Escola d'Enginyeria de Telecomunicació i Aeroespacial de Castelldefels
UNIVERSITAT POLITÈCNICA DE CATALUNYA

TREBALL FINAL DE GRAU

Título: Captura y análisis de datos para inferir el estado de una ciudad

Autor: Miquel Sánchez Omenac

Director: José Manuel Yúfera

Fecha: 08 de setiembre del 2017

AGRADECIMIENTOS

Antes de empezar este TFG, quiero agradecer el soporte y la ayuda de las diversas personas que han estado imprescindibles durante estos meses para llevar a cabo este trabajo.

En primer lugar, quiero agradecer a mi tutor, José Manuel Yúfera, su plena disponibilidad, la orientación constante, el positivismo, la confianza que ha depositado en mí y su criterio, que han sido esenciales.

Es segundo lugar, quiero dar las gracias a Borja Ràfols, que gracias a su visión más profesional en el lenguaje computacional me ha ayudado mucho a la hora de escribir el código de mi programa.

Por último, quiero agradecer el soporte de mis familiares, amigos y compañeros por su empatía y predisposición a la hora de ayudarme en el desarrollo de mi trabajo.

RESUMEN

En este trabajo realizaré un estudio con ayuda de una aplicación previamente diseñada y realizada que se encargue de la obtención, junto a un análisis de tweets delimitados en una geolocalización determinada, en este caso la localización será la ciudad de Barcelona.

Para lograr esto utilizaré la minería de datos, que permite obtener información de la red social Twitter. Junto con un análisis que proporcionará una idea aproximada del estado anímico de la ciudad condal.

Palabras clave: Geolocalización, minería de datos, análisis, Twitter, Barcelona.

RESUM

En aquest treball realitzaré un estudi amb ajuda d'una aplicació prèviament dissenyada i realitzada que s'encarregui de l'obtenció, juntament amb un anàlisi de tweets delimitats en una geolocalització determinada, en aquest cas la localització serà la ciutat de Barcelona.

Per aconseguir això utilitzaré la mineria de dades, que permet obtenir informació de la xarxa social Twitter. Juntament amb una anàlisi que proporcionarà una idea aproximada de l'estat anímic de la ciutat comtal.

Paraules clau: Geolocalització, minería de dades, anàlisi, Twitter, Barcelona.

ABSTRACT

In this work I will make a study with the help of a previously designed and performed application that is in charge of obtaining and analysing tweets delimited in a certain geolocation, in this case the location will be the city of Barcelona.

To achieve this I will use data mining, which allows to obtain information from the social network Twitter. Then I will do an analysis that will provide an approximate idea of the mood of the city.

Keywords: Geolocation, data mining, analysis, Twitter, Barcelona.

ÍNDICE DE CONTENIDO

INDICE DE FIGURAS

1. INTRODUCCIÓN

Las redes sociales son estructuras donde la gente puede comunicarse de forma virtual. Los usuarios pueden compartir información, enviar mensajes, publicar fotos y videos, etc.

Grandes empresas y negocios han puesto sus focos en estas herramientas virtuales para darse a conocer o conseguir publicidad extendiéndose a través de las redes.

En este proyecto me voy a focalizar en la red social Twitter, ésta permite la emisión de mensajes de un máximo de 140 caracteres y cuenta con más de 328 millones de usuarios activos.[1]

El objetivo del proyecto consiste en aplicar técnicas para extraer datos de la red social y después analizar los mensajes extraídos a partir de unos filtros determinados. Los usuarios registrados, llamados twitteros, pueden seguir a otros con el fin de recibir las publicaciones que estos realicen. Además, un usuario puede utilizar hashtags, mencionar a otros usuarios, o retuit de los mensajes que le parezcan interesantes. Por eso voy a buscar aquellos mensajes que vayan acompañados con hashtags que estén relacionados con el estado de animo de las personas y con una ubicación determinada.

Para obtener los tweets voy a utilizar la minería de datos. Este proceso se define como el descubrimiento de nuevas y significantes relaciones, patrones y tendencias al examinar grandes cantidades de datos. Esto lo voy a hacer con la ayuda del programa informático *iPhyton* utilizando la librería *tweepy* para vincular la aplicación con el código.

1.1 Motivación del proyecto

Hace ya años que las redes sociales se han convertido en uno de los medios de comunicación más importantes y extendidos.

Actualmente, aproximadamente el 50% de la población mundial dispone de acceso a internet. Cada vez más, los usuarios de estas redes sociales se alejan del uso cotidiano y las grandes empresas, gracias a la minería de datos extraen información para predecir ventas, tendencias y comportamientos o para analizar diversos sectores tanto de la sociedad como del sector empresarial.

[1] https://about.twitter.com/es/company

1.2 Objetivos

A continuación describo los objetivos de este proyecto:

1. **Selección de las herramientas de trabajo**: Lo primero que deberé hacer será realizar un análisis para escoger las herramientas con las que voy a trabajar, desde lenguajes de programación, *APIs* de descarga de mensajes y consultas, hasta gestores de bases de datos.

2. **Geolocalización:** Debo buscar aquella zona geográfica dónde la obtención de los mensajes cumpla unas condiciones determinadas para el proyecto.

3. **Extracción de tweets**: Tras haber escogido que librería voy a utilizar para trabajar con la API de Twitter, procederé a extraer aquellos mensajes que me interesen.

4. **Filtrado a partir de unas prioridades**: En este punto trataré de filtrar los mensajes obtenidos para después quedarme con aquellos tweets que contengan las palabras que me interesan y almacenarlos en una tabla.

5. **Análisis y conclusiones:** Analizaré los mensajes seleccionados para agruparlos en dos grupos: mensajes positivos y negativos. Finalmente, y como resultado de todo el trabajo, extraeré las conclusiones acerca de los análisis realizados.

2. ESTADO DEL ARTE

En este apartado de mi proyecto introduzco el estado del arte, dando a conocer estudios ya realizados sobre la minería de datos en Twitter.

2.1 Twitter y la minería de datos[2]

Tal y como todos sabemos twitter es una red social con servicio de *microblogging*, fundada en 2006 en California.[3] Por ello y por el gran volumen de usuarios de esta red social, he decidido que utilizaré esta plataforma social para realizar la minería de datos.

Actualmente, Twitter[4] es una de las fuentes más utilizadas por grandes marcas y empresas que utilizan esta red para saber intereses que van apareciendo entorno a las nuevas generaciones, además de utilizarlo para saber las opiniones de los usurarios sobre sus productos. Los tweets relacionados con estas marcas van desde el sector automovilístico hasta el textil pasando por sectores como el hostelero, logístico, etc.

Además, a través de los años ha ido apareciendo el concepto de *Twitstars*, usuarios no famosos que se convirtieron en estrellas en el ámbito de Twitter y han conseguido crear cuentas con miles de seguidores. Estos usuarios se dedican a publicar opiniones sobre todo tipo de temas en la red social.

Hace apenas unos años han aparecido varios estudios relacionados con Twitter que se dedican a estudiar posibles brotes de enfermedades en una geolocalización determinada.

La universidad de *Northwestern*[5], basándose en esta idea ha desarrollado un programa que a través de la lectura de tweets es capaz de detectar brotes de gripe. Este sistema va extrayendo mensajes o tweets continuamente relacionados con enfermedades utilizando la técnica de minería de datos para acabar mostrando gráficos sobre los posibles contagios.

---

[2] https://blog.es.logicalis.com/analytics/mineria-de-datos-aplicaciones-que-ya-son-una-realidad

[3] http://www.cad.com.mx/historia_de_twitter.htm

[4] https://www.importancia.org/twitter.php

[5] http://www.lavanguardia.com/salud/20131213/54395494664/recrean-como-pandemia-podria-propagarse-mundo.html

En marzo de 2015, gracias a la minería de datos se encontraron a los más fervientes seguidores de ISIS[6] y así facilitar a la policía la ubicación exacta de estos individuos en caso de tener indicios de pertinencia al grupo terrorista.

En mi caso voy a utilizar la minería de datos para obtener una idea del estado anímico de una comunidad determinada estudiando y analizando los tweets que suben los usuarios en una zona en concreto.

2.2 Hashtags por analizar

El hecho de conocer el estado anímico de una comunidad en concreto permite hacer una idea a políticos, comercios, servicios y demás para saber si realmente están haciendo bien su trabajo.

La firma *GfK Custom Research*[7] realizó el pasado 2013 un estudios basado en encuestas a 10.000 personas de 29 países diferentes con el objetivo de hacer un ranking de las ciudades más felices.

Otros estudios realizados rebelan que los atractivos culturales, la oferta comercial, los servicios y el clima de una ciudad resultan claves para contribuir con la felicidad de una ciudad.

Por eso en este proyecto he decidido que los filtros que utilizaré, para analizar los mensajes enviados por los usuarios de Twitter, serán en forma de hashtags, pero no tan relacionados con el hecho de que las personas publiquen mensajes como #*happy* o #alegre. Sino que buscaré esos mensajes de usuarios que estén dedicando su tiempo a viajar, hacer deporte, ir al cine, conciertos, teatro, disfrutando del buen clima que hace en un sitio determinado, etc para conectar con el hecho de estar contento o alegre.

Es decir buscaré hashtags como por ejemplo #*travel*, #escapada, #sport, #*gym*, #cine, #*music*, #*sun*, #*sunny* o #sol.

Al contrario para gente triste o no tan contenta buscaré esos hashtags relacionados con atascos, días lluviosos u oscuros, rutinas, trabajo, madrugar, problemas de renfe, etc.

Algunos ejemplos podrían ser #*traffic*, #atasco, #*raining*, #lluvia, #rutina, #*work*, #*early*, #temprano, #retrasos o #*tenfe*.

[6] https://mulherolhodepeixe.wordpress.com/2015/11/22/terrorismo-internet-y-data-mining/

[7] http://www.ntn24america.com/noticia/buscando-donde-vivir-ranking-de-las-10-mejores-ciudades-del-mundo-para-vivirbuscando-donde-vivir-las-146605

2.3 ¿Cómo está una ciudad?

Actualmente se realizan muchos estudios y sondeos para hacer rankings y valoraciones de cuales son aquellos países en los que mejor se vive y más gente feliz habita.

Happy Planet Index[8] se encarga de hacer eso precisamente para después representarlo gráficamente en su página web. Según ellos, el índice de la felicidad se mide teniendo en cuenta el bienestar, la esperanza de vida, la propia satisfacción de la gente y la media del impacto ecológico propio de cada país.

Estos índices y estos rankings que muchas empresas obtienen a través de sondeos son necesarios para observar que el mundo se va convirtiendo en un sitio mejo. Sin embargo estudios actuales realizados en USA y Europa revelan que sus habitantes no consideran que sus comunidades vayan mejorando.

La siguiente fotografía muestra el mapa mundial representando con colores el índice de felicidad de cada país, dónde rojo es el más bajo y verde el más elevado.

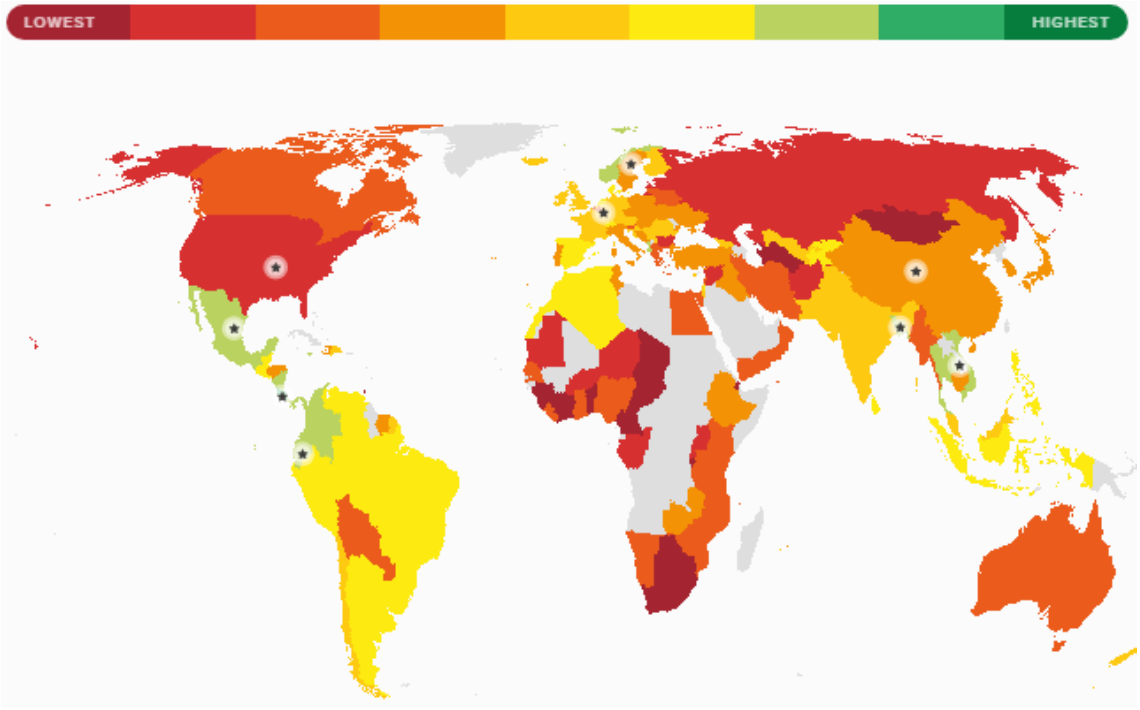


Figura 2.1 *Happy Planet Index* Score en mapa mundial. Este mapa muestra el nivel de felicidad del mundo.

[8] http://happyplanetindex.org/about/

2.4 Trabajos relacionados[9]

Durante los últimos años, el estudio y análisis de redes sociales ha acabado ocasionando grandes avances. Redes sociales como Twitter, Facebook, Instagram o *Snapchat* han abierto nuevas vías de negocio e investigación para compañías y empresas.

El análisis de opinión sobre productos o servicios de grandes empresas, donde se obtienen mensajes con opiniones tanto positivas como negativas es uno de los temas más importantes que se presentar para estudiar en un futuro muy cercano.

La minería de datos permite a grandes negocios obtener la opinión de sus clientes sin necesidad de preguntarles directamente, y este hecho se da con fotos en las redes o simplemente mensajes etiquetando las empresas o con el hecho de relacionar directamente con la ubicación de dichas empresas.

Seguidamente voy a comentar algunos proyectos o trabajos que van algo relacionados con mi estudio:

- **Twitterfall.com:** Visualiza en tiempo real varios hashtag pudiendo asignar a cada uno con un color distinto, ideal para seguir varios eventos simultáneos. En la siguiente imagen muestro el funcionamiento de la aplicación, para este ejemplo he buscado las palabras '*travel*', '*sun*' y '*raining*' de todos los mensajes de Twitter realizados en Barcelona hasta una distancia de 10 km.

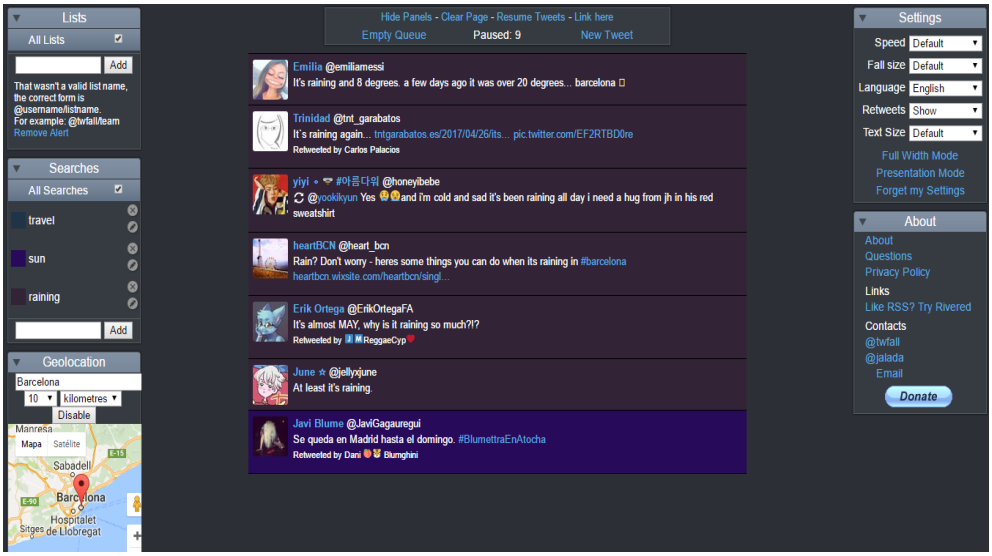


Figura 2.2 Aplicación de Twitter llamada *Twitterfall.* Muestra los '*trending topic*' en un lugar determinado diferenciados por colores.

[9] http://www.socialblabla.com/25-aplicaciones-para-twitter.html

- **TweepsMap.com:** Analiza tus seguidores de Twitter y te muestra estadísticas por ciudades, regiones o países, para saber de donde es tu comunidad mayormente.
- **Trendsmap.com**: *'Trending Topic'* locales en tiempo real en tu ciudad, o en cualquier otra ciudad o región que te interese del mundo. En la siguiente imagen muestro las palabras o hashtags más mencionados en los comentarios de los usuarios geolocalizados en Barcelona y cercanías.

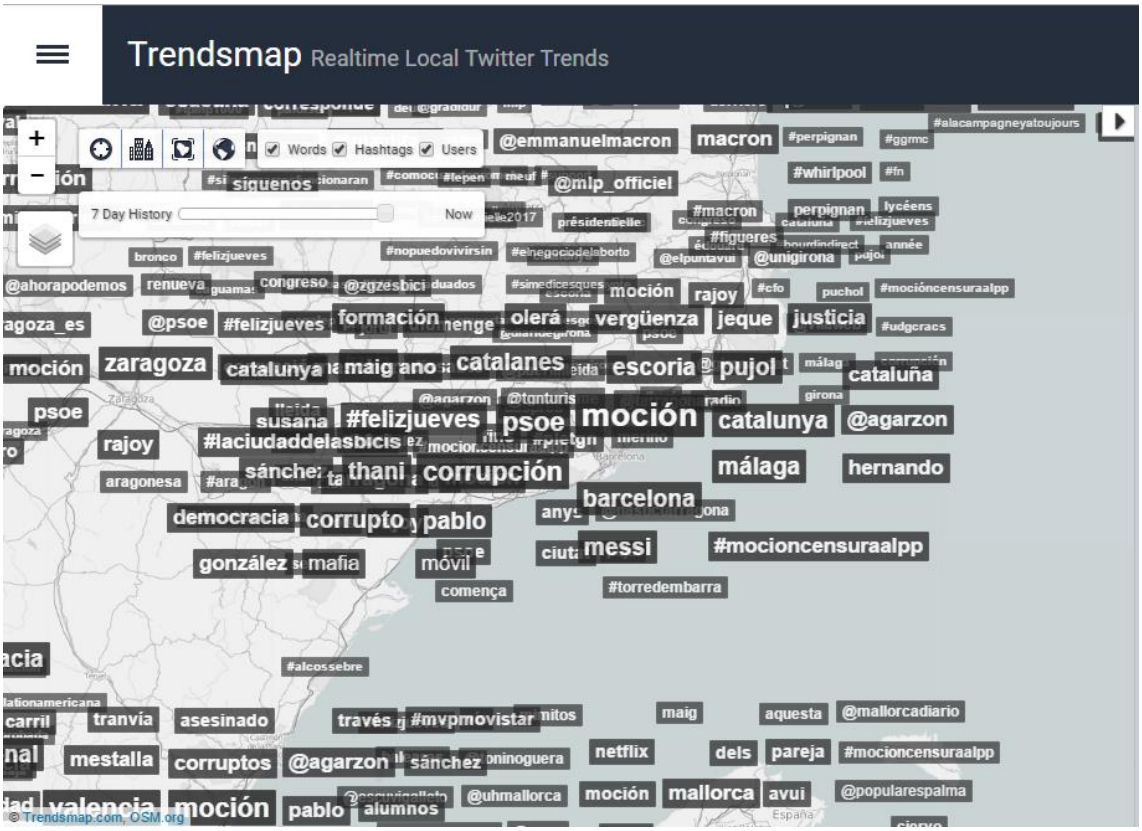


Figura 2.3 Aplicación de Twitter llamada Trendsmap

3. ARQUITECTURA DE LA APLICACIÓN

Una vez comentados los objetivos de mi proyecto y varias aplicaciones relacionadas él, voy a pasar a hablaros de twitter y de mi programa.

3.1Twitter y su API

3.1.1 ¿Qué es Twitter?[10]

Twitter es un servicio de *microblogging*, con sede en San Francisco, California, con filiales en San Antonio (Texas) y Boston (Massachusetts) en Estados Unidos. Twitter, Inc. fue creado originalmente en California, pero está bajo la jurisdicción de Delaware desde 2007.

Este servicio permite a sus usuarios enviar mensajes de texto con un máximo de 140 caracteres. Cada usuario tiene su propia página principal donde le aparecen los mensajes que han enviado todas aquellas personas a las que dicho usuario sigue.

En el recuento realizado en enero de 2016 se contaron unos 328[11] millones de usuarios activos.

En la siguiente imagen muestro la cuenta de Twitter de la Universidad Politécnica de Catalunya.

Figura 3.1 Vista desde la web del Twitter de la UPC

[10] https://es.wikipedia.org/wiki/Twitter

[11] https://about.twitter.com/es/company

3.1.2 Términos de Twitter

A continuación hago un pequeño glosario de algunas de las palabras más utilizadas dentro de la red social Twitter:

- **Tweet:** Mensaje o publicación en Twitter.
- ***Following*:** Todos aquellas cuentas o usuarios que un usuario sigue con el objetivo de informarse sobre sus publicaciones.
- ***Followers*:** Todos aquellos usuarios que siguen una determinada cuenta y leen sus publicaciones.
- ***Timeline*:** Lista de publicaciones enviadas por las cuentas que cada usuario sigue ordenadas cronológicamente
- **Retuit:** Función de la red social Twitter que permite a los usuarios volver a publicar un mensaje de otra cuenta citando el autor.

3.1.3 Información que podemos extraer de Twitter

La aplicación Twitter nos permite extraer diversos de datos de cada publicación, a continuación hago una breve explicación de que datos podemos extraer de cada tweet.

- ¿*What*? (¿Qué?): El contenido en sí de cada publicación. Puede contener además de texto, imágenes, videos, links o emoticonos.
- ¿*Who*? (¿Quién?): La persona o cuenta que ha escrito el mensaje o bien a retuitetado una publicación. Esta información contiene nombre completo y lenguaje.
- ¿*When*? (¿Cuándo?): Fecha y hora de la publicación.
- ¿*Where*? (¿Dónde?): Esta información no aparece en todas las publicaciones, es opcional. Por tanto, cada usuario decide si quiere mostrar su ubicación o no. En caso de mostrar la ubicación podemos obtener las coordenadas geográficas desde dónde se ha publicado el tweet.

En el caso de mi proyecto, en el cual me quiero centrar en una zona determinada es muy importante que uno de los filtros a la hora de escoger los mensajes sea que contengan datos geográficos.

3.1.4 Información extraíble de la API de Twitter[12]

La red social Twitter proporciona su propia API, actualmente en la versión 1.1, que permite la comunicación entre diferentes componentes software. En concreto permite controlar tu cuenta y obtener información desde código.

A continuación explico un poco las dos vías para extraer información desde Twitter:

- **API *Rest*:** Permite realizar las mismas operaciones que desde la web o aplicación. El acceso a los datos se realiza a través de un sistema en forma de caja negra realizando peticiones de tipo GET y POST.

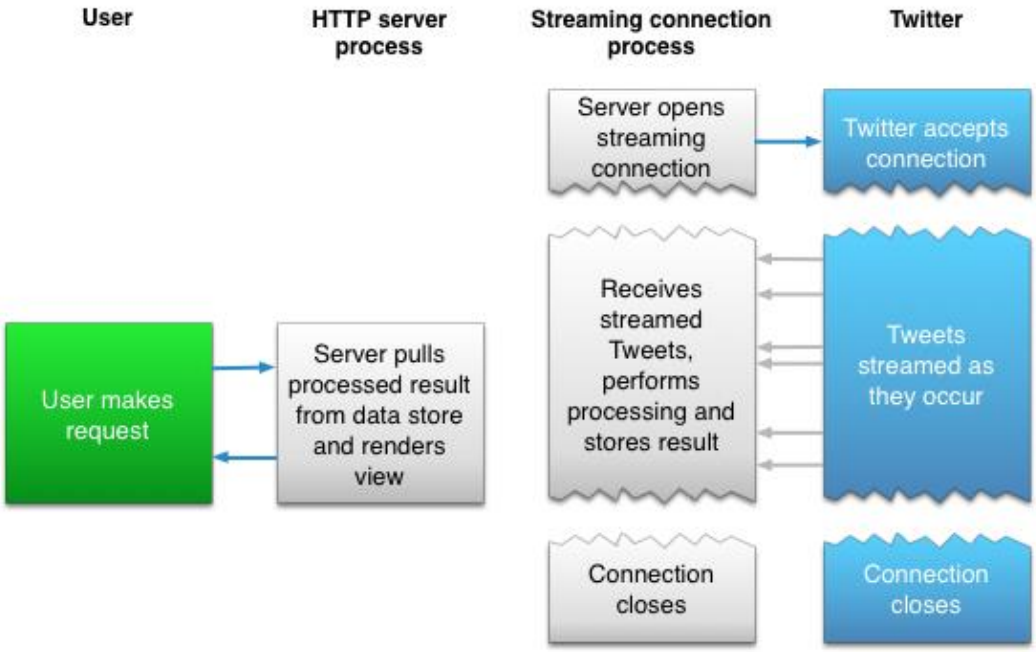


Figura 3.2 Funcionamiento API *Rest*. Proporciona acceso mediante programación para leer y escribir datos de Twitter. Crear un nuevo Tweet, leer perfil y datos de seguidores y más

[12] https://dev.twitter.com/streaming/overview

- ***Streaming* API:** Permite recibir información publicada posterior a nuestra petición. Se inicia con Twitter con una conexión entre el servidor y nuestro sistema.

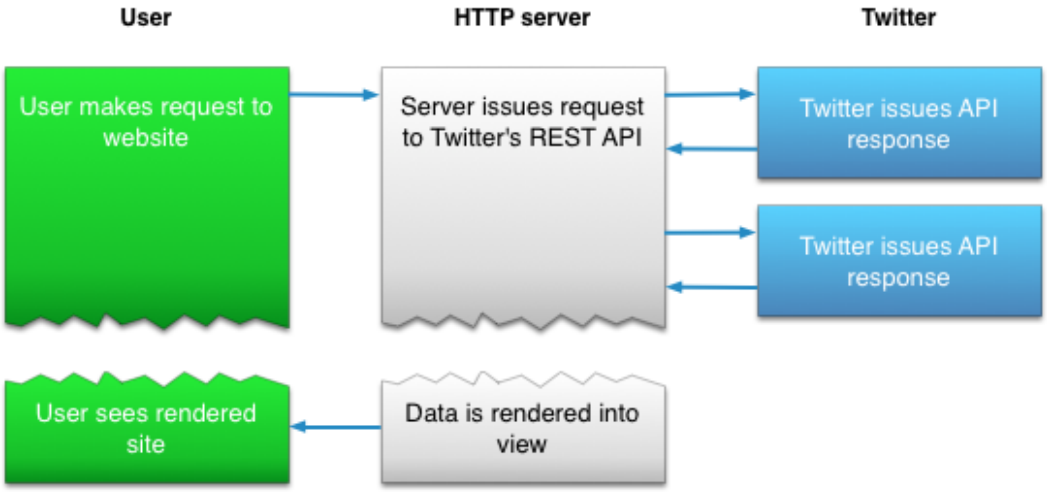


Figura 3.3 Funcionamiento API *Streaming.* Proporciona un subset de tweets en casi tiempo real. Se establece una conexión permanente por usuario con los servidores de Twitter y mediante una petición http se recibe un flujo continuo de tweets en formato json.

3.1.5 Obtención de *tokens*[13]

Para enlazar el código de '*Phyton*' con la red social twitter necesitaremos unas claves de seguridad, por eso lo primero que debemos realizar será crear una cuenta de Twitter, en caso de que no tengamos, y después nos dirigiremos a la web de las apps de Twitter donde crearemos nuestra aplicación.

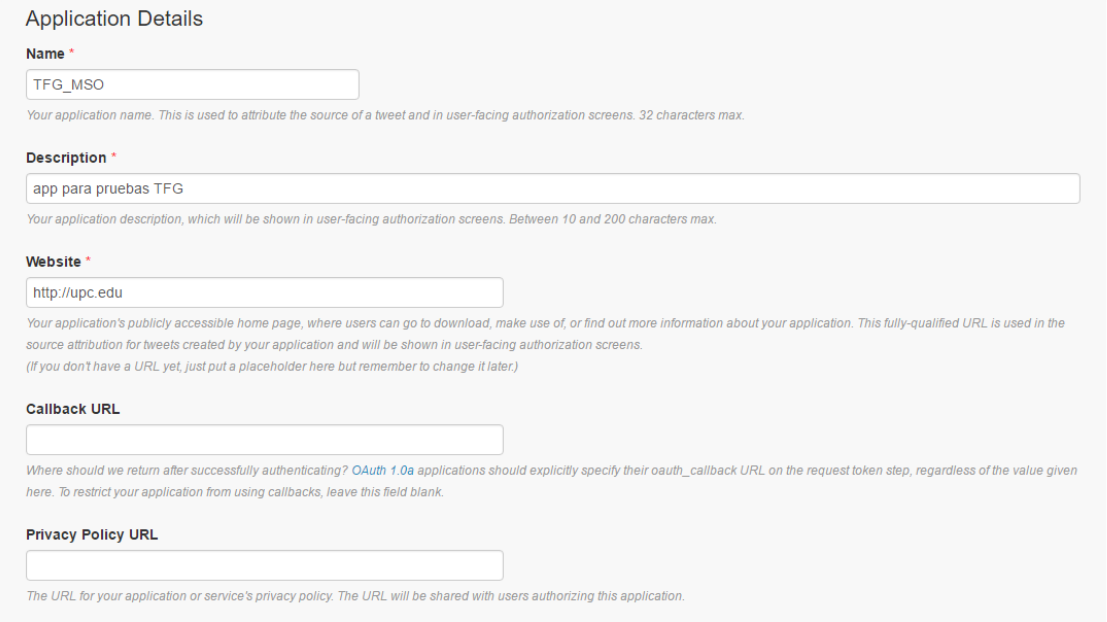


Figura 3.4. Formulario de mi aplicación de Twitter. Este es el formulario que nos aparece en la web oficial de twitter cuando queremos hacer una aplicación.

[13] http://codygo.es/redes-sociales/conseguir-las-consumer-key-y-access-token-de-twitter/

A continuación deberemos introducir que clase de permisos le damos a la aplicación. En mi caso, lo único que quiero es leer publicaciones de otros usuarios para después analizarlas. Por tanto sólo le daremos permisos de lectura tal y como muestro en la siguiente imagen (Figura 3.5).

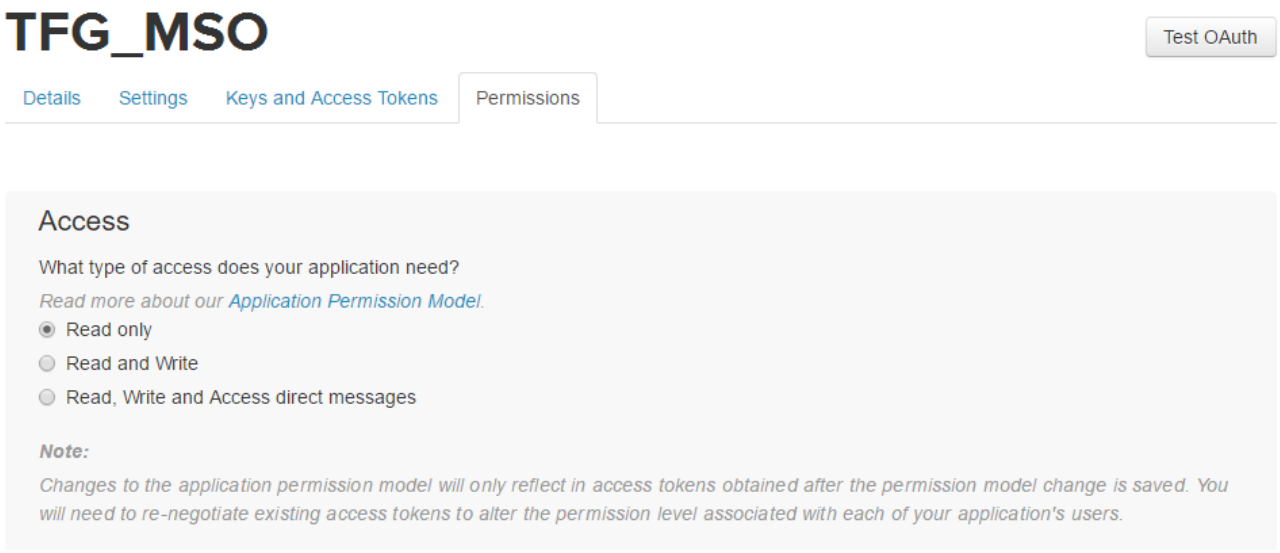


Figura 3.5 Definición de permisos para mi aplicación de Twitter. En mi caso, me sirve con que sea una aplicación de sólo lectura, pues lo que quiero es leer los mensajes de los usuarios, ni enviar nada.

Una vez hemos acabo el proceso de rellenar el formulario y hemos dado permisos de sólo lectura, necesitaremos algunos *tokens*, que son los que darán acceso a nuestra cuenta Twitter a través del código. Estos valores son: *Consumer Key, Consumer Secret, Acces Token* y *Acces Token Secret*. Estos valores los proporciona Twitter de forma automática y son los que después tendré que introducir en el código para vincularlo con mi cuenta de Twitter (Figura 3.6).

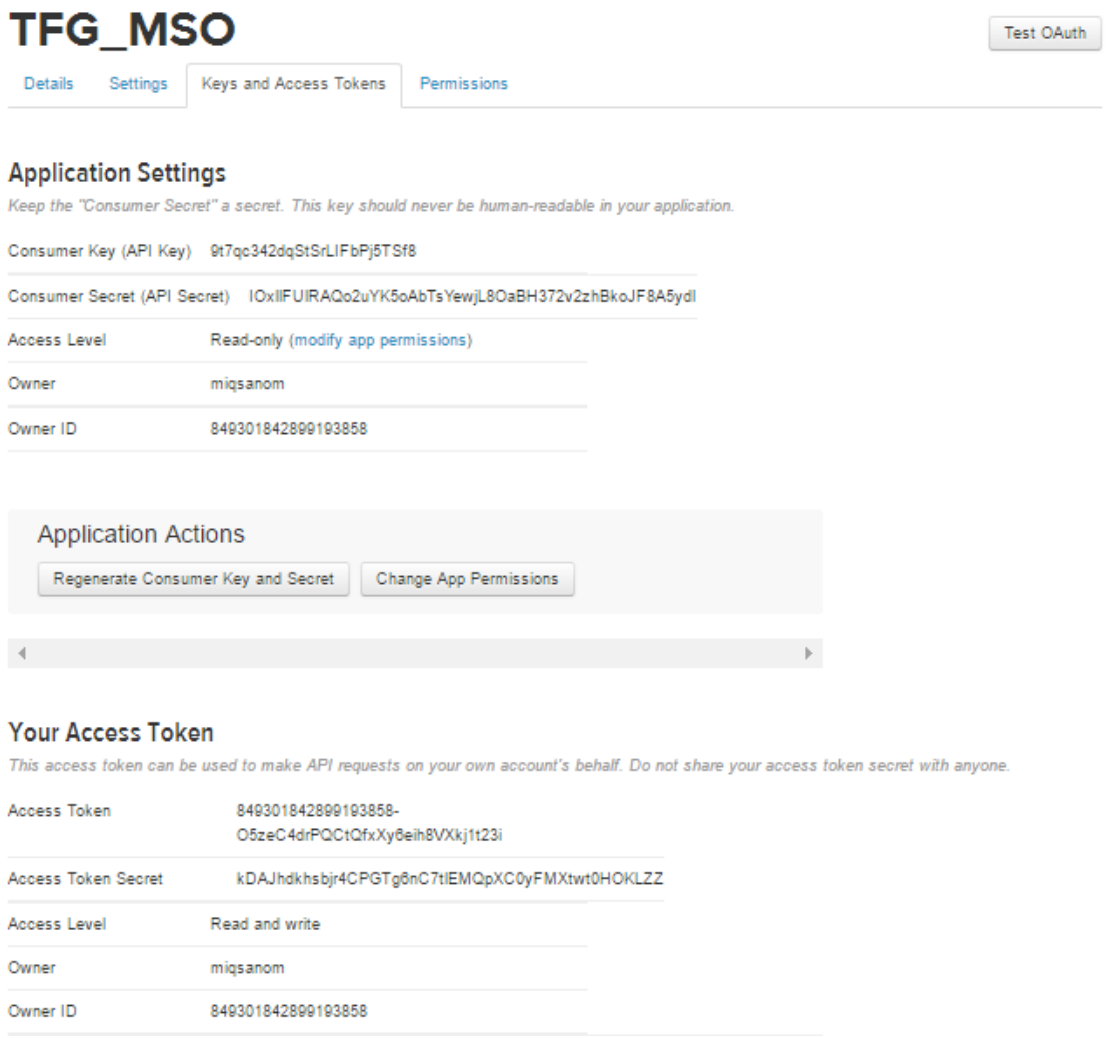


Figura 3.6 *Tokens* de mi aplicación de Twitter. Vinculan el código con la red social Twitter.

3.1.6 Localización de Tweets

En cuanto a cómo asegurarnos de que los tweets que estoy filtrando han sido realizados desde la ubicación o localización que me interesa, disponemos de dos vías principales, además de una secundaria que explicaré pero no utilizaré.

- **Mensaje**: El usuario muestra su localización en el momento de la emisión del mensaje. Esta ubicación puede ser exacta, latitud y longitud, o bien, utilizando un Twitter Place, marca un lugar determinado conocido, p.ej. 'Pl. Catalunya'.
- **Perfil**: Cada usuario puede indicar la ciudad o población donde vive en su perfil.
- **Contenido**: Esta vía consiste en analizar los tweets que contengan el lugar que buscamos citado en el mensaje. Ej. "El Barcelona se impuso por 6 a 1".

La prioridad será buscar aquellos mensajes que contengan la ubicación desde donde han sido emitidos, sin embargo únicamente un 20% de los tweets mundiales contienen este dato[14].

Sin embargo, dado a que los mensajes que contengan la palabra del lugar que buscamos pueden no ser significativos, descartamos esta tercera vía.

3.2 Librerías código[15]

Lo primero que deberemos hacer será declara aquellas librerías que vayamos a utilizar en el programa.

En la siguiente imagen muestro las librerías que importo para utilizarlas seguidamente (Figura 3.7).

```
In [12]: import tweepy
         import pandas as pd
         import matplotlib.pyplot as plt

         pd.options.display.max_columns = 50
         pd.options.display.max_rows= 50
         pd.options.display.width= 120
```

Figura 3.7 Importación de librerías para mi aplicación. Importo las librerías que voy a utilizar en el código y después limito el tamaño del cuadro de datos que utilizo más tarde

[14] https://pressroom.usc.edu/twitter-and-privacy-nearly-one-in-five-tweets-divulge-user-location-through-geotagging-or-metadata/
[15] https://jarroba.com/pandas-python-ejemplos-parte-i-introduccion/

A continuación voy a hacer un breve resumen de las librerías que he utilizado a lo largo del código.

3.2.1 *Tweepy*

Tweepy es la librería más conocida para acceder a la API de Twitter desde Python. La podemos encontrar fácilmente en *github* y tiene ejemplos y explicaciones para todo tipo de operaciones que deseemos realizar.

Su instalación es muy sencilla y se puede hacer desde la consola a través del comando *pip*.

3.2.2 *Pandas*

Pandas es una librería destinada al análisis de datos, lo que nos proporcionará una visión de la información recopilada estructurada según lo que le pidamos.

Ofrece distintas estructuras, desde series, *dataframes*, *panel*, *panel4d* y *panelIND*.

Igual que *Tweepy* su instalación se hace a través de la consola con el comando *pip*.

3.2.3 *Matplotlib*

Matplotlib es la librería destinada a la representación gráfica a partir de una serie de datos.

Proporciona una interfaz orientada a la representación de objetos.

3.3 Declaración de *tokens*

La declaración de *tokens* nos permite conectar la aplicación de *python*, el código, con la app de twitter que hemos creado previamente.

En este caso, únicamente necesitamos el *consumer key* y el *consumer secret* para vincularlo con la app.

Primero declaramos y después lo añadimos a la Api de búsqueda para no tener que hacerlo cada paso tal y como muestro en la siguiente imagen (Figura 3.8).

```
In [13]: consumer_key = "9t7qc342dqStSrLIFbPj5TSf8" # Use your own key. To get a key https://apps.twitter.com/
         consumer_secret = "IOxllFUlRAQo2uYK5oAbTsYewjL8OaBH372v2zhBkoJF8A5ydl"

         auth = tweepy.OAuthHandler(consumer_key=consumer_key, consumer_secret=consumer_secret)

         api = tweepy.API(auth)
```

Figura 3.8 Declaración de *tokens. Aquellos tokens que había obtenido de la página de Twitter son los que ahora he de introducir para vincular el código con la red social.*

3.4 Búsqueda [16]

Utilizaremos el proceso de búsqueda *api.search* para buscar cada una de las palabras clave relacionadas con los términos de análisis de proyecto.

La entrada de las palabras la haremos de forma directa, integrada el código, y cada vez que queramos buscar otra palabra lo haremos modificando los parámetros de búsqueda.

Gracias a esta aplicación podremos filtrar que la búsqueda sea en Barcelona, coordenadas 41.38, 2.16, y a un radio de 2 kilómetros. Igual que la entrada de la palabra a buscar, la localización también la entraremos directamente en el código.

Además de este filtro podríamos filtrar por idioma, retuits, páginas, usuario, etc.

```
In [14]: results = api.search(q="travel",geocode="41.3818,2.1685,2km")

In [15]: len(results)
Out[15]: 15
```

Figura 3.9 Búsqueda palabra *'travel'* en Barcelona.

[16] http://docs.python.org.ar/tutorial/pdfs/TutorialPython2.pdf

3.5 Muestreo

Una vez realizada la búsqueda, haremos una prueba para comprobar que esta se ha hecho correctamente.

Pediremos que nos muestre el nombre de usuario, nombre real de la persona, fecha y hora de emisión del mensaje y mensaje del primer tweet que hayamos encontrado. La siguiente imagen muestra el primer resultado que hemos obtenido en buscar la palabra *travel* (Figura 3.10).

```
In [16]: def print_tweet(tweet):
             print "@%s - %s (%s)" % (tweet.user.screen_name, tweet.user.name, tweet.created_at)
             print tweet.text

         tweet=results[1]
         print_tweet(tweet)

@BrandonsTravel - David Brandon (2017-05-28 07:48:25)
The skyline of Barcelona #spain #luxurylife #architecture #architecturephotography #travel… https://t.co/svMXN1ALOd
```

Figura 3.10 Muestreo del primer tweet que hemos encontrado al realizar la búsqueda de la palabra *travel.*

3.6 Inspección de resultados

Comprobamos que el segundo mensaje que hemos encontrado en nuestra búsqueda contiene tanto la información relacionada con el mensaje enviado como la información relacionada con la emisión del mensaje; usuario, fecha, hora, perfil, etc.

```
In [17]: tweet=results[2]

         for param in dir(tweet):
             if not param.startswith("_"):
                 print "%s : %s" % (param, eval("tweet." + param))

rofile_background_tile=False, favourites_count=131, screen_name=u'RandyRas01', url=None, created_at=datetime.dateti
me(2011, 2, 3, 17, 30, 9), contributors_enabled=False, location=u'Portland, Oregon', profile_sidebar_border_color=
u'C0DEED', translator_type=u'none', following=False), _json={u'contributors': None, u'truncated': False, u'text':
u'Not quite the midnight hour. #spain #travel #travelphotography #cat #barcelona #blackcat @\u2026 https://t.co/clA
u1bxZHV', u'is_quote_status': False, u'in_reply_to_status_id': None, u'id': 868710640172617728L, u'favorite_coun
t': 0, u'entities': {u'symbols': [], u'user_mentions': [], u'hashtags': [{u'indices': [29, 35], u'text': u'spain'},
{u'indices': [36, 43], u'text': u'travel'}, {u'indices': [44, 62], u'text': u'travelphotography'}, {u'indices': [6
3, 67], u'text': u'cat'}, {u'indices': [68, 78], u'text': u'barcelona'}, {u'indices': [79, 88], u'text': u'blackca
t'}], u'urls': [{u'url': u'https://t.co/clAu1bxZHV', u'indices': [92, 115], u'expanded_url': u'https://www.instagra
m.com/p/BUoBuFNFkh6/', u'display_url': u'instagram.com/p/BUoBuFNFkh6/'}]}, u'retweeted': False, u'coordinates':
{u'type': u'Point', u'coordinates': [2.18333, 41.3833]}, u'source': u'<a href="http://instagram.com" rel="nofollo
w">Instagram</a>', u'in_reply_to_screen_name': None, u'in_reply_to_user_id': None, u'retweet_count': 0, u'id_str':
u'868710640172617728', u'favorited': False, u'user': {u'follow_request_sent': None, u'has_extended_profile': False,
u'profile_use_background_image': True, u'default_profile_image': False, u'id': 246887230, u'profile_background_imag
e_url_https': u'https://abs.twimg.com/images/themes/theme1/bg.png', u'verified': False, u'translator_type': u'non
e', u'profile_text_color': u'333333', u'profile_image_url_https': u'https://pbs.twimg.com/profile_images/123384465
4/Ct.Randy_Rasmussen.online_normal.jpg', u'profile_sidebar_fill_color': u'DDEEF6', u'entities': {u'description':
{u'urls': [{u'url': u'https://t.co/LiKhHF3r1i', u'indices': [107, 130], u'expanded_url': u'http://randyrasmussen.co
m', u'display_url': u'randyrasmussen.com'}]}}, u'followers_count': 745, u'profile_sidebar_border_color': u'C0DEED',
```

```
In [18]: user=tweet.author

         for param in dir(user):
             if not param.startswith("_"):
                 print "%s : %s" % (param, eval("user." + param))
```

```
contributors_enabled : False
created_at : 2011-02-03 17:30:09
default_profile : True
default_profile_image : False
description : Editorial and commercial photographer / multimedia specialist in Portland, Oregon. To contact me: ran
dy{at}https://t.co/LiKhHF3r1i or 503 926 3536
entities : {u'description': {u'urls': [{u'url': u'https://t.co/LiKhHF3r1i', u'indices': [107, 130], u'expanded_ur
l': u'http://randyrasmussen.com', u'display_url': u'randyrasmussen.com'}]}}
favourites_count : 131
follow : <bound method User.follow of User(follow_request_sent=None, has_extended_profile=False, profile_use_backgr
ound_image=True, _json={u'follow_request_sent': None, u'has_extended_profile': False, u'profile_use_background_imag
e': True, u'default_profile_image': False, u'id': 246887230, u'profile_background_image_url_https': u'https://abs.t
wimg.com/images/themes/theme1/bg.png', u'verified': False, u'translator_type': u'none', u'profile_text_color': u'33
3333', u'profile_image_url_https': u'https://pbs.twimg.com/profile_images/1233844654/Ct.Randy_Rasmussen.online_norm
al.jpg', u'profile_sidebar_fill_color': u'DDEEF6', u'entities': {u'description': {u'urls': [{u'url': u'https://t.c
o/LiKhHF3r1i', u'indices': [107, 130], u'expanded_url': u'http://randyrasmussen.com', u'display_url': u'randyrasmus
sen.com'}]}}, u'followers_count': 745, u'profile_sidebar_border_color': u'C0DEED', u'id_str': u'246887230', u'profi
le_background_color': u'C0DEED', u'listed_count': 82, u'is_translation_enabled': False, u'utc_offset': -25200, u'st
atuses_count': 1848, u'description': u'Editorial and commercial photographer / multimedia specialist in Portland, O
regon. To contact me: randy{at}https://t.co/LiKhHF3r1i or 503 926 3536', u'friends_count': 159, u'location': u'Port
```

Figura 3.11 Inspección de resultados *Status Object* y *User Object.* Comprobación de que la búsqueda se ha hecho de forma correcta y cada tweet contiene la información que le corresponda.

3.7 Cursor

Una vez establecida la conexión con twitter y la búsqueda ha sido realizada con éxito, hay que crear un cursor. Un cursor es una estructura de control que se usa para recorrer (y eventualmente procesar) los resultados que hemos obtenido anteriormente.

El método para crear el cursor se llama, originalmente, cursor(), dónde volveremos a decirle lo que queremos que recorra y cuantos resultados querremos, estos los encontraremos ordenados de forma cronológica y cómo máximo con una antigüedad de 7 días.

```
In [19]: results = []
         for tweet in tweepy.Cursor(api.search,q="travel",geocode="41.3818,2.1685,2km").items(100):
             results.append(tweet)

         print len(results)
```

```
88
```

Figura 3.12 Creación de un “cursor” para recorrer los resultados. El objetivo es poder utilizar todos los resultados que hemos obtenido para visualizarlos juntos y realizar una comparativa. Por eso es necesario el uso de un cursor que vaya recorriendo los resultados para después mostrarlos todos.

3.8Data Frame[17]

Gracias al uso de un *data frame* procesamos los resultados obtenidos para seguidamente verlos ordenados, tal y como nosotros le pedimos al programa.

Cada resultado obtenido se guardará con el mismo identificador de la red social.

Este procesado se divide en dos partes:

- *Tweet Data*: contiene texto del tweet, fecha y hora de creación, contador de retuits, contador de favoritos y la fuente original del mensaje.
- *User Data*: contiene nombre real del usuario, *user ID*, nombre de usuario, fecha de creación de la cuenta, descripción del usuario, número de seguidores, etc.

En este caso, únicamente hemos pedido que nos muestre los 5 primeros resultados de los encontrados.

En las siguientes imágenes muestro el código empleado para el *Data Frame (*Figura 3.13*)*.

```
In [20]: def process_results(results):
             id_list = [tweet.id for tweet in results]
             data_set = pd.DataFrame(id_list, columns=["id"])

             # Processing Tweet Data

             data_set["text"] = [tweet.text for tweet in results]
             data_set["created_at"] = [tweet.created_at for tweet in results]
             data_set["retweet_count"] = [tweet.retweet_count for tweet in results]
             data_set["favorite_count"] = [tweet.favorite_count for tweet in results]
             data_set["source"] = [tweet.source for tweet in results]

             # Processing User Data
             data_set["user_id"] = [tweet.author.id for tweet in results]
             data_set["user_screen_name"] = [tweet.author.screen_name for tweet in results]
             data_set["user_name"] = [tweet.author.name for tweet in results]
             data_set["user_created_at"] = [tweet.author.created_at for tweet in results]
             data_set["user_description"] = [tweet.author.description for tweet in results]
             data_set["user_followers_count"] = [tweet.author.followers_count for tweet in results]
             data_set["user_friends_count"] = [tweet.author.friends_count for tweet in results]
             data_set["user_location"] = [tweet.author.location for tweet in results]

             return data_set
         data_set = process_results(results)
```

```
In [21]: data_set.head(5)
```

Out[21]:

	id	text	created_at	retweet_count	favorite_count	source	user_id	user_screen_name	user_name
0	868777140011106305	なんでもない街並みが、とても好きでして\n #traveling #travel #trip...	2017-05-28 10:33:05	0	0	Instagram	2914711448	mousou_trip_jp	Caori
1	868735696852459520	The skyline of Barcelona #spain #luxurylife #a...	2017-05-28 07:48:25	0	0	Instagram	1439916589	BrandonsTravel	David Brandon
2	868710640172617728	Not quite the midnight hour. #spain #travel #t...	2017-05-28 06:08:51	0	0	Instagram	246887230	RandyRas01	Randy L. Rasmussen

Figura 3.13 Código *Data Frame* y visualización 5 primeros resultados encontrados.

[17] https://pandas.pydata.org/pandas-docs/stable/dsintro.html

3.9Búsqueda de dos palabras

Para saber más sobre los usuarios de la red social Twitter buscaré dos palabras en un mismo lugar y así tener una mejor idea de la situación.

Esto lo podemos hacer utilizando la palabra 'OR' entre las dos palabras que quiero buscar en el momento en el que realizamos la búsqueda.

Estas dos palabras que usaré para conocer mejor el estado de los usuarios pueden ser palabras con una idea anímica similar, '*happy* OR *travel*', ambas palabras definen un estado de ánimo positivo, o por el contrario, de ánimo negativo, '*work* OR *exams*'.

A raíz de lo ocurrido el 17 de agosto, he decidido realizar una búsqueda de dos palabras relacionadas con el atentado terrorista. Las palabras que buscaré son *por* y *terrorism*.

Figura 3.14 Búsqueda de dos palabras y muestreo del primer resultado que contiene una de las dos palabras.

Podemos observar el retuit realizado por ERC Sants-Montjuic del tuit de Carles García que dice: la catástrofe ha sido devastadora pero la respuesta ha sido aún más impresionante. Barcelona ha apostado..

Este tuit muestra el claro mensaje de todo el pueblo Barcelonés en contra del terrorismo y a favor de la lucha de un pueblo unido contra los terroristas. Lo cual podemos considerar que pesé al desastre ocurrido, la ciudad de Barcelona es fuerte y no se va a dejar doblegar por el atentado.

A continuación muestro algunos de los mensajes de apoyo y lucha que han transmitido usuarios de la red social Twitter (Figura 3.15).

```
In [10]: data_set.head(5)
```

	text	created_at	retweet_count	favorite_count	source	user_id	user_screen_name	user_name	user_created_at	user_descri
654131200	RT @CarlesGarcias: L'@AlfredBosch 'La catástro...	2017-08-20 08:45:52	12	0	Twitter for iPhone	329294908	ERChg7	ERC Horta-Guinardó	2011-07-04 20:58:03	Casal Josep
776003072	RT @CarlesGarcias: L'@AlfredBosch 'La catástro...	2017-08-20 08:36:34	12	0	Twitter Lite	961043671	ErcSantsMon	ERC Sants-Montjuïc	2012-11-20 19:11:06	
865990144	#batoni excelente noche en el #pasapalo bar de...	2017-08-20 06:45:25	0	0	Instagram	53885827	fernandobatoni	fernando batoni	2009-07-05 10:41:31	Director Crea Diseñador . Músico\nhttp
192773633	RT @yellowseahawk: @NessyHeil @Fabada_Heydrich...	2017-08-20 02:01:12	1	0	Twitter for iPhone	1511019176	Unrutilofilomas	So quick	2013-06-12 16:44:23	A veces retw sobre política

Figura 3.15 Resultados de la búsqueda de las palabras '*por*' y '*terrorism*' después del atentado en Barcelona

3.10 Procesado de tweets

Para este apartado he decidido realizar a búsqueda de una sola palabra, se trata del hashtag más utilizado días después de los atentados, '*notincpor*'.

Este lema surgió de forma totalmente espontanea el día 18 de agosto después del minuto de silencio, que se hizo por las víctimas del atentado, en la plaza Cataluña.

A continuación muestro uno de los resultados que he obtenido en esta búsqueda, es un retuit de Silvia Rodríguez que dice; '*mig milió de persones van participar dissabte en la manifestació #notincpor demanant pau per combatre la violència'.*

```
In [5]: def print_tweet(tweet):
            print "@%s - %s (%s)" % (tweet.user.screen_name, tweet.user.name, tweet.created_at)
            print tweet.text

        tweet=results[1]
        print_tweet(tweet)
```

```
@SilviaRBen - Sílvia Rodríguez Ben (2017-08-28 11:53:40)
RT @Bcn_Eixample: Mig milió de persones van participar dissabte en la manifestació #Notincpor demanant pau per combat
re la violència https:…
```

Figura 3.16 Muestreo del primer resultado obtenido en la búsqueda de la palabra 'notincpor'. Realizado por Silvia Rodríguez el 28 de agosto.

Seguidamente muestro otros resultados de esta misma búsqueda.

```
In [11]: data_set.tail(5)
```

Out[11]:

	id	text	created_at	retweet_count	favorite_count	source	user_id	user_screen_name	user_nam
95	901505117505155072	#notincpor #notenimpor #lamillorrespostalapau ...	2017-08-26 18:02:23	0	1	Instagram	251330998	5_josep	J SEP S LER
96	901504955080515585	#notincpor #Barcelona #Cambrils en La Rambla-F...	2017-08-26 18:01:45	0	1	Instagram	75355895	Isaacpecino	Isaac Pecino
97	901504520282411010	A la Rambles de Barcelona, una "riuada" de gen...	2017-08-26 18:00:01	8	6	Instagram	307474069	jroca34	Joan Roca Tió
98	901503999043670016	Incident menor pel tema de les banderes a la #...	2017-08-26 17:57:57	0	0	Instagram	2370998347	a_velasco_gomez	Antonio Velasco
99	901503875382947841	#igers #igersbarcelona #notincpor #manifestati...	2017-08-26 17:57:27	0	1	Instagram	269896542	deivisviloria	Deivis Viloria

Figura 3.17 Algunos de los resultados de la búsqueda de la palabra '*notincpor*' realizada el 28 de agosto.

Una vez realizada la búsqueda, pasamos al procesado de los resultados obtenidos. Este proceso se basará en mostrar gráficamente la red social de origen de los tuits obtenidos.

Me explico, en twitter cualquier usuario puede compartir mensajes que han sido previamente publicados en otra red social, como *instagram, facebook, hootsuite*, etc.

Para conseguir el gráfico que muestro a continuación, he utilizado la tabla que relleno con los mensajes obtenidos en mi búsqueda y toda la información de cada mensaje. Una vez utilizada esta tabla, específico que la única columna que me interesa es aquella que contiene la fuente de origen de los mensajes, '*source*'.

Después utilizo la función '*plt'* para dibujar las columnas de las fuentes más usadas, y dibujan el porcentaje de cantidad de tuits por red social.

```
In [13]: sources = data_set["source"].value_counts()[:5][::-1]

plt.barh(xrange(len(sources)), sources.values)
plt.show()
```

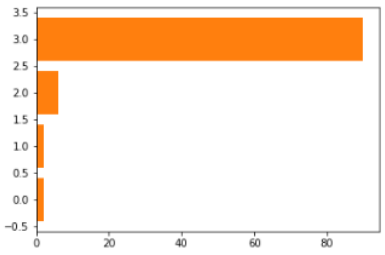


Figura 3.18 Grafica que muestra el porcentaje de cantidad de tuits por red social. El resultado es bastante coherente pues Twitter es la fuente más utilizada.

En el caso de la búsqueda de la palabra '*notincpor*', las redes sociales más utilizadas han sido las siguientes, de más a menos, *Instagram, Twitter, Twitter for iphone* y *Hootsuite*.

A continuación muestro otra vía de procesado de tuits que he querido realizar para mostrar la diversidad de personas de diferentes nacionalidades que se han solidarizado con Barcelona después de los atentados. Para esto he realizado una gráfica, que en vez de basarse en la fuente de origen de los tuits, me mostrará la cantidad de tuits realizados a través de cuentas creadas en un país determinado.

```
In [34]: sources = data_set["user_location"].value_counts()[:5][::-1]

         plt.barh(xrange(len(sources)), sources.values)
         plt.show()
```

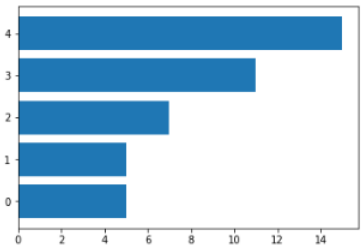


Figura 3.19 Grafica que muestra la cantidad de tuits por país de creación de la cuenta. He utilizado la función plt para mostrar los países asociados a los mensajes obtenidos en los resultados.

De más a menos los países donde se crearon las cuentas vinculadas a los mensajes de esta búsqueda son España, Reino Unido, Francia, América y Alemania.

4. EXPERIMENTACIÓN Y ANÁLISIS

Una vez generado el procesado de tweets, podemos proceder a su análisis y experimentación. El objetivo del análisis de los tweets encontrados es validar si estos datos están relacionados con los datos registrados por diversas páginas webs y diarios, validar cuán efectiva es nuestra aplicación. A continuación, se aplicarán algunas métricas como la precisión, para comprobar si la tendencia que se puede sacar de las opiniones de los usuarios posteriormente se ven reflejadas en los resultados registrados en diarios y páginas webs.

4.1 Definición de prioridades

Las principales prioridades de mi proyecto es buscar aquellos tweets que sean publicados en el lugar determinado que buscamos y no buscar aquellos que mencionan el nombre de la ubicación que queremos.

Además de esta prioridad para anular aquellas publicaciones que no interesan, también buscaré todos aquellos tweets que vienen relacionados con palabras o hashtags que vayan relacionados con un estado anímico determinado. Esto lo he realizado haciendo diversas búsquedas, algunas con un fondo positivo y otras con un fondo más bien negativo. Seguidamente he juntado aquellas búsquedas para compararlas entre ellas y obtener una idea aproximada sobre la opinión de los usuarios de diversas redes sociales ubicados en Barcelona.

4.2 Análisis de resultados

Para analizar los resultados obtenidos en este trabajo, se van a comparar los resultados obtenidos con otras fuentes basadas en diferentes parámetros que definen la forma de vivir de una ciudad determinada.

Antes voy a estudiar la precisión de los resultados obtenidos, que toma un valor entre 0 y 1. Cuanto más se acerque a 1 la precisión, más acertados serán los resultados.

4.2.1 Precisión de los resultados

El cálculo que se ha hecho en el análisis de los resultados ha sido la precisión de los resultados obtenidos en mis búsquedas. Dicho valor mide el acierto al predecir si el tweet es correcto o no, es decir si realmente un tweet con un hashtag 'positivo' es un mensaje positivo o no. También he realizado el mismo sistema para aquellos tweets negativos. La fórmula de la precisión queda recogida en la siguiente ecuación:

$$Precisión = {}^{TP}/_{TP + FP}$$

Donde TP (True Positive) son las búsquedas consideradas como correctas y FP (False Positive) son las búsquedas consideradas como incorrectas. A continuación pongo algunos ejemplos de tweets considerados como TP y FP:

Tweets obtenidos en la búsqueda de la palabra *travel*:

- **TP:** `@rowanrowrowan - Rowan Row (2017-09-08 08:54:40)- Buenos dias Barcelona ES.Have a good day my friends #livelovelaugh #travel… https://t.co/6P1OEodc8a`
- **FP:** `@tmj_spn_cstsrv - TMJ-SPN CstSrv Jobs (2017-09-08 10:13:24)- We're #hiring! Click to apply: Senior business travel counsellor - Swedish speaker - https://t.co/95G14mXg6r #CustomerService #Job #Jobs`

Tweets obtenidos en la búsqueda de la palabra *work*:

- **TP:** `@frvdbosch - Frank (2017-09-08 07:42:01) - Where is the sun today #letslookforthesun #lifegoals #work #nobluesky #nosun #thatslife… https://t.co/elCSwvSiE5`
- **FP:** `@tmj_spn_adv - TMJ-SPN Advert. Jobs (2017-09-08 06:04:12) If you're looking for work in #Barcelona, CT, check out this #job : https://t.co/H0xscfVCxb #Marketing #Hiring #CareerArc`

Para estudiar la precisión de mis búsquedas he utilizado dos palabras para cada una de las búsquedas:

- Positivas: '*travel*' y 'sol'.
- Negativas: '*work*' y *'por'.*

He decidido que las palabras fuesen en catalán y en inglés porque, como todos sabemos, Barcelona es una de las capitales del turismo mundial y el idioma más globalizado es el inglés.

En caso de sólo haber buscado palabras en castellano y catalán me hubiese dejado un gran volumen de tweets emitidos desde Barcelona.

Seguidamente muestro un fragmento de la tabla (Figura 4.1) que he realizado para obtener la precisión de mis resultados, he buscado 100 resultados para cada caso de mensajes, positivos y negativos. Después de leerlos uno a uno, he identificado como correctos aquellos marcados en verde con las letras TP y como incorrectos los marcados en rojo con las letras FP.

Tweet 90	FP	1			Tweet 90	TP	1
Tweet 91	TP	1			Tweet 91	TP	1
Tweet 92	TP	1			Tweet 92	TP	1
Tweet 93	TP	1			Tweet 93	TP	1
Tweet 94	TP	1			Tweet 94	FP	1
Tweet 95	TP	1			Tweet 95	TP	1
Tweet 96	TP	1			Tweet 96	TP	1
Tweet 97	FP	1			Tweet 97	TP	1
Tweet 98	TP	1			Tweet 98	TP	1
Tweet 99	TP	1			Tweet 99	TP	1
Tweet 100	TP	1			Tweet 100	TP	1
	TP	82				TP	76
	FP	18				FP	24
	P	0,82				P	0,76

Figura 4.1 Tabla de análisis de precisión de los resultados obtenidos. La tabla muestra los últimos 10 resultados analizados distinguidos en colores para identificar los TP y los FP. Al final de la tabla se puede observar la precisión obtenida en casa caso.

Como conclusiones acerca de estos resultados, podemos decir que uno de los factores que han influido en ellos ha sido el idioma empleado en el proceso de minería de datos. Las búsquedas de las palabras utilizadas las he hecho combinando dos idiomas, español e inglés, lo cual afecta directamente a las diversas connotaciones que una misma palabra puede tener.

Además he de tener en cuenta que muchos de los usuarios de estas redes sociales han hecho retuits de otros usuarios, me refiero a que he encontrado varios mensajes que son repetidos al haber utilizado la herramienta de retuitear mensajes de otros usuarios, lo cual afecta directamente a los resultados obtenidos.

Una de las vías para obtener una precisión más elevada de mis resultados sería eliminar aquellos mensajes repetidos e incluir más palabras y no sólo dos que están directamente relacionadas con na opinión positiva o negativa.

Aun así considero que la precisión de los resultados ha sido bastante elevada y puedo considerar que mi código es fiable.

4.3 Líneas futuras de la minería de datos

El futuro de la minería de datos[18], tanto dentro de las redes sociales cómo dentro de diferentes bases de datos que podemos encontrar en internet, está creciendo durante los últimos años a gran escala.

La capacidad de consultar grandes bases de datos para después utilizarlos con un fin determinado nos abre un futuro que nos permitirá rastrear, analizar y filtrar grandes volúmenes de datos.

Desde grandes plataformas comerciales, servicios policiales, médicos, aeronáuticos, navales, económicos, gobiernos, etc utilizarán esta herramienta para dirigirla, según su elección a la vía que más les interese. Podrán consultar una gran masa de información para focalizar en aquello que más les interesé.

Por eso, la minería de datos es una tecnología emergente, tanto para investigadores y negocios, como para aquellos que necesiten valorar hasta el más mínimo detalle en la toma de decisiones.

Actualmente destacan algunos proyectos basados en la minería de datos[19], voy a comentar algunos:

- SKYCAT: se trata de un proyecto en el que el Second Palomar Observatory Sky Survey se dedicó a coleccionar tres terabytes de imágenes durante seis años. Los resultados de un estudio realizado con esta base de datos ha permitido descubrir dieciséis nuevos quásares a los astrónomos.
- ANALISIS DE SUEÑO: varias universidades están estudiando a través de las variables obtenidas durante las horas de sueño la calidad del sueño de los usuarios. Esto lo hacen mediante minería de datos registrando movimiento, temperatura, flujo térmico, etc.
- ALGORITHMIA: con el crecimiento de la minería de datos las empresas están comprando algoritmos en lugar de programarlos y añadir sus propios datos, es el caso de servicios como Algorithmia.
- KNIME: Ofrece a los usuarios la capacidad de crear de forma visual flujos o tuberías de datos, ejecutar selectivamente algunos o todos los pasos de análisis, y luego estudiar los resultados, modelos y vistas interactivas.

Hay que tener en cuenta que el crecimiento de esta herramienta puede ser mucho más elevado de lo que tenemos pensado, pues el poder de gestionar información de un ordenador no es comparable al de ningún ser humano.

[18] http://www.tuataratech.com/2016/03/17-predicciones-sobre-el-futuro-de-big.html
[19] http://www.que.es/tecnologia/201306211308-lado-oscuro-mineria-datos-rc.html

Pues ya ha habido casos de hackers que a través de la minería de datos han obtenido información confidencial a través de vías fuera de la ley. [20]

[20] http://www.que.es/tecnologia/201306211308-lado-oscuro-mineria-datos-rc.html

5. CONCLUSIONES Y LINEAS FUTURAS

A partir de la elaboración de este trabajo, pienso que he llegado a los objetivos que me marqué en un principio. Creo que la elaboración del programa propuesto podría ser efectivo y útil.

La recopilación de grandes volúmenes de información, proporcionados por las redes sociales, es una fuente de datos con gran proyección a la hora de ser analizada. Tanto la geolocalización de los mensajes como la propia búsqueda de estos y su procesado han sido los puntos más importantes del proyecto.

Si recordamos los objetivos del proyecto, vemos que se han podido recopilar los tweets a través del API que proporciona Twitter, pudiendo mostrar los tweets por terminal así como después procesarlos para obtener las gráficas que me interesaban. La dependencia de los resultados enfocada al lugar de origen de cada cuenta, así como la probable repetición de tweets o resultados debido a la función que nos permite reenviar mensajes que han sido enviados previamente, puede ser algo negativo en casos concretos de la búsqueda.

La recopilación de los tweets según su zona geográfica ha estado fuertemente condicionada por la escasez de tweets que incluyan su ubicación. Además he evitado aquellos mensajes que contuvieran la palabra de la geolocacalización que he buscado a lo largo del proyecto, esto lo he hecho para evitar mensajes que poco tuviesen que ver con lo que realmente estaba buscando. La solución a este problema ha llevado a la obtención del lugar del usuario que aparece en el perfil, con la pérdida de exactitud que esto supone. No obstante, dado que muchos usuarios no indican un lugar válido tampoco en su perfil, el número de tweets geolocalizados en territorio nacional sigue siendo pequeño. Pues tal y como he dicho antes, no son todos los usuarios que comparten sus mensajes señalando la ubicación en la que se encuentran.

El análisis de los mensajes buscando unas palabras o hashtags determinados, de forma automática, se ha basado en el código de la aplicación *“api.search”*. He utilizado un *timeline* de 7 días, por lo tanto, en una misma búsqueda he encontrado resultados totalmente diferentes, sobretodo resaltan antes y después de los atentados que atacaron Barcelona el pasado mes de agosto.

En cuanto a proporcionar una visualización significativa de los datos obtenidos, he clasificado todos los resultados de mis búsquedas mostrando diversos parámetros propios de cada mensaje, a continuación indico los más significativos:

- Fecha de emisión del mensaje
- Nombre del usuario
- Contador de retuits de cada mensaje
- ID del usuario
- País de origen de la cuenta que emite el mensaje

Pienso que tanto la forma en la que muestro los tweets como los gráficos que realizo al procesarlo cumplen su objetivo de dar una idea aproximada de lo que busco, sin embargo, seguir un paso más allá y hacer el código más interactivo, hubiese facilitado de cara al usuario una mayor comprensión de la aplicación y un manejo de esta más sencillo.

El mayor problema al que me he enfrentado al embarcarme con este proyecto, ha sido mi inexistente formación de las herramientas que he tenido que utilizar. Puesto que no había utilizado el API de Twitter, ni había trabajado con Python nunca, y nunca. Pese a ello, con el debido tiempo de estudio y formación, puedo decir que el resultado del proyecto ha sido satisfactorio. Sin embargo, se podría haber mejorado substancialmente la parte de código, así como reducido el tiempo e intentos para que esté funcionase con algo de formación de este lenguaje en asignaturas como Informática 1 o Informática 2, cabe decir que Python no deja de utilizar un lenguaje de programación y como bien todos sabemos, los lenguajes suelen ser bastante parecidos entre sí y al haber estudiado varios a lo largo del grado, el aprendizaje de este nuevo lenguaje no se ha hecho tan largo.

En cuanto a posible desarrollo de este trabajo, me hubiera gustado poder proporcionar un mapa de la ciudad de Barcelona indicando en qué punto exacto es emitido cada mensaje. Así como permitir al usuario entrar la o las palabras que desea buscar y no tener que entrarlas a través del código. Serían estos dos puntos los que podrían dar pie a mejoras futuras de mi proyecto.

En mi opinión, sus posibles aplicaciones en el campo del estudio de los perfiles tanto de negocios, venta de billetes, calidad de servicios, calidad de vida son muy significativos. Otras posibles situaciones de estudio serían los flujos turísticos por un territorio en concreto, el avance en rastreo de posibles grupos radicales o el registro de diversos parámetros médicos para evitar y prevenir enfermedades.

Por último, quiero decir que este trabajo me ha servido para hacer un recopilatorio de gran parte de lo que aprendido durante mis últimos años y para poder aplicarlo en un futuro muy próximo.

6. BIBLIOGRAFÍA Y REFERENCIAS

Durante la realización de este proyecto he consultado la información en internet, artículos de revista y libros. A continuación nombro las webs, libros o revistas que he utilizado:

- Páginas webs

[1] / [11] https://about.twitter.com/es/company

(consultada el 06/04/2017)

[2]https://blog.es.logicalis.com/analytics/mineria-de-datos-aplicaciones-que-ya-son-una-realidad (consultada el 06/04/2017)

[2] http://www.cad.com.mx/historia_de_twitter.htm

(consultada el 06/04/2017)

[3] https://www.importancia.org/twitter.php (consultada el 08/04/2017)

[4]http://www.lavanguardia.com/salud/20131213/54395494664/recrean-como-pandemia-podria-propagarse-mundo.html

(consultada el 11/04/2017)

[5]https://mulherolhodepeixe.wordpress.com/2015/11/22/terrorismo-internet-y-data-mining/ (consultada el 13/04/2017)

[6]http://www.ntn24america.com/noticia/buscando-donde-vivir-ranking-de-las-10-mejores-ciudades-del-mundo-para-vivirbuscando-donde-vivir-las-146605 (consultada el 19/04/2017)

[7] http://happyplanetindex.org/about/ (consultada el 13/04/2017)

[8] http://www.socialblabla.com/25-aplicaciones-para-twitter.html

(consultada el 19/04/2017)

[9] https://es.wikipedia.org/wiki/Twitter (consultada el 24/04/2017)

[10] https://dev.twitter.com/streaming/overview

(consultada el 10/05/2017)

[12]http://codygo.es/redes-sociales/conseguir-las-consumer-key-y-access-token-de-twitter/ (consultada el 16/05/2017)

[13] https://jarroba.com/pandas-python-ejemplos-parte-i-introduccion/

(consultada el 23/05/2017)

[14] http://docs.python.org.ar/tutorial/pdfs/TutorialPython2.pdf

(consultada el 11/06/2017)

[15] https://pandas.pydata.org/pandas-docs/stable/dsintro.html

(consultada el 23/06/2017)

[16] http://python-twitter.readthedocs.io/en/latest/getting_started.html

(consultada el 12/07/2017)

[17]http://blog.jmacoe.com/gestion_ti/base_de_datos/5-mejores-software-mineria-datos-codigo-libre-abierto/

(consultada el 01/08/2017)

[18]http://www.tuataratech.com/2016/03/17-predicciones-sobre-el-futuro-de-big.html

(consultada el 18/08/2017)

[19]http://www.que.es/tecnologia/201306211308-lado-oscuro-mineria-datos-rc.html

(consultada el 21/08/2017)

[20]http://www.que.es/tecnologia/201306211308-lado-oscuro-mineria-datos-rc.html (consultada el 25/08/2017)

- Libros

[1] Learning Phyton, Lutz Marck (1999) (consultado continuamente a lo largo de la realización del proyecto)

[2] Python for Dummies, Stef Maruch & Aahz Maruch (2006) (consultado continuamente a lo largo de la realización del proyecto)

7. ANEXO

Importo las librerías, previamente instaladas a través del terminal, para poderlas llamar a ser utilizadas posteriormente en el código. Después introduzco los *tokens* para poder vincular el código con la red social twitter.

Seguidamente introduzco mi búsqueda y pido la cantidad de resultados obtenidos.

```
In [1]: import tweepy
        import pandas as pd
        import matplotlib.pyplot as plt

        pd.options.display.max_columns = 50
        pd.options.display.max_rows= 50
        pd.options.display.width= 120
```

```
In [2]: consumer_key = "9t7qc342dqStSrLIFbPj5TSf8" # Use your own key. To get a key https://apps.twitter.com/
        consumer_secret = "IOxllFUlRAQo2uYK5oAbTsYewjL8OaBH372v2zhBkoJF8A5ydl"

        auth = tweepy.OAuthHandler(consumer_key=consumer_key, consumer_secret=consumer_secret)

        api = tweepy.API(auth)
```

```
In [7]: results = api.search(q="work",geocode="41.3818,2.1685,2km")
```

```
In [8]: len(results)
```

En el caso que buscase dos palabras.

```
In [7]: results = api.search(q="por OR work",geocode="41.3818,2.1685,2km")
```

Pido un muestreo del primer resultado obtenido en mi búsqueda.

```
In [11]: def print_tweet(tweet):
             print "@%s - %s (%s)" % (tweet.user.screen_name, tweet.user.name, tweet.created_at)
             print tweet.text

         tweet=results[1]
         print_tweet(tweet)
```

Ahora quiero comprobar que el resultado dos de mi búsqueda contiene todos los parámetros de un mensaje emitido a través de twitter para después poder saber cuáles son los que realmente me interesan y rellenar la tabla con aquellos que quiero.

```
In [27]: tweet=results[2]

         for param in dir(tweet):
             if not param.startswith("_"):
                 print "%s : %s" % (param, eval("tweet." + param))
```

A continuación muestro un pequeño fragmento de los resultados que he tenido que analizar de la parte del mensaje.

```
author : User(follow_request_sent=None, has_extended_profile=False, profile_use_background_image=True, _json={u'follo
w_request_sent': None, u'has_extended_profile': False, u'profile_use_background_image': True, u'default_profile_imag
e': False, u'id': 421293889, u'profile_background_image_url_https': u'https://pbs.twimg.com/profile_background_image
s/576670672191033346/bsBtGUcx.jpeg', u'verified': False, u'translator_type': u'none', u'profile_text_color': u'B8813
0', u'profile_image_url_https': u'https://pbs.twimg.com/profile_images/495032565234159616/0GshRD6O_normal.jpeg', u'pr
ofile_sidebar_fill_color': u'141214', u'entities': {u'description': {u'urls': []}}, u'followers_count': 94, u'profile
_sidebar_border_color': u'121112', u'id_str': u'421293889', u'profile_background_color': u'1A1B1F', u'listed_count':
7, u'is_translation_enabled': False, u'utc_offset': 7200, u'statuses_count': 3341, u'description': u'', u'friends_cou
nt': 297, u'location': u'Catalunya', u'profile_link_color': u'DD2E44', u'profile_image_url': u'http://pbs.twimg.com/p
rofile_images/495032565234159616/0GshRD6O_normal.jpeg', u'following': None, u'geo_enabled': True, u'profile_banner_ur
l': u'https://pbs.twimg.com/profile_banners/421293889/1469662497', u'profile_background_image_url': u'http://pbs.twim
g.com/profile_background_images/576670672191033346/bsBtGUcx.jpeg', u'screen_name': u'Bredotoijo', u'lang': u'ca', u'p
rofile_background_tile': False, u'favourites_count': 1077, u'name': u'Bredotoijo', u'notifications': None, u'url': No
ne, u'created_at': u'Fri Nov 25 19:18:57 +0000 2011', u'contributors_enabled': False, u'time_zone': u'Madrid', u'prot
ected': False, u'default_profile': False, u'is_translator': False}, time_zone=u'Madrid', id=421293889, description=
u'', _api=<tweepy.api.API object at 0x0AF48BD0>, verified=False, profile_text_color=u'B88130', profile_image_url_http
s=u'https://pbs.twimg.com/profile_images/495032565234159616/0GshRD6O_normal.jpeg', profile_sidebar_fill_color=u'14121
4', is_translator=False, geo_enabled=True, entities={u'description': {u'urls': []}}, followers_count=94, protected=Fa
lse, id_str=u'421293889', default_profile_image=False, listed_count=7, lang=u'ca', utc_offset=7200, statuses_count=33
41, profile_background_color=u'1A1B1F', friends_count=297, profile_link_color=u'DD2E44', profile_image_url=u'http://p
bs.twimg.com/profile_images/495032565234159616/0GshRD6O_normal.jpeg', notifications=None, default_profile=False, prof
ile_background_image_url_https=u'https://pbs.twimg.com/profile_background_images/576670672191033346/bsBtGUcx.jpeg', p
rofile_banner_url=u'https://pbs.twimg.com/profile_banners/421293889/1469662497', profile_background_image_url=u'htt
p://pbs.twimg.com/profile_background_images/576670672191033346/bsBtGUcx.jpeg', name=u'Bredotoijo', is_translation_ena
bled=False, profile_background_tile=False, favourites_count=1077, screen_name=u'Bredotoijo', url=None, created_at=dat
etime.datetime(2011, 11, 25, 19, 18, 57), contributors_enabled=False, location=u'Catalunya', profile_sidebar_border_c
olor=u'121112', translator_type=u'none', following=False)
contributors : None
coordinates : {u'type': u'Point', u'coordinates': [2.18166758, 41.38348889]}
created_at : 2017-08-29 17:35:16
```

Después realizo lo mismo para comprobar la cuenta del usuario que lo ha emitido.

```
In [28]: user=tweet.author

         for param in dir(user):
             if not param.startswith("_"):
                 print "%s : %s" % (param, eval("user." + param))
```

A continuación muestro un pequeño fragmento de los resultados que he tenido que analizar de la parte del usuario.

```
contributors_enabled : False
created_at : 2011-11-25 19:18:57
default_profile : False
default_profile_image : False
description :
entities : {u'description': {u'urls': []}}
favourites_count : 1077
follow : <bound method User.follow of User(follow_request_sent=None, has_extended_profile=False, profile_use_backgr
ound_image=True, _json={u'follow_request_sent': None, u'has_extended_profile': False, u'profile_use_background_imag
e': True, u'default_profile_image': False, u'id': 421293889, u'profile_background_image_url_https': u'https://pbs.t
wimg.com/profile_background_images/576670672191033346/bsBtGUcx.jpeg', u'verified': False, u'translator_type': u'non
e', u'profile_text_color': u'B88130', u'profile_image_url_https': u'https://pbs.twimg.com/profile_images/4950325652
34159616/0GshRD6O_normal.jpeg', u'profile_sidebar_fill_color': u'141214', u'entities': {u'description': {u'urls':
[]}}, u'followers_count': 94, u'profile_sidebar_border_color': u'121112', u'id_str': u'421293889', u'profile_backgr
ound_color': u'1A1B1F', u'listed_count': 7, u'is_translation_enabled': False, u'utc_offset': 7200, u'statuses_coun
t': 3341, u'description': u'', u'friends_count': 297, u'location': u'Catalunya', u'profile_link_color': u'DD2E44',
u'profile_image_url': u'http://pbs.twimg.com/profile_images/495032565234159616/0GshRD6O_normal.jpeg', u'following':
None, u'geo_enabled': True, u'profile_banner_url': u'https://pbs.twimg.com/profile_banners/421293889/1469662497',
u'profile_background_image_url': u'http://pbs.twimg.com/profile_background_images/576670672191033346/bsBtGUcx.jpe
```

A continuación creo una lista que se va a ir llenando con los 100 primeros resultados que he obtenido en mi búsqueda.

```
In [29]: results = []
         for tweet in tweepy.Cursor(api.search,q="barcelona",geocode="41.3818,2.1685,2km").items(100):
             results.append(tweet)

         print len(results)
```

```
100
```

```
In [30]: def process_results(results):
             id_list = [tweet.id for tweet in results]
             data_set = pd.DataFrame(id_list, columns=["id"])

             # Processing Tweet Data

             data_set["text"] = [tweet.text for tweet in results]
             data_set["created_at"] = [tweet.created_at for tweet in results]
             data_set["retweet_count"] = [tweet.retweet_count for tweet in results]
             data_set["favorite_count"] = [tweet.favorite_count for tweet in results]
             data_set["source"] = [tweet.source for tweet in results]

             # Processing User Data
             data_set["user_id"] = [tweet.author.id for tweet in results]
             data_set["user_screen_name"] = [tweet.author.screen_name for tweet in results]
             data_set["user_name"] = [tweet.author.name for tweet in results]
             data_set["user_created_at"] = [tweet.author.created_at for tweet in results]
             data_set["user_description"] = [tweet.author.description for tweet in results]
             data_set["user_followers_count"] = [tweet.author.followers_count for tweet in results]
             data_set["user_friends_count"] = [tweet.author.friends_count for tweet in results]
             data_set["user_location"] = [tweet.author.location for tweet in results]

             return data_set
         data_set = process_results(results)
```

Pido una muestra de los 5 primeros resultados

```
In [31]: data_set.head(5)
```

		Barcelona P...	17:36:00						Alonso	0
1	902585640298176512	✶ à Barcelona, Spain https://t.co/IHTfihzfVK	2017-08-29 17:36:00	0	0	Instagram	2943878883	l_dplqs	lau✿	2 1
2	902585455291576320	Acaba de publicar una foto en El Born Barrie G...	2017-08-29 17:35:16	0	0	Instagram	421293889	Bredotoijo	Bredotoijo	2 1
3	902584519005356032	Didn't think this shot was anything special un...	2017-08-29 17:31:33	0	0	Instagram	61194772	nickpunxxx	niko	2 1
4	902584184098697221	finally🏠🌃 (@ Hotel Casa Fuster in Barcelon...	2017-08-29 17:30:13	0	0	Foursquare	1176841746	durraalth	دُرء	2 2

Pido una muestra de los 5 últimos resultados

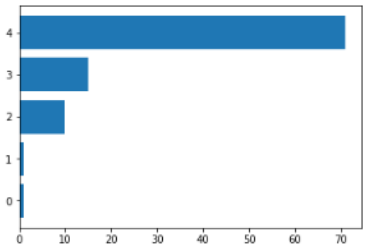
```
In [32]: data_set.tail(5)
```

Out[32]:

	id	text	created_at	retweet_count	favorite_count	source	user_id	user_screen_name	user_n
95	902527729987444736	RT @Bcn_Eixample: Aquest estiu gaudeix amb la ...	2017-08-29 13:45:53	3	0	Twitter for iPhone	828565613379538944	FinquesFutura	Finque Futura
96	902527615424237568	Barcelona was increíble🔌 but now we are che...	2017-08-29 13:45:26	0	0	Instagram	128765745	ashleyImahaffey	Ashley Mahaff
97	902527552358686720	I'm at Museu Nacional d'Art de Catalunya (MNAC...	2017-08-29 13:45:11	0	0	Foursquare	113367393	erhardignity	+
98	902527419764047872	'rosie', 'contaros' y 'remachadora' es ahora u...	2017-08-29 13:44:39	0	0	Trendsmap Alerting	187804659	TrendsBarcelona	Trends Barcel
99	902527389791657985	Uecastellbisbal, @uecastellbisbal es ahora una...	2017-08-29 13:44:32	0	0	Trendsmap Alerting	187804659	TrendsBarcelona	Trends Barcel

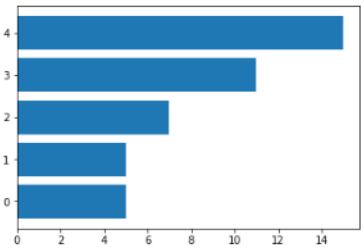
Pido un plot que muestre las 5 primeras fuentes de donde provienen los tweets.

```
In [33]: sources = data_set["source"].value_counts()[:5][::-1]

         plt.barh(xrange(len(sources)), sources.values)
         plt.show()
```



Pido un plot que muestre las 5 primeras nacionalidades donde se crearon las cuentas que han emitido los mensajes que he encontrado.

```
In [34]: sources = data_set["user_location"].value_counts()[:5][::-1]

         plt.barh(xrange(len(sources)), sources.values)
         plt.show()
```



A continuación muestro el cuadro que realicé para calcular la precisión de mis resultados con el Excel.

	Tweets Positivos			Tweets Negativos	
Tweet 1	TP	1	Tweet 1	FP	1
Tweet 2	TP	1	Tweet 2	FP	1
Tweet 3	TP	1	Tweet 3	FP	1
Tweet 4	TP	1	Tweet 4	TP	1
Tweet 5	TP	1	Tweet 5	TP	1
Tweet 6	TP	1	Tweet 6	TP	1
Tweet 7	FP	1	Tweet 7	TP	1
Tweet 8	TP	1	Tweet 8	TP	1
Tweet 9	TP	1	Tweet 9	FP	1
Tweet 10	TP	1	Tweet 10	TP	1
Tweet 11	TP	1	Tweet 11	FP	1
Tweet 12	TP	1	Tweet 12	FP	1
Tweet 13	TP	1	Tweet 13	TP	1
Tweet 14	TP	1	Tweet 14	TP	1
Tweet 15	FP	1	Tweet 15	TP	1
Tweet 16	TP	1	Tweet 16	TP	1
Tweet 17	TP	1	Tweet 17	TP	1
Tweet 18	FP	1	Tweet 18	TP	1
Tweet 19	TP	1	Tweet 19	TP	1

Tweet 20	TP	1	Tweet 20	TP	1
Tweet 21	TP	1	Tweet 21	TP	1
Tweet 22	TP	1	Tweet 22	TP	1
Tweet 23	TP	1	Tweet 23	TP	1
Tweet 24	TP	1	Tweet 24	FP	1
Tweet 25	TP	1	Tweet 25	FP	1
Tweet 26	FP	1	Tweet 26	TP	1
Tweet 27	TP	1	Tweet 27	TP	1
Tweet 28	TP	1	Tweet 28	TP	1
Tweet 29	TP	1	Tweet 29	TP	1
Tweet 30	TP	1	Tweet 30	FP	1
Tweet 31	FP	1	Tweet 31	TP	1
Tweet 32	FP	1	Tweet 32	FP	1
Tweet 33	FP	1	Tweet 33	TP	1
Tweet 34	TP	1	Tweet 34	TP	1
Tweet 35	TP	1	Tweet 35	TP	1
Tweet 36	FP	1	Tweet 36	TP	1
Tweet 37	TP	1	Tweet 37	TP	1
Tweet 38	TP	1	Tweet 38	TP	1
Tweet 39	TP	1	Tweet 39	FP	1
Tweet 40	TP	1	Tweet 40	TP	1
Tweet 41	TP	1	Tweet 41	FP	1
Tweet 42	TP	1	Tweet 42	TP	1
Tweet 43	TP	1	Tweet 43	FP	1
Tweet 44	TP	1	Tweet 44	TP	1
Tweet 45	TP	1	Tweet 45	FP	1
Tweet 46	TP	1	Tweet 46	FP	1
Tweet 47	TP	1	Tweet 47	TP	1
Tweet 48	TP	1	Tweet 48	TP	1
Tweet 49	TP	1	Tweet 49	TP	1
Tweet 50	TP	1	Tweet 50	TP	1
Tweet 51	TP	1	Tweet 51	TP	1
Tweet 52	TP	1	Tweet 52	TP	1
Tweet 53	TP	1	Tweet 53	TP	1
Tweet 54	TP	1	Tweet 54	TP	1
Tweet 55	FP	1	Tweet 55	TP	1
Tweet 56	TP	1	Tweet 56	TP	1
Tweet 57	TP	1	Tweet 57	FP	1
Tweet 58	FP	1	Tweet 58	TP	1
Tweet 59	FP	1	Tweet 59	TP	1
Tweet 60	TP	1	Tweet 60	TP	1
Tweet 61	TP	1	Tweet 61	TP	1
Tweet 62	FP	1	Tweet 62	TP	1
Tweet 63	FP	1	Tweet 63	FP	1
Tweet 64	TP	1	Tweet 64	FP	1
Tweet 65	TP	1	Tweet 65	TP	1

Tweet 66	TP	1
Tweet 67	TP	1
Tweet 68	TP	1
Tweet 69	TP	1
Tweet 70	TP	1
Tweet 71	FP	1
Tweet 72	TP	1
Tweet 73	TP	1
Tweet 74	FP	1
Tweet 75	TP	1
Tweet 76	FP	1
Tweet 77	TP	1
Tweet 78	TP	1
Tweet 79	TP	1
Tweet 80	TP	1
Tweet 81	TP	1
Tweet 82	TP	1
Tweet 83	TP	1
Tweet 84	TP	1
Tweet 85	TP	1
Tweet 86	TP	1
Tweet 87	TP	1
Tweet 88	FP	1
Tweet 89	TP	1
Tweet 90	FP	1
Tweet 91	TP	1
Tweet 92	TP	1
Tweet 93	TP	1
Tweet 94	TP	1
Tweet 95	TP	1
Tweet 96	TP	1
Tweet 97	FP	1
Tweet 98	TP	1
Tweet 99	TP	1
Tweet 100	TP	1
	TP	82
	FP	18
	P	0,82

Tweet 66	TP	1
Tweet 67	TP	1
Tweet 68	TP	1
Tweet 69	TP	1
Tweet 70	TP	1
Tweet 71	FP	1
Tweet 72	TP	1
Tweet 73	TP	1
Tweet 74	FP	1
Tweet 75	TP	1
Tweet 76	TP	1
Tweet 77	TP	1
Tweet 78	TP	1
Tweet 79	TP	1
Tweet 80	TP	1
Tweet 81	FP	1
Tweet 82	TP	1
Tweet 83	TP	1
Tweet 84	TP	1
Tweet 85	FP	1
Tweet 86	TP	1
Tweet 87	FP	1
Tweet 88	FP	1
Tweet 89	TP	1
Tweet 90	TP	1
Tweet 91	TP	1
Tweet 92	TP	1
Tweet 93	TP	1
Tweet 94	FP	1
Tweet 95	TP	1
Tweet 96	TP	1
Tweet 97	TP	1
Tweet 98	TP	1
Tweet 99	TP	1
Tweet 100	TP	1
	TP	76
	FP	24
	P	0,76