

Sprint 2: Búsqueda de funcionalidades para la herramienta

Jorge Martín Arévalo
Guillermo Novillo Díaz

Índice

1. Introducción	3
2. Funcionalidades identificadas	3
3. Control de versiones	4
4. Bibliografía	4

1. Introducción

El objetivo principal de nuestra aplicación es ofrecer a los usuarios de la red social Twitter una plataforma en la que poder visualizar de forma estructurada y separada su actividad en la aplicación a partir de los datos generados.

Para poder lograr este objetivo, vamos a identificar en este apartado las distintas funcionalidades que queremos implementar en nuestra herramienta, de modo que en primer lugar vamos a realizar un estudio de los aspectos más importantes que hemos podido identificar dentro del dataset que Twitter nos ofrece.

Este dataset consta de varios archivos JSON, que contienen información acerca de distintos aspectos generados dentro de la red social, como un listado de los tweets que han sido publicados por el usuario, un listado de los seguidores que tiene y los anuncios que han sido generados en base a la actividad, entre otras funcionalidades destacables.

2. Funcionalidades identificadas

Como vimos anteriormente y tras el análisis en detalle de cada uno de los archivos generados por Twitter, vamos a ver algunas de las funcionalidades que hemos pensado que podrían resultar de gran ayuda al usuario y no son ofrecidas directamente por la plataforma:

1. **Tracking del usuario durante el último período:** uno de los principales campos que podemos encontrar en el archivo *tweets.js* es la ubicación del usuario. Todavía no tenemos ningún perfil que tenga este campo relleno, pero tenemos varias alternativas en función del formato:
 - **Coordenadas latitud - longitud:** en caso de que se nos ofrezca en este formato, podemos utilizarlas para representarlas geográficamente en un mapa y poder realizar un listado de los lugares desde los que ha twiteado el usuario.
 - **Ciudad y País:** es otro de los posibles formatos en los que se puede mostrar la información. A partir de ellos es posible obtener las coordenadas (latitud y longitud) mediante el uso de la librería Geopy.
2. **Perfilar al usuario según sus gustos y anuncios:** las impresiones de los anuncios, contenidas en el fichero *ad-engadgements.js* guarda un registro de la configuración de los anuncios que le aparecen al usuario. Dentro de estas características tenemos una que es *matchedTargetingCriteria*, que clasifica los distintos criterios por tipo y valor. De este modo, para el análisis podemos hacer una especie de tabla en la que se guarden estas variables y así poder conocer el perfil de usuario que se trata. Por ejemplo si todos los *targetingTypes* de Age son 18 to 24 supondremos que la edad del usuario está en ese rango. La idea es hacer esto con todas las posibilidades de types que aparezcan.
3. **Círculo de amigos:** la idea de esta funcionalidad es poder realizar un recuento de todos los usuarios que se ha tenido una mayor actividad. Vamos a contar como actividad las distintas menciones, likes, retweets, conversaciones... Para ello, vamos a recurrir a distintos ficheros de los generados, como *tweets.js*, *twitter-circle-tweet.js*, *direct-messages.js*, *group-messages.js*.
4. **Análisis de sentimientos de los tweets:** la idea de esta visualización es mostrar al usuario un histórico de los últimos tweets que ha publicado (a lo mejor unos 50) y en base al contenido, poner al lado del tweet si conderamos que es positivo, negativo o neutral. Estos tweets los sacaremos del fichero *tweets.js*. Además de esto se puede hacer un PieChart en el que se muestre con porcentajes los tweets que se tienen positivos, negativos y neutrales.
5. **Mapa de calor con la actividad del usuario:** uno de los principales estudios que podemos mostrar al usuario es su actividad en forma de un mapa de calor. Este mapa de calor se movería con dos variables: los días de la semana y las horas, permitiendo ver las horas de mayor actividad durante el día, y los días de la semana con más actividad.

Dentro de esta representación, la idea es que podamos cambiar la visualización en función del tiempo que queramos (tentativamente última semana, último mes y últimos 15 días).

Esta actividad la vamos a considerar como todo aquello que aparezca una fecha. Los ficheros que tendremos que consultar van a ser *ad-impresions.js*, *direct-messages.js*, *tweets.js*, *group-messages.js* y *user-link-clicks.js*.
6. **Usuarios a los que más has mencionado:** para los 10 usuarios que más hayas mencionado en tus tweets publicados, mostrar un gráfico de barras que, para cada uno de ellos, represente la cantidad de menciones que se han hecho.

7. **Porcentaje de lenguajes en los que se ha twiteado:** la idea es reconocer mediante librerías de Python los idiomas en los que se encuentran los tweets, representando en una piechart los porcentajes con los idiomas en los que se ha tweeteado.
8. **Perfil:** No es una funcionalidad como tal, pero tener como un miniperfil de Twitter, que salga el correo que tienes asociado, tu nombre de usuario y la fecha en la que se creó la cuenta (*account.js*). También puede consultarse la edad real del usuario en el fichero *age.js*.
También se puede hacer algo con los JSON del fichero *key-registry.js*, que guarda los datos de los dispositivos en los que se tiene abierta una sesión de la cuenta.
9. Por último, una frase que simbolice la huella digital que tiene el usuario. Por ejemplo, tus datos ocupan lo mismo que el Quijote (se haría o bien con las imágenes o con el tamaño total del fichero de datos).

3. Control de versiones

Para realizar un correcto control de las distintas versiones del proyecto y realizar recuperaciones en caso de que sean necesarias vamos a utilizar Github. Dentro de Github vamos a almacenar una copia del código del proyecto, así como una copia de varios datasets generados por Twitter que vamos a utilizar para realizar las sucesivas pruebas.

4. Bibliografía

- Ejemplo de datos extraídos de Twitter