crear de forma rápida y sencilla, un cuadro de mando.

El cuadro de mando se compone de dos capítulos:

- Análisis Temporal.
 - KPIS principales
 - Por Fechas
 - Detalle
- Análisis Followings.
 - Followings
 - Info Followings.
- Análisis Covid

- KPIS principales
- Por Fechas
- Detalle

El origen del cuadro de mando son ficheros CSV generados por los notebooks de Python. Dichos ficheros se encuentran disponibles en el repo del proyecto [7.5]



2.4. Ejecución del Programa

Para poder ejecutar el programa, el único notebook que se tiene que ejecutar es:

`tfm_Data_Science.ipynb`

Este a su vez, irá convocando al resto de notebooks que contienen los métodos necesarios para su correcta ejecución.

Consta de varios flujos de ejecución:

- **Online=1**
 - Conexión mediante GetOldTweets3 y descarga de las fechas disponibles en el calendario.
- **Online=0 y all_tweets=1**
 - Lectura de los ficheros CSVs ya almacenados en el directorio CSVs sin conexión a Twitter.
- **Online=0 y all_tweets=0**
 - Lectura de un único fichero.
- **Online_user = 1**

- Conexión mediante Tweepy a Twitter para recuperar la información de las cuentas following.

Método Principal - TFM Data Science

```
if __name__ == '__main__':

    #Control horario programa
    fecha_inicio = datetime.now().strftime("%m/%d/%Y, %H:%M:%S")
    print('Comienzo del programa: ', fecha_inicio)

    # Definición de variables
    fichero = 'CSVs/info_tweets_el_corte_ingles_202003.csv'
    pattern = 'CSVs/info_tweets_el_corte_ingles*.csv'
    fichero_user = 'CSVs/following_info_el_corte_ingles.csv'
    fichero_tweets = 'CdM/Tweets_Info.csv'
    fichero_fact = 'CdM/Tweets_Fact_Info.csv'
    fichero_covid = 'CdM/Tweets_Covid_Info.csv'

    following_ids = []
    calendario = calendar()
    df_tweets, df_tweets_final, df_following, df_fact_following = creacion_DataFrame()

    #Flujo de la información
    online = 0
    all_tweets = 0
    online_user = 0
```

3 Investigación

A continuación, se detallan cada uno de los componentes de investigación del TFM:

3.1. Data Engineering

Al conectarnos a Twitter mediante la librería de Python Tweepy, observamos algunas limitaciones de la API de Twitter. Algunas de éstas restricciones ^[7.6] son las siguientes:

- Acceso a Tweet de más de 7 días.
- Limitación de recuperar información de cuentas following/followers.
- Timeout de la conexión.

Para solventar algunos de estos problemas, nos hemos apoyado en las siguientes soluciones:

```
auth = tweepy.OAuthHandler(CONSUMER_KEY, CONSUMER_SECRET)
auth.set_access_token(OAUTH_TOKEN, OAUTH_SECRET)
api = tweepy.API(auth, wait_on_rate_limit=True, wait_on_rate_limit_notify=True, timeout=3600)
```

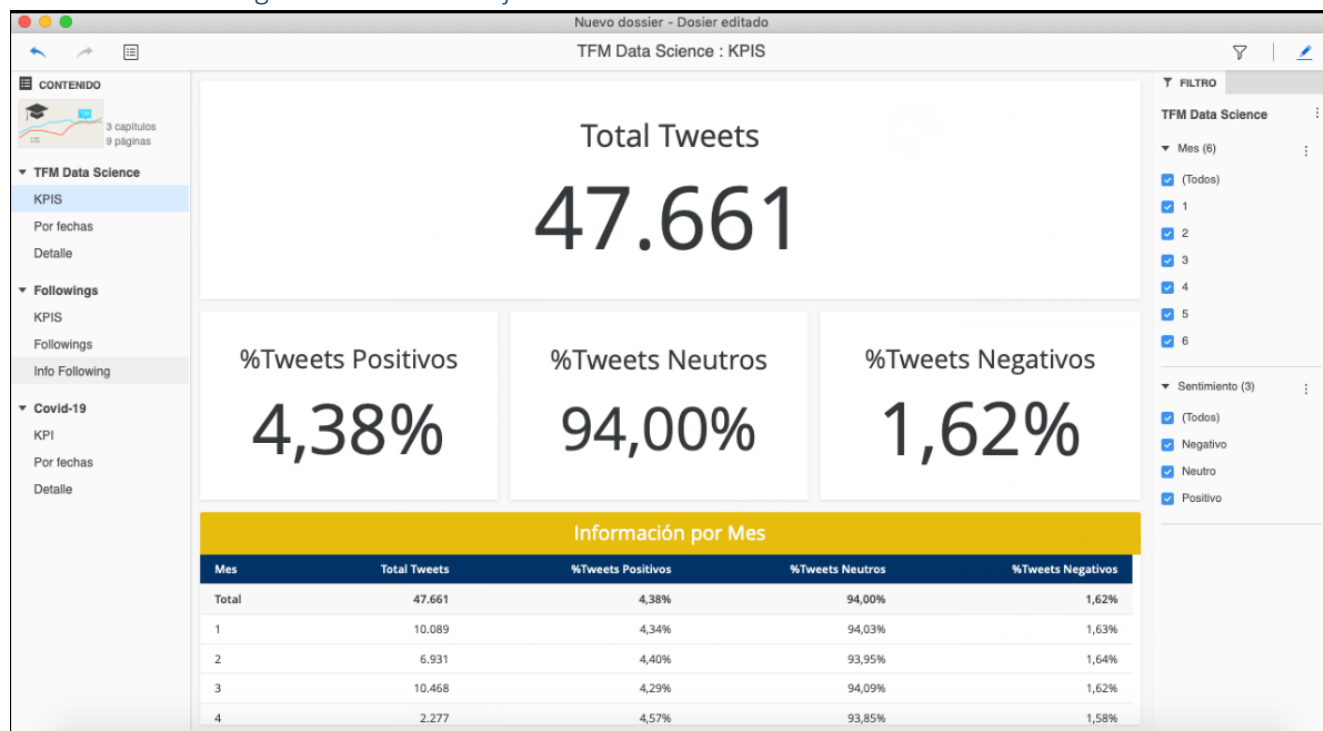
Inicializando estos parámetros, nos ayuda a que sea la propia API quien gestione los posibles errores de ejecución. Sin embargo, algunos códigos los hemos incluido dentro de una Try/Catch dejando un tiempo de reposo.

En el caso de la limitación de aquellos tweets con más de siete días de antigüedad, como ya se ha explicado en el apartado de Alcance TFM, se ha procedido a utilizar la librería de Python GetOldTweets3.

4 Resultados

Una vez terminado el proceso de extracción y análisis de los tweets generados con la etiqueta *El Corte Inglés* para un determinado rango de fechas, el cuadro de mando arroja los siguientes resultados.

Análisis del estado general de enero a junio del 2020

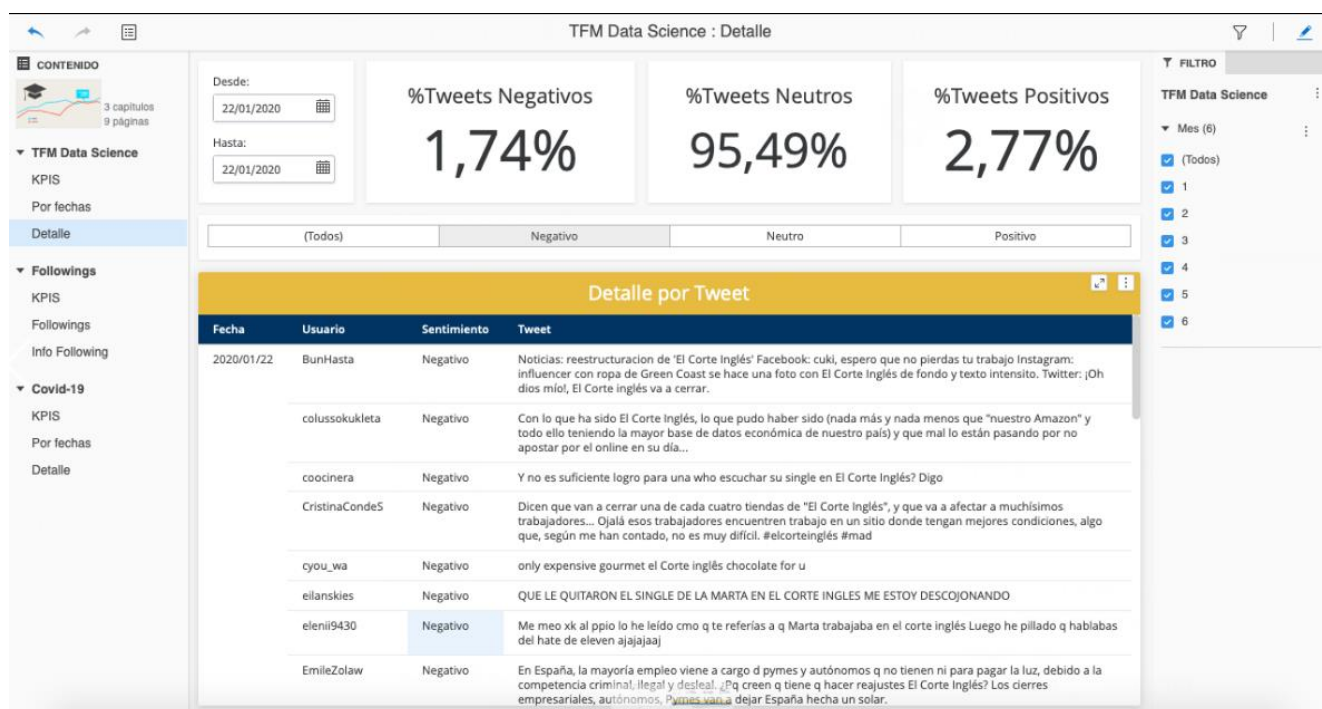
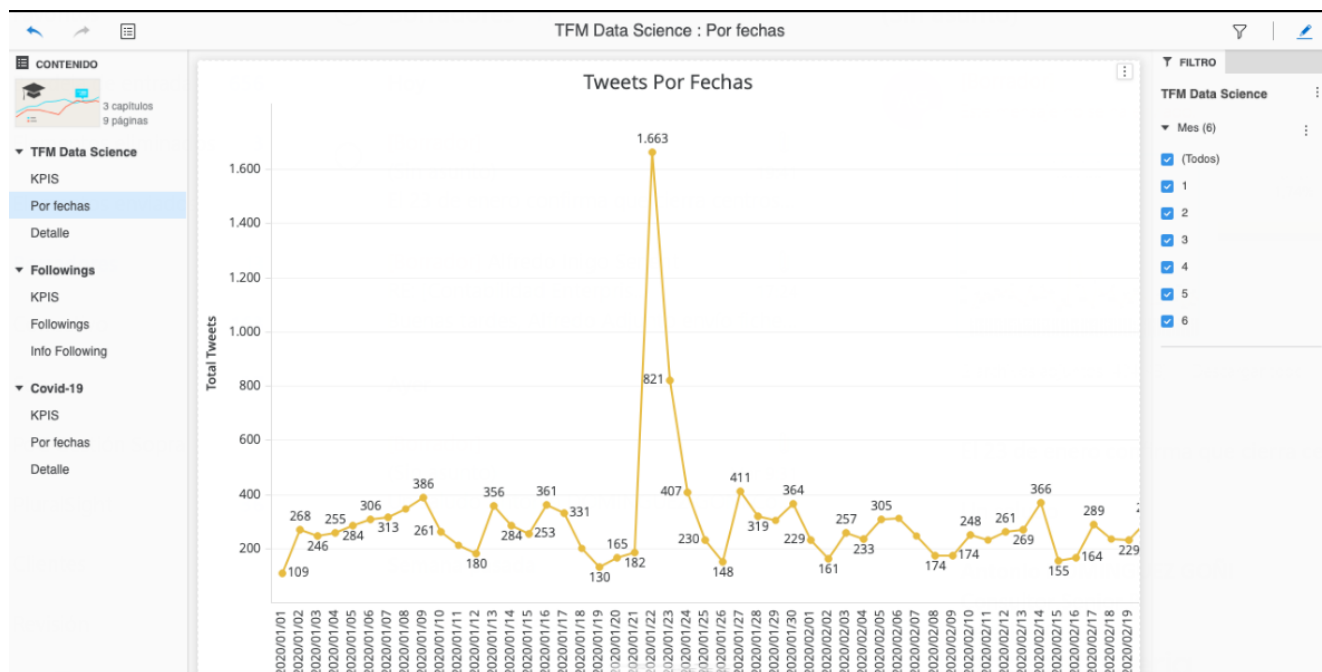


En términos generales, la opinión de la red social ha sido bastante neutra, incluso siendo positiva para algunos usuarios (representando el 5%) y prácticamente sin comentarios negativos.

También como se puede apreciar el número de Tweets disminuye a partir de febrero debido a la finalización de la campaña de navidad y las rebajas. Sin embargo, se vuelve a apreciar un incremento de Tweets en el mes de marzo, superando incluso al mes de enero, debido a la crisis sanitaria del Covid-19.

Otro dato que se aprecia es para el día 23/01 el gran impacto que tuvo en la red social el anuncio del cierre de algunos centros comerciales en España donde se ve un pequeño despunte de Tweets con sentimiento negativo. A continuación, se expone la gráfica donde se ve el gran número de Tweets y el sentimiento.

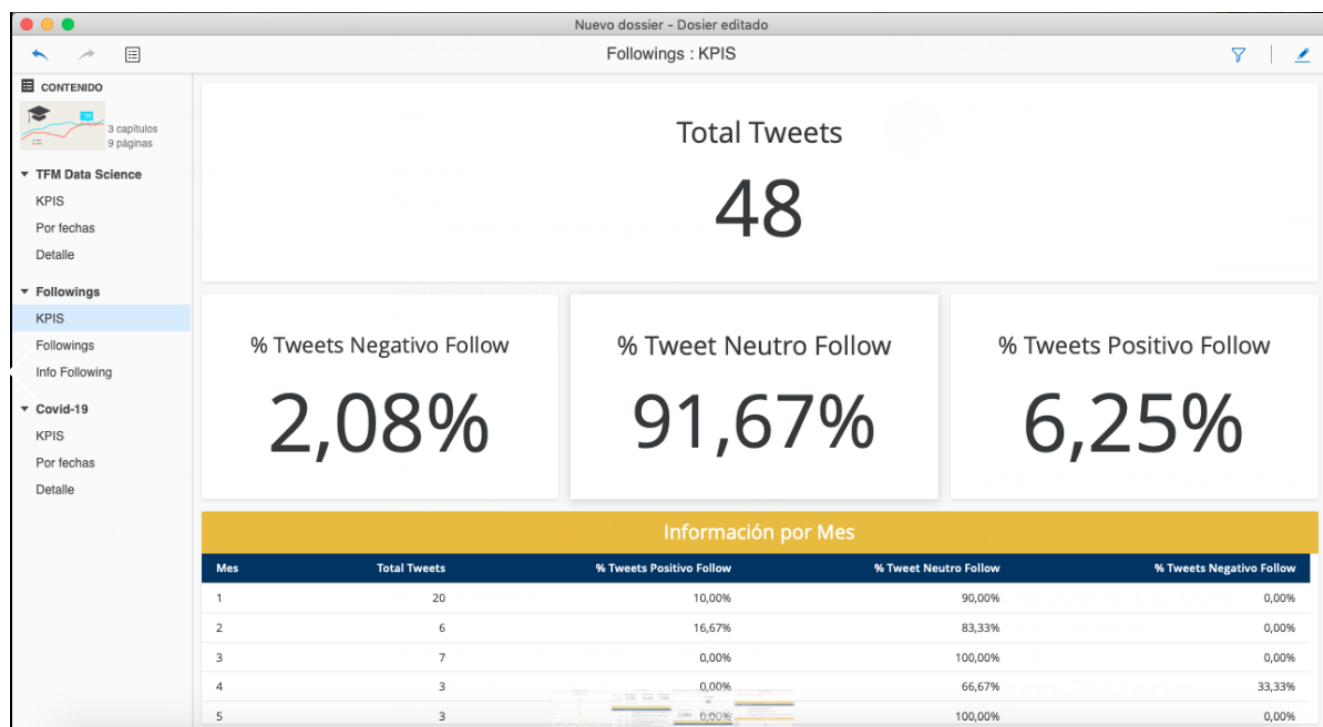
24 de abril de 20190



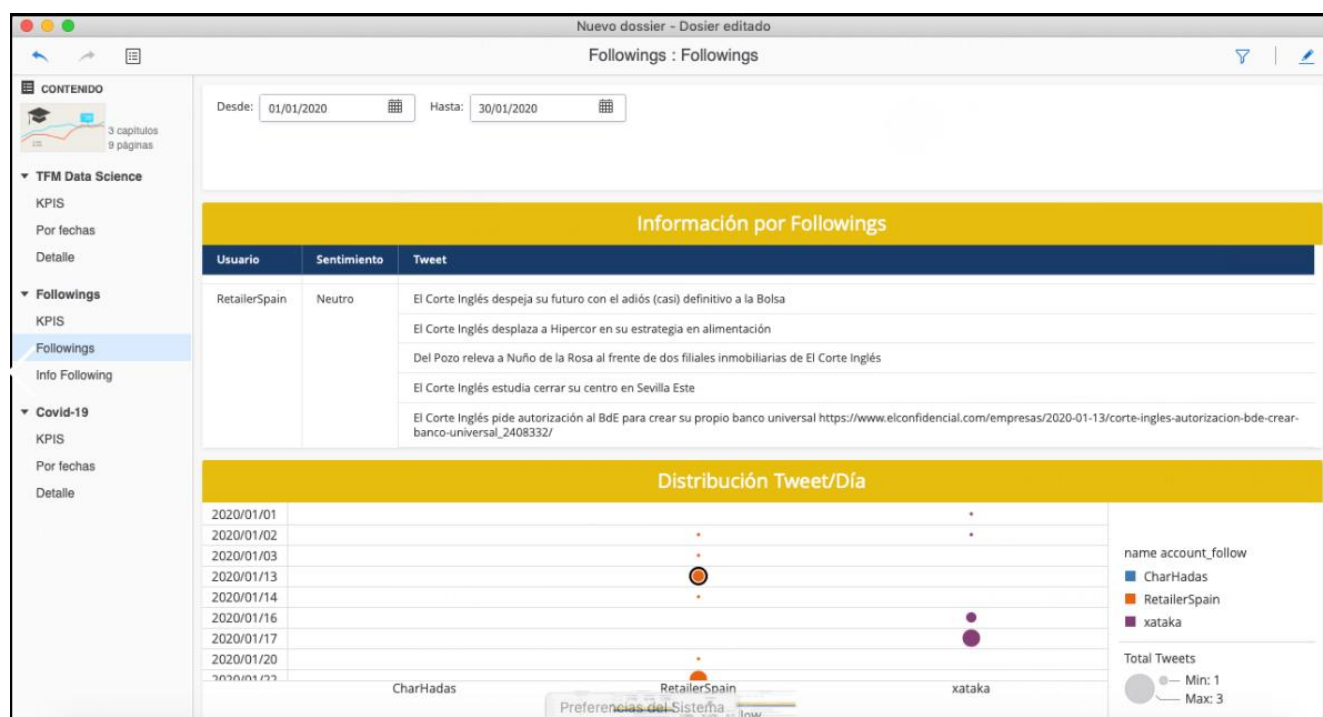
Análisis opinión cuentas Following

Para mi empresa, puede ser muy interesante la información de las cuentas que hago following para poder destinar recursos a dichas cuentas. Preocupa la poca actividad de dichas cuentas una vez pasado la campaña de Navidad y rebajas y periodo de confinamiento. Aunque la actividad es poca, a opinión de dichas cuentas, sigue la misma tendencia que el conjunto total de Tweets pero sí se ve un pequeño incremento de comentarios positivos (cerca del 7%)

24 de abril de 20190



En el siguiente gráfico, vemos la actividad por día de dichas cuentas para el mes de enero.



Y para complementar la información, se muestra la información de cada una de las cuentas para poder comparar las cuentas.

24 de abril de 20190

Nuevo dossier - Dossier editado

Followings : Info Following

CONTENIDO

3 capítulos
9 páginas

TFM Data Science

KPIS

Por fechas

Detalle

Followings

KPIS

Followings

Info Following

Covid-19

KPIS

Por fechas

Detalle

Información Following

ID Cuenta	Usuario	NIK Usuario	Followers	Following	Creación Cuenta
44739010	chicadelatele	chicadelatele	6862.0	184.0	2007-04-13 08:53:35
182004000	xataka	xataka	1407454.0	45.0	2008-12-17 21:28:48
363972740	CincoDiascom	CincoDiascom	347861.0	536.0	2009-04-29 16:19:02
471007240	KiehlsSpain	KiehlsSpain	18040.0	372.0	2009-06-14 14:32:22
1063860770	CharHadas	CharHadas	12954.0	830.0	2010-01-19 11:36:40
2298240810	Sportown	Sportown	1041.0	834.0	2010-12-23 12:32:12
12412170900	Biotherm_es	Biotherm_es	5393.0	341.0	2013-03-04 12:20:21
21831873860	CarlosOnRetail	CarlosOnRetail	1175.0	655.0	2013-11-08 23:24:37
26142620960	RetailerSpain	RetailerSpain	2543.0	1637.0	2014-07-09 21:38:20

Análisis sobre COVID-19

Nuevo dossier - Dossier editado

Covid-19 : KPIS

CONTENIDO

3 capítulos
9 páginas

TFM Data Science

KPIS

Por fechas

Detalle

Followings

KPIS

Followings

Info Following

Covid-19

KPIS

Por fechas

Detalle

Total Tweets Covid

1.696

%Tweets Negativo Covid

1,65%

%Tweets Neutros Covid

94,46%

%Tweets Positivos Covid

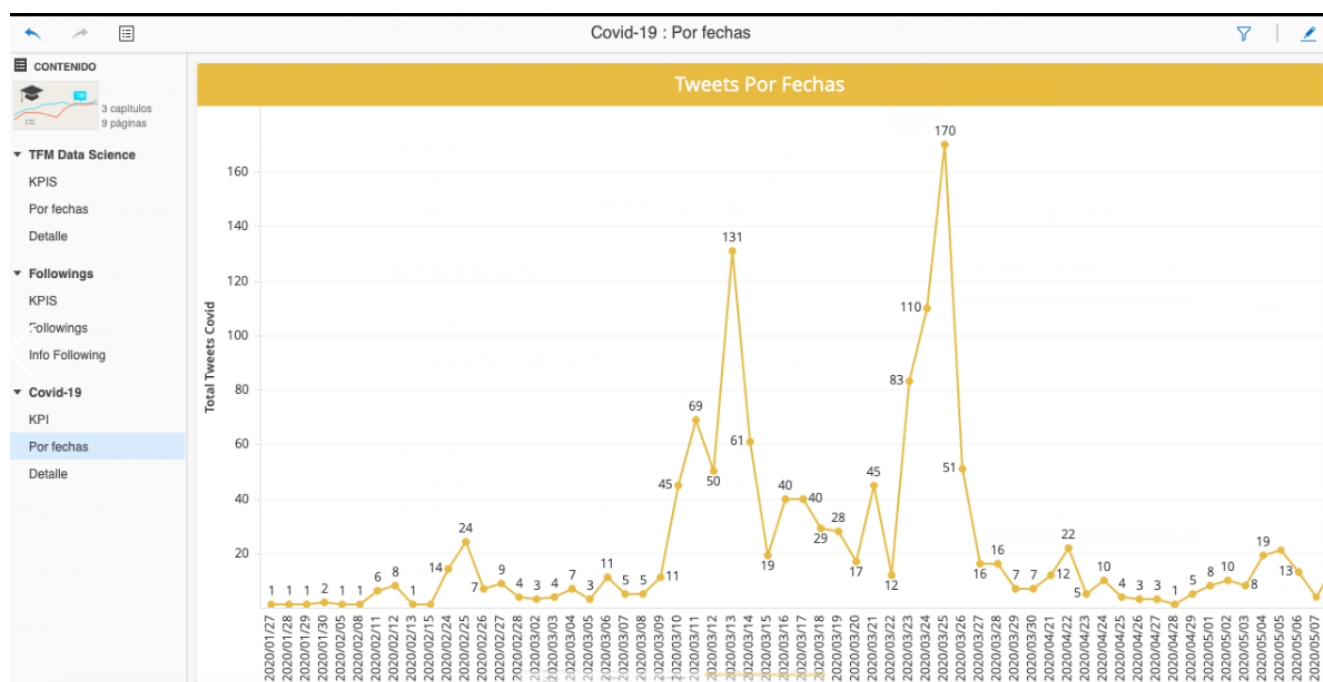
3,89%

Información por Mes

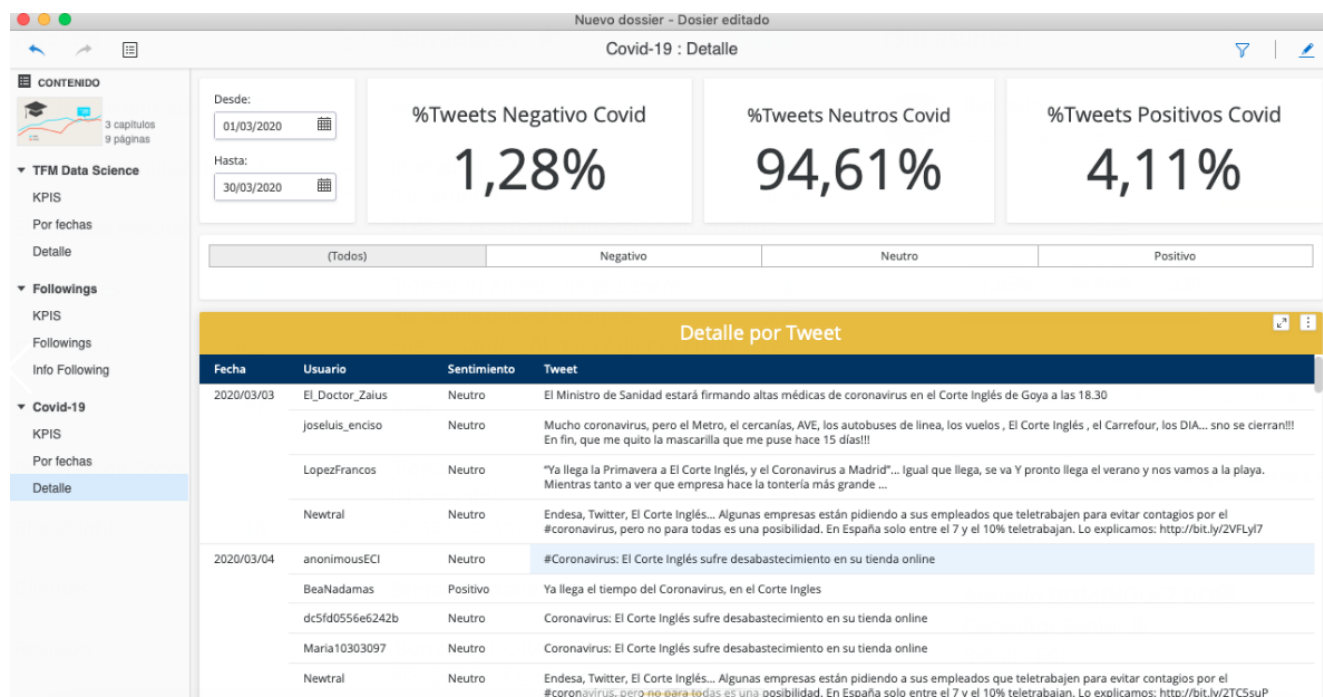
Mes	Total Tweets Covid	%Tweets Positivos Covid	%Tweets Neutros Covid	%Tweets Negativo Covid
1	5	0,00%	100,00%	0,00%
2	76	7,89%	92,11%	0,00%
3	1.095	4,11%	94,61%	1,28%
4	65	0,00%	95,38%	4,62%
5	302	4,30%	93,71%	1,99%

Profundizando en los datos del Covid, vemos un despunte de los tweets generados sobre los días 13/03 al 25/03.

24 de abril de 20190



Con mejor opinión de los Tweets en Negativo.



Dichos días coinciden con:

- 13/03/2020: Estado de alarma. Confinamiento.



Agencia Estatal Boletín Oficial del Estado

Castellano ▼


Buscar 🔍

Mi BOE 👤

Menú ☰

Está Ud. en > [Inicio](#) > [BOE](#) > [Calendario](#) > [14/03/2020](#) > Documento BOE-A-2020-3692

Real Decreto 463/2020, de 14 de marzo, por el que se declara el estado de alarma para la gestión de la situación de crisis sanitaria ocasionada por el COVID-19.

 [Ver texto consolidado](#)

- 25/03/2020: Publicación por parte de El Corte Inglés que aplica ERTE en su organización

El Corte Inglés

[Quiénes somos](#) ▼ [Información financiera](#) ▼ [Gobierno corporativo](#) ▼ [RSC](#) ▼ [Comunicación](#) ▼ [Talento](#) ▼



25/03/2020 | Empleo / Resultados económicos

[Descargar noticia](#)

[Descargar recursos gráficos](#) ↓

La empresa se acoge a esta medida habilitada por el Gobierno para aquellas empresas que, por causa de fuerza mayor, han debido cesar su actividad tras la declaración del estado de alarma.

5 Conclusión

La conclusión de este TFM presentar de manera clara y sencilla al usuario lo que está opinando las redes sociales sobre su organización, profundizando a nivel de Tweet, cuenta, día y opinión.

Teniendo accesible dicha información, los usuarios que se encarguen del análisis de esta información podrán actuar de manera eficaz para poder identificar la raíz del problema de una manera rápida y concisa y destinar los recursos necesarios para mejorar la opinión pública.

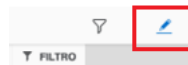
Respecto a los datos que devuelve el algoritmo de TextBlob para obtener el sentimiento de los tweets, es cierto que el volumen de tweets neutros es muy considerable, considerando el aplicar otro algoritmo de machine learning para su tratamiento y comparando resultados.

6 Interfaz Gráfica

La forma de interactuar con el cuadro de mando es muy sencilla. Una vez ejecutado, consta de 3 capítulos y varias páginas cada uno de ellos.

- Capítulo: Análisis Temporal.
 - Pág: KPIS principales
 - Pág: Por Fechas
 - Pág: Detalle
- Capítulo: Análisis Followings.
 - Pág: Followings
 - Pág: Info Followings.
- Capítulo: Análisis Covid
 - Pág: KPIS principales
 - Pág: Por Fechas
 - Pág: Detalle

Para ver el cuadro de mando ejecutado, se tendrá que ejecutar mediante el botón de la esquina superior derecha.



Para poder filtrar sobre los datos, consta de dos tipos de filtros.

- 1- Filtros de capítulo: Aplica a todos los objetos de ese capítulo
- 2- Filtro de página: Aplica filtros sobre esa página.

7 Anexo

Para más información https://es.wikipedia.org/wiki/El_Corte_Ingl%C3%A9s

Para más información <https://es.wikipedia.org/wiki/Twitter>

Para más información <https://pypi.org/project/GetOldTweets3/>

Para más información <http://docs.tweepy.org/en/latest/>

El repositorio de código se encuentra en la siguiente URL: [Github.com/antdomgoni/tfm](https://github.com/antdomgoni/tfm)

Información sobre las limitaciones API Twitter:

<https://developer.twitter.com/en/docs/twitter-api/v1/rate-limits>

7.7. MicroStrategy

MicroStrategy: Software empresarial que permite crear informes, documentos y tableros sobre datos almacenados en diferentes bases de datos, CSVs y otras fuentes para posteriormente realizar análisis sobre los mismos.

Wikipedia: <https://es.wikipedia.org/wiki/MicroStrategy>

Para poder descargar MicroStrategy Desktop se puede realizar desde la siguiente URL:

<https://www.microstrategy.com/es/get-started/desktop>

MicroStrategy Intelligence Everywhere

Producto Soluciones Servicios Recursos Empresa

EMPEZAR AHORA EXPERT.NOW

PRIMEROS PASOS | Desktop

La empresa inteligente comienza con usted.

¿Está listo para poner a trabajar sus datos? MicroStrategy Desktop ofrece todo lo que necesita para acceder, visualizar y analizar sus datos, de forma gratuita. Sin claves de licencia. Sin versión de prueba. Simplemente análisis rápidos, flexibles y de autoservicio para ayudarlo a maximizar el impacto de sus datos,

Descargue MicroStrategy Desktop gratis hoy mismo.

NOMBRE

APELLIDOS

EMAIL CORPORATIVO

TELÉFONO CORPORATIVO

Leave us a message.