UNIVERSITAT POLITÈCNICA DE VALÈNCIA

etsinf

Escola Tècnica Superior d'Enginyeria Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica
Universitat Politècnica de València

Desarrollo de un sistema de seguimiento para Instagram

TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

Autor: Jose Sebastián Canós

Tutor: Ferran Pla Santamaría

Cotutor: Lluís Felip Hurtado Oliver

Curso 2016-2017

Resum

Instagram és una xarxa social que permet crear i compartir imatges i vídeos amb altres usuaris d'Internet. Això permet fer un seguiment d'usuaris, trobar contigut específic i estudiar la seua reacció. Instagram s'ha convertit en l'aplicació més popular d'aquest tipus de contingut a smartphones, per això resulta interessant desenvolupar eines d'anàlisi. L'objectiu del projecte és desenvolupar un sistema de seguiment dels posts dels usuaris d'Instagram i el seu anàlisi. El seguiment es realitzarà per usuari o per etiqueta i l'anàlisi inclourà un estudi dels comentaris dels usuaris i les seues interaccions.

Paraules clau: Anàlisi de xarxes socials, Instagram, Extracció de dades web

---

Resumen

Instagram es una red social que permite crear y compartir imágenes y vídeos con otros usuarios de Internet. Esto permite hacer un seguimiento de usuarios, encontrar contenido específico y estudiar su reacción. Instagram se ha convertido en la aplicación más popular de este tipo de contenido en smartphones, por ello resulta interesante desarrollar herramientas de análisis. El objetivo del proyecto es desarrollar un sistema de seguimiento de los posts de los usuarios de Instagram y su análisis. El seguimiento se realizará por usuario o por etiqueta y el análisis incluirá un estudio de los comentarios de los usuarios y las interacciones entre estos.

Palabras clave: Análisis de redes sociales, Instagram, Extracción de datos web

---

Abstract

Instagram is a social network that allows us to create and share images and videos with other users of Internet. This allows us to track users, find specific content and study their reaction. Instagram has become the most popular application of this type of content in smartphones. Therefore it is interesting to develop analysis tools. The objective of the project is to develop a system to monitor the posts of Instagram users and their analysis. The monitoring will be done by user or by tag and the analysis will include a study of the comments of the users and the interactions between them.

Key words: Social Network analysis, Instagram, Web scraping

Índice general

Introducción

1.1 Motivación

Actualmente algunas redes sociales son utilizadas para comunicar información y respuestas sobre prácticamente cualquier ámbito, tanto privado como público. Los usuarios de las redes sociales pueden usar texto y también signos no lingüísticos, como emoticonos o imágenes. En ocasiones ambas formas de comunicación verbal y no verbal pueden ser clasificadas, posibilitando a su vez la clasificación del mensaje. Conocer la clase de experiencias de los usuarios de las redes sociales, de manera automática, puede ser útil para extraer conclusiones, a lo largo del tiempo, sobre el comportamiento de los usuarios o la reputación de diferentes personajes públicos, tendencias, marcas comerciales, posturas políticas, etc.

Instagram se ha convertido en una de las aplicaciones de publicación de fotos más populares en smartphones, con 700 millones de usuarios activos (Constine, 2017). En las publicaciones esta aplicación puede registrar comentarios y reacciones no verbales, como el botón «me gusta». Por lo expuesto anteriormente, puede ser interesante desarrollar una herramienta capaz de captar y almacenar esta actividad de Instagram, para posteriormente facilitar su análisis en conjunto.

Para captar y almacenar la actividad de Instagram, podría no ser necesario un sistema de seguimiento, hay redes sociales, como por ejemplo Twitter, que en su plataforma permiten la búsqueda del contenido existente desde el origen de la red social, y/o especificando un período de tiempo. En ese caso, podría ser suficiente cada vez que se deseara realizar un análisis, buscar y almacenar todo el contenido deseado. Sin embargo, Instagram no ofrece esta posibilidad de búsqueda especializada, y en ocasiones, se limita a mostrar el contenido más reciente. Por esta razón, en este proyecto se desarrolla un sistema de seguimiento, que nos permite capturar periódicamente la actividad deseada de Instagram.

1.2 Objetivos

El objetivo de este trabajo es desarrollar un sistema de seguimiento de publicaciones de Instagram, permitiendo como criterio de búsqueda un usuario o etiqueta, en este caso *hashtag*. Se busca facilitar el acceso a estas publicaciones específicas mediante una recopilación de ellas, incorporando automáticamente nuevas publicaciones a medida que ocurran, e incluyendo las reacciones y comentarios que llevan consigo, para luego poder ser analizadas en conjunto. Sin embargo, este sistema no es completamente autónomo, depende del

funcionamiento de otros agentes, por lo que se deben hacer asunciones para evitar que el alcance del trabajo sea demasiado ambicioso:

1. Se asume que Instagram utiliza las mismas estructuras y formatos para representar y comunicar datos a lo largo del tiempo.
2. Se asume que el sistema de seguimiento dispondrá de una conexión ininterrumpida a los servicios de Instagram, a través de Internet, para su correcto funcionamiento.
3. Se asume que las solicitudes resueltas por Instagram devuelven todos los objetos y datos individuales que cumplen los requisitos deseados.
4. Se ignora el control de seguridad del sistema mediante autentificación. Este es un aspecto a tener en cuenta, especialmente en el uso de un servicio de base de datos, pero se asume que sólo el sistema de seguimiento y posteriores servicios de análisis están autorizados para utilizar el sistema de almacenamiento.

Una vez definidas las restricciones y supuestos, se puede definir los objetivos concretos a alcanzar. El sistema de seguimiento debe ser capaz de:

1. Obtener la información de publicaciones de Instagram de un usuario público, a partir de su nombre de usuario.
2. Obtener la información de publicaciones de Instagram públicas que utilicen un *hashtag*, a partir del propio *hashtag*.
3. Escoger los servicios de Instagram que permitan obtener la mayor cantidad de información no redundante sobre cada publicación.
4. Almacenar la información de las publicaciones de Instagram obtenidas, en un servicio de base de datos, evitando la duplicación de información.
5. Buscar periódicamente publicaciones nuevas, asociadas a un usuario o *hashtag* determinado, que no hayan sido almacenadas previamente.
6. Actualizar las publicaciones de Instagram almacenadas, cuando se considere que han sufrido cambios o incorporación de información nueva.

Contexto general

En esta sección se explica qué es Instagram, el funcionamiento para sus usuarios, y qué alternativas proporciona Instagram a desarrolladores de software para acceder a la información sobre sus usuarios y publicaciones. También se incluye un último apartado sobre el estado del arte en sistemas de extracción de datos web.

2.1 Instagram

Actualmente Instagram es una aplicación dirigida principalmente a ser usada en *smartphones*, es compatible con las últimas versiones de los sistemas operativos iOS, Android y Windows. Esta aplicación permite a sus usuarios registrados aplicar filtros digitales a fotos y vídeos y publicarlos. Un usuario puede en sus publicaciones etiquetar a otros usuarios y añadir un pie de foto, que de aquí en adelante llamaremos *caption*.

Un usuario de Instagram puede seguir a otros usuarios, y así en su canal de noticias, dentro de la aplicación, aparecen las publicaciones de aquellos usuarios a los que sigue. Además, cada cuenta de usuario puede ser pública o privada. Si una cuenta es privada es necesario el permiso de su usuario para poder observar sus publicaciones. Si una cuenta es pública, cualquier usuario de Instagram puede seguirla y sólo es necesario tener acceso a Internet para observar sus publicaciones.

Los usuarios de Instagram, en aquellas publicaciones que son capaces de visitar, pueden escribir comentarios, que quedan asociados a la publicación, así como marcar la publicación con un *like* o «me gusta». Los comentarios y las *captions* tienen los mismos límites y funcionan como formas de texto equivalentes. Tanto en una *caption* como en un comentario, el límite de caracteres que lo forman es de dos mil doscientos caracteres, estos caracteres pueden ser cualquiera presente en el estándar Unicode.

En el texto que forman las *captions* y comentarios, Instagram puede identificar usuarios y etiquetas usando dos tipos de cadenas especiales. Un usuario se puede identificar con su nombre de usuario precedido por el símbolo arroba, tal que «@nombre.de.usuario». Por otra parte, una etiqueta o *hashtag*, se identifica con cualquier cadena de caracteres precedida por el símbolo almohadilla, sin distinguir entre mayúsculas y minúsculas, tal que «#etiqueta». Los *hashtags* son utilizados para etiquetar publicaciones o textos con un tema, identificado por el propio *hashtag*.

A continuación, en la figura 2.1 hay un ejemplo de publicación de la usuaria «laurarosely», que en su *caption* utiliza el *hashtag* «#helloworld», y en uno de los comentarios de la publicación, se dirigen expresamente a ella utilizando «@laurarosely».

Figura 2.1: Ejemplo de publicación de Instagram.

Instagram incorpora un motor de búsqueda que puede devolver diferentes tipos de resultados. En la figura 2.2 se muestra un ejemplo de búsqueda con el término «glasgow», a continuación se explican los resultados de la búsqueda:

- Con el símbolo de localización, aparecen dos resultados: «Glasgow, United Kingdom» y «University of Glasgow». En Instagram las localizaciones geográficas se identifican, y recogen publicaciones recientes que se hayan asociado a la localización.
- Con el símbolo de almohadilla aparece el resultado «glasgow», el cual recoge publicaciones recientes que contienen el *hashtag* «#glasgow».
- Con imágenes enmarcadas en círculos aparecen «paesanopizzaglasgow» y «glasgowphotography1», los cuales son usuarios de Instagram, y sirven de enlace hacia las propias páginas en Instagram de los usuarios, donde se pueden visitar todas sus publicaciones y su perfil.

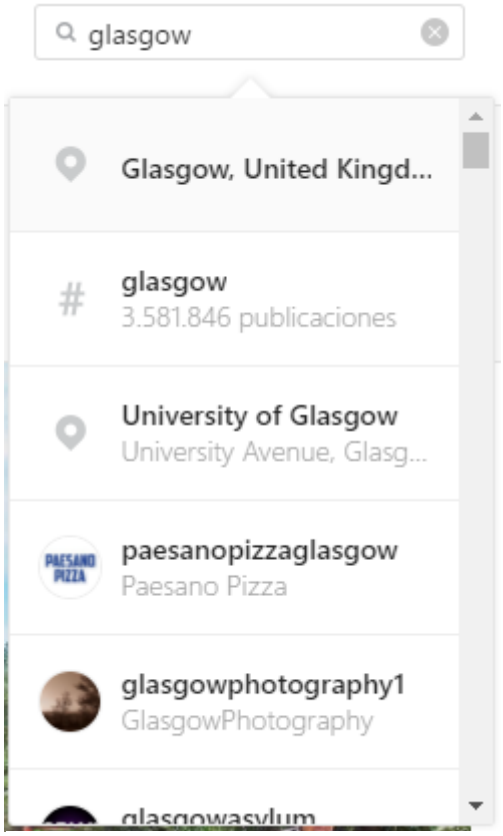

Figura 2.2: Ejemplo de búsqueda en Instagram.

Acceder a uno de estos resultados mostraría aquellas publicaciones que cada resultado recoge, ordenadas de más reciente a más antigua.

Instagram, aparte de las publicaciones, ofrece otros tipos de actividad a sus usuarios, como las *Instagram Stories*, pero es un tipo de contenido con una mayor base audiovisual y que se elimina a las veinticuatro horas de su creación, por lo que no se toman en consideración por el sistema de seguimiento a desarrollar.

2.2 Instagram Endpoints

En las redes de telecomunicaciones, el término *endpoint* se entiende como un tipo de nodo en la red, que presenta una interfaz a través de un canal de comunicación. Un *endpoint* facilitaría, a sistemas software, una capa de abstracción con la que comunicarse.

Instagram utiliza y provee varias clases de *endpoints* que permiten manejar cuentas de usuarios, publicaciones, buscar contenido, etc. A continuación, se exponen estas clases de *endpoints*, con algunos de sus requisitos e inconvenientes, con tal de mostrar al lector qué diferentes alternativas tiene un desarrollador para trabajar con la plataforma de Instagram.

2.2.1 Instagram API Platform

Instagram proporciona oficialmente la *Instagram API Platform* a desarrolladores de software, con el requisito de que se produzcan aplicaciones y servicios, no automatizados, que según ellos mismos (Instagram, 2015), ayuden a que:

- Los **usuarios publiquen su propio contenido** en Instagram a través de estas aplicaciones de terceros, y no con la aplicación oficial.
- Las **marcas y anunciantes** comprendan y manejen su audiencia y derechos de contenido.
- Las **emisoras y canales** descubran contenido, obtengan derechos sobre su propio material y permitan la publicación de este material con derechos de autor de forma apropiada.

Esta API proporciona *endpoints* para buscar y manejar: cuentas de usuario, publicaciones, comentarios, *likes*, *hashtags*, localizaciones, etc. Entre estas funciones se encuentra la búsqueda de publicaciones por usuario o *hashtag*.

Para utilizar esta API es necesario registrar previamente la aplicación a desarrollar, una vez hecho esto, la aplicación debe iniciar sesión con un usuario de Instagram cada vez que se utilice. Al principio del desarrollo, la API permite a la aplicación acceder a la información compuesta por las publicaciones y cuentas de un máximo de veinte usuarios, los cuales deben haber dado su permiso. Más adelante, una vez finalizado el desarrollo de la aplicación, se puede pedir permisos para que la aplicación sea capaz de acceder a toda la información pública de Instagram. Y finalmente, después de una evaluación de la aplicación, en caso satisfactorio, Instagram concederá sólo ciertos permisos según el funcionamiento de la aplicación.

Esta API está dirigida al desarrollo de aplicaciones que sean usadas por cualquier usuario de Instagram, iniciando sesión con su propia cuenta y credenciales, de un modo similar a la propia aplicación de Instagram para *smartphones*. Sin embargo, el sistema a desarrollar en este trabajo no busca ser este tipo de aplicación, ni necesitar el permiso expreso de Instagram, en consecuencia, el sistema de seguimiento no utilizará la *Instagram API Platform*.

2.2.2 Instagram Private API

La aplicación oficial de Instagram en *smartphones* no utiliza la *Instagram API Platform* vista anteriormente, sino que utiliza una API privada de Instagram, que para ser usada requiere las credenciales de un usuario de Instagram y una clave secreta de encriptación incorporada en la propia aplicación oficial. Nos referiremos a esta API como *Instagram Private API* de aquí en adelante.

La *Instagram API Platform* utiliza el protocolo OAuth 2.0 para autenticar a sus usuarios, sin entrar en gran detalle, esto significa que, un usuario que desee utilizar una aplicación que se apoye en esta API, debe ser redirigido a una dirección URL de autenticación de Instagram (Instagram, 2015). En esta dirección, el usuario debe iniciar sesión con su cuenta de Instagram y autorizar a la aplicación el acceso y uso de su cuenta. Una vez autorizada la aplicación, Instagram otorga un identificador de acceso, con el cual la aplicación puede hacer todas sus solicitudes a la API.

Por el contrario, cuando un usuario inicia sesión en la aplicación oficial de Instagram, el usuario escribe sus credenciales, estas son cifradas utilizando una clave secreta y el resultado se envía a los servidores de Instagram. Si las credenciales son válidas, el servidor devuelve un identificador de sesión que puede ser usado para las siguientes solicitudes por la aplicación oficial (Balochi, 2016). Sin embargo, esto significa que la clave secreta de cifrado está almacenada en la aplicación oficial, si la clave es extraída de la aplicación, otros agentes podrían usarla para tramitar con éxito solicitudes a los servidores de Instagram, emulando la aplicación oficial de Instagram. De hecho, en Internet se encuentran decenas de librerías para diferentes lenguajes de programación que usan la *Instagram Private API* con estas claves.

En las librerías mencionadas, el único requisito para utilizar la *Instagram Private API* es iniciar sesión con un usuario de Instagram. En estas librerías, los *endpoints* de la API utilizados se han obtenido mediante ingeniería inversa, e incluyen prácticamente todos los implementados por la *Instagram API Platform*. Estos *endpoints* permiten el acceso a usuarios, publicaciones, comentarios, *likes*, etc., e incluyen la búsqueda de publicaciones por usuario, *hashtag* o localización.

2.2.3 Instagram Web API

Aparte de la aplicación oficial, Instagram ofrece navegar por el contenido de su plataforma a través de su propio dominio web:

https://www.instagram.com

Si visitas la página, te invitará a crear una cuenta de usuario en Instagram o iniciar sesión con una existente. Sin embargo, esto no es necesario para explorar el contenido, puedes acceder a otra página del dominio, por ejemplo:

https://www.instagram.com/explore/tags/glasgow/

Esta última página muestra las publicaciones que contienen el *hashtag* «#glasgow», y también incluye, en el espacio superior, un buscador como el mostrado anteriormente en la figura 2.2. Este buscador es el que permite desplazarse y encontrar contenido por Instagram.

La página web de Instagram también tiene su propia API, distinta de la *Instagram API Platform* y de la *Instagram Private API*, nos referiremos a ella como *Instagram Web API* de aquí en adelante. Cuando navegamos por una página web de Instagram, internamente se realizan solicitudes a esta *Instagram Web API*, si se utiliza un navegador web, con algunas herramientas para desarrolladores, es posible conocer los parámetros de estas solicitudes y por ingeniería inversa deducir su funcionamiento (Martin, 2017).

Las solicitudes a la *Instagram Web API* en general van dirigidas a unas URLs que comienzan con el siguiente formato:

https://www.instagram.com/graphql/query/?query_id=<num_query>&...

Se observan las cadenas «graphql» y «query». *GraphQL* es un lenguaje de consultas para APIs, que puede operar a través de un servidor HTTP. En este caso, la *Web API* utiliza *GraphQL* a través de esta URL, como único *endpoint*.

La mayoría de las solicitudes utiliza esta URL intercambiando <num_query> por un número entero, que varía según el tipo de consulta, y añadiendo posteriormente, otros parámetros pertinentes para el tipo de consulta. Por ejemplo, cuando se piden las publicaciones asociadas a un *hashtag*, <num_query> es siempre un número determinado y se añade la cadena del *hashtag* en los demás parámetros de la consulta. En el caso anterior del *hashtag* «#glasgow», se realizaría una solicitud a la *Instagram Web API* con la siguiente terminación de URL:

.../?query_id=17882293912014529&tag_name=glasgow&first=10

Se han añadido los parámetros: «tag_name», que corresponde a la cadena del *hashtag* sin la almohadilla, y «first», que sirve para especificar el número de objetos mínimos deseados, en este caso, número de publicaciones. El número entero, o <num_query>, del tipo de consulta y los demás parámetros pertinentes, se pueden deducir por ingeniería inversa como se mencionaba anteriormente (Martin, 2017).

Las solicitudes a la *Web API*, de búsqueda de contenido público, no requieren la autenticación de un usuario de Instagram, y se pueden realizar accediendo a la URL del *endpoint* de *GraphQL* desde, por ejemplo, un navegador web corriente. La *Web API* proporciona tipos de consulta para buscar publicaciones por nombre de usuario o por *hashtag*, comentarios de una publicación, seguidores de un usuario, los usuarios seguidos por un usuario, etc. Estas solicitudes devuelven un número finito de objetos, por lo que puede haber más objetos que pertenecen a la consulta y no se han devuelto. La *Web API* soluciona este problema con la paginación de las consultas por medio de un cursor, utilizando un parámetro «after» en la URL. Cada respuesta incluye el valor que «after» debe tomar, para que se devuelva la siguiente página de resultados de una misma consulta. A pesar de esta paginación, en la búsqueda de publicaciones por *hashtag* con cualquier API, Instagram limita las publicaciones devueltas a sólo las seis mil más recientes aproximadamente.

2.3 JSON

La respuesta de las solicitudes a las APIs de Instagram generalmente incluye un objeto en formato JSON que contiene la información pedida. JSON es un formato de texto, utilizado para transmitir colecciones de pares nombre-valor, y listas ordenadas de valores. Instagram representa la información de, por ejemplo una publicación, usando sólo esta estructura; pero las imágenes y vídeos de la publicación, no se transmiten en el objeto JSON, sino que se incluye una URL desde donde descargarlos, Instagram siempre almacena el contenido audiovisual de las publicaciones en sus propios servidores.

Sin embargo, Instagram no utiliza los mismos pares nombre-valor de JSON para enviar la información de una misma publicación por las diferentes APIs, *endpoints*, o tipos de consulta, incluso desde de una misma API. Según el método utilizado se puede obtener una mayor riqueza de información sobre un mismo objeto de Instagram. En la figura 2.3 se muestra una representación JSON de una publicación, generada por la *Web API*:


```
{
  "graphql": {
    "shortcode_media": {
      "__typename": "GraphSidecar",
      "id": "1550572512717725918",
      "shortcode": "BWEvXwrBmDe",
      "dimensions": { … }, // 2 items
      "gating_info": null,
      "media_preview": null,
      "display_url": "https://scontent-mad1-1.cdninstagram.com/t51.2885-
      15/e35/19534603_228784527641511_3527434611317538816_n.jpg",
      "is_video": false,
      "edge_media_to_tagged_user": { … }, // 1 item
      "edge_media_to_caption": { … }, // 1 item
      "caption_is_edited": false,
      "edge_media_to_comment": {
        "count": 3,
        "page_info": { … }, // 2 items
        "edges": [ … ] // 3 items
      },
      "comments_disabled": false,
      "taken_at_timestamp": 1499062669,
      "edge_media_preview_like": { … }, // 2 items
      "edge_media_to_sponsor_user": { … }, // 1 item
      "location": null,
      "viewer_has_liked": false,
      "owner": { … }, // 10 items
      "is_ad": false,
      "edge_web_media_to_related_media": { … }, // 1 item
      "edge_sidecar_to_children": { … } // 1 item
    }
  }
}
```


Figura 2.3: Ejemplo de publicación de Instagram en formato JSON.

Los identificadores de las publicaciones de Instagram son los valores de los atributos “id” y “shortcode”, las APIs usan prevalentemente uno de los dos.

2.4 Estado del arte

2.4.1 Sistemas de extracción de datos web

El sistema de seguimiento para Instagram a desarrollar en este trabajo se podría clasificar como un ejemplo muy específico de sistema de extracción de datos web, o *web scraping*. Uno de los trabajos recientes que recoge el estado del arte sobre estos sistemas es el artículo de (Grigalis & Čenys, 2013). Se trata de un artículo de recopilación, que presenta diferentes usos de estos sistemas, distingue las funciones de un sistema de extracción de datos web, y revisa algunas soluciones comerciales y propuestas. Divide las tareas de extracción en cinco funciones:

- **Interacción con un sitio web**, incluye la navegación por las páginas web predeterminadas que contienen datos deseados.
- **Identificación, extracción y conversión** de los datos deseados en las páginas web a un formato estructurado.
- **Programación de tareas** de extracción, visitando periódicamente las páginas web en busca de cambios e información nueva.
- **Transformación de los datos**, incluye el filtro y transformación de la información a un formato de salida, generalmente uno compatible con un sistema de bases de datos.
- **Suministro** de los datos extraídos y estructurados, a un sistema de almacenamiento y/o análisis.

Hoy en día hay propuestas para que los sistemas de extracción de datos web se aproximen a una menor necesidad de supervisión humana, a cambio de que exista un mayor grado de estructuración de la información en la *World Wide Web*. Con ello quieren facilitar tareas complejas, como por ejemplo la búsqueda de restaurantes cercanos que sirvan cierta receta, reduciendo el tiempo invertido personalmente, respecto a introducir palabras clave en un buscador web actual. Algunas de estas propuestas estriban su éxito en la publicación generalizada de objetos web que sigan una metodología o formato determinados. En cambio, otras propuestas se apoyan más en la colaboración de multitud de personas coleccionando y compartiendo información, de forma análoga a, por ejemplo, cómo se mantiene Wikipedia.

2.4.2 Sentiment Analysis

Este sistema de seguimiento para Instagram está enfocado a uno de los usos u objetivos de los sistemas de extracción de datos web recogidos por (Grigalis & Čenys, 2013). Se trata del *sentiment analysis*, muchos usuarios de redes sociales como Facebook, Twitter o Instagram comparten sus experiencias en estas

plataformas. Estas experiencias pueden ser sobre productos comerciales, personajes públicos o ideas políticas.

El *sentiment analysis* se refiere al uso del procesamiento de lenguaje natural, por ejemplo el texto de estas experiencias en redes sociales, con tal de extraer, del lenguaje, la opinión subjetiva subyacente. En el caso de un negocio, analizar la opinión de los clientes es fundamental para mantener y mejorar la competitividad de un producto o servicio, el uso del *sentiment analysis* podría otorgar una ventaja o facilitar este análisis del cliente.

Extraer la opinión de un texto es un problema complejo, el *sentiment analysis* se ha convertido en una de las áreas de investigación más activas en el procesamiento del lenguaje natural. Existen libros de texto que explican la complejidad de este problema y los métodos más generalizados de abordarlo, uno de ellos es el de (Liu, 2012).

En este libro se exponen tres subproblemas o niveles del *sentiment analysis*. El primero de ellos es clasificar si todo un documento de opinión expresa un sentimiento positivo o negativo acerca de una sola entidad, como por ejemplo cuál es la opinión final de la reseña de un producto. El segundo nivel es clasificar si cada frase individualmente expresa una opinión positiva, negativa, o no expresa ninguna opinión, en cuyo caso sería neutral. Este último nivel está relacionado con distinguir si una frase presenta hechos y es objetiva, o expresa una opinión subjetiva. El tercer nivel es la identificación de entidades y aspectos en una opinión. Una opinión consiste de un sentimiento, positivo o negativo, y de un objetivo, aquello sobre lo que el sentimiento se expresa. En la frase «la pantalla de mi móvil es buena, pero no le dura nada la batería» expresa dos opiniones sobre los aspectos «pantalla» y «batería» de la entidad «mi móvil». La meta de esta última tarea es la estructuración de cada opinión en un sentimiento, y el aspecto y/o entidad sobre el que se expresa.

La clasificación de un documento como positivo o negativo es una clase de problema para la que se pueden utilizar métodos de aprendizaje automático supervisado. Estos métodos usan unos elementos de entrenamiento, cada elemento es un par que consta de un objeto de entrada y un valor de salida deseado. Con los ejemplos de entrenamiento, estos métodos crean una función que intenta predecir el valor deseado a partir del objeto de entrada. El objetivo final es que la función prediga correctamente el valor de salida no sólo para los elementos de entrenamiento, sino también para elementos de la misma clase que no estuvieran en este conjunto de entrenamiento. Los métodos de aprendizaje automático supervisado pueden considerar muchas características del texto: frecuencia de los términos, categorías gramaticales (un adjetivo puede tener más importancia que una preposición), palabras con sentimiento (por ejemplo: amar, idiota, mierda, increíble, etc.), negaciones (un «no» podría cambiar el sentimiento de toda una frase)…

Existen otras clases de métodos para clasificar documentos y aplicar *sentiment analysis*. Pero para redes sociales como Twitter, cuyas publicaciones constan principalmente de texto y están limitadas a ciento cuarenta caracteres, numerosos equipos de investigación han desarrollado métodos de aprendizaje automático supervisado. Estos métodos de *sentiment analysis* para Twitter generalmente se valen de corpus y léxicos, extraídos de publicaciones de Twitter, que han sido debidamente procesados para su uso por los métodos.

En cambio en Instagram, aunque se utilice texto, el elemento principal de las publicaciones son las imágenes y vídeos. Estos elementos audiovisuales aumentan la complejidad de la aplicación de *sentiment analysis*, en comparación con Twitter. Esto es debido a que las imágenes pueden ser esenciales para un correcto análisis del texto, aún si sólo consideráramos cada comentario individualmente.

Por lo expuesto anteriormente, aún no se han publicado avances en investigación sobre soluciones holísticas de *sentiment analysis* en Instagram, que integren un análisis de las imágenes. Sin embargo, existen ejemplos de trabajos que sólo tienen en cuenta los comentarios de publicaciones para aplicar *sentiment analysis*, como el desarrollado por (AbdelFattah, Galal, Hassan, Doaa, & Tallent, 2017). El objetivo de este trabajo es desarrollar una herramienta que, de forma automática, pueda recomendar a un usuario cuáles son las mejores características a incluir en las imágenes de sus publicaciones. Para ello intentan cuantificar el *valor social* de una publicación de Instagram a partir de sus comentarios. Sin embargo, esta herramienta aún no se considera validada, no han mostrado ningún resultado que pueda ser juzgado por conocimiento experto.

Para desarrollar la herramienta han concentrado su análisis en las reacciones generadas por las publicaciones de las cincuenta marcas de moda más seguidas en Instagram. La herramienta, para cuantificar el *valor social* de una publicación extrae los últimos ciento cincuenta comentarios de una publicación. Preprocesa los comentarios para extraer sólo aquellas partes que puedan expresar una opinión. Después, con un método de *sentiment analysis*, puntúa cada palabra con un valor de 1 a -1, de más positivo a más negativo respectivamente, siendo 0 el valor neutro. Para la herramienta, la puntuación de un comentario es la suma de las puntuaciones de sus palabras. Y por último, el *valor social* de una publicación es el promedio de las puntuaciones de sus comentarios.

Implementación de un sistema de seguimiento

En esta sección se explica la estructura y el funcionamiento del sistema de seguimiento desarrollado. Sin embargo, no se van a mostrar los detalles del código porque no son estrictamente imprescindibles para comprender el sistema. Los métodos están comentados en inglés en el propio código, pero tampoco se va a incluir el código en un apéndice debido a su extensión, el autor cree que puede ser un gasto de papel innecesario. Por estos motivos se deja a disposición del lector la siguiente dirección donde puede encontrar el código:

https://github.com/ep3p/TFG-ep3p

3.1 Tecnologías empleadas

Para el desarrollo del código se han utilizado conceptos vistos a lo largo de la carrera como programación orientada a objetos, estructuras de datos, concurrencia, expresiones regulares, etc. Pero a la hora de concretar tecnologías empleadas, se pueden resumir en las siguientes:

- **Python**: Se ha escogido este lenguaje de programación por su alto nivel de abstracción, dinamismo, sencillez, y ser un lenguaje de propósito general. Python tiene una librería estándar muy completa y soporta la ejecución de scripts, además de facilitar la creación y uso de librerías no estándar. Existen marcos de trabajo, o *frameworks,* web para Python, como Django, que permitirían usar el sistema de seguimiento alrededor de un entorno web. Y también existen librerías orientadas a la computación científica, como NumPy, que facilitarían el uso del *sentiment analysis*.
- **ping/instagram_private_api**: Es una librería para Python poco extendida, que se puede encontrar en el portal GitHub, y sólo tiene un único autor. Se actualiza frecuentemente y está documentada. Ha sido utilizada porque ofrece métodos para realizar solicitudes a la *Instagram Private API*, facilitando una capa de abstracción y soporte de errores. Cabe añadir que actualmente hay una alternativa más apropiada para este sistema de seguimiento, sólo que ha sido creada y desarrollada a la par que este sistema, y descubierta durante la realización de la memoria del trabajo.
- **MongoDB**: Es un sistema de base de datos NoSQL, de código abierto y multiplataforma. Se ha escogido porque está orientada a almacenar documentos de tipo JSON o similar, a través de una especificación llamada BSON, y porque puede trabajar con Python a través de la librería PyMongo. Este sistema de base de datos agrupa sus documentos en colecciones, y a su vez estas colecciones en bases de datos. Un solo servicio MongoDB puede mantener varias bases de datos, a través de una única dirección.

3.2 Estructura del código

El sistema de seguimiento para Instagram utiliza un servicio MongoDB con el que se comunica en cada ejecución. El sistema trabaja a través de varias bases de datos del mismo servicio MongoDB, cuyos propósitos son almacenar publicaciones y comentarios de Instagram por separado. En el sistema de seguimiento intervienen los siguientes archivos:

- **searcher.py**: Contiene la clase *InstagramSearcher*, una instancia de esta clase puede realizar solicitudes en la *Instagram Web API* y la *Instagram Private API*. A partir de una cadena y un periodo de tiempo, puede buscar publicaciones asociadas a la cadena, tanto si la cadena es el nombre de un usuario o un *hashtag*. También puede descargar, a partir de la identificación de una publicación de Instagram, la información de la publicación y sus comentarios. Puede utilizar múltiples hilos de ejecución para acelerar la descarga de una cola de publicaciones, y se encarga de manejar las excepciones más comunes durante la descarga.
- **mongo_frontend.py**: Contiene la clase *MongoFrontEnd*, una instancia de esta clase sirve para comunicarse con un servicio MongoDB, y utilizar las bases de datos y colecciones del servicio. Puede almacenar en una base de datos documentos en formato JSON, buscar y actualizar documentos, y unir colecciones de documentos en una sola colección evitando la repetición de documentos.
- **monitor.py**: Contiene la clase *InstagramMonitor*, una instancia de esta clase utiliza instancias de *InstagramSearcher* y *MongoFrontEnd* para obtener y almacenar publicaciones asociadas a una consulta, que sería la cadena de un nombre de usuario o de un *hashtag*. Detecta las publicaciones previamente almacenadas de una consulta, y puede actualizarlas si se considera que han sufrido cambios o recibido nuevos comentarios. Marca las publicaciones, que considera que no sufrirán cambios, como archivadas.

  También puede generar archivos asociados a una consulta para facilitar un análisis externo: archivos que contienen todas las *captions* y comentarios, o archivos que representan grafos cuyos nodos son usuarios y las aristas son menciones de un usuario a otro en los textos de las publicaciones.
- **queries.txt**: Contiene la lista de las consultas, en forma de cadenas, a seguir por el sistema.
- **__main__.py**: Es el punto de entrada del sistema de seguimiento, utiliza una instancia de *Instagram Monitor* para realizar y actualizar consultas. Lee el archivo *queries.txt* para obtener las cadenas de las consultas a seguir. Pide un usuario y contraseña de Instagram con los que iniciar sesión y poder utilizar la *Instagram Private API*. También puede funcionar sin las credenciales, utilizando sólo la *Instagram Web API*. Permite programar la ejecución periódica de las consultas, definiendo la duración de la espera entre iteraciones.

3.3 Ejemplo de ejecución

Después de introducir los archivos del sistema de seguimiento y explicar brevemente su función, vamos a mostrar un ejemplo de ejecución por pasos de la búsqueda y actualización de una consulta, que como hemos repetido, sería un nombre de usuario o *hashtag* de Instagram, con ayuda de la Figura 3.1:

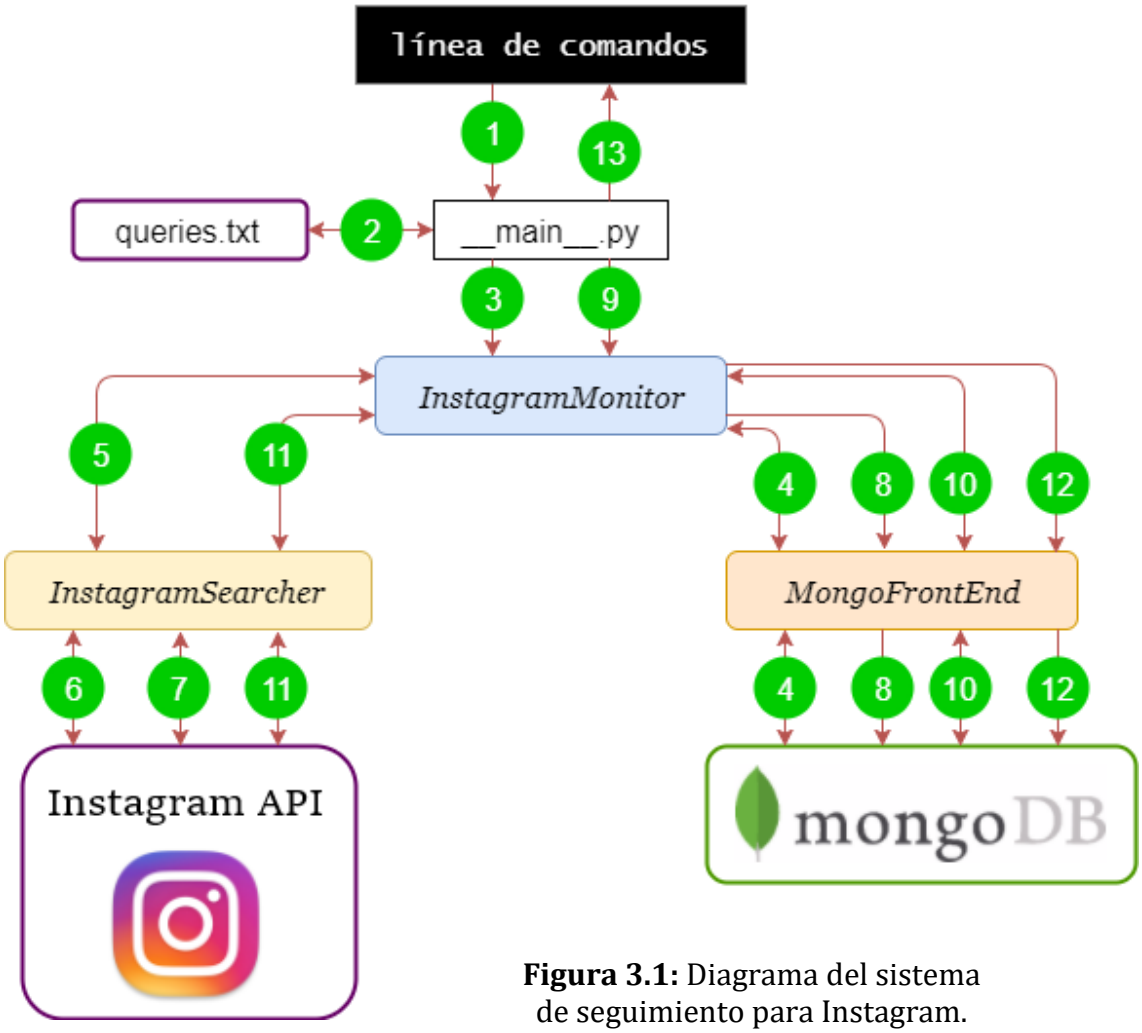

Figura 3.1: Diagrama del sistema de seguimiento para Instagram.

1. Se inicia el sistema desde una interfaz de línea de comandos, indicando opcionalmente en los argumentos las credenciales de un usuario de Instagram u otras preferencias.
2. Se ejecuta el punto de entrada del sistema, o función *main*, identificando los argumentos de la línea de comandos. La función abre el archivo *queries.txt* que contiene las consultas a seguir, y lee los nombres de usuario y *hashtags* en él.
3. La función *main* crea una instancia de la clase *InstagramMonitor*, y a través de ella, pide buscar nuevas publicaciones, para cada consulta.
4. La instancia de *InstagramMonitor*, a través de una instancia de *MongoFrontEnd*, comprueba si hay publicaciones de la consulta previamente almacenadas en el servicio de base de datos MongoDB. Si las hubiera, encuentra la publicación más reciente y guarda su fecha.

5. La instancia de *InstagramMonitor*, a través de una instancia de *InstagramSearcher*, solicita la búsqueda de publicaciones de la consulta posteriores a la fecha de la publicación almacenada más reciente. Si no hubiera ninguna publicación previamente almacenada, solicita arbitrariamente las publicaciones del último día transcurrido.
6. La instancia de *InstagramSearcher* solicita en la *Instagram Web API*, la búsqueda de publicaciones recientes asociadas a la consulta. Debido a que las publicaciones devueltas por este tipo de solicitud no aportan toda la información deseada, se extrae solamente la identificación en Instagram de las publicaciones, de aquellas en el período de tiempo deseado.
7. La instancia de *InstagramSearcher*, con la lista de identificaciones de publicación, solicita a la *Instagram Private API* la información completa de cada publicación, y a la *Instagram Web API* todos los comentarios de cada publicación. En el caso de no haberse indicado en el paso 1 unas credenciales de usuario, tanto para la publicación como los comentarios se usa la *Web API*. Este paso puede utilizar varios hilos de ejecución para acelerar la descarga y se encarga de resolver posibles errores HTTP, como la superación del límite de solicitudes, por período de tiempo, a las APIs de Instagram.
8. Las publicaciones de la consulta han sido descargadas y devueltas en formato JSON a la instancia de *InstagramMonitor*. A través de *MongoFrontEnd*, las publicaciones son almacenadas en el servicio MongoDB comprobando si alguna de las publicaciones a almacenar ya se encuentra guardada.
9. Una vez terminada la adquisición de las publicaciones recientes, para cada consulta, la función *main* solicita a la instancia de *InstagramMonitor* la actualización de aquellas publicaciones que pudieran haber sufrido cambios o recibido comentarios nuevos. El sistema asume que las publicaciones no sufren cambios a partir de un período de tiempo desde su creación, por defecto se asume que este período es de dos días. Si una publicación es descargada después de transcurrir este período, es marcada como archivada.
10. La instancia de *InstagramMonitor* busca, a través de *MongoFrontEnd*, la identificación de las publicaciones almacenadas a actualizar. El sistema busca publicaciones no archivadas y que se consideren que no sufrirán cambios.
11. Se solicita a la instancia de *InstagramSearcher* la descarga de las publicaciones a partir de su identificación, de forma análoga al paso 7.
12. La instancia de *InstagramMonitor* almacena las publicaciones descargadas de forma análoga al paso 8. Pueden existir publicaciones que se haya solicitado descargar pero cuya información no haya sido encontrada por las APIs de Instagram. Estos casos pueden deberse a la eliminación en Instagram de publicaciones que fueron previamente almacenadas por el sistema de seguimiento. El sistema actualiza estas publicaciones particulares, en el servicio MongoDB, como archivadas y «no encontradas».
13. El sistema de seguimiento ofrece mensajes descriptivos del funcionamiento a la interfaz de línea de comandos, y puede esperar un período de tiempo para volver a ejecutar la búsqueda y actualización de consultas, de forma reiterada en un bucle perpetuo.

Conclusiones

4.1 Evaluación de objetivos

Ahora que ya se ha presentado el sistema de seguimiento para Instagram desarrollado y un ejemplo de su funcionamiento, es necesario comprobar si se han cumplido los objetivos presentados en la introducción:

- **¿El sistema desarrollado es capaz de obtener la información de publicaciones públicas de Instagram a partir de un nombre de usuario o *hashtag*?**

  Sí, puede obtener todas las publicaciones de un usuario público, y puede obtener las últimas seis mil publicaciones que contengan un *hashtag*. Actualmente es necesario conocer previamente el nombre de usuario exacto para poder encontrar sus publicaciones.

- **¿El sistema desarrollado es capaz de escoger los servicios de Instagram que permitan obtener la mayor cantidad de información no redundante sobre cada publicación?**

  Parcialmente, las publicaciones y comentarios ofrecidos por la *Instagram Private API* contienen más información que sus equivalentes en la *Instagram Web API*. Si se aportan credenciales de Instagram, las publicaciones se descargan a través de la *Private API*. Pero los comentarios por defecto siempre se descargan usando la *Web API*, a favor de una menor duración de descarga respecto a la *Private API*. Sin embargo, se ha implementado una opción para especificar que los comentarios sean descargados con la *Private API*.

  En cuanto a los datos audiovisuales se ha optado por no almacenarlos directamente a favor de no sobrecargar el sistema de base de datos, y en cambio, guardar una dirección URL de Instagram desde donde descargarlos si se necesitan, ya que se asumía, en la introducción de este trabajo, que el sistema de seguimiento dispondrá de una conexión ininterrumpida a los servicios de Instagram.

- **¿El sistema desarrollado es capaz de** a**lmacenar la información de las publicaciones de Instagram obtenidas, en un servicio de base de datos, evitando la duplicación de información?**

  Sí. Se utiliza un servicio de base de datos MongoDB, que almacena los documentos en formato JSON obtenidos de Instagram. Se han tomado medidas para evitar la repetición de publicaciones y comentarios con el mismo identificador de Instagram.

- **¿El sistema desarrollado es capaz de buscar periódicamente la aparición de publicaciones nuevas, asociadas a un usuario o *hashtag* determinado, que no hayan sido almacenadas previamente?**

  Sí, el sistema utiliza la fecha de la última publicación almacenada como referencia para buscar publicaciones posteriores.

  Se asume que, si el sistema capta publicaciones con la suficiente asiduidad, no existe ninguna publicación en Instagram, asociada a una consulta, que no se haya almacenado entre las publicaciones almacenadas más antigua y más reciente. Se hace la anterior asunción porque el sistema descarga publicaciones, de forma individual, a partir de una cola de identificadores. Durante la descarga, en caso de manejar una excepción por un problema de conexión, el sistema vuelve a encolar la publicación como pendiente de descarga, y en caso de error fatal no se almacena ninguna publicación, porque ello no ocurre hasta que se ha descargado toda la cola.

- **¿El sistema es capaz de actualizar publicaciones de Instagram almacenadas, cuando se considere que han sufrido cambios o incorporación de información?**

  Sí. El sistema marca como archivadas aquellas publicaciones que considera que no van a sufrir cambios. Aquellas publicaciones no marcadas, y que el sistema considera que ya podrían marcarse como archivadas, son descargadas de nuevo, sustituyendo su versión anterior en el servicio de base de datos. En caso de no poder ser descargadas porque la API no encuentra la publicación, como en el caso de la eliminación de una publicación, la última versión almacenada es marcada como archivada y no encontrada.

4.2 Consideraciones futuras

En la introducción de este trabajo se ha expresado que la motivación final de este sistema de seguimiento para Instagram es el análisis de la actividad de Instagram captada. Posteriormente se ha explicado que uno de los usos específicos de los sistemas de extracción web, al cual está enfocado este sistema de seguimiento, es el del *sentiment analysis*. Sin embargo implementar un servicio de análisis no se encontraba entre los objetivos específicos de este trabajo, debido a la falta de tiempo disponible, en comparación a la magnitud del desarrollo un servicio de análisis.

En cambio, este sistema abre la posibilidad a la implementación futura de un análisis de la actividad de Instagram, a través del servicio MongoDB que almacena el contenido captado. También se ha implementado una opción para exportar los textos de las *captions* y comentarios, almacenados en el servicio de MongoDB, a un archivo, por usuario o *hashtag* seguidos. Además se ha implementado la creación de archivos con la información de un tipo de grafos, cuyos nodos representan usuarios de Instagram, y cuyos arcos representan menciones de un usuario a otro en los textos de comentarios y *captions*, de todas las publicaciones almacenadas de una consulta.

Integrar un servicio de *sentiment analysis* al sistema de seguimiento requeriría la aplicación de sucesivas tareas. Sería necesario la extracción de los textos de *captions* y comentarios de las publicaciones a analizar. Posteriormente, se debería realizar un preprocesado de los textos, eliminando caracteres innecesarios y palabras o cadenas irreconocibles, entre otras transformaciones. Opcionalmente, se podría idear una herramienta que aporte información sobre las características más importantes de las imágenes y vídeos de una publicación, sin embargo esta es una tarea muy compleja. En el caso de decidir usar un método de aprendizaje automático supervisado, antes de analizar el contenido objetivo, sería necesario elaborar un corpus de comentarios y/o publicaciones de Instagram, debidamente etiquetado, y que considere imágenes si también se analizan. Con este corpus se entrenaría y validaría la función de clasificación del método de aprendizaje, no obstante, este paso sería complicado, y podría costar muchos meses de investigación sin una garantía de éxito.

4.3 Ejemplos de seguimiento

Para probar el sistema de seguimiento se escogieron unas consultas a seguir, durante el mes de junio, que pudieran suscitar un buen tráfico de opiniones a través de comentarios, y tuvieran tanto una cuenta de usuario oficial como un *hashtag* particular, fueron elegidas las siguientes consultas:

- **Donald Trump**: su nombre de usuario en Instagram es **@realdonaldtrump**, y se escogió el *hashtag* **#donaldtrump**.
- **Playoffs de la NBA**: son las fases finales del campeonato de la NBA de baloncesto. Se escogió el usuario **@nba** y el *hashtag* **#nbaplayoffs**.
- **RuPaul's Drag Race**: es una competición televisiva estadounidense de drag queens donde se van eliminando concursantes hasta quedar uno. Se escogió el usuario **@rupaulsdragrace** y el *hashtag* **#rupaulsdragrace**.

En la tabla 4.1 se ha incluido información sobre el número total, y medias de comentarios y *likes*, de las publicaciones almacenadas para las consultas seguidas durante el mes de junio.

Tabla 4.1: Características de las publicaciones almacenadas en junio**.**

Consultas seguidas	nº publicaciones	media comentarios	media *likes*
@realdonaldtrump	174	2.955,86	99.013,81
@nba	384	1.310,62	240.752,40
@rupaulsdragrace	99	629,22	23.956,32
#donaldtrump	103.757	6,54	210,34
#nbaplayoffs	25.841	9,80	287,00
#rupaulsdragrace	49.902	5,30	226,62

Las publicaciones de *hashtags* tienen una menor media de comentarios y *likes* debido a que en estas consultas se incluyen las publicaciones de usuarios que pueden tener muy pocos seguidores, y por lo tanto sus publicaciones han sido vistas por menos usuarios, en comparación con los usuarios oficiales de Instagram.

A continuación se van a mostrar gráficas sobre el número de publicaciones diarias que contienen un *hashtag* determinado, que se crearon durante el periodo del 7 de junio de 2017 al 4 de julio de 2017. Se va a intentar relacionar las variaciones en la cantidad diaria de publicaciones con sucesos relacionados con el asunto del *hashtag*. Las gráficas se han obtenido con la librería matplotlib para Python.

En la figura 4.2, se muestran el número de publicaciones diarias que contienen el *hashtag* **#donaldtrump**, se observan ciertos rangos de días en los que aumentan significativamente el número de publicaciones creadas. El 14 de junio se produjo un tiroteo en Virginia sobre unos miembros del congreso estadounidense, Donald Trump afirmó personalmente haber visitado uno de los afectados, posteriormente el mismo afectado desmintió que Trump hubiera hablado con él (Shear, Goldman, & Cochrane, 2017). El 2 de julio Trump publicó un vídeo paródico en Twitter de él mismo golpeando simbólicamente al canal de noticias CNN, luego además la CNN intimidó al creador del vídeo (Washington, 2017).

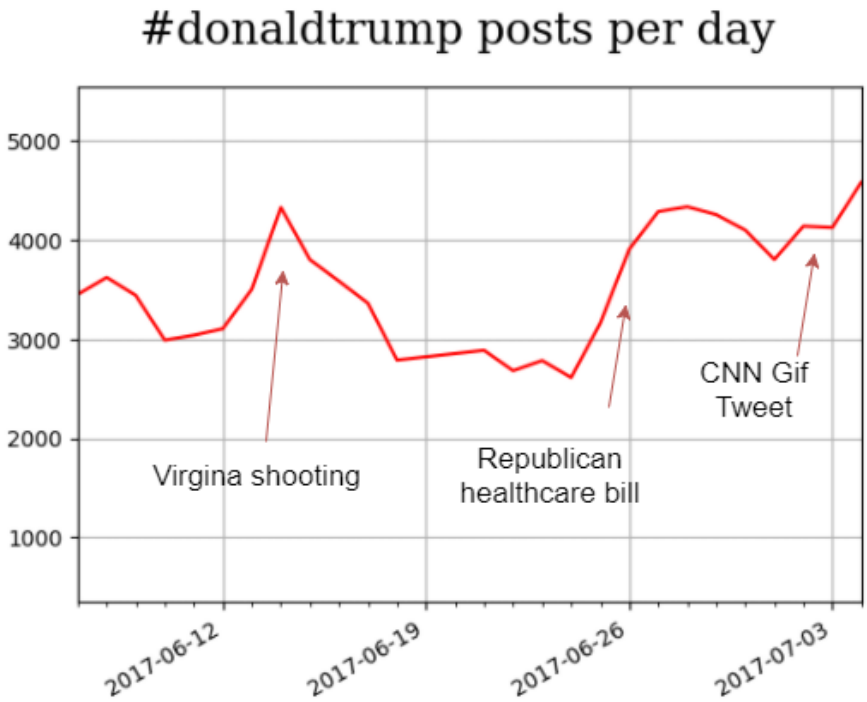

Figura 4.2: Número de publicaciones diarias con el hashtag **#donaldtrump**.

En la figura 4.3, se muestran el número de publicaciones diarias que contienen el *hashtag* **#nbaplayoffs**. El 18 y el 19 de junio se realizaron cambios en el sistema de seguimiento, se provocaron errores que pasaron inadvertidos y que conllevaron a limitar las publicaciones obtenidas para ciertas consultas. El 1 de junio comenzó la final del campeonato de la NBA, que consta de varios partidos entre dos equipos, el 7 y 9 de junio se disputaron dos partidos de esta final, y el 12 de junio se disputó el último partido en el que uno de los dos equipos acumuló las suficientes victorias para ganar el campeonato.

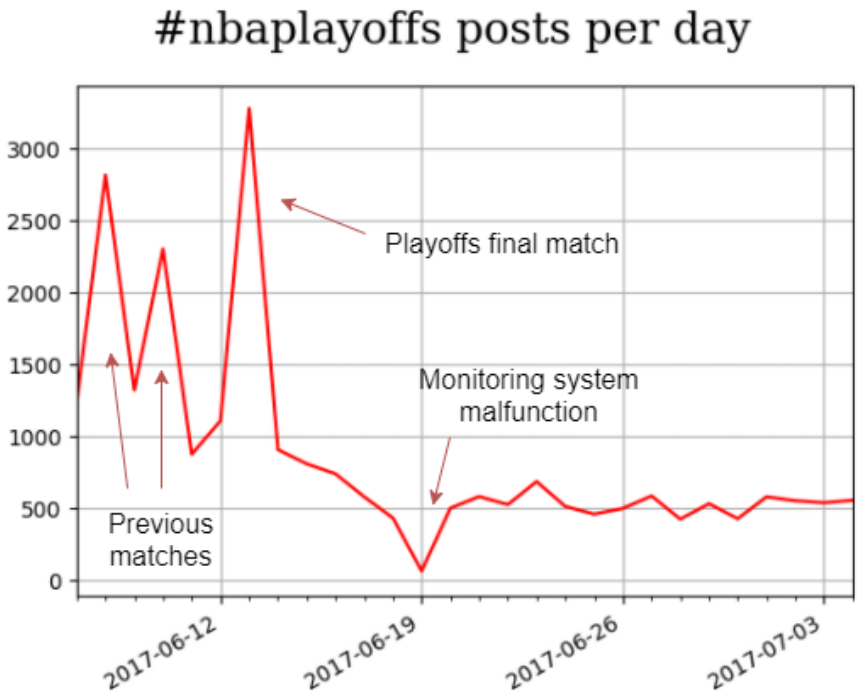

Figura 4.3: Número de publicaciones diarias con el hashtag **#nbaplayoffs**.

En la figura 4.4, se muestran el número de publicaciones diarias que contienen el hashtag **#rupaulsdragrace**. El error en el sistema de seguimiento el 18 y el 19 de junio también afectó a esta consulta. Este programa de televisión se emite los viernes, esta temporada constó de 14 episodios, el episodio final donde se conoció al ganador de la competición se transmitió el 23 de junio.

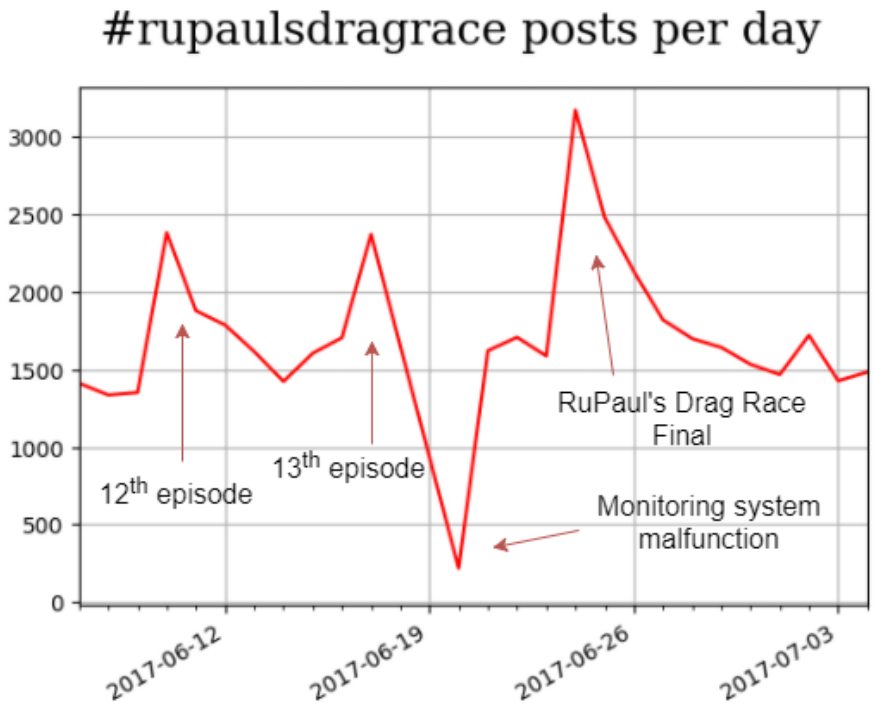

Figura 4.3: Número de publicaciones diarias con el hashtag **#rupaulsdragrace**.

A continuación, en la figura 4.5, se muestra un ejemplo de grafo dirigido, generado a partir de las publicaciones que contienen el *hashtag* **#rupaulsdragrace**. Los nodos representan usuarios de Instagram, y los arcos representan menciones de un usuario a otro en estas publicaciones.

La visualización del grafo se ha obtenido con una herramienta denominada Gephi que ha leído el archivo generado por el sistema de seguimiento. El nombre de cada nodo es el nombre de usuario que representa. El tamaño de un nodo es mayor si es mayor su grado de entrada, es decir, cuántas menciones ha recibido. Se ha aplicado un algoritmo de detección de comunidades, que ha servido para colorear los nodos y los arcos que salen de ellos. Los nodos han sido colocados en la imagen por un algoritmo de proximidad que toma en cuenta muchas propiedades del grafo.

En amarillo aparecen los dos usuarios oficiales del programa Rupaul's Drag Race junto a los finalistas de la última edición, el nodo más grande, «sashavelour» es el usuario del ganador. En la zona inferior, con distintos colores, se encuentran los usuarios de los concursantes eliminados de la última edición. Y en la zona superior, en rosa, se encuentran los usuarios de concursantes de pasadas ediciones.

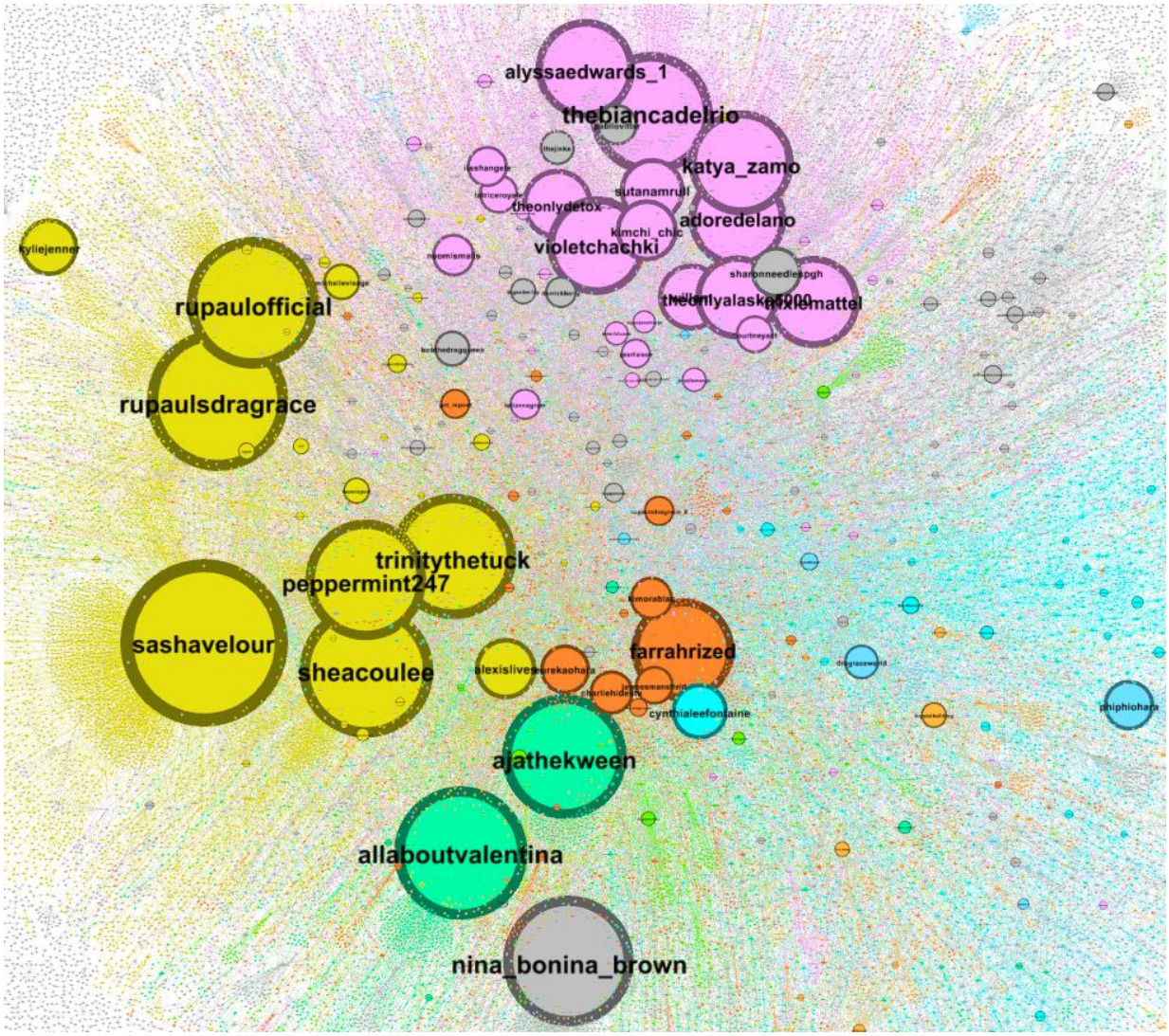

Figura 4.5: Grafo de las menciones a usuarios de Instagram en publicaciones de que contienen el *hashtag* **#rupaulsdragrace**.

Conocer qué usuarios, de los concursantes, son más mencionados en Instagram mientras la competición sigue activa, podría servir al programa televisivo para concentrar u otorgar más tiempo de pantalla a ciertos concursantes. En la edición del siguiente episodio se mostraría preferentemente a un concursante más que a otro, con tal de que el programa aumente su cuota de pantalla.

No se han incluido figuras relacionadas con los usuarios de Instagram de los ejemplos de seguimiento debido a que tienen un menor flujo de publicaciones respecto a sus *hashtags*. Se podría haber mostrado qué publicaciones del usuario han tenido más *likes* o comentarios, pero averiguar la explicación entraría dentro del servicio de análisis que se no ha podido implementar.

4.4 Resultados de aprendizaje

Durante el desarrollo del sistema se cometieron errores, por ejemplo, se asumió que si no surgían errores de ejecución, la información obtenida era toda la existente. Por otra parte, al principio, los comentarios se almacenaban dentro del documento de las publicaciones, cuando pueden existir publicaciones con más de cien mil comentarios.

También surgieron problemas externos. A principios de junio Instagram cesó el servicio de un *endpoint* de la *Web API*. Algunas de las solicitudes disponibles a través de este *endpoint* eran utilizadas por el sistema de seguimiento, esto provocó la búsqueda de *endpoints* alternativos y la implementación de dos métodos independientes por cada tipo de solicitud del sistema a las APIs, por si ocurrieran más cesiones de servicios.

Esta última experiencia me ha enseñado que es prudente considerar la posibilidad de que una plataforma deje de prestar determinados servicios, y por ende puede ser beneficioso preparar métodos de reserva, y adaptar el código para un posible reciclaje.

Gracias a la realización del trabajo y los problemas originados durante el mismo creo que he aprendido o reforzado algunas competencias:

- He reforzado la programación con Python, he aprendido mejor la declaración de sus clases y atributos, he descubierto partes valiosas de su librería estándar, como concurrencia y estructuras de datos entre otras. Además he conocido algunas de las guías de estilo y documentación de Python más usadas.
- He conocido mejor cómo Instagram atiende las solicitudes de contenido de su plataforma, las distintas APIs que sus aplicaciones oficiales utilizan y cómo acceden a ellas.
- He aprendido el funcionamiento básico de un servicio MongoDB, cómo desplegarlo, qué cantidad de recursos puede acaparar, y cómo buscar, filtrar y actualizar documentos MongoDB.
- He aprendido a buscar mejor información sobre la que no existe una documentación oficial, y librerías de código abierto, así como importarlas y usarlas.

Agradecimientos

Quisiera dar las gracias a todos los profesores que me han dado clase durante el grado, especialmente a aquellos que han hecho más de cuanto era necesario para que aprendiéramos más y mejor. También a mis amigos en el grado por su compañía y ayuda desinteresada.

Y quisiera dar las gracias especialmente a mis tutores Ferran Pla y Lluís Hurtado por haber aguantado mis impertinencias y caos generado, además de haberse preocupado porque pudiera sacar adelante este proyecto.

Bibliografía

AbdelFattah, M., Galal, D., Hassan, N., Doaa, E. S., & Tallent, G. (2017). Sentiment Analysis Tool for Determining the Promotional Success of Fashion Images on Instagram. *International Journal of Interactive Mobile Technologies*, 11(2).

Bagley, E. (2 de Diciembre de 2016). *Reverse Engineering the Android Instagram App's Private API*. Obtenido de Technomancy: https://eliasbagley.github.io/reverseengineering/2016/12/02/reverse-engineering-instagram-api.html

Balochi, E. (10 de Marzo de 2016). *How Google, Instagram and LinkedIn Authenticate Users in their Native Mobile Apps*. Obtenido de Essa Karam: https://www.essakaram.com/how-google-instagram-and-linkedin-authenticate-users-in-their-native-mobile-apps/

Constine, J. (26 de Abril de 2017). *Instagram's growth speeds up as it hits 700 million users*. Obtenido de TechCrunch: https://techcrunch.com/2017/04/26/instagram-700-million-users/

Dickinson, T. (15 de Diciembre de 2016). *Extracting Instagram Data – Part 1*. Obtenido de TomKDickinson: http://tomkdickinson.co.uk/2016/12/extracting-instagram-data-part-1/

ECMA. (Octubre de 2013). *Introducing JSON*. Obtenido de json.org: http://json.org/

Evans, W. (24 de Diciembre de 2013). *Reverse Engineering the Android Instagram App's Private API*. Obtenido de Blog will3942: http://blog.will3942.com/reverse-engineering-instagram

Facebook Inc. (2016). *Serving over HTTP*. Obtenido de GraphQL: http://graphql.org/learn/serving-over-http/

Grigalis, T., & Čenys, A. (2013). State-of-the-art web data extraction systems for online business intelligence. *Informacijos mokslai*, 64.

Instagram. (17 de Noviembre de 2015). *Instagram API Platform*. Obtenido de Instagram API Platform: https://www.instagram.com/developer/

Liu, B. (2012). Sentiment Analysis and Opinion Mining. En G. Hirst, *Synthesis Lectures on Human Language Technologies* (págs. 1-167). Morgan & Claypool Publishers.

Martin, E. (22 de Abril de 2017). *Scraping difficult sites using private API's*. Obtenido de Edmund Martin: http://edmundmartin.com/scraping-difficult-sites-using-private-apis/

Shear, M. D., Goldman, A., & Cochrane, E. (14 de Junio de 2017). *Steve Scalise Among 4 Shot at Baseball Field; Suspect Is Dead.* Obtenido de The New York Times: https://www.nytimes.com/2017/06/14/us/steve-scalise-congress-shot-alexandria-virginia.html

Washington, A. (2 de Julio de 2017). *Trump Tweets Video of Himself Attacking CNN.* Obtenido de The Hollywood Reporter: http://www.hollywoodreporter.com/news/trump-tweets-video-him-attacking-cnn-1018417