UNIVERSITAT POLITÈCNICA DE VALÈNCIA

etsinf
Escola Tècnica Superior d'Enginyeria Informàtica

Escola Tècnica Superior d'Enginyeria Informàtica Universitat Politècnica de València

Desarrollo de una herramienta para la planificación y seguimiento de una campaña de comunicación en Twitter

TREBALL FI DE GRAU

Grau en Enginyeria Informàtica

Alumno: Ignacio Díaz Arellano

Tutores: Lluís Felip Hurtado Oliver y Ferran Pla Santamaría

2017-2018

Resum

Les xarxes socials són cada vegada més usades per tot tipus d'organitzacions per arribar a persones de tot el món. Twitter destaca entre elles com una de les més populars, dinàmiques i útils per a la ràpida difusió de missatges. No obstant això, per la complexitat, és difícil realitzar campanyes de comunicació efectives a Twitter.

Amb l'objectiu de remeiar aquesta situació, el present projecte permet, donada una subxarxa de Twitter, fer una anàlisi del seu context i ajudar a planificar una campanya en xarxes que millore la comunicació d'una organització quantitativament o qualitativament. Posteriorment, possibilita realitzar un seguiment d'aquesta campanya.

Paraules clau: Twitter, campanya comunicació, graf

Resumen

Las redes sociales son cada vez más usadas por todo tipo de organizaciones para llegar a personas de todo el mundo. Twitter destaca entre ellas como una de las más populares, dinámicas y útiles para la rápida difusión de mensajes. Sin embargo, por su complejidad, es difícil realizar campañas de comunicación efectivas en Twitter en Twitter.

Con objeto de remediar esta situación, el presente proyecto permite, dada una subred de Twitter, realizar un análisis de su contexto y ayudar a planificar una campaña en redes que mejore la comunicación de una organización cuantitativa o cualitativamente. Posteriormente, también hace posible realizar un seguimiento de dicha campaña.

Palabras clave: Twitter, campaña comunicación, grafo

Abstract

Social media are increasingly used by all types of organizations to reach people around the world. Twitter stands out among them as one of the most popular, dynamic and useful for the rapid diffusion of messages. However, because of the complexity, it is difficult to carry out effective communication campaigns on Twitter.

In order to solve this situation, this project allows, given a Twitter subnet, to make an analysis of the context and help to plan a network campaign that improves the communication of an organization quantitatively or qualitatively. Subsequently, it makes it possible to keep track of this campaign.

Key words: Twitter, communication campaign, graph

Índice

Índice 6

Lista de figuras 10

Lista de tablas 11

Lista de algoritmos 11

1 INTRODUCCIÓN 13

1.2 Objetivo 15

1.3 Estructura de la memoria 15

2 CONCEPTOS PREVIOS 17

2.1 Relacionados con Twitter 17

2.2 Relacionados con teoría de grafos 18

2.3 Árbol 19

2.4 API 19

2.5 Biblioteca (o librería) 19

2.6 NoSQL 19

2.7 Front-end 20

2.8 Back-end 20

2.9 Templating web 20

3 TECNOLOGÍAS UTILIZADAS 21

3.1 Twitter API 21

3.2 Tweepy 21

3.3 MongoDB 21

3.4 PyMongo 22

3.5 D3.js 22

3.5 Django 23

3.7 HTTP (Hypertext Transfer Protocol) 24

3.8 HTTPResponse 24

3.9 XMLHttpRequest 24

4 ENFOQUE 25

4.1 El entorno de una cuenta de Twitter 25

4.2 Visualización estática o interactiva 26

4.3 Usuarios influyentes 28

4.4 Seguimiento de la actividad 30

5 FASES 32
5.1 Extracción y almacenamiento de datos 33
5.1.1. Funcionalidad 33
5.1.2. Seguidores 34
5.1.3. Tuits 35
5.1.4. Peticiones a la API de Twitter 35
5.1.5. Almacenamiento 36
5.2 Tratamiento de datos 37
5.2.1. Selección de los vértices con más seguidores 37
5.2.2. Centralidad 38
5.2.3. Retuits por usuario 39
5.3 Visualización de información 39
5.3.1. Estructura 39
5.3.2. Tamaño de los vértices 41
5.3.3. Centralidad 41
5.3.4. Interactividad 41
5.3.5. Actividad
43
6 CASOS DE ESTUDIO 46
6.1 D.A. ETSINF 46
6.2 Compromís La Pobla de Vallbona 52
6.3 Delegación de Valencia de la Asociación de Estudiantes de Videojuegos 56
57
7 CONCLUSIONES 60
7.1 Trabajo realizado 62
7.2 Posibles mejoras

8 BIBLIOGRAFÍA 65

Lista de figuras

- Figura 1. Ejemplo de estadísticas de Twitter Analytics
- Figura 2. Ejemplo de estadísticas de Twitstats. Densidad de tuits a lo largo de la semana
- Figura 3. Ejemplo de grafo dirigido
- Figura 4. Ejemplo de árbol
- Figura 5. Diferentes visualizaciones hechas con D3.js
- Figura 6. Templating en Django
- Figura 7. Seguidores más próximos. Adyacentes
- Figura 8. Seguidores inmediatos (primer nivel) y sus respectivos seguidores (segundo nivel)
- Figura 9. Seguidores inmediatos (primer nivel) y sus respectivos seguidores (segundo nivel) cohesionados
- Figura 10. Usuarios de Twitter hablando sobre el Centre de Cultura Contemporània de Barcelona. Espinós, Alex. (2014)
- Figura 11. Un nodo y sus arcos resaltados en un grafo
- Figura 12. "*Measuring user influence on Twitter: A survey*". Fabián Riquelme, Pablo González-Cantergiani. (2015)
- Figura 13. El servicio oficial de analíticas de Twitter. Valora como "seguidor principal" al usuario con mayor número de seguidores
- Figura 14. Usuarios según los retuits hechos a la cuenta de estudio
- Figura 15. Flujo general de la aplicación
- Figura 16. Interfaz de la aplicación desde la vista Extract
- Figura 17. Búsqueda en profundidad de usuarios
- Figura 18. Medidas de centralidad
- Figura 19. Interfaz de la vista Struct mostrando un grafo
- Figura 20. Tonos de centralidad. De azul a rojo según menos y más centralidad respectivamente
- Figura 21. El usuario *@AppTraceCentre* focalizada con sus seguidores destacados
- Figura 22. El usuario *@eamnupv* siendo arrastrado
- Figura 23. Información adicional del usuario *@CompromisPobla*
- Figura 24. *Inputs* de la interfaz *Activity*
- Figura 25. Actividad. Azul más intenso conforme más retuits ha hecho un usuario
- Figura 26. Grafo del usuario *@CompromisPobla* con los usuarios que más retuitean en un color azul más intenso
- Figura 27. Los 70 usuarios con más seguidores y los usuarios que están relacionados con ellos
- Figura 28. Entorno más cercano a *@dai_etsinf*. Sus vértices adyacentes
- Figura 29. Seguidores de Eduardo Vendrell (*@evendrell*)
- Figura 30. Usuarios del entorno de *@dai_etsinf* que han retuiteado alguno de sus tuits. Datos históricos
- Figura 31. Usuarios del entorno de @dai_etsinf que han retuiteado alguno de sus tuits. Datos desde el 17/11/16 al 17/11/17
- Figura 32. Los 50 usuarios con más seguidores y los usuarios que están relacionados con ellos
- Figura 33. Entorno más cercano a *@CompromisPobla*. Sus vértices adyacentes

- Figura 34. Usuarios del entorno de *@CompromisPobla* que han retuiteado alguno de sus tuits. Datos históricos
- Figura 35. Los 90 usuarios con más seguidores y los usuarios que están relacionados con ellos
- Figura 36. Grupo de usuarios relacionados con el desarrollo de videojuegos
- Figura 37. Usuario *@jonllaguno* focalizado
- Figura 38. Usuarios del entorno de *@AEVvalencia* que han retuiteado alguno de sus tuits. Datos históricos
- Figura 39. Grafo del entorno de un usuario con vértices agrupados

Lista de tablas

- Tabla 1. Datos recogidos de la cuenta @dai_etsinf
- Tabla 2. Datos recogidos de la cuenta @CompromisPobla
- Tabla 3. Datos recogidos de la cuenta @AEVvalencia

Lista de algoritmos

- Algoritmo 1. Exploración de usuarios de forma recursiva
- Algoritmo 2. Selección de los usuarios del subgrafo con los *n* usuarios con más seguidores

1 INTRODUCCIÓN

Twitter es una de las redes sociales más importantes. A pesar de estar de capa caída, al menos en comparación con sus competidores, en julio de 2017 contaba con 328 millones de usuarios registrados[1]. Además, es una de las más usadas por partidos políticos y organizaciones de todo tipo por su potencial a la hora de difundir rápida y masivamente mensajes (Parmelee & Bichard, 2012). No obstante, han sido fundamentales en una buena cantidad de movimientos sociales (Gil & Guilleumas 2017) (Torres 2009)

De su estudio han surgido dos principales usos. Por una parte, los relacionados con investigaciones científicas, y por otro el de aplicaciones que buscan mejorar los resultados en Twitter de una organización. En este último caso, todas las herramientas de uso abierto no van más allá de las estadísticas de la actividad del usuario de estudio, similar a las que puede haber sobre analíticas web, ignorando el entorno de la cuenta y por tanto, no enmarcándola en su contexto, omitiendo así mucha información que podría ser útil. Ejemplos de estas aplicaciones pueden ser Twitter Analytics (la oficial de la compañía, que puede verse en la Figura 1) o Twitstats (Figura 2).

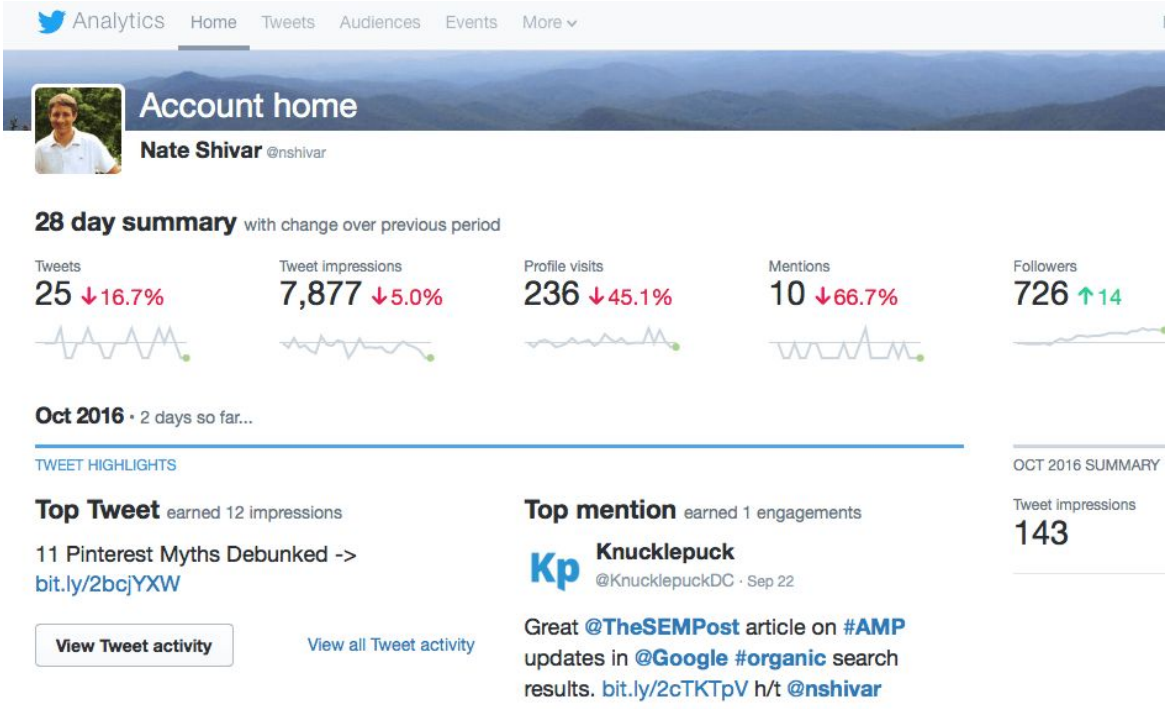

Figura 1. Ejemplo de estadísticas de Twitter Analytics

[1] Toledano, B. (2017, Julio 27). El número de usuarios que ha sumado Twitter en el último trimestre asciende a cero. *El Mundo*. Obtenido en: http://www.elmundo.es

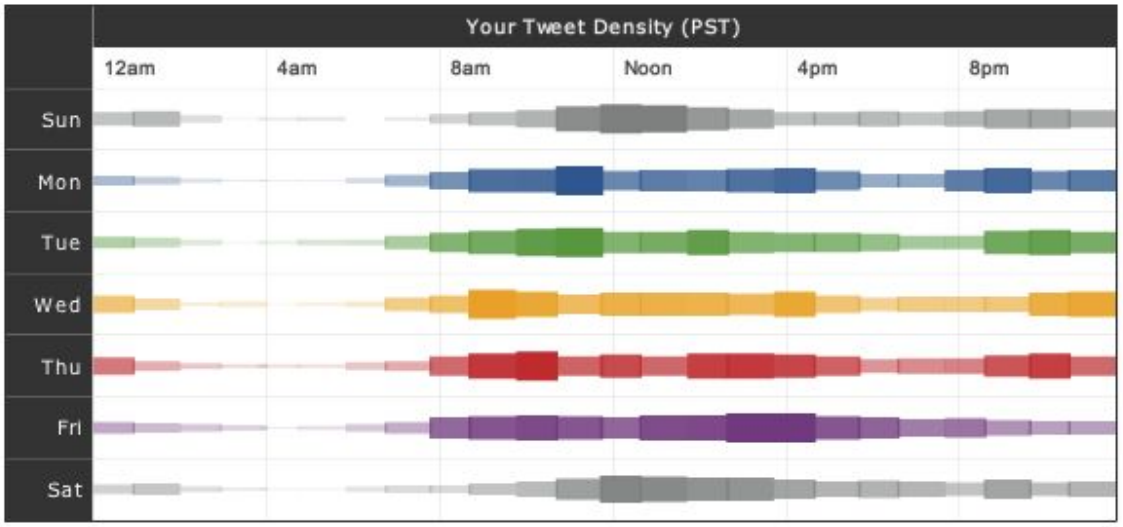

Figura 2. Ejemplo de estadísticas de Twitstats. Densidad de tuits a lo largo de la semana

1.1 Motivación

El proyecto surge primero como una inquietud personal. Una de las vertientes que más me atrae del tratamiento masivo de datos es la relacionada con el uso en redes sociales. El uso que le dió Cambridge Analytica en favor de Ted Cruz como precandidato republicano para las elecciones presidenciales de Estados Unidos de 2016, es una demostración del potencial que este campo tiene para ser usado en la política u otras tantas cuestiones de índole social.

Segundo, como una necesidad práctica, por lo que el proyecto no busca sólo ser aprobado como TFG sino que será usado más adelante. Como administrador de la cuenta de Twitter de algunas organizaciones, he experimentado (y experimento) serias dificultades gestionándolas. ¿Cómo puedo incrementar mis seguidores, ya no sólo en número, sino en calidad (influyentes)? ¿Cuáles son los usuarios más importantes de mi entorno y cómo me relaciono con ellos? ¿Y cómo puedo ver si están difundiendo los mensajes? Busca, de peor o mejor forma, dar respuesta a estas preguntas.

Tercero, es una oportunidad para aprender nuevas tecnologías. Es la primera vez que uso muchas de las tecnologías que se usan en este proyecto. Lo único que había hecho con anterioridad que se pueda relacionar con este trabajo es extraer datos de Twitter y almacenarlos en una base de datos cuyo funcionamiento es bastante distinto al que se usa en este proyecto. Por lo demás, es la primera vez que trabajo haciendo una aplicación web y con tecnologías como MongoDB, Django o D3.js, muy dispares entre sí tanto en sus usos como en sus funciones dentro de la aplicación.

Cuarto y último, debido a que este año he comenzado el Máster Universitario en Gestión de la Información, he querido aprovechar la oportunidad para tratar un tema y tecnologías que serán recurrentes a lo largo del posgrado.

1.2 Objetivo

El objetivo general es desarrollar una herramienta que ayude a una organización a entender el contexto en el que se encuentra, su entorno, para así crecer cualitativa y cuantitativamente, a la vez que puede verificar si esto se logra mediante un seguimiento de la actividad. Para lograrlo, el proyecto debe abarcar todas las fases que lo hagan posible: desde extraer los datos existentes en Twitter hasta crear una visualización de interés.

Desgranado este objetivo general en otros más específicos, serían los siguientes:

1. Crear la plataforma web donde integrar las funcionalidades y permitir su futura expansión

2. Extraer la información necesaria del entorno de la cuenta objetivo de Twitter

3. Almacenar estos datos de forma adecuada para su tratamiento posterior

4. Tratar los datos para extraer información a partir de ellos. Por una parte, habrá que descartar los datos recogidos menos relevantes y construir la visualización a partir del resto. Por otra, analizar estos datos para aumentar su significado a la hora de que el usuario los trate de entender

5. Visualizar la información de tal forma que permita analizarla de forma sencilla. Debe ser, primero, inteligible, y segundo, aportar la suficiente información como para que la herramienta sea útil

Es difícil detallar más los objetivos por dos cuestiones ya comentadas: 1) no tengo experiencia en las tecnologías que voy a usar y 2) hasta donde sé, no hay aplicaciones similares de acceso abierto, por lo que estudiar cómo lo han hecho previamente otros desarrolladores se vuelve imposible.

Esta memoria no entra concretar la estrategia comunicativa que una organización ha de realizar para sacarle el máximo partido a la aplicación. Esto sale del ámbito del trabajo. Esta herramienta aporta información al usuario y es éste el que, atendiendo a su caso específico, formula una estrategia para darle uso. A este respecto, se aportan algunas ideas en el capítulo 6. Casos de estudio, pero únicamente a modo de esbozo, sin pretender agotar el potencial de la herramienta.

1.3 Estructura de la memoria

1. **Introducción**. Se introduce brevemente la red social y se plantea por qué es necesario el presente proyecto. Posteriormente se especifican los objetivos del

mismo.

2. **Conceptos previos**. Se explicarán los conceptos que puedan ser ajenos al lector, de tal forma que posteriormente pueda hacer una lectura adecuada.

3. **Tecnologías utilizadas**. Se expondrán las tecnologías que se han usado y se explicará brevemente cada una.

4. **Enfoque**. Se explicará cómo se ha abordado el problema de forma general, en todos aquellos aspectos que no se circunscriban únicamente a una de las fases de todo el proceso, sino que afecten a su totalidad. Se justifican aquí muchas de las decisiones adoptadas en base algunas investigaciones recientes en este campo.

5. **Fases.** Se explica detalladamente en qué consiste cada fase del proyecto, abordando los aspectos técnicos relevantes de cada uno. Estas fases son: extracción de datos, selección de datos y visualización de datos

6. **Casos de estudio.** Para comprobar la utilidad de la aplicación en una situación real, se prueba su uso con distintas cuentas y se documenta qué información se obtiene de cada una de ellas y cómo podría una organización sacarle provecho

7. **Conclusiones.** Finalmente, se extrae una valoración y conclusiones del trabajo. Además, como proyecto en continuo desarrollo, se aportan posibles mejoras de cara al futuro.

2 CONCEPTOS PREVIOS

Con el fin de que el lector conozca una serie de conceptos específicos de la temática del trabajo que se utilizarán a lo largo de la memoria, se definen a continuación.

2.1 Relacionados con Twitter

- **Tuit** (del inglés: *tweet*): Es un mensaje de, como máximo, 280 caracteres, que emiten los usuarios de Twitter. Este es el formato esencial mediante el cual se genera la información en esta red social. Son públicos a no ser que el usuario decida hacer su cuenta privada, algo muy infrecuente.

- **Seguidor** (del inglés: *follower*): Es la relación principal de Twitter entre dos usuarios y el método mediante el cual un usuario se suscribe a los tuits de otro. De otra forma, es relativamente difícil que un usuario vea tus mensajes. Es por este motivo por lo que se vuelve tan importante tener un gran número de seguidores: para que el contenido sea visto por otros.

- **Retuit** (del inglés: retweet): Es el mecanismo mediante el cual los usuarios de Twitter pueden difundir un tuit ajeno. Cuando un usuario retuitea un tuit, los usuarios que le siguen podrán verlo en su forma y autor original, aunque no sigan directamente a dicho autor.

- ***Followback***: Es una práctica mal vista por la comunidad de usuarios pero extendida en Twitter en la cual los usuarios siguen automáticamente a todos aquellos que les siguen a ellos. Esta es una forma de aumentar rápidamente de seguidores. Este ímpetu por buscar más seguidores es a consecuencia de querer aparentar ser más influyente de lo que realmente se es.

- **Mención**: Es una forma de interactuar mediante los tuits. Se realiza escribiendo el nombre del usuario que se quiere mencionar, precedido por un arroba. Permite referenciar a otro usuario, de forma que quien lo lee puede conocer fácilmente a quien se menciona.

- **Comentario**: Es un tuit enlazado como respuesta a otro tuit, propio o ajeno, publicado con anterioridad.

- **Cuenta protegida**: Una de las opciones de Twitter en cuanto a la privacidad es la posibilidad de "proteger tu cuenta". El usuario que decide protegerla mantiene privados, para todo aquel que no le sigue, sus tuits, la información de sus seguidores y a quién sigue.

- ***Token* de acceso**: Cada usuario de Twitter con cuenta verificada mediante un número telefónico puede acceder a la API de Twitter. Este acceso lo proveen mediante unos datos públicos y privados que identifican cada aplicación registrada

en la plataforma. Estos datos son *tokens* de acceso.

2.2 Relacionados con teoría de grafos

- **Grafo dirigido:**

Los grafos dirigidos (o digrafos) son formas habituales de representar redes sociales y también es como se representan las redes de Twitter en este proyecto. Un grafo dirigido se define de la siguiente forma:

$$G = (V, E)$$
$$V \neq \{\}$$
$$E \subseteq \{(a,b) \in VxV : a \neq b\}$$

En este caso, los vértices (V) serán los usuarios y los arcos (E) representan la relación en la que un usuario sigue a otro. En la Figura 3 puede verse un ejemplo sencillo de grafo dirigido.

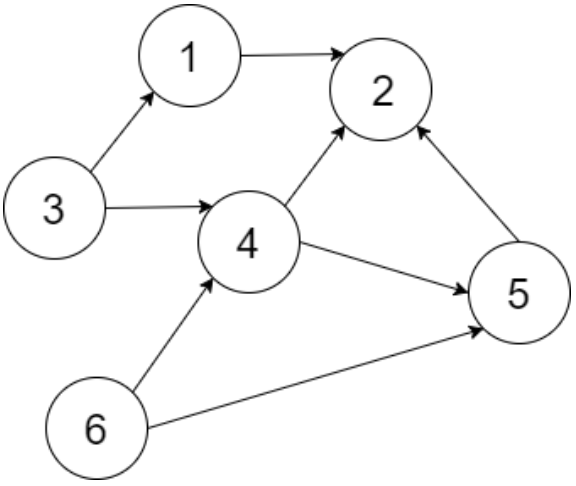

Figura 3. Ejemplo de grafo dirigido

- **Grafo dirigido:** Un subgrafo de un grafo G es un grafo cuyo conjunto de vértices es un subconjunto del de G y cuyo conjunto de arcos es un subconjunto del de G.

- **Grado:** El grado de un vértice v de un conjunto V, denotado como $grado(v)$, $g(v)$ o $d(v)$, es el número de aristas que inciden en el vértice. Como en este trabajo se tratarán únicamente grafos dirigidos, hay que tomar en consideración que el grado podrá ser tanto el grado de entrada, $grado_E(v)$, los arcos cuyo destino es el vértice v, o el grado de salida, $grado_S(v)$, los arcos cuyo origen es el vértice v.

- **Camino**: En un grafo dirigido es una secuencia de vértices de un grafo tal que exista un arco cuyo origen sea el vértice anterior y cuyo destino el vértice posterior. Es

decir, todos los arcos están orientados en una misma dirección.

- **Ciclo**: En un grafo dirigido, es un camino cerrado en la que no se repite ningún vértice a excepción del primero (que constituye el principio y el final del ciclo).

2.3 Árbol

Un árbol es una estructura que puede definirse como un caso especial de grafo dirigido en el que cada nodo sólo tiene un padre y no hay ningún ciclo (Figura 4).

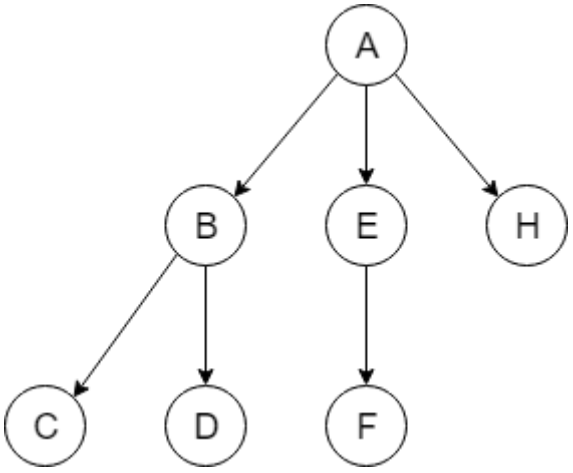

Figura 4. Ejemplo de árbol

2.4 API

Una API, siglas en inglés de interfaz de programación de aplicaciones (*Application Programming Interface*) es un conjunto de comandos, funciones, protocolos y objetos que los programadores usan para crear software o interactuar con sistemas externos.

2.5 Biblioteca (o librería)

En el contexto de la programación, una biblioteca es un conjunto de funciones de un determinado ámbito, no funcional por sí mismo, cuya finalidad es ser utilizado por un programa.

2.6 NoSQL

NoSQL refiere a los sistemas de bases de datos que no son sólo RDBMS, siglas en inglés de sistema de gestión de bases de datos relacionales (*Relational Database Management System*). Son populares en el uso de redes sociales por su capacidad para escalar horizontalmente.

2.7 *Front-end*

Front-end es una abstracción común en el desarrollo de software referida a la capa de presentación mediante la cual interactúa el usuario y recibe la información del sistema. Por lo general, al introducir el usuario un *input*, el *front-end* lo comunica al back-end. Éste genera una respuesta que se encarga de transmitir el *front-end*.

2.8 *Back-end*

Complementario al *front-end*, el *back-end* es la capa con acceso a los datos de la aplicación. En el desarrollo web, como es el caso de este trabajo, mientras que el *front-end* es ejecutado por el cliente en el navegador, el *back-end* se computa en el servidor.

2.9 *Templating* web

Es una forma de representar datos en la web de diferentes maneras. Normalmente es una estructura fácil de entender donde se emplazarán los datos para constituir la forma final de la página[2].

[2] Python Software Foundation. Templating in Python. Obtenido en: https://wiki.python.org/moin/Templating

3 TECNOLOGÍAS UTILIZADAS

En este capítulo se pasa a enumerar y describir cada una de las tecnologías usadas en el proyecto a fin de asegurar que el lector tiene un mínimo conocimiento de ellas antes de comenzar su exposición.

3.1 Twitter API

Escrita en Ruby, permite que los programadores desarrollen software que interactúe con la red social. Ofrece muchas más funcionalidades que las API de la mayoría de redes sociales, motivo que contribuye a que Twitter sea una de las redes sociales más utilizadas para investigación.

Tratando de evitar un uso abusivo que podría acarrearles muchos costes, la compañía limita el número de peticiones por rango de tiempo que puede hacerse a su API. De esta forma, se ve dificultado extraer información de forma masiva.

3.2 Tweepy

Tweepy es una biblioteca de código abierto que permite, mediante código escrito en Python, hacer uso de la API de Twitter de forma sencilla. Su uso está muy extendido.

3.3 MongoDB

MongoDB es un sistema de base de datos NoSQL de código abierto. Es orientado a documentos y probablemente el más popular de este tipo, lo que facilita la continuidad del proyecto en el futuro, buena documentación y buena comunidad de desarrolladores. Concretamente, estos documentos son de una especificación de JSON llamada BSON. Fue distribuida públicamente por primera vez en 2009.

Algunas de las ventajas que ofrece con respecto a otros sistemas de bases de datos RDBMS son:

- Flexibilidad en la información almacenada. De esta forma, los campos de cada documento pueden ser distintos y la estructura de los datos variar con el tiempo
- Muy buena escalabilidad horizontal
- El mapeo de los documentos a objetos es sencillo y cómodo para el programador
- Búsquedas *ad hoc*. Proporcionan mucha flexibilidad a la hora de realizar búsquedas

3.4 PyMongo

PyMongo es una librería de Python que contiene herramientas para trabajar con MongoDB y es la forma recomendada de trabajar con MongoDB desde Python[3]. Es de código abierto y su uso está muy extendido.

3.5 D3.js

Las siglas del inglés Data-Driven Documents dan nombre a D3.js. Es una biblioteca escrita en Javascript para manipular documentos basados en datos. Es de código abierto y trabaja con elementos HTML, SVG y CSS para crear visualizaciones de datos interactivos. Fue distribuida públicamente por primera vez en 2011.

Pese a ser una herramienta limitada a programadores, D3.js es muy popular para crear todo tipo de visualizaciones interactivas por su potencial y versatilidad (varios ejemplos pueden verse en la Figura 5). Es por esto, por su capacidad de ir al detalle, por lo que será la tecnología encargada de visualizar la información de este proyecto. Además, es una biblioteca muy documentada, lo que facilita aprender a usarla.

Figura 5. Diferentes visualizaciones hechas con D3.js

[3] MongoDB, Inc. Pymongo 3.5.1 Documentation. Obtenido en: https://api.mongodb.com/python/current/

3.5 Django

Django es un framework para crear aplicaciones web. Es de código abierto y está escrito en Python. Sigue el patrón de diseño MVC (Modelo-Vista-Controlador).

Proporciona muchas facilidades para crear sitios web complejos. Algunas son:

- Facilidad para conectar módulos de Django desarrollados por otros programadores
- Mapeador objeto-relacional, que facilita mucho la utilización de la información de las bases de datos
- Sistema para usuarios administradores incluido por defecto
- Direccionamiento de URLs basado en expresiones regulares
- Sistema de plantillas que incorpora herencia
- Funcionalidades diversas como cacheo de objetos o protección CSRF

En este proyecto, Django es el *framework* sobre el que se asientan tanto la fase de tratamiento de datos como la de visualización de información. La elección de este framework y no otro es, por una parte, porque está escrito en Python, un lenguaje que hace muy cómodo el tratamiento de datos y es muy popular en este campo; por otra, debido a que algunas de las facilidades listadas anteriormente serán aprovechadas (por ejemplo, el uso de plantillas o la protección CSRF).

Hay que destacar que el uso de Django en la visualización de datos tiene que ver, sobre todo, con el *templating* web. Este *framework* cuenta con un lenguaje específico y sencillo[4]. Hay que tener en cuenta que ha de servir para dar forma al HTML, no para crear lógica de sistema. Un ejemplo bastante instructivo y propio de este trabajo puede verse en la Figura 6.

```
<form method="POST" action="">
    {% csrf_token %}
    <p>Select <select id="graphs" name="graphname">
        {% for g in graphs %}
            <option value="{{ g.name }}">{{ g.name }}</option>
        {% endfor %}
    </select></p>
    <p>Search for <input type="number" placeholder="0" min="1"
         name="n_users_query"/> users with more followers.</p>
    <p>Eliminate users that follow >= <input type="number"placeholder="0"
         min="1" name="user_friends_threshold"/> users</p>
    <input type="submit" value="Enviar"/>
</form>
```

Figura 6. Templating en Django

[4] MongoDB, Inc. The Django template language. Obtenido en: https://docs.djangoproject.com/en/1.11/ref/templates/language/

3.7 *HTTP (Hypertext Transfer Protocol)*

Es un protocolo de la capa de aplicación que permite transferir información en la *World Wide Web*. Sigue un esquema petición-respuesta entre el cliente (normalmente un navegador) y el servidor.

3.8 *HTTPResponse*

Es una clase de Python utilizada para enviar información de forma muy fácil mediante el protocolo HTTP.

3.9 *XMLHttpRequest*

Es una API en forma de objeto cuyo fin es transferir datos entre el navegador y el servidor. En la mayoría de casos, esto lo hace mediante el protocolo HTTP. Normalmente es usado para comunicaciones asíncronas.

4 ENFOQUE

En este capítulo se tratará la aproximación general al proyecto y a las cuestiones que pese a que parezca que afectan sólo a una parte, tiene repercusiones en el resto y determinan cómo funcionarán todas en conjunto.

4.1 El entorno de una cuenta de Twitter

En el Capítulo 1, Introducción, se mencionaba por primera vez la voluntad de conocer "el entorno" de un usuario de Twitter porque no hay aplicaciones gratuitas que faciliten conocerlo. ¿Pero qué es el entorno de un usuario de Twitter? El entorno de un usuario es el contexto en el que se enmarca, que está definido por otros usuarios próximos y sus relaciones, tanto con él como con otros usuarios.

De esta forma, el entorno más inmediato de un usuario sería el compuesto por sus seguidores (vértices adyacentes en la denominación propia de la teoría de grafos). Así se obtiene, a mayor escala, un grafo similar al de la Figura 7.

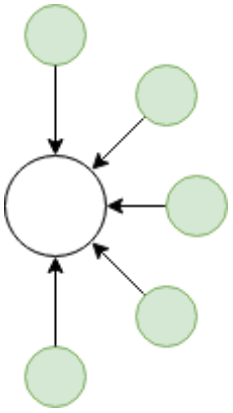

Figura 7. Seguidores más próximos. Adyacentes.

Pero obtener solamente estos usuarios no tiene ningún sentido. Primero, porque no obtenemos más información de la que el gestor de la cuenta de Twitter puede obtener inmediatamente consultando el listado de usuarios que le siguen. Segundo, porque la estructura resultante en forma de diente de león no aporta ningún significado adicional.

Por tanto, obtener el entorno significa ir más allá de los usuarios más inmediatos. El siguiente paso es, además de obtener a los seguidores del usuario, los adyacentes (seguidores de primer nivel), a los seguidores de los seguidores (seguidores de segundo nivel). Orientativamente queda ilustrado en la Figura 8.

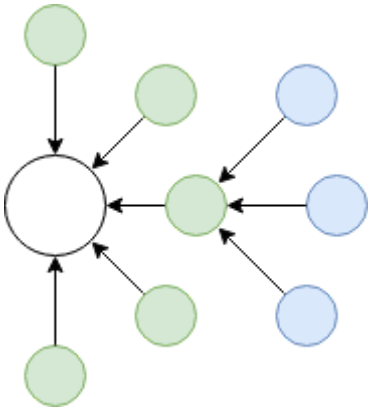

Figura 8. Seguidores inmediatos (primer nivel) y sus respectivos seguidores (segundo nivel)

Si bien la Figura 8 sirve para expresar la idea anteriormente comentada, ilustra de forma irreal cómo se expresan estas estructuras. Por diversos motivos: porque hay usuarios que se se conocen fuera de internet y forman pequeños grupos; por la existencia de filtros de burbuja (del inglés: *filter bubble*) (Nguyen *et al.* 2014)... Por todo esto, su forma es mucho más cohesionada y con un aspecto más orgánico, donde existen más arcos entre distintos vértices. Esta noción queda mejor ilustrada en la Figura 9.

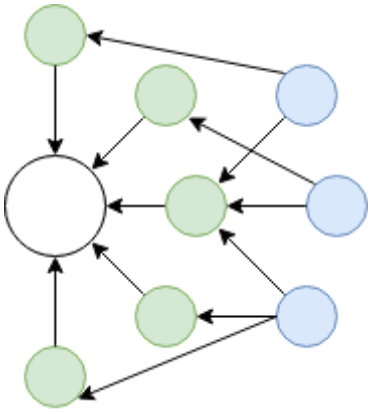

Figura 9. Seguidores inmediatos (primer nivel) y sus respectivos seguidores (segundo nivel) cohesionados

La delimitación de este entorno, su profundidad, es variable. En este proyecto se limita a dos niveles, es decir, hasta que se han obtenido de la cuenta objetivo los seguidores y después los seguidores de los seguidores.

4.2 Visualización estática o interactiva

La visualización del grafo es la principal información de la que se provee al usuario, así que cómo se hace es una cuestión fundamental. En la fase de extracción se almacenan en torno a cien mil usuarios. Para grafos tan grandes, aparece una disyuntiva: generar una visualización estática, una imagen, donde estén presentes todos los vértices y sus arcos o por contra una más pequeña, donde se muestren sólo los elementos más importantes, y con la que se pueda incluso interactuar.

Es importante tratar esta cuestión en el presente capítulo porque no sólo afecta a la fase de visualización, sino al tratamiento de datos. Si se va a visualizar un grafo con un número muy elevado de nodos o uno con menos, afecta directamente tanto a la estrategia de selección de datos como a la información que se va a extraer para visualizarlo. No todas las forma de medir la centralidad de un vértice son igual de válidas para todos los tamaños de grafo (Wasserman & Faust 1994).

Como se comentaba antes, en el caso de una visualización estática se podría ver todo el grafo en una sola imagen que se generaría en el *back-end*. Hacerlo así presenta dos problemas. Primero, visualizar todo el grafo sirve para ver cuestiones de carácter muy general y pero no es posible reparar en cuestiones concretas (muchos elementos que no tienen interés ocupan espacio, restando protagonismo a los que sí). Simplemente es imposible ver los arcos de un vértice específico. Segundo, una visualización estática no permite interactuar con ella e inhabilita al usuario para explorar de forma cómoda y natural los datos que son de su interés. La Figura 10 ejemplifica una de este tipo.

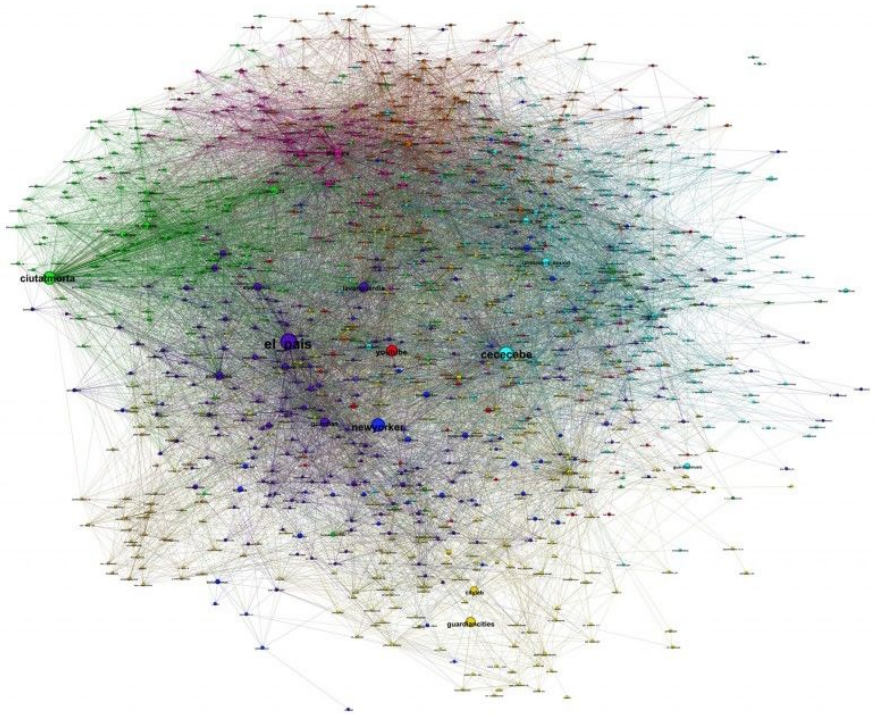

Figura 10. Usuarios de Twitter hablando sobre el Centre de Cultura Contemporània de Barcelona. Espinós, Alex. (2014). “Museums on Twitter: Three case studies of the relationship between a museum and its environment” Recuperado de mw2015.museumsandtheweb.com

Por contra, si se opta por una visualización interactiva, no se puede mostrar un gran número de vértices y arcos por cuestiones de rendimiento. Que el ordenador deba tener guardado en memoria RAM cada nodo y cada arista lo hace inviable. La mayor parte de grafos de este trabajo contienen entre 200 y 300 vértices. Por este motivo, es importante elegir adecuadamente qué parte del grafo mostrar. La Figura 11 muestra una visualización generada por este proyecto en la que un vértice y sus arcos son resaltados, fruto de la posibilidad de interacción.

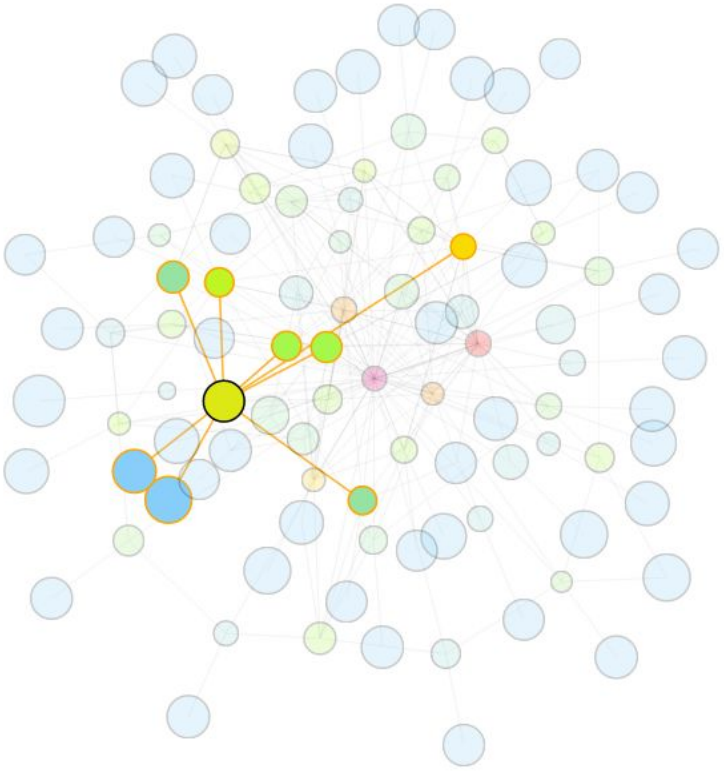

Figura 11. Un nodo y sus arcos resaltados en un grafo

Este proyecto se ha desarrollado utilizando la alternativa de la visualización interactiva por los motivos ya planteados.

4.3 Usuarios influyentes

Como se acaba de ver, sólo se visualizará una pequeña parte del grafo a fin de hacerlo interactivo. Por tanto, cabe tener muy presente qué objetivos tiene el proyecto para saber qué hay que mostrar.

El objetivo es mostrar el entorno de un usuario concreto de Twitter para ayudar a que crezca cuantitativa y cualitativamente. Este crecimiento ha de concretarse en una mayor capacidad para difundir mensajes que, al fin y al cabo, es el objetivo principal de la actividad en Twitter. Esta difusión se materializa mediante los retuits. Entonces, la cuestión radica en llegar a los usuarios con esa alta capacidad de difusión de mensajes, que son los usuarios influyentes.

Determinar quiénes son los usuarios influyentes de un grafo no es sencillo. Es un campo sobre el que se ha investigado mucho. Es necesario determinar qué métricas son las que se considerarán (Wasserman & Faust 1994): número de seguidores, retuits, cantidad de me gusta, comentarios, menciones... (Figura 12) Incluso crear índices mediante su combinación. También cabe la posibilidad de ir más allá y valorar el contenido de los tuits y si son más o menos adecuados para determinado público.

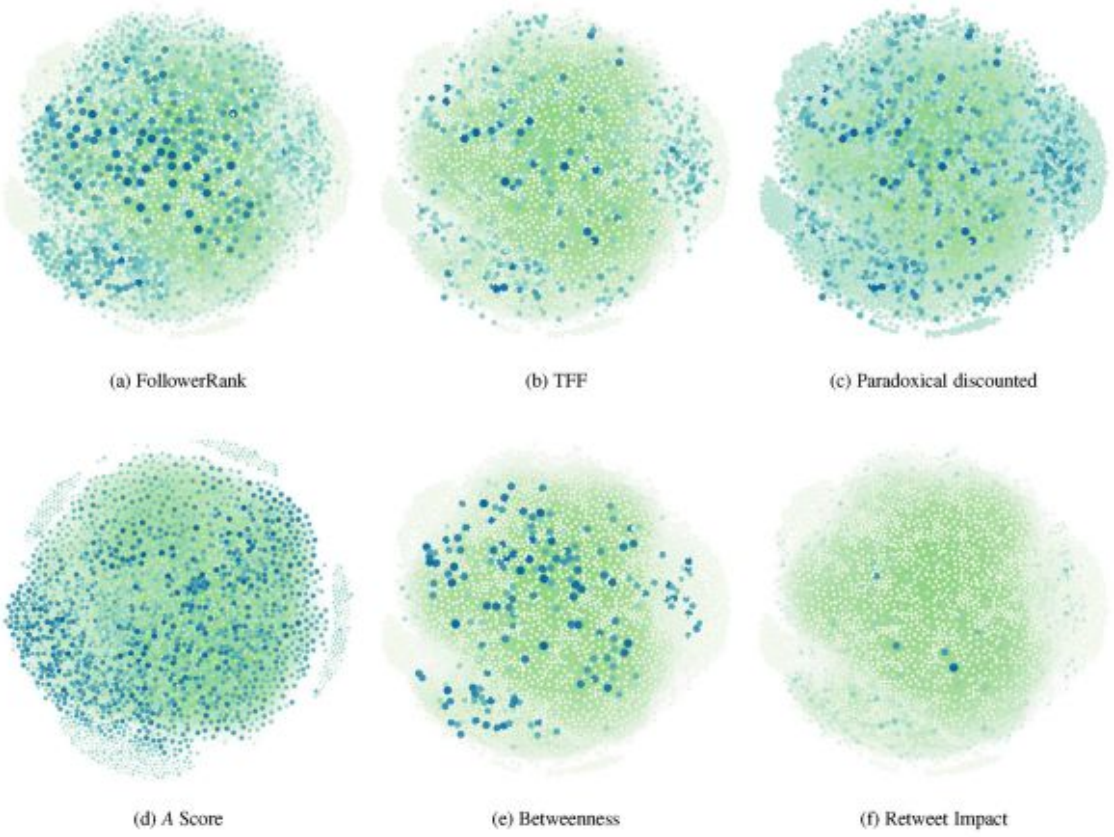

Figura 12. "Measuring user influence on Twitter: A survey". Fabián Riquelme, Pablo González-Cantergiani. (2015)

En general, hay cierto consenso en que el número de retuits es uno de los mejores indicadores para saber qué usuario es realmente influyente en una red (Riquelme & González-Cantergiani, 2015). Al fin y al cabo, el retuit es la acción que de forma más evidente sirve para difundir un mensaje. Otra manera más sencilla de saber quiénes son los usuarios más influyentes es mediante el número de seguidores que tiene y ésta es por la que se opta en este proyecto (al igual que hace la herramienta oficial de métricas de Twitter tal y como se ve en la Figura 13). Sin embargo, esta métrica tiene el problema de que nada garantiza que los seguidores sean de calidad; es decir, que sean cuentas activas o que estén predispuestas a retuitear al usuario (Cha *et al.* 2010). La circunstancia más evidente en la que esto puede ocurrir es que la cuenta puede estar haciendo de forma sistemática *followback* y por tanto, quienes le sigan no lo hagan por suscribirse a su contenido, por lo que serían poco influyentes. Para evitar esto, una medida muy sencilla pero que funciona razonablemente bien es ignorar quienes sigan a un número muy alto de usuarios, entendiendo que quien sigue a un número desproporcionado de personas lo hace

únicamente buscando el *followback* porque es inviable que puedan leer los mensajes de los otros usuarios.

Figura 13. El servicio oficial de analíticas de Twitter. Valora como "seguidor principal" al usuario con mayor número de seguidores

4.4 Seguimiento de la actividad

El objetivo de la herramienta no es únicamente determinar cómo es el entorno de un usuario determinado de Twitter. También se requiere poder hacer un seguimiento de una campaña, sobre todo desde la perspectiva de si está surtiendo efecto entre los usuarios más importantes del contexto de la cuenta. Este seguimiento se realiza obteniendo los tuits de la cuenta y almacenando quiénes son los usuarios que lo han retuiteado. En base a esta información, posteriormente se verá en el grafo de la visualización qué nodos están interactuando más y quiénes menos (Figura 14).

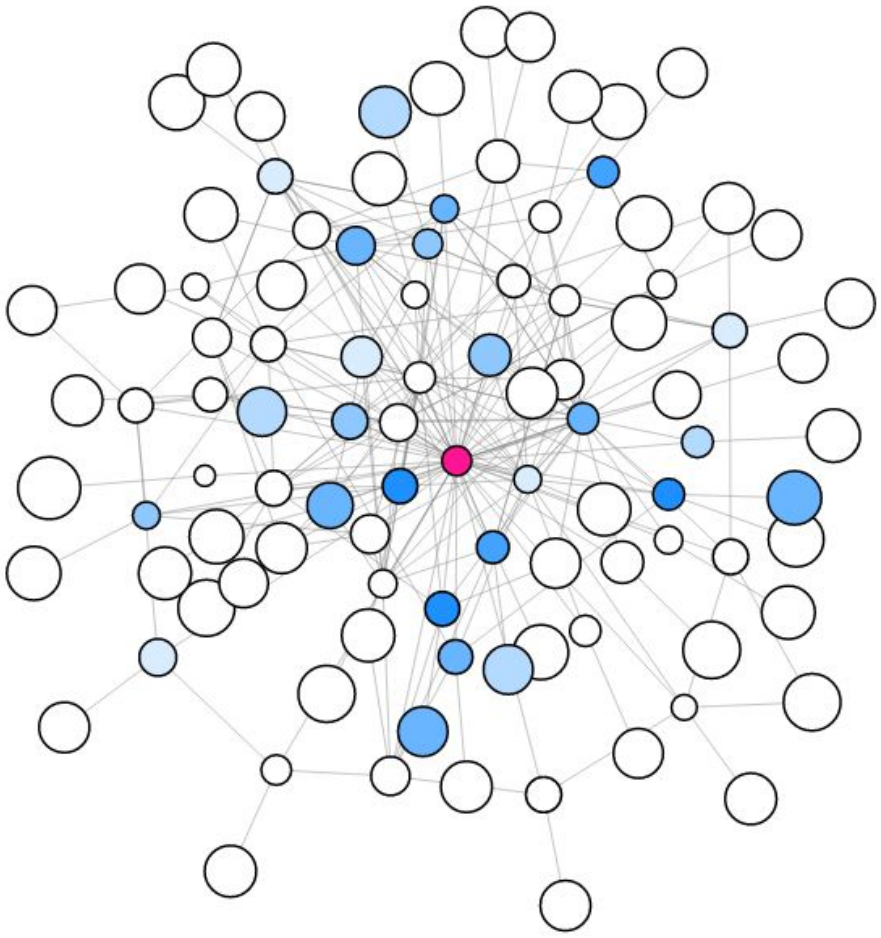

Figura 14. Usuarios según los retuits hechos a la cuenta de estudio

5 FASES

Con el fin de cumplir los objetivos planteados, y teniendo en cuenta el enfoque decidido, se desarrolla una aplicación web que dispondrá de toda la funcionalidad y mediante la cual el usuario interactuará.

Para que los datos que se encuentran en Twitter finalmente puedan ser visualizados en la aplicación web, se pasa por tres fases: primero, la extracción de los datos y su almacenamiento; segundo, tratamiento de los datos a fin de seleccionar los de mayor interés y obtener información a partir de los datos; tercero y último, visualizarlos y preparar el front-end para poder interactuar con ellos.

De forma simplificada, puede verse el proceso general por el que pasan los datos en la Figura 15.

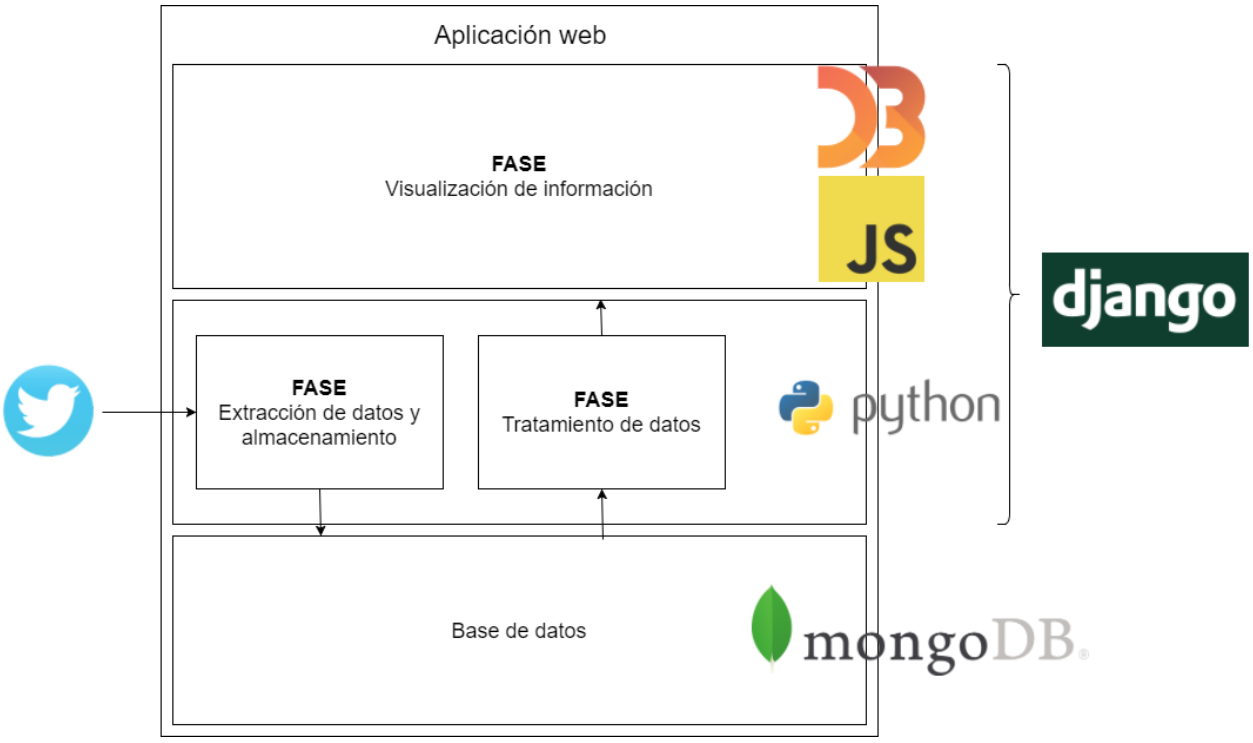

Figura 15. Flujo general de la aplicación

Desde la interfaz (Figura 16) tendremos tres opciones principales: *Extract*, que corresponde a la fase de extracción y almacenamiento de datos; Structure, donde se visualiza el grafo haciendo énfasis en su información estructural y que corresponde a las fases de tratamiento de datos y visualización de la información; y finalmente *Activity*, que muestra el grafo desde la perspectiva de los retuits que han hecho los usuarios y que también requiere de las fases de tratamiento de datos y visualización de información.

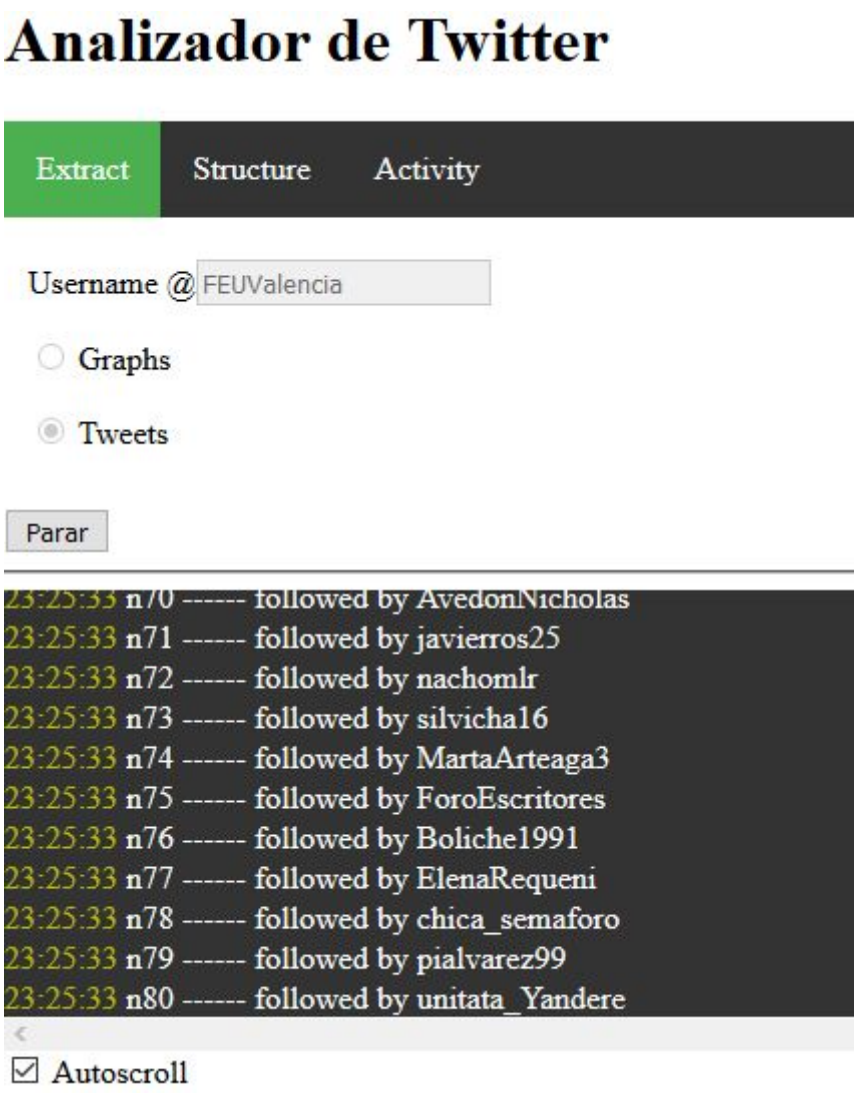

Figura 16. Interfaz de la aplicación desde la vista *Extract*.

5.1 Extracción y almacenamiento de datos

La primera fase del trabajo es la extracción de los datos de Twitter. Consiste en obtener los datos de la red social y adecuarla a la base de datos, dejándolos perfectamente disponibles para que posteriormente sean leídos.

5.1.1. Funcionalidad

El primer paso es introducir el nombre del usuario sobre el que se desea extraer los datos. Luego, hay que concretar qué información se desea extraer de él: si el grafo (usuarios y sus relaciones) o bien los tuits. Cuando se pulsa el botón *Explore*, comienza la extracción. Este proceso puede ser seguido por el usuario mediante la información que en tiempo real se muestra por pantalla. Esta información, en el caso de la extracción del grafo, se compone de: el usuario que acaba de ser registrado, el nivel de profundidad al que se encuentra, la hora a la que ha sido obtenido y el número seguidores del usuario que se han

extraído. Cuando se extraen los tuits, la información mostrada es la misma que la que se guarda en la base de datos. Estas cuestiones serán luego tratadas con mayor profundidad.

Toda esta información aparece en pantalla en tiempo real (Figura 16). Esto se lleva a cabo mediante la conexión HTTP entre ambos lados, servidor y cliente. El servidor envía el log de datos mediante el objeto Python HTTPResponse. El cliente, con un objeto XMLHttpRequest, se actualiza periódicamente esperando nuevos datos para reflejarlos en pantalla.

5.1.2. Seguidores

Para crear el grafo se necesitan dos tipos de datos: los usuarios y la relación entre ellos (cuáles siguen a cuáles). El algoritmo que obtiene estos datos, Algoritmo 1, lo hace mediante una búsqueda en profundidad. Es decir: comenzando por la raíz del árbol (el usuario de estudio), se van expandiendo cada vértice adyacente de forma recurrente hasta que no queden más en ese camino. Entonces, regresa y se repite el proceso con cada uno de los hermanos.

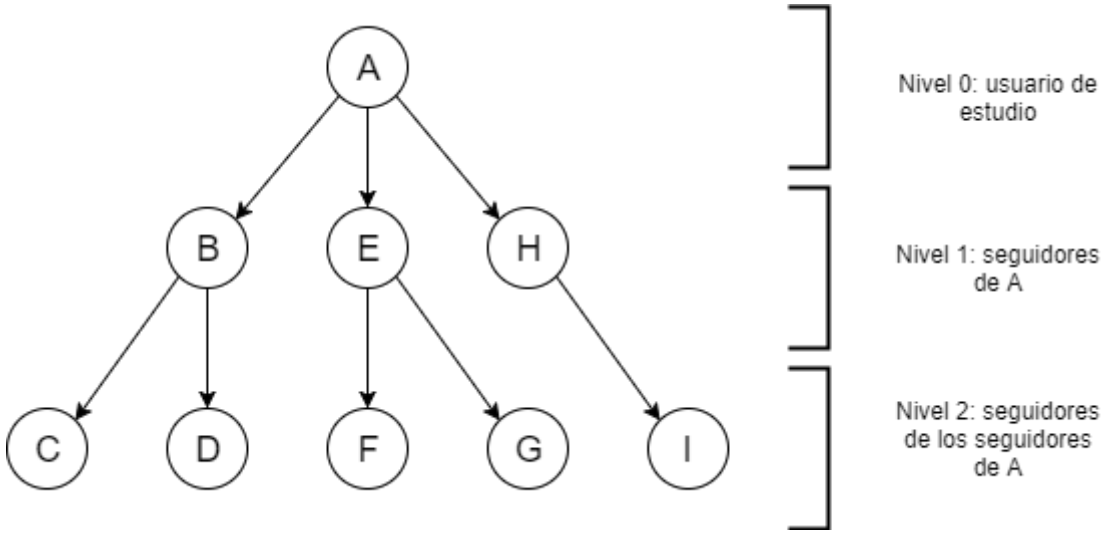

Figura 17. Búsqueda en profundidad de usuarios

Tomando el ejemplo de la Figura 17, en el que A es seguido por B, E y H; B es seguido por C y D; E es seguido por F y G; y H es seguido por I, ejecutando el algoritmo, el orden de exploración es el mismo que el alfabético. Hay que tener en cuenta que en el algoritmo ha de indicarse el nivel de profundidad hasta el que queremos explorar. Como se decía en el apartado 4.1 El entorno de una cuenta de Twitter, en este documento sólo se exploran usuarios hasta el segundo nivel de profundidad.

Otra cuestión a tener presente es la anchura máxima que se explora de cada usuario, es decir, cuántos de sus seguidores se está dispuesto a explorar. Es necesario poner un máximo razonable porque si intenta explorar un usuario con, por ejemplo, cientos de miles de seguidores, desperdiciaremos muchos recursos y tardará días en acabar.

```
Input: user to explore, current depth, max depth and max width

FUNCTION explore_user
     insert_db_user
     FOR each follower
          IF users_count = max_width
               BREAK
          insert_db_relation
          IF current_depth = max_depth
               insert_db_user
          ELSE
               current_depth = current_depth + 1
               explore_user
```

Algoritmo 1. Exploración de usuarios de forma recursiva

El algoritmo requiere como *inputs* el usuario a explorar, la profundidad actual en la que se encuentra el usuario y la profundidad y anchura máximas a las que se desea llegar. Para cada seguidor, comprueba si ya se ha llegado a la máxima anchura en la exploración. En tal caso, interrumpe el bucle. Si no lo ha hecho, guarda la relación entre ambos usuarios (usuario B sigue a usuario A, en el ejemplo de la Figura 17). Luego se comprueba si el usuario que se está explorando en este momento ya se encuentra en la profundidad máxima. De ser así, se guarda el usuario en la base de datos. Por contra, si esa profundidad aún no se ha alcanzado, se vuelve a hacer una llamada a la función con los parámetros del nuevo usuario.

Aunque no figure en el Algoritmo 1 puesto que sólo aparece lo fundamental, hay que tener en consideración que los usuarios con cuenta protegida no podrán ser explorados. Sólo se puede registrar que existe, pero no extraer la información de sus seguidores.

En caso de interrupción, si volvemos a iniciar la recogida de datos de los usuarios, los que ya hayan sido explorados en la colección no volverán a explorarse. Esto es debido a que, siendo un proceso que puede tardar horas y por tanto es probable que se detenga sin haber acabado, se guarda como información de cada usuario si ya ha sido completamente explorado o no.

5.1.3. Tuits

El caso de la extracción de tuits es más sencillo. Sólo es necesario recorrer todos los tuits generados por el usuario de estudio y guardar la información que consideremos necesaria en la base de datos.

5.1.4. Peticiones a la API de Twitter

En el proyecto, la comunicación con la API de Twitter se desarrolla a través de la librería Tweepy. Pese a que permite muchas acciones distintas como por ejemplo publicar tuits o retuitarlos, únicamente se van a recoger datos. Grosso modo, este proceso consiste en

realizar una petición mediante una serie de parámetros y Twitter devuelve un archivo JSON, un formato de texto con el que se trabaja de forma muy cómoda.

Uno de los mayores problemas a la hora de recabar datos de Twitter son los límites que esta empresa impone para el uso de su API. El principal motivo de su existencia es que ningún desarrollador pueda hacer un uso abusivo y, por tanto, generar costes a la empresa. Tal y como se hacen las peticiones en el Algoritmo 1 se extraen 300 usuarios cada 15 minutos, haciendo que podamos tardar varios días en construir un grafo de un usuario con pocos seguidores. ¿Cómo aumentar en estas condiciones la capacidad de extracción?

Los límites de número de peticiones por tiempo se aplican por cada *token* de acceso. Esto significa que con el uso de varios *tokens* se puede aumentar la velocidad de extracción, pero surge otro problema: ¿cuando una *token* ha excedido el límite de su uso, cómo continuar ese trabajo con otra? La respuesta a esta cuestión reside en hacer un buen uso de los cursores, un objeto que permite iterar sobre una colección de forma sencilla. Cuando un *token* ha llegado a su límite, ha de almacenarse el *id* del siguiente cursor para que se siga iterando desde ahí pero con un nuevo *token* de acceso.

Llegado este momento, aumentar la velocidad de esta extracción se limita a obtener un mayor número de cuentas. El proyecto funciona en estos momentos con seis cuentas, pero perfectamente pueden aumentarse para mejorar el rendimiento de la herramienta. Esto es así hasta que se alcanza un número aproximado de 45 cuentas (según la velocidad a la hora de enviar peticiones a la API de Twitter), momento a partir del cual se vuelve necesario paralelizar y hacer una planificación más compleja del reparto de datos a extraer.

5.1.5. Almacenamiento

Los datos se almacenan en MongoDB, un sistema de base de datos basado en documentos. Hay una serie de factores que hacen que sea una decisión adecuada. Más allá de que es sencillo de manejar para el uso que le da esta herramienta, el principal argumento es que puede alcanzar un tamaño muy grande. No obstante, para estudiar una única cuenta pueden llegar a almacenarse cientos de miles de nodos y miles de tuits.

Cuando se extrae el entorno y los tuits a partir del nombre de un usuario, los datos se guardan en tres colecciones distintas: users, que contiene los datos de cada usuario; followers, la relación entre estos; y *tweets*, todos los tuits del usuario que

Datos almacenados de cada usuario:

- ***id_str***: una cadena con el *id* de cada usuario. Siempre es el mismo, a diferencia del nombre de usuario que sí puede cambiarse
- ***screen_name***: nombre del usuario que sirve para hacer menciones y va precedido de "@". Puede ser cambiado.
- ***name***: a diferencia los anteriores, este sí puede contener espacios y ser más largo. Puede ser cambiado.
- ***followers_count***: número de seguidores.

- ***friends_count***: número de usuarios a los que sigue.
- ***location***: localización del usuario introducida por él mismo. No es obligatorio introducirla ni tiene por qué ser una localización real.
- ***explored***: indica si el usuario ya ha sido explorado

Datos almacenados de cada tuit:

- ***id_str***: *id* único de cada tuit
- ***created_at***: fecha de publicación del tuit en el formato internacional definido por ISO (IS0 8601)
- ***text***: texto del tuit
- ***retweet_count***: número de retuits que ha recibido
- ***favorite_count:*** número de me gusta que ha recibido
- ***rt_users_id***: un array con los *id_str* de los usuarios que lo han retuiteado

5.2 Tratamiento de datos

Después de extraer los datos con los que se va a trabajar, toca tratarlos. Esta es la fase en la que se parte de los datos guardados en la base de datos y acaba cuando son enviados al *front-end* para que se cree la visualización. Es en esta parte, de hecho, cuando se puede decir que se transforman los datos en información porque se toman los valores de la base de datos para discriminar unos y juntar otros, aportando un mayor significado como resultado.

5.2.1. Selección de los vértices con más seguidores

Puesto que no se va a mostrar la totalidad del grafo, se visualizará un subgrafo de éste donde aparezcan los *n* usuarios con más seguidores. La problemática de esto surge cuando por una parte está el usuario de estudio, los *n* usuarios con más seguidores y no necesariamente una relación entre ellos. Es preciso entonces reconstruir el subgrafo con todos los vértices existentes entre unos y otros.

```
Input: most followed users and users adjacent to main user
Output: list with subgraph users

subgraph_users += most_followed_users

FOR each user in most_followed_users
     IF is_not_adyacent_to_main_user
          subgraph_users += adyacent_to_most_followed_users
```

Algoritmo 2. Selección de los usuarios del subgrafo con los *n* usuarios con más seguidores

Una vez están claros cuáles son los usuarios que forman parte del subgrafo, obtener los arcos que los unen es trivial, poco más que una serie de consultas a la base de datos.

5.2.2. Centralidad

Para añadir significado a la visualización, la herramienta también muestra la centralidad de grado de cada vértice del subgrafo. La centralidad de grado es una medida muy simple de centralidad, es decir, la noción de cómo de central es un vértice dentro de un grafo. Se opta por esta medida de centralidad por dos motivos: 1) es computacionalmente muy poco costosa en comparación con otras alternativas; 2) en grafos pequeños, como son los que esta aplicación visualizará, esta medida es suficientemente buena. En otros más grandes es mejor optar por otras que tengan en cuenta la estructura global del grafo y no sólo la local (Scott 2017). Otras medidas de centralidad pueden verse en la Figura 18.

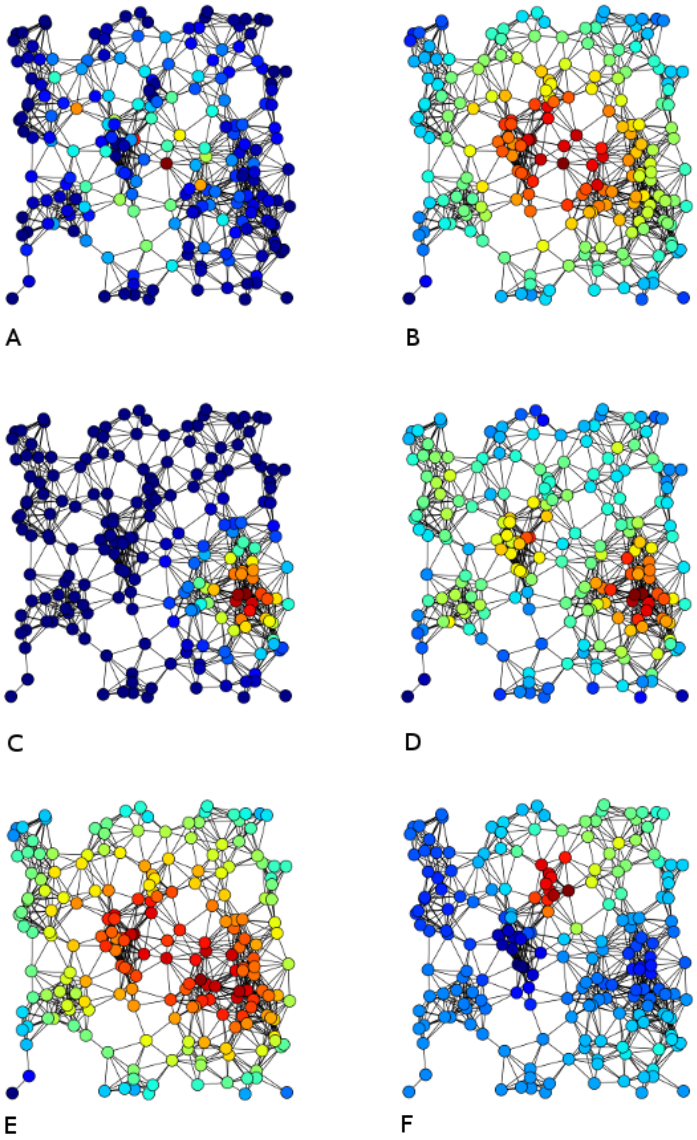

Figura 18. Medidas de centralidad. A) intermediación, B) cercanía, C) valor propio, D) grado, E) centralidad armónica, F) centralidad de Katz. Las tonalidades van del azul al rojo e indican menor a mayor centralidad respectivamente.

Dado que en este caso se trata de un grafo dirigido, la centralidad tiene que ser el grado de salida o el de entrada. El enfoque de este trabajo es que cuantos más seguidores tiene un usuario, más central es, así que se considerará sólo el grado de entrada. Tomando la definición de grafo dirigido propuesta en 2. Conceptos previos, la centralidad de grado queda definida de la siguiente manera:

$$C_{DEG}(v) \ = grado_E(v)$$

Como este dato se verá como un tono de color y no de forma numérica, es preciso calcular el valor normalizado por cada uno de ellos. Esto se hace dividiendo cada centralidad por la centralidad máxima.

5.2.3. Retuits por usuario

Como se dijo en el capítulo 4. Enfoque, el seguimiento de la actividad se hará en base a los retuits que cada usuario del subgrafo haya hecho a la cuenta de estudio en cierto periodo de tiempo. Esto se hace, primero, obteniendo el subgrafo (vértices y arcos) como se ha visto antes. Después se ha de hacer una consulta a la base de datos de tal forma que devuelva todos aquellos tuits publicados en el rango de tiempo especificado. A partir de esta información, se genera un diccionario donde los usuarios son la clave y el valor es el número de retuits que ha hecho.

5.3 Visualización de información

La correcta visualización de datos es fundamental para que la herramienta sea útil y sirva para analizar la información. La visualización está compuesta de muchos detalles que tratan de potenciar el significado de cada elemento y aportar funcionalidad en la interacción. En este apartado se explicarán.

5.3.1. Estructura

La estructura del grafo puede verse desde la vista *Structure* de la interfaz (Figura 19). Desde ahí se selecciona cualquiera de los usuarios explorados previamente, se indica el número de usuarios con más seguidores a partir de los que se reconstruye el grafo y luego el umbral de usuarios a los que pueden seguir para no ser descartados.

La visualización comienza con una pequeña animación en la que todos los elementos se van colocando en relación unos a otros para que sea lo más fácil posible entender cómo se relacionan todos los usuarios. Esto es importante porque pese a que no se use ninguna técnica específica de agrupamiento, puede apreciarse que la distribución de los usuarios sí guarda una relación estructural. De esta manera, pueden verse grupos más o menos cohesionados según la

temática, sector profesional o localización geográfica. Además, en cada nodo puede leerse el nombre del usuario al que representa (Figura 19).

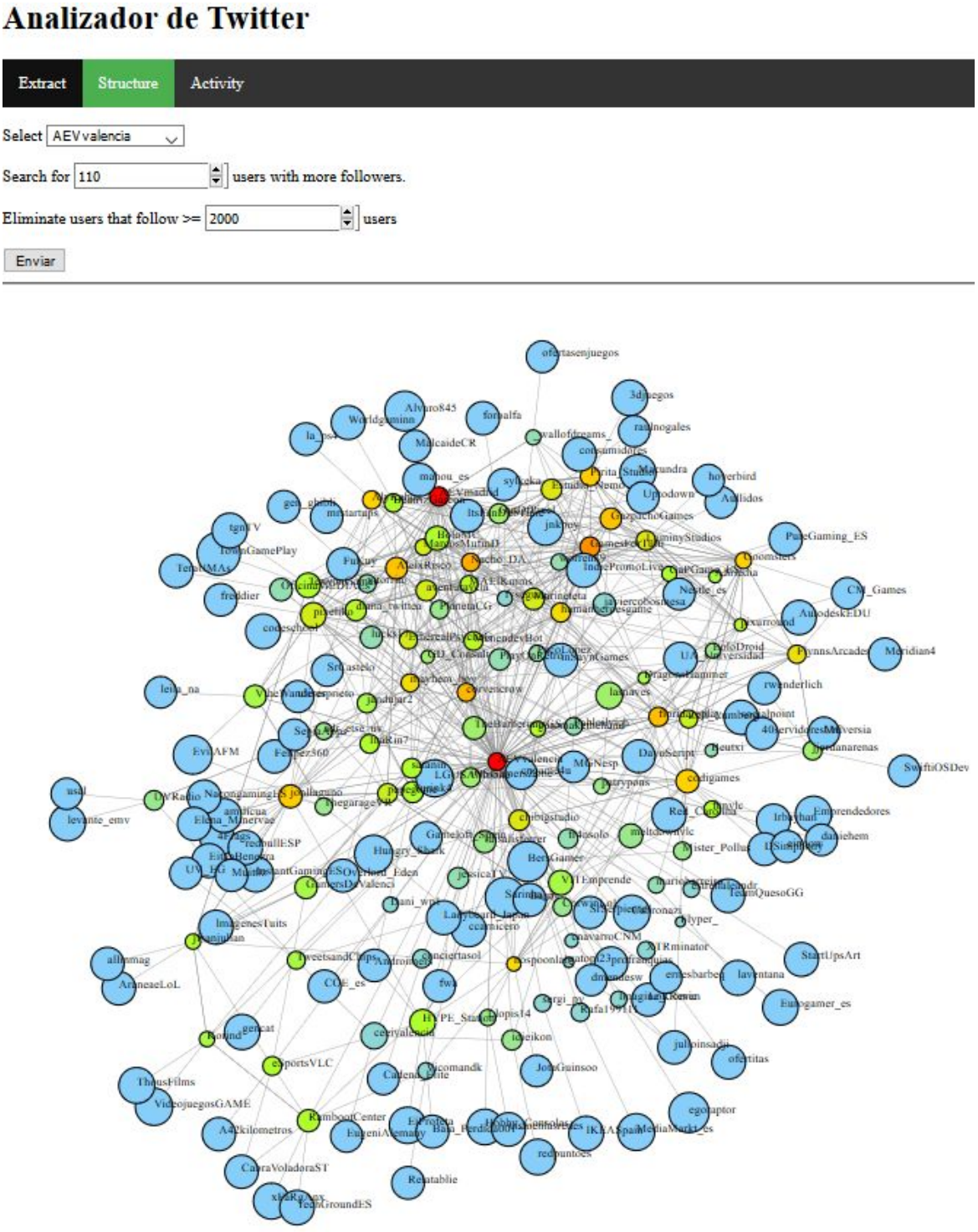

Figura 19. Interfaz de la vista *Struct* mostrando un grafo

5.3.2. Tamaño de los vértices

Los usuarios influyentes, como se explica en el capítulo 4 Enfoque, se entienden en este trabajo como los usuarios con un mayor número de seguidores una vez se han eliminado aquellos que se considera que hacen *followback*. Esta noción de influencia es necesaria remarcarla en la visualización y se hace mediante el tamaño de los vértices. Su radio responde a la siguiente fórmula siempre que no exceda de un máximo o no llegue a cierto mínimo, caso en el que adoptaría unos valores fijos:

$$radio = log_2(seguidores)$$

5.3.3. Centralidad

Como puede observarse en la Figura 20 del grafo, los vértices con mayor número de arcos en el subgrafo son los que cuentan con un color más cálido.

Figura 20. Tonos de centralidad. De azul a rojo según menos y más centralidad respectivamente

Puede apreciarse cómo normalmente los usuarios con mayor número de seguidores (los más grandes) son, a su vez, los que menor centralidad tienen. Lo que en un primer momento podría parecer paradójico es en realidad normal: los que tienen un alto número de followers en Twitter no son del entorno más directo del usuario de estudio; están poco relacionados con el resto de usuarios del subgrafo. Si en lugar de un subgrafo se trabajara con toda la red de Twitter, los vértices con un mayor número de seguidores sí serían los que más arcos tendrían, pero la visualización toma sólo el contexto en el que se encuentra determinada cuenta.

5.3.4. Interactividad

Con la finalidad de facilitar la capacidad de análisis, la visualización permite la posibilidad de focalizar la atención sobre un vértice concreto haciendo clic izquierdo en él (Figura 21). Cuando esto ocurre, se activa una suave animación en la que el vértice aumenta de tamaño un 40% y todos sus seguidores presentes en el grafo quedan resaltados en color naranja: tanto sus perímetros como los arcos que los unen. A la vez, el resto de elementos se hacen translúcidos.

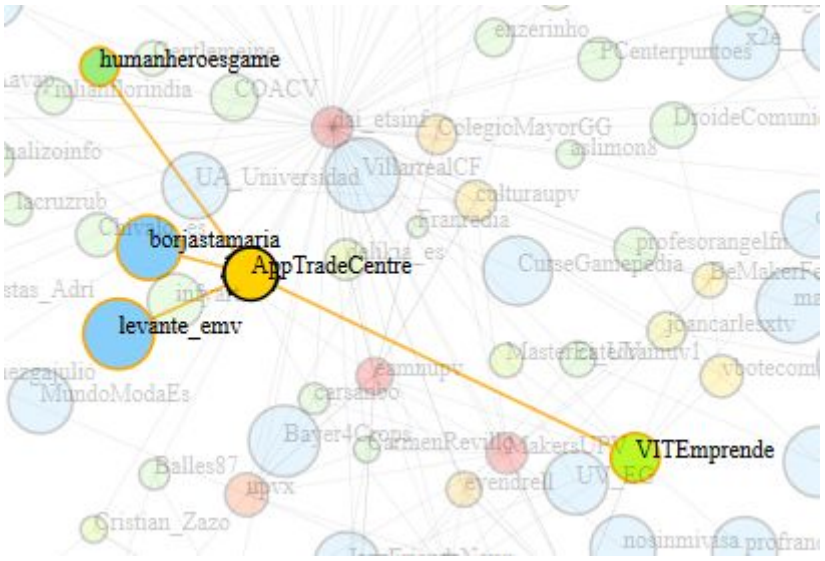

Figura 21.El usuario @*AppTraceCentre* focalizada con sus seguidores destacados

De forma similar, pero manteniendo el botón izquierdo del ratón pulsado, se pueden arrastrar los vértices (Figura 22). Esto sirve, por ejemplo, para emplazarlos en otro sitio con el fin de poder verlo con más atención.

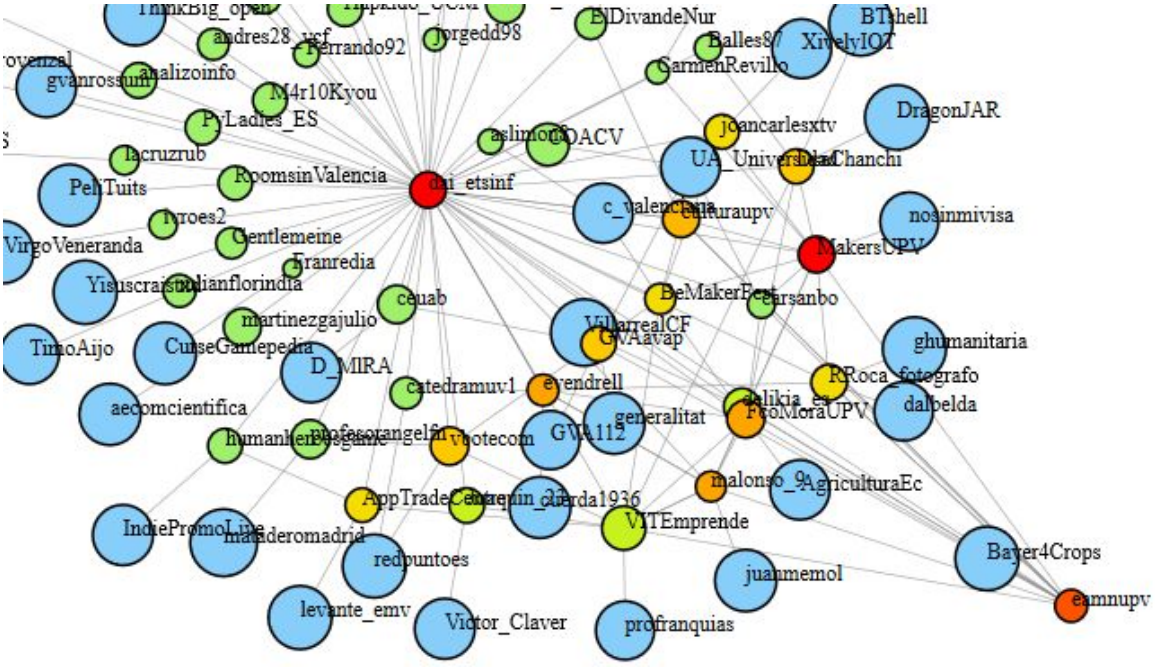

Figura 22. El usuario @*eamnupv* siendo arrastrado

Si se mantiene el ratón sobre cualquiera de los vértices, aparece un recuadro con información adicional del usuario: el número de seguidores que tiene, a cuántos sigue él y su localización (en caso de que el usuario la haya introducido). Esto puede verse en la Figura 23.

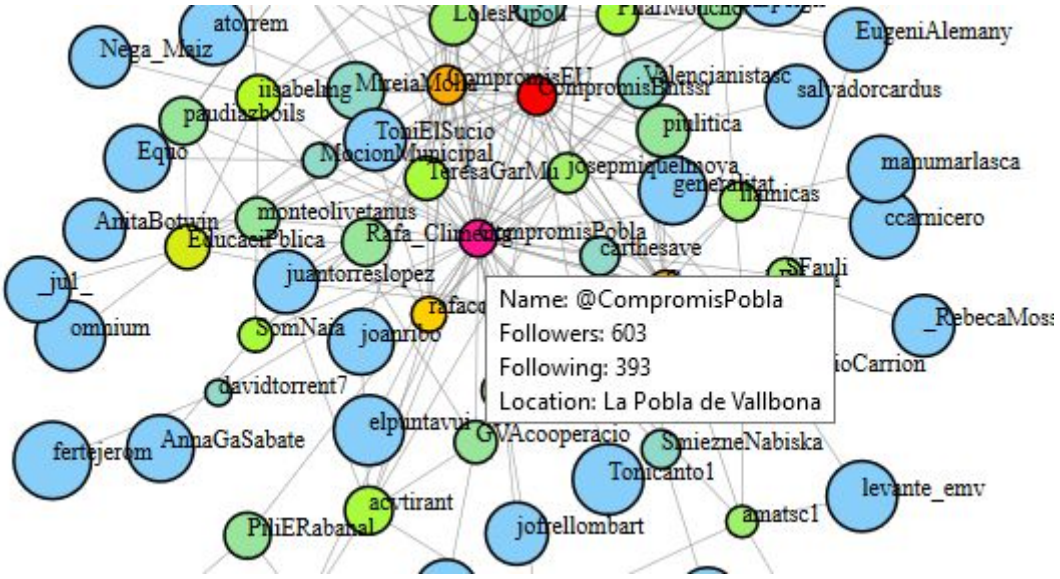

Figura 23. Información adicional del usuario *@CompromisPobla*

Si se desea obtener más información de un usuario, otra opción muy útil es hacer doble click en los vértices. Al hacerlo, se abre una nueva pestaña en el navegador correspondiente a la página de Twitter de la cuenta seleccionada. De esta forma, rápidamente se puede conocer en profundidad cualquiera de los usuarios del subgrafo.

5.3.5. Actividad

La interfaz para mostrar el grafo de actividad es muy similar a la de estructura. Por la parte de los *inputs*, tiene los mismos con dos añadidos: fecha de inicio y fecha fin (Figura 24). Estos dos campos sirven para indicar el intervalo de fechas en los que estarán los retuits que se visualizarán en el grafo. Si no se introduce ningún intervalo, se usarán todos los retuits existentes en la base de datos.

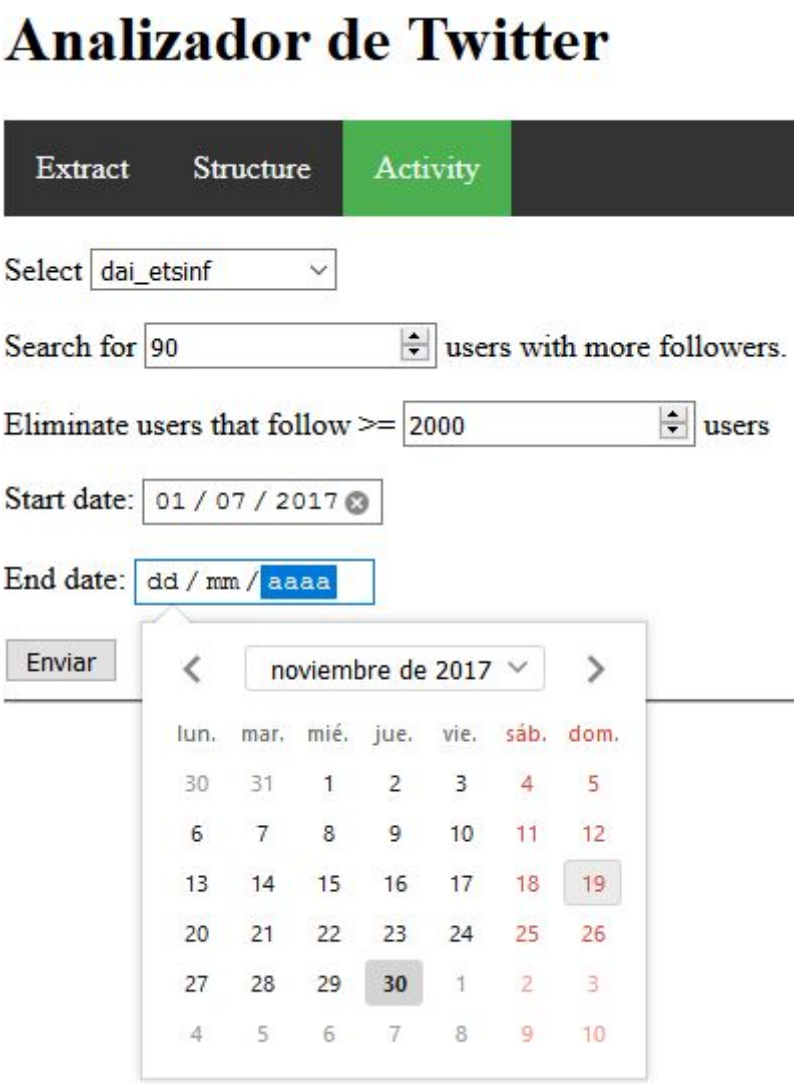

Figura 24. *Inputs* de la interfaz *Activity*

La visualización del grafo de actividad es muy similar a la de la estructura. Comparten muchos elementos: vértices, su tamaño, arcos e interactividad. Lo que cambia es básicamente el color de los vértices. Se sustituye azul, más o menos intenso conforme más o menos retuits han hecho en el periodo seleccionado a la cuenta de estudio (Figura 25).

Figura 25. Actividad. Azul más intenso conforme más retuits ha hecho un usuario

Este degradado queda finalmente en el grafo como se indica en la Figura 26.

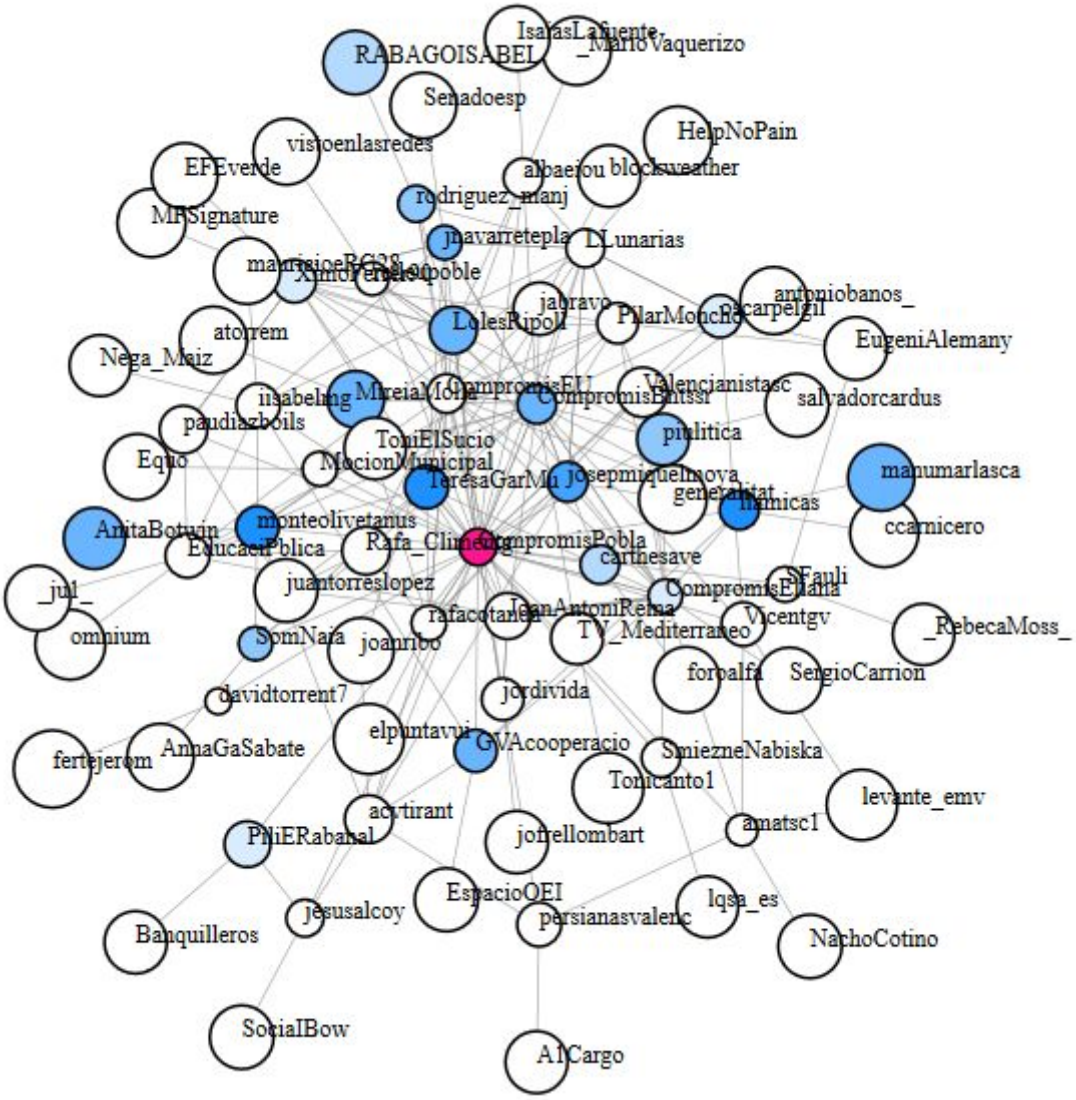

Figura 26. Grafo del usuario @*CompromisPobla* con los usuarios que más retuitean en un color azul más intenso

6 CASOS DE ESTUDIO

Dada la intención de que el proyecto sea una herramienta práctica para que una organización mejore su comunicación en Twitter, en este capítulo se exponen tres casos de estudio, concretamente de las cuentas de la Delegación de alumnos de la ETSINF (*@dai_etsinf*), una agrupación local de un partido político (@CompromisPobla) y una asociación de estudiantes de videojuegos de Valencia (*@AEVvalencia*).

La intención no es detallar la estrategia comunicativa que una organización ha de seguir para exprimir el potencial de la aplicación en cada uno de los casos de estudio. Lo que se quiere en este capítulo es analizar brevemente algunos usuarios y su entorno como ejemplo del funcionamiento de la herramienta. Este análisis puede servir, entre otras cosas, para entender qué tipos de usuarios hay en el contexto del usuario estudiado y generar contenido afín y/o planear cómo llegar a los actores del entorno más importantes (mediante publicidad, menciones, retuits u otras formas de interacción)

6.1 D.A. ETSINF

Dado que el lector de esta memoria es alguien potencialmente vinculado a la UPV, se comienza con un caso de estudio relacionado con la universidad. De esta forma, la información que aparezca en el grafo será familiar para el lector. Concretamente se ha optado por la Delegación de Alumnos de la ETSINF puesto que es un usuario abordable por su tamaño (1.031 seguidores) y puede ser interesante por su relación con los alumnos, empresas y entorno UPV.

6.2.1 Descripción

Los datos de la base de datos recogidos de la cuenta *@dai_etsinf* son los que figuran en la Tabla 1.

Nombre de la cuenta:	*@dai_etsinf*
Usuarios (vértices)	128.173
Relaciones (arcos)	158.306
Tweets	806

Tabla 1. Datos recogidos de la cuenta *@dai_etsinf*

La siguiente vista (Figura 27) la componen los 70 usuarios con más seguidores y que al mismo tiempo siguen a menos de 2000 (para evitar tener en cuenta usuarios que hagan *followback*). A partir de estos usuarios, los relacionados con ellos hasta componer el grafo.

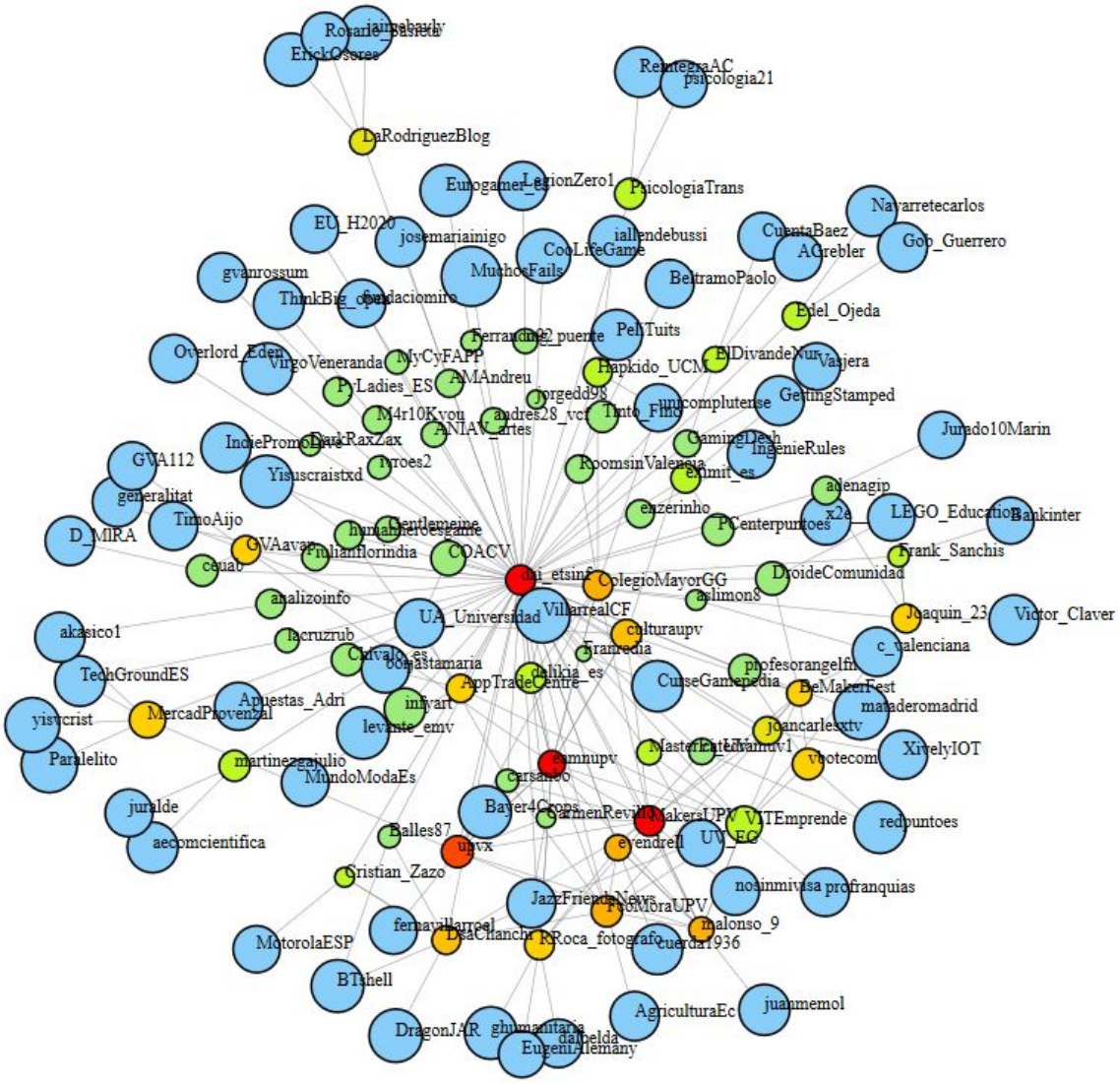

Figura 27. Los 70 usuarios con más seguidores y los usuarios que están relacionados con ellos

6.2.2 Análisis

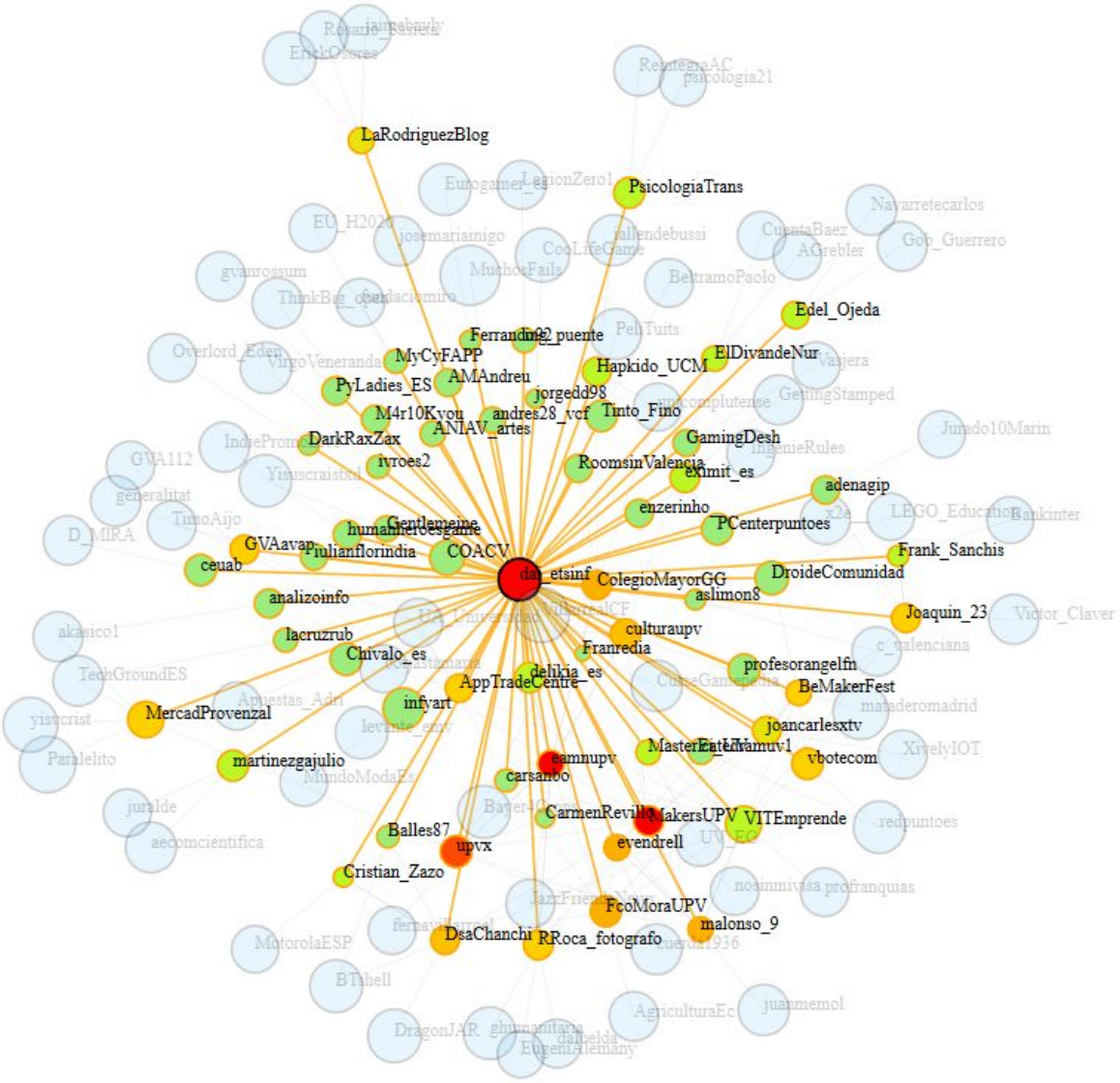

Figura 28. Entorno más cercano a @dai_etsinf. Sus vértices adyacentes

Del entorno más cercano a la cuenta (Figura 28), podemos apreciar instituciones u organizaciones de la UPV, personalidades de la universidad y alumnos actuales o pasados de la escuela.

Los nodos con mayor centralidad en el grafo, los de color rojo, están directamente relacionados con la UPV. Son la *Escuela Técnica Superior d'Ingenyieria Agronòmica i del Medi Natural* (*@eamnupv*), una asociación de estudiantes bastante vinculada a la Escuela de Informática (*@MakersUPV*) y la plataforma de cursos abiertos masivos online de la UPV (*@upvx*). En un tono menos cálido pero aún así con una alta centralidad, en naranja, siguen apareciendo este tipo de cuentas: el Área de Actividades Culturales (*@culturaupv*) y la residencia Galileo Galilei (*@ColegioMayorGG*).

También pueden verse tres personalidades de la universidad, como son el actual rector Francisco Mora (*@FcoMoraUPV*), Eduardo Vendrell, exdirector de la ETSINF (*@evendrell*), y Marina Alonso Díaz, profesora titular de la universidad que imparte clases en la escuela. Es relevante comprobar que en la visualización se encuentran posicionados muy cerca, evidenciando la relación que existe entre ellos (Figura 29).

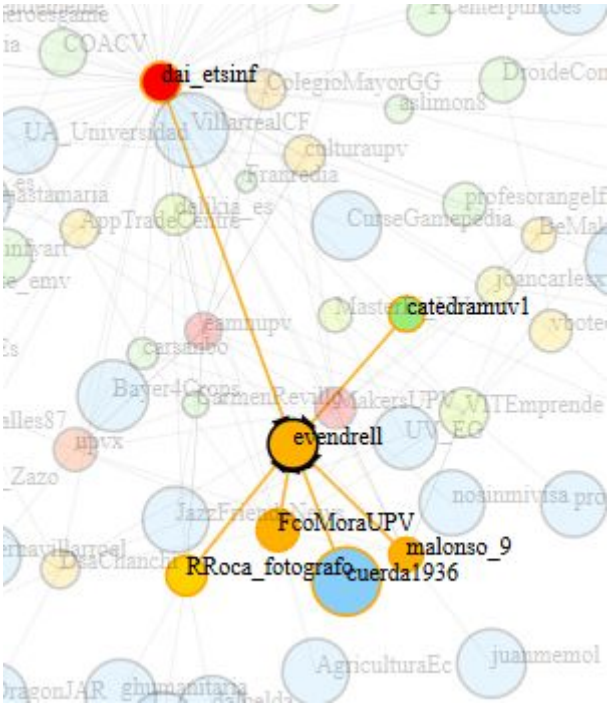

Figura 29. Seguidores de Eduardo Vendrell (*@evendrell*)

En la parte superior del grafo (Figura 28) puede apreciarse muchos vértices verdes. Esto significa que tienen poca centralidad. Se debe a que, a diferencia de lo que ocurre con los vértices inferiores, tienen una baja relación orgánica con la universidad y por tanto están poco cohesionados. Muchos de esos vértices superiores son alumnos o exalumnos (*@andres28_vcf*, *@DarkRaxZax* o *@lacruzrub*). Otros son proyectos relacionados con la informática e instalados en la Comunidad Valenciana, y aunque no se puede tener la seguridad, probablemente muchos de ellos vinculados a alumnos o exalumnos de la escuela (así es, por ejemplo, en el caso de *@humanheroesgame*).

Con el mundo empresarial tecnológico también se aprecia un estrecho vínculo. El más fuerte con el *@AppTradeCenter*, *@VITemprende* o *@delikia_es*, que si bien ésta última no es puramente tecnológica, sí tiene mucha relación con la universidad y ha patrocinado eventos de estudiantes en la escuela. Otras empresas que aparecen son Motorola España (*@MotorolaESP*), Bankinter (*@Bankinter*) o Telefónica (a través de su blog corporativo *@ThinkBig_open*).

Por último, destacar la presencia de medios de comunicación, dos relacionados con la tecnología (*@TechGround* o *@Eurogamer_es*) y uno generalista local (@levante_emv). Además, aparecen cuentas oficiales de instituciones públicas autonómicas como son la de la propia Generalitat (*@generalitat*) o la *Agència Valenciana d'Avaluació i Prospectiva* (*@GVAavap*), que está relacionada con la educación.

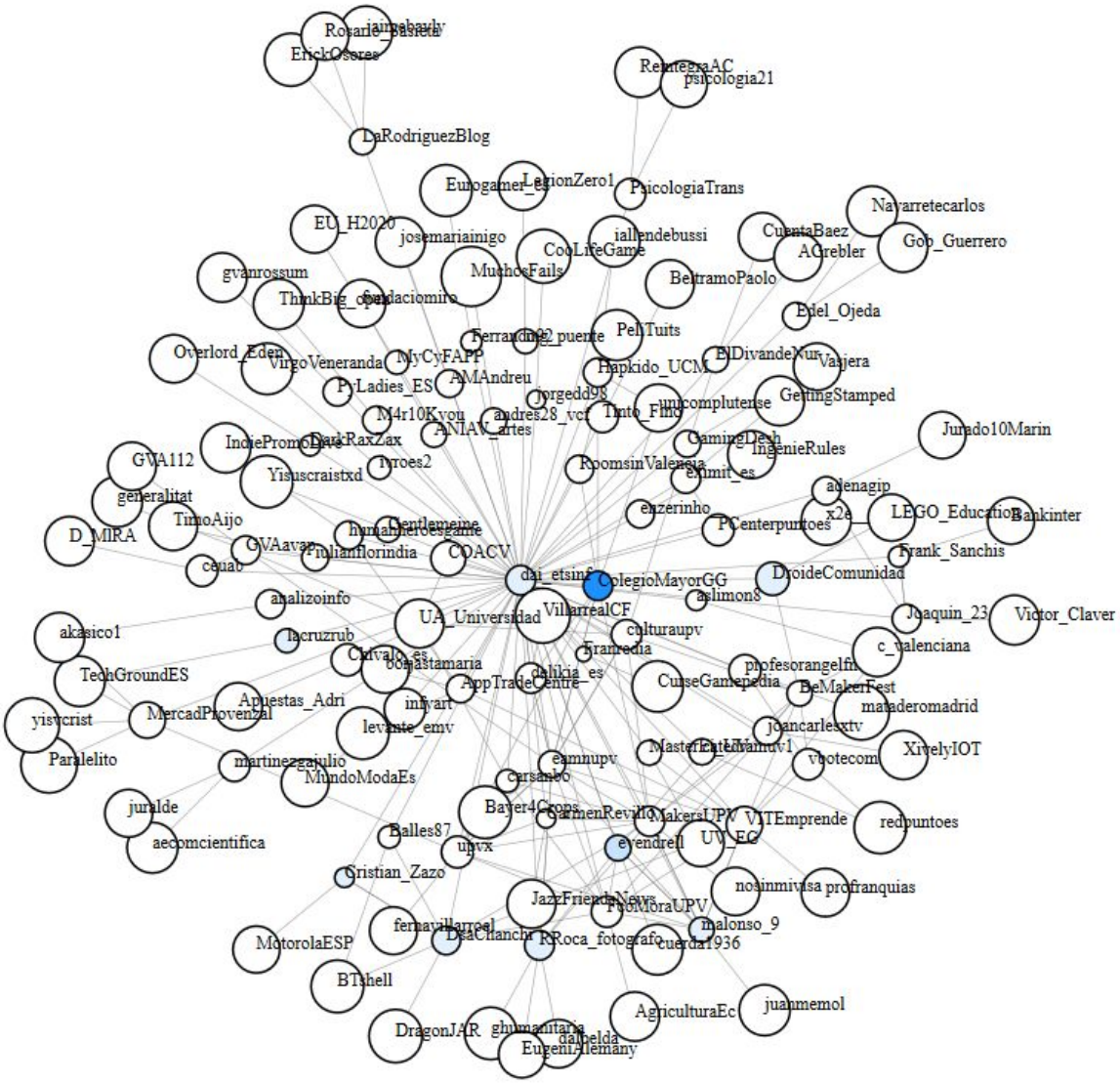

Figura 30. Usuarios del entorno de *@dai_etsinf* que han retuiteado alguno de sus tuits. Datos históricos

Si analizamos la actividad (Figura 30), podemos ver cómo la mayoría de vértices son de color blanco. Esto significa que se trata de una organización poco ligada a su comunidad y como se verá, los usuarios importantes de su entorno normalmente sólo realizan retuits cuando tienen un motivo bastante concreto para hacerlo.

Aunque con un bajo número de retuits, Eduardo Vendrell (@evendrell) y Marina Alonso Díaz (@malonso_9) son de los pocos usuarios que han difundido mensajes de la Delegación de Alumnos. Esto es sin duda debido a su estrecha relación con las actividades promovidas o difundidas por ella.

También es apreciable que, por mucho, la residencia Galileo Galilei (*@ColegioMayorGG*) es el usuario que más retuitea la cuenta. Esto probablemente se deba a la difusión de eventos organizados en la ETSINF que desde la residencia se consideraron interesantes para sus

estudiantes residentes. Una situación similar ocurre con @DroideComunidad, un centro de enseñanza tecnológico con el que es probable que se organizara alguna actividad. Sin embargo, si observamos los retuits hechos por estas dos cuentas en el último año, concretamente desde el 17 de noviembre de 2016 al 17 de noviembre de 2017 (Figura 31), se aprecia nula actividad. Esto probablemente tenga que ver con haber dejado de hacer actividades que pudieran resultar de su interés.

Figura 31. Usuarios del entorno de *@dai_etsinf* que han retuiteado alguno de sus tuits. Datos desde el 17/11/16 al 17/11/17

6.2 Compromís La Pobla de Vallbona

Es usual el estudio de fenómenos políticos en las redes sociales y un caso de estudio de este tipo resulta interesante. Además, es el único caso de estudio que está circunscrito a un lugar determinado: La Pobla de Vallbona.

La elección concreta de Compromís dentro del resto de partidos con cuenta de Twitter del pueblo se debe a que tiene un número manejable de seguidores y que es el actual partido en el gobierno, por lo que se espera más interactividad que el resto de cuentas y puede llegar a ser más interesante.

6.2.1 Descripción

Los datos de la base de datos recogidos de la cuenta *@CompromisPobla* son los que figuran en la Tabla 2.

Nombre de la cuenta:	*@CompromisPobla*
Usuarios (vértices)	88.427
Relaciones (arcos)	186.522
Tweets	375

Tabla 2. Datos recogidos de la cuenta *@CompromisPobla*

La siguiente vista (Figura 32) la componen los 50 usuarios con más seguidores y que al mismo tiempo siguen a menos de 2000 (para evitar tener en cuenta usuarios que hagan *followback*). A partir de estos usuarios, los relacionados con ellos hasta componer el grafo.

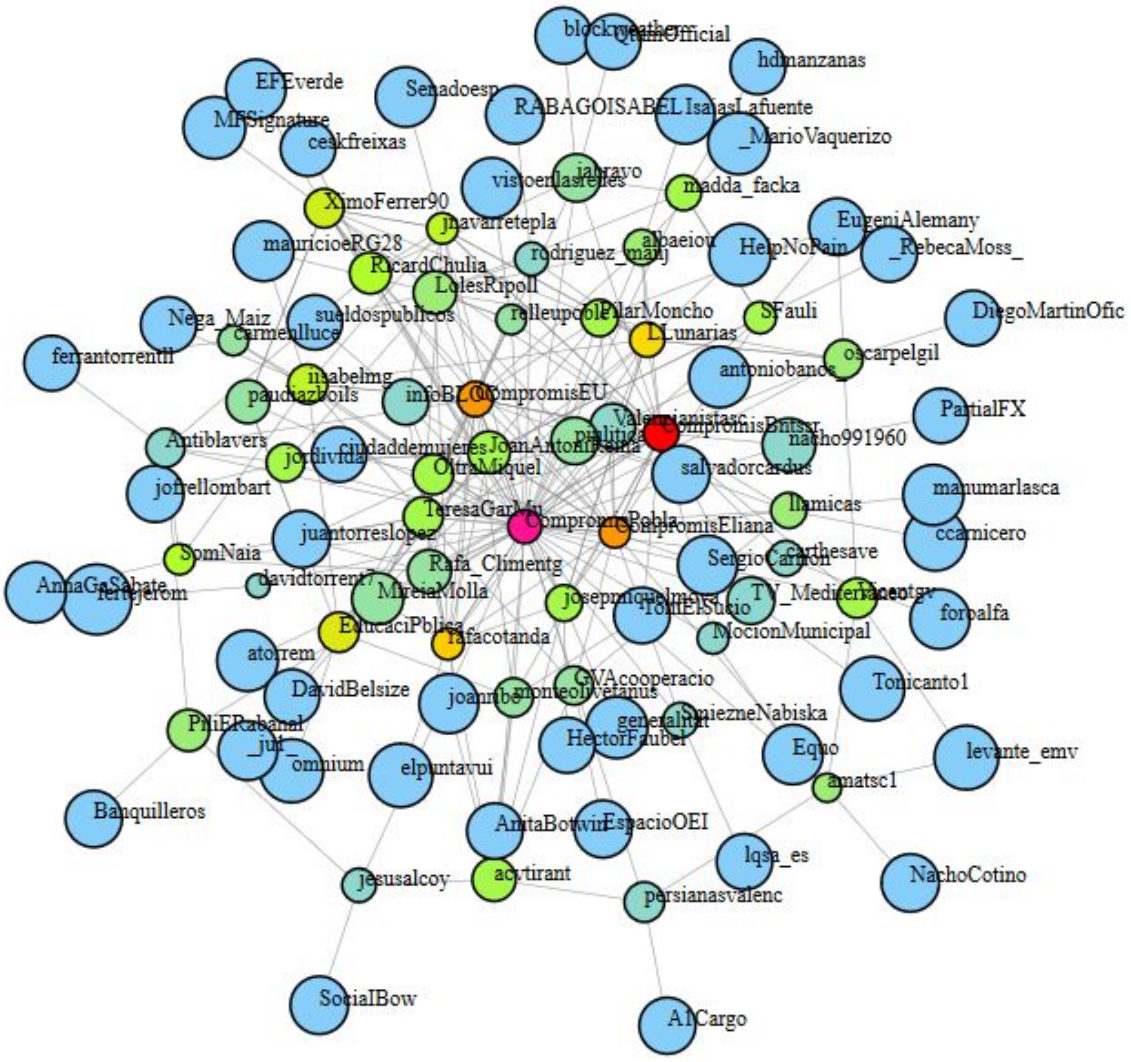

Figura 32. Los 50 usuarios con más seguidores y los usuarios que están relacionados con ellos

6.2.2 Análisis

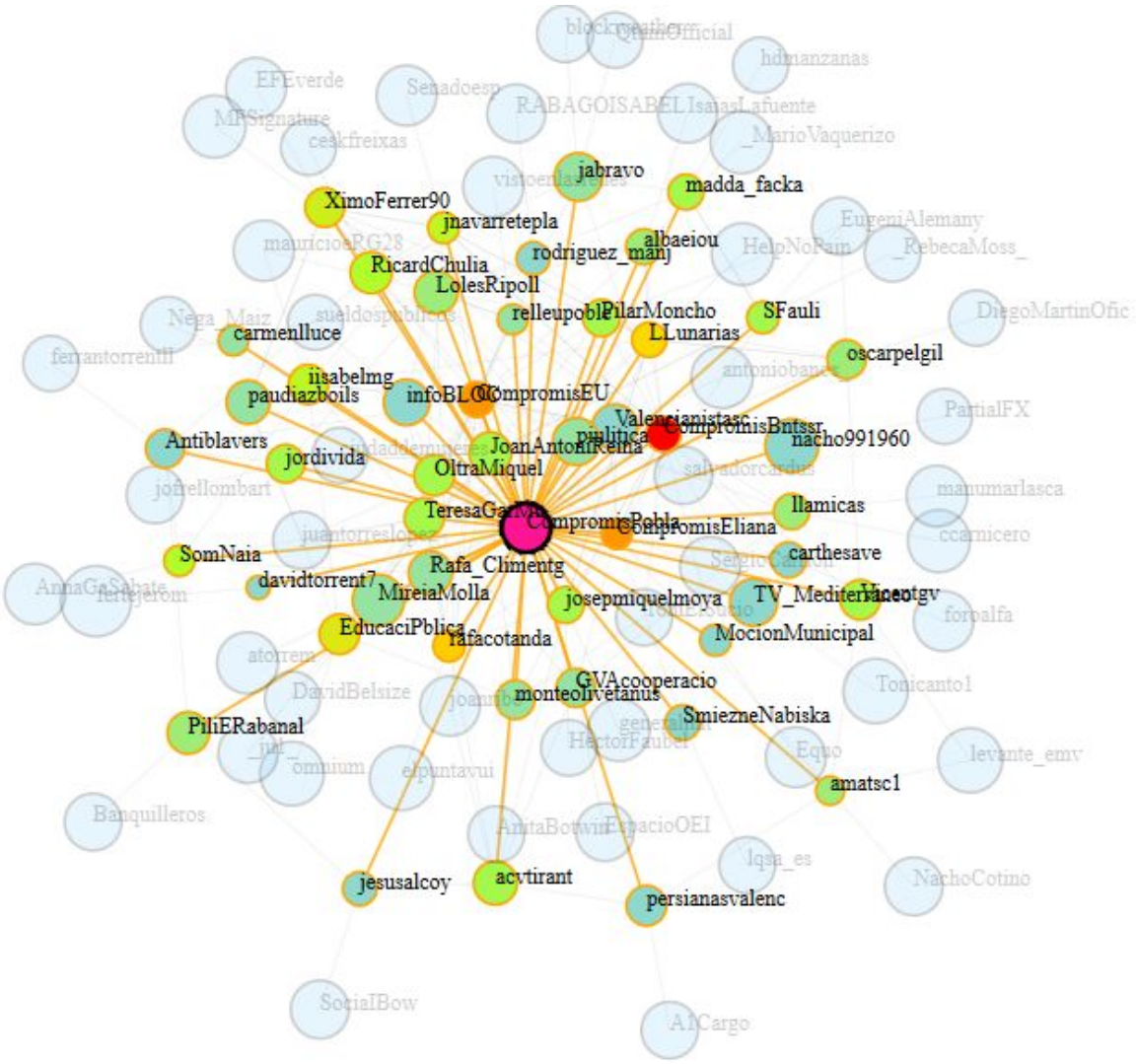

Figura 33. Entorno más cercano a *@CompromisPobla*. Sus vértices adyacentes

Lo primero fácilmente apreciable es que se trata de un grafo más cohesionado (Figura 33) que el caso de estudio de la Delegación de Alumnos de Informática. Esto quiere decir que los usuarios tienen más relaciones entre ellos, formando un entorno más cerrada. Los que tienen mayor centralidad son *@CompromisEliana* y *@CompromisBntssr*, que además siguen directamente a *@CompromisPobla*. Es destacable que ambos usuarios son agrupaciones locales del grupo político que se está estudiando y además de poblaciones relativamente cercanas (a menos de 30 kilómetros). En el entorno pueden encontrarse también personalidades u organizaciones políticas tanto de Comprimís (Joan Ribó, Lluna Àrias Cortinam o Bloc Nacionalista Valencià) como de fuera de esta organización (Anna Gabriel Sabate, Antonio Baños, Equo, Nega, Juan Torres López o *@_ju1_*).

Se pueden ver cuentas relacionadas con el gobierno valenciano (*@GVAcooperacio*) o el español (*@senadoesp*). También se encuentran algunos medios de comunicación: El Punt Avui (*@elpuntavui*), TV Mediterráneo (*@TV_Mediterraneo*) o EFE verde (*@EFEverde*).

La mayor parte de los usuarios del subgrafo corresponden a cuentas estrechamente vinculadas a Compromís (grupos locales o personalidades destacadas), a instituciones valencianas, medios de comunicación o usuarios famosos por su actividad política. Son estos últimos dos tipos de usuarios, medios y personalidades, los que normalmente no están directamente relacionados con la cuenta de estudio y a los que sería conveniente tratar de llegar a través de las cuentas que directamente siguen a *@CompromisPobla*.

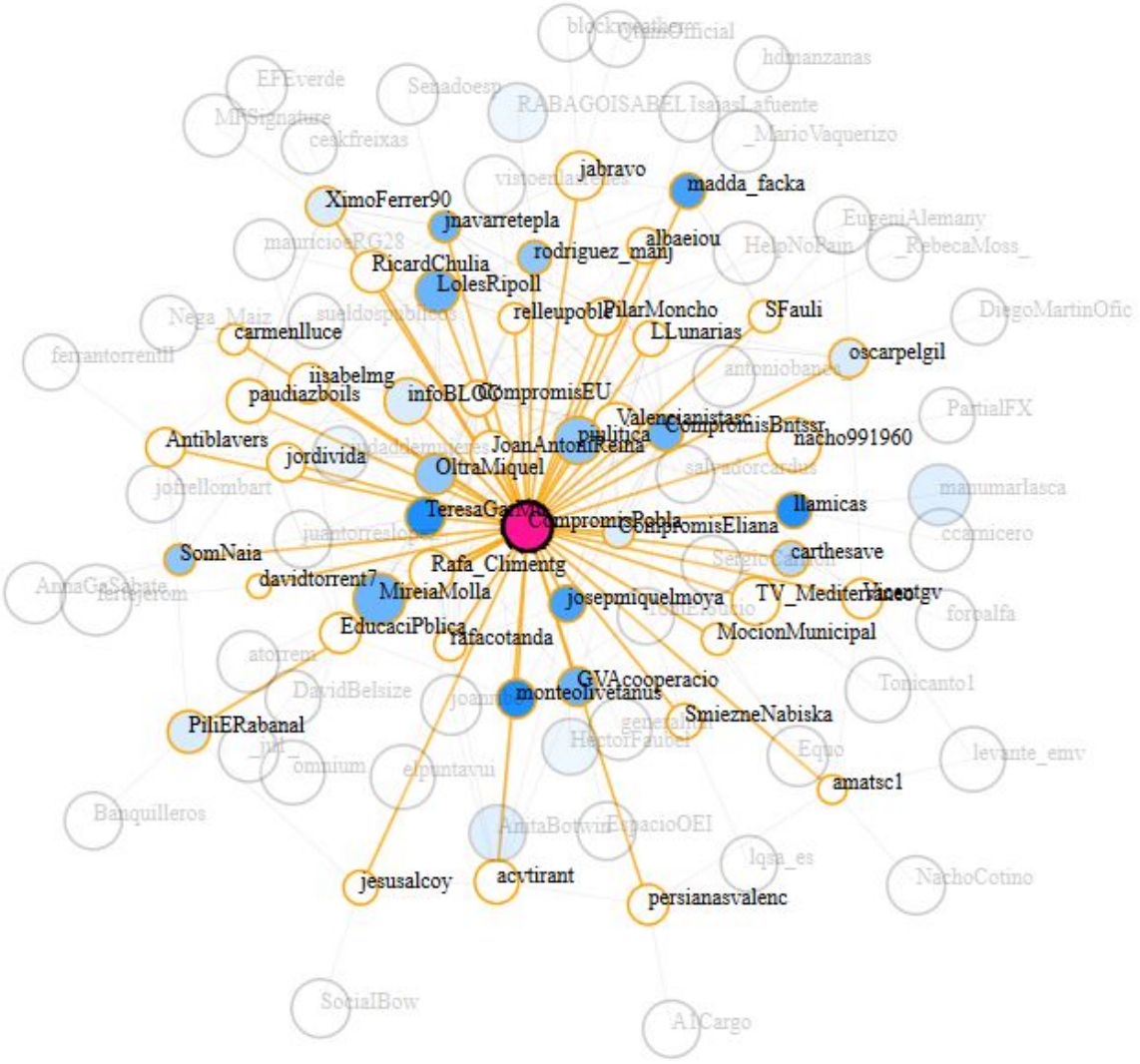

Figura 34. Usuarios del entorno de *@CompromisPobla* que han retuiteado alguno de sus tuits. Datos históricos

Sobre la actividad (Figura 34), es patente que el contenido de *@CompromisPobla* es interesante para su entorno (al menos en comparación con lo visto en el caso de la Delegación de Alumnos de la ETSINF). Como era de esperar, las cuentas que más retuitean a *@CompromisPobla* son las adyacentes a la cuenta de estudio. Hay otras cuentas importantes como *@ciudaddemujeres* con más de 35.000 seguidores que también han retuiteado a la cuenta, presumiblemente a través de *@OltraMiquel*, que es seguidor de *@CompromisPobla*.

6.3 Delegación de Valencia de la Asociación de Estudiantes de Videojuegos

Este caso de estudio es la delegación valenciana de la Asociación de Estudiantes de Videojuegos (*@AEVvalencia*), una organización que cofundé y presidí durante tres años. Por tanto, creo que es buena elección porque la conozco bien, tengo capacidad para razonar sobre ella y pertenece al ámbito estudiantil y tecnológico.

La Delegación de Valencia tiene una antigüedad aproximada de año y medio y es la parte de la organización que se dedica a hacer actividades en Valencia. Se desarrollan talleres, conferencias, maratones de desarrollo y fiestas y reuniones de desarrolladores. Su público objetivo son estudiantes que quieran dedicarse al desarrollo de videojuego o desarrolladores no muy destacados del sector.

6.2.1 Descripción

Los datos de la base de datos recogidos de la cuenta *@CompromisPobla* son los que figuran en la Tabla 3.

Nombre de la cuenta:	*@AEVvalencia*
Usuarios (vértices)	96.851
Relaciones (arcos)	143.581
Tweets	441

Tabla 3. Datos recogidos de la cuenta *@AEVvalencia*

La siguiente vista (Figura 35) la componen los 90 usuarios con más seguidores y que al mismo tiempo siguen a menos de 2000 (para evitar tener en cuenta usuarios que hagan *followback*). A partir de estos usuarios, los relacionados con ellos hasta componer el grafo.

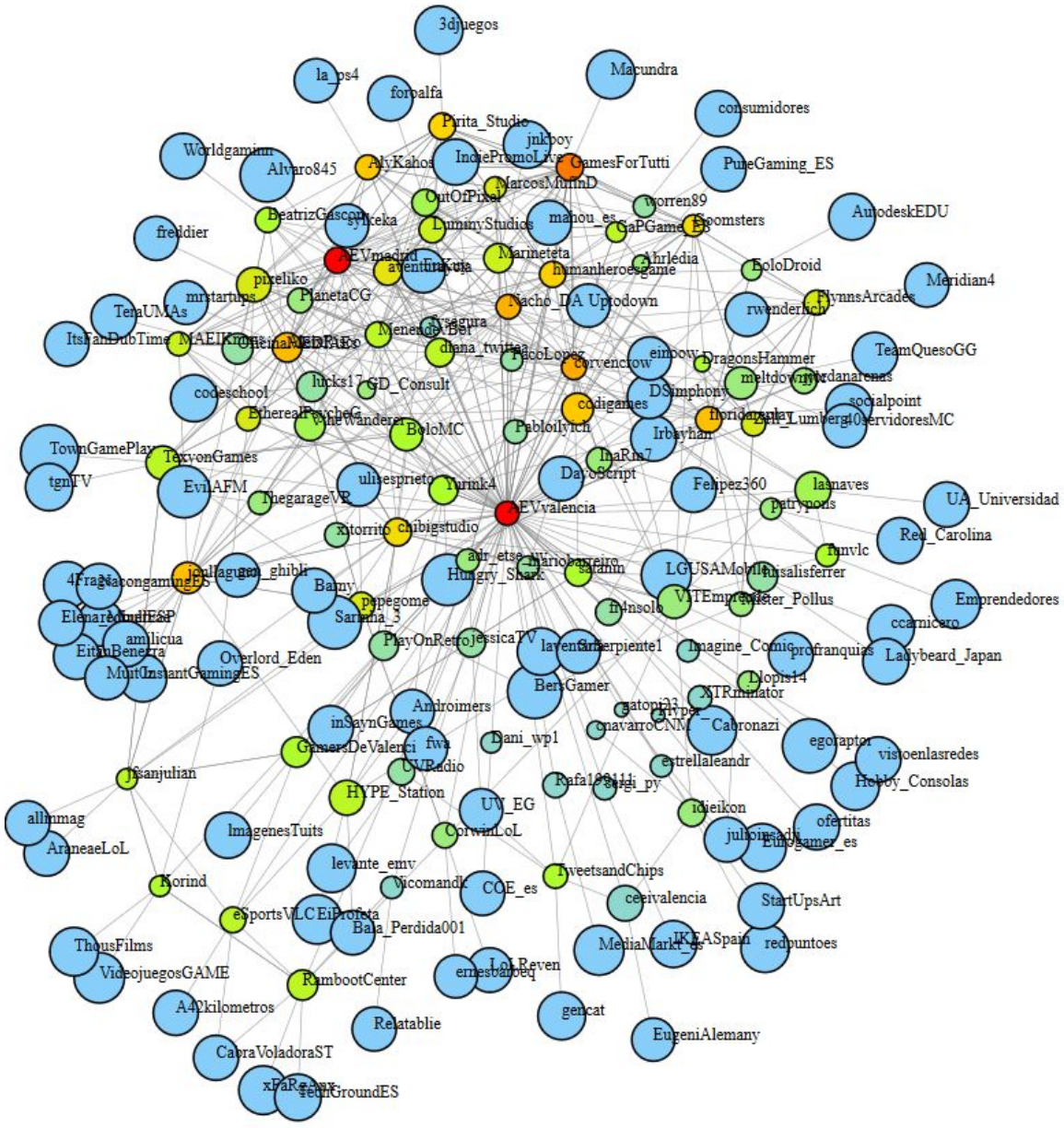

Figura 35. Los 90 usuarios con más seguidores y los usuarios que están relacionados con ellos

6.2.2 Análisis

Es fácilmente visible a través de la Figura 35 que en todo el grafo hay un grupo de vértices arriba que se encuentran muy cohesionados. Como se va a ver a continuación, este grupo lo componen principalmente personas ligadas al desarrollo de videojuegos (Figura 36).

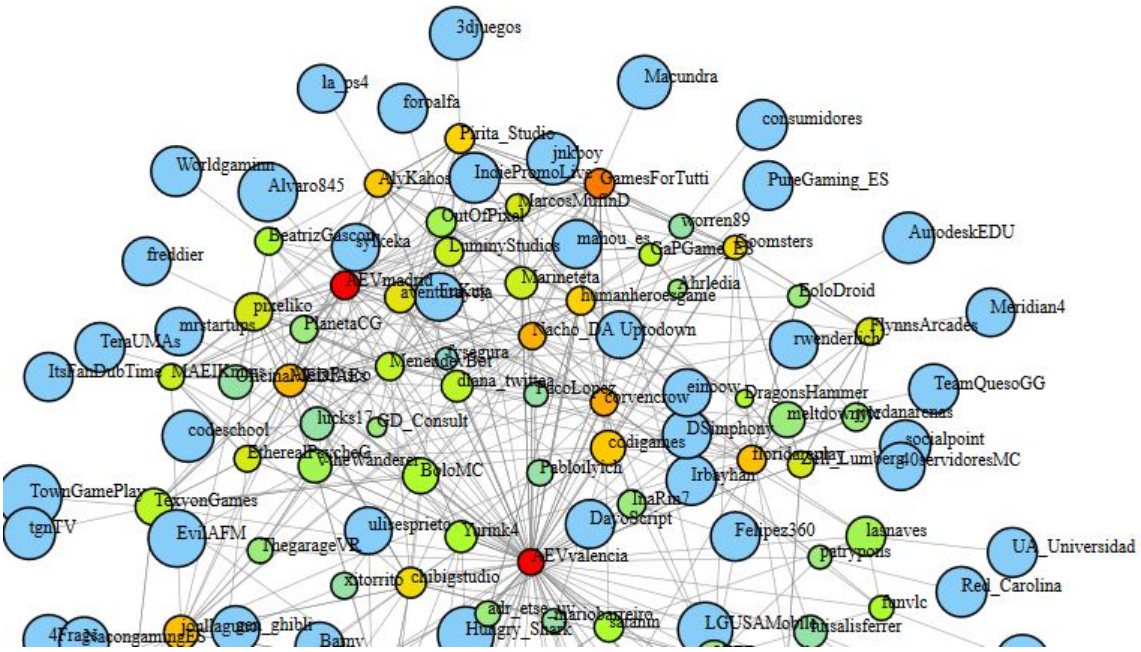

Figura 36. Grupo de usuarios relacionados con el desarrollo de videojuegos

Dentro de esta agrupación, es notable la presencia de vértices con una alta centralidad. Los que más tienen son la delegación madrileña de la asociación (*@AEVmadrid*); *@GamesForTutti*, un estudio de videojuegos de Valencia en el que trabajaron dos personas de la dirección de la asociación; y *@Nacho_DA* y *@covencrow*, dos personas que son o fueron parte de su directiva.

Es ese grupo podemos ver también una buena cantidad de profesionales del sector, sobre todo valencianos: *@AlyKahos*, *@MarcosMufinD*, *@Marineteta*, *@diana_twittea*, *@PacoLopez* o *@Pabloilyich*. También varios estudios de desarrollo de la autonomía: *@humanheroesgame*, *@codigames* o *@GapGames_ES*.

En el caso de *@AEVMadrid*, aunque por su posición dentro del grupo es evidente su ligazón con esta comunidad de desarrolladores, se distingue por tener seguidores más localizados en Madrid. Es el caso de *@BeatrizGascon, @pixeliko* o *@MAEIKmms*.

Tomando de nuevo una perspectiva general (Figura 35), pueden verse otras cuentas que, aunque tengan un vínculo menos estrecho con la cuenta de estudio, son muy importantes en el grafo por la cantidad de seguidores que tienen. Estas cuentas son, sobre todo, de medios de comunicación, sector empresarial y sector educativo.

Medios de comunicación los hay sobre todo especializados en la industria del videojuego: Eurogamer España (*@Eurogamer_es*), Hobby Consolas (*@Hobby_Consolas*), 3Djuegos (*@3djuegos*), Laps4 (*@la_ps4*) o PureGaming España (*@PureGaming_ES*). Y no sólo medios especialistas: también aparece Levante-EMV (*@levante_emv)*. Los *youtubers* populares, usuarios de YouTube que generan contenido para esa plataforma, también tienen presencia: *@Alvaro845* o *@TownGamePlay* son ejemplos. Este tipo de perfil es de los más interesantes porque tienen cientos de miles o millones de seguidores, por lo que planear cómo hacer que les llegue nuestro

contenido sería muy acertado. Para terminar el sector de la comunicación, destacar la presencia de Jon Llaguno (*@jonllaguno*), un profesional del márketing al que también sería muy importante llegar puesto que le siguen una buena cantidad de usuarios con muchos seguidores (Figura 37).

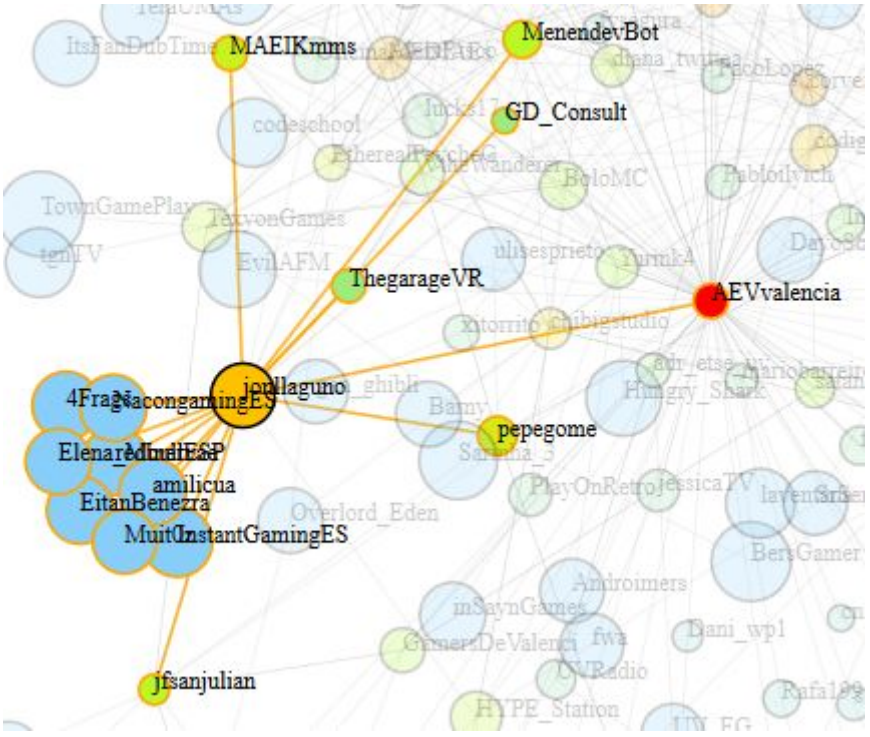

Figura 37. Usuario *@jonllaguno* focalizado

Respecto a cuentas del sector empresarial están, entre otras, *@VITEmprende*, *@Emprendedores* o *@socialpoint*. Relacionadas con la educación, se ven centros de educación superior como la *Universitat de València* (*@UV_EG*), Universidad de Alicante (*@UA_Universidad)* o Florida Replay *(@floridareplay*), que aglutina los estudios de videojuegos de la universidad privada Florida Universitària.

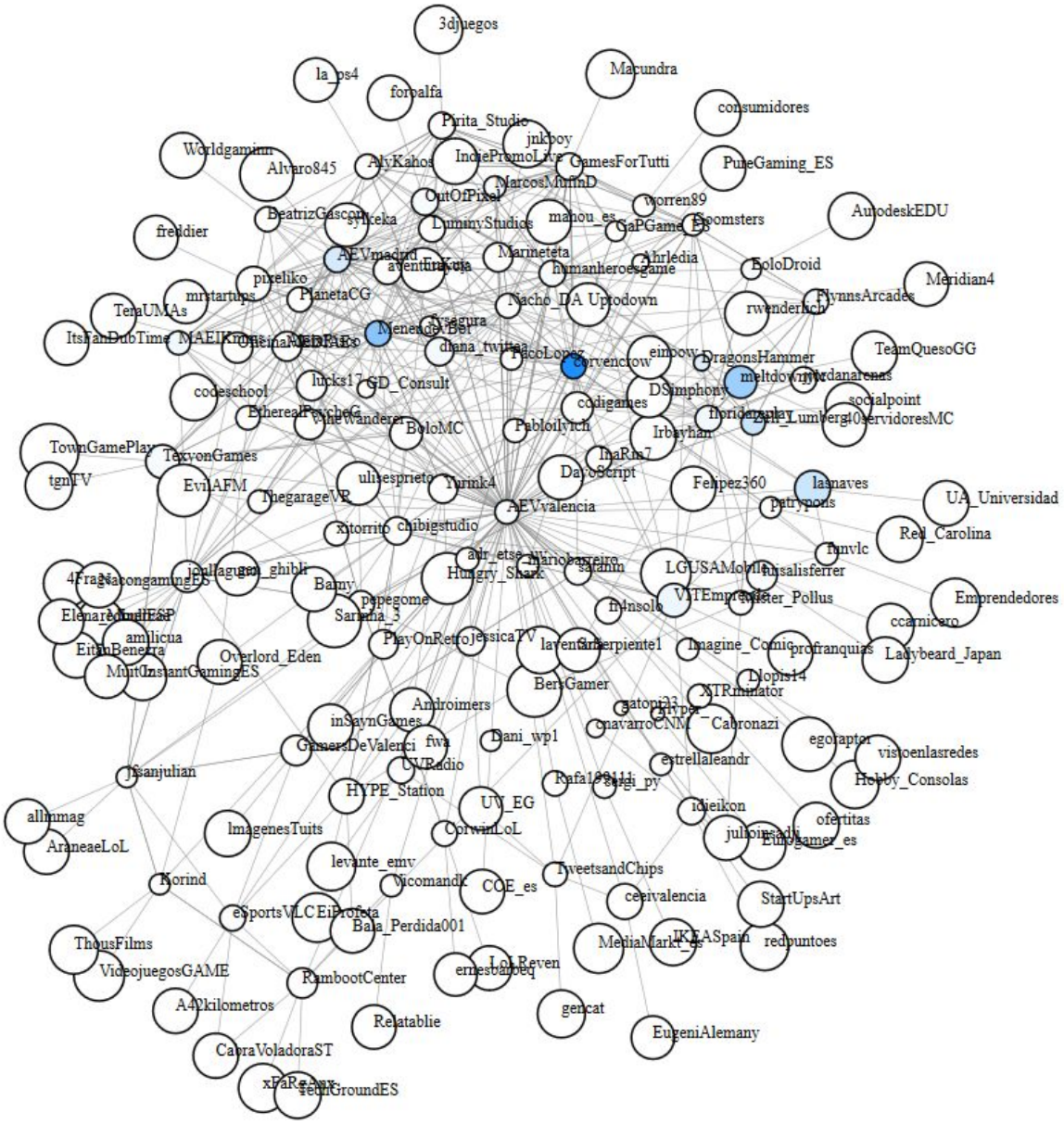

Figura 38. Usuarios del entorno de *@AEVvalencia* que han retuiteado alguno de sus tuits. Datos históricos

Analizando la actividad del entorno de la cuenta (Figura 38), volvemos a ver que, como pasaba en el caso de la Delegación de Alumnos de la ETSINF, son pocos los usuario que retuitean a *@AEVvalencia*. Principalmente lo hacen *@covercrown* y *@DragonsHammer*, ambos miembros de la organización, y *@AEVmadrid*, que como se ha indicado anteriormente es la parte de la asociación instalada en Madrid. En segundo lugar, una cuenta que retuitea automáticamente información relacionada con el sector del desarrollo de videojuegos. Por último, *@meltdownvlc* y *@lasnaves*, los lugares donde se realizan la casi totalidad de las actividades y por tanto, tienen especial interés en darlas a conocer.

7 CONCLUSIONES

En este capítulo primero se valora el trabajo realizado, se estudia si los objetivos se han cumplido y si finalmente la herramienta resulta de utilidad. Se discute también sobre las dificultades encontradas y cómo y en qué medida los estudios del Grado de Ingeniería Informática han sido útiles para la realización del proyecto. En último término, se valoran posibles mejoras futuras a la herramienta desarrollada.

7.1 Trabajo realizado

Los objetivos planteados se han cumplido. Se repasan a continuación:

1. Toda la funcionalidad de la aplicación está integrada en una plataforma web mediante la que el usuario interactúa. Esta plataforma, en buena medida gracias al framework utilizado, Django, permite ser extendida con facilidad, por lo que integrar en el futuro nuevas funcionalidades no debería ser difícil.

2. Se extrae toda la información necesaria de Twitter y el sistema puede obtener más información de forma sencilla, con ligeros cambios. Además, como se comenta en el apartado 5.1.4 Peticiones a la API de Twitter, la aplicación está preparada para aumentar su velocidad de extracción sólo obteniendo más *tokens* de acceso.

3. Todos los datos son almacenados en una base de datos adecuada para ello. La tecnología elegida, MongoDB, es idónea para datos como tuits y facilita mucho guardar la información obtenida de Twitter puesto que ambos trabajan con la misma estructura de datos: JSON la API de Twitter y BSON MongoDB, un derivado de JSON.

4. Se tratan los datos y se extrae información como se ha podido ver en capítulos anteriores. Por una parte, se visualiza un subgrafo concreto, el que es de nuestro interés. Por otra, se hacen algunos cálculos como el de la centralidad de grado para aportar significado a la visualización.

5. La visualización es útil debido a que se han cumplido los dos objetivos marcados a este respecto: 1) la información que se muestra se puede entender; 2) se aporta información que de otra forma es muy difícil obtener.

El enfoque decidido para el proyecto se ha demostrado, al menos, en buena medida acertado. Esto se desprende de lo visto en el capítulo 6. Casos de estudio. Por una parte, se ha comprobado cómo cada cuenta guarda una relación lógica con su entorno, ya fuera por temática, intereses o localización geográfica. Por otra parte, es fácil ampliar la información que se tenía sobre el contexto de una cuenta mediante la visualización: los nodos más importantes son destacados y su lugar dentro del subgrafo es coherente según las relaciones entre los usuarios. Finalmente, la interactividad ayuda mucho en el trabajo de análisis. Pese a que en un primer momento una de las vistas puede parecer confusa debido

a la cantidad de información que hay reflejada, es sencillo focalizar la atención en determinados nodos o abrir una nueva pestaña en el navegador y seguir explorando en su página de Twitter.

Es destacable que el proyecto no sólo sirve para estudiar una cuenta propia sino cualquiera. Por ejemplo, la de la competencia. En este sentido, es muy interesante poder conocer cuáles son los usuarios más importantes del entorno de un competidor o cuáles son los que más interactúan con él. Dentro de las muchas utilidades que ofrece esta posibilidad, una podría ser realizar un acercamiento a estos usuarios.

Utilizando la herramienta para mi propio interés, he topado con otro uso inesperado: recoger los datos de una red para conocer la comunidad que hay en torno a un determinado usuario. Esto me podría servir, por ejemplo, para saber cuáles son los actores clave de una determinada temática. Si estoy interesado en el periodismo de datos, puedo visualizar el entorno de la cuenta de Twitter @GuardianData y, puesto que The Guardian es uno de los periódicos más reputados en cuanto al trabajo con los datos, con toda seguridad los usuarios con una mayor centralidad serán importantes en este sector.

El potencial concreto de la herramienta es difícil de determinar. Es posible que otras personas de disciplinas menos técnicas y más relacionadas con la comunicación tengan enfoques distintos sobre cómo utilizarla o incluso sobre posibles ampliaciones. Será tras su uso prolongado cuando sus límites, carencias, puntos fuertes y posibilidades futuras queden bien marcados.

Los errores que se han cometido en la realización del proyecto son sobre todo los que se dan cuando se utilizan tecnologías por primera vez. La falta de experiencia dificulta la concreción del diseño antes de comenzar a programar, lo que deriva en un código poco optimizado, a veces difícil de leer o innecesariamente complejo.

Otro de los limitantes, que ya se tuvo en cuenta en los objetivos, se ha comprobado cierto: es inviable el uso de esta aplicación para usuarios con un gran número de seguidores. La extracción no es el mayor problema puesto que como se explica en el apartado 5.1.4 Peticiones a la API de Twitter, aumentar la velocidad de obtención de datos es sencillo. El mayor problema radica en realizar una visualización interactiva de un grafo con un número elevado de nodos puesto que sencillamente un ordenador doméstico potente se queda muy lejos de disponer de la suficiente memoria.

Para la creación del proyecto han sido útiles, en mayor o menor medida, varias de las asignaturas del grado. No por haber realizado prácticas con estas tecnologías, porque como he dicho son nuevas para mí, sino por conocimientos que considero que son más importantes que transmita un grado universitario, como conceptos, abstracciones y en definitiva bases de conocimiento donde posteriormente se enmarcan las tecnologías concretas, todas ellas al fin y al cabo caducas en el futuro. Las asignaturas cuyos conocimientos he podido aplicar de forma clara son las relacionadas con el diseño de software (“Diseño de software”), las estructuras de datos (“Estructuras de datos y algoritmos”), las redes e internet (“Redes de computadores” y “Tecnologías de sistemas de

información en la red”), los grafos (“Grafos. Modelos y aplicaciones”) y evidentemente las de programación.

7.2 Posibles mejoras

Las posibles mejoras son muchas. Se repasan aquellas que se centran en cuestiones directamente tratadas en esta memoria.

El actual sistema de extracción de datos puede mejorarse en cuanto a velocidad sólo introduciendo más *tokens* de acceso, pero hasta determinado punto (tal y como se puntualiza en el apartado 5.1.4 Peticiones a la API de Twitter). El sistema funciona alternando distintos *tokens* mientras otros han agotado su tiempo de acceso, de forma que con el suficiente número de *tokens*, en un determinado momento el tiempo de acceso a la API será mayor que el tiempo en el que se le restringe. En ese momento, se estaría perdiendo potencia de extracción. La solución es realizar un sistema de obtención de datos más complejo en el que haya un reparto inteligente de tareas y la distribución del trabajo se paralelice.

Desde luego también hubiera sido interesante, y se proyecta como un cambio futuro, exprimir más los datos para obtener más información: por ejemplo, agrupando los nodos según relación estructural dentro del grafo (ya sea por la relación de seguimiento o por otras características, de manera similar a la Figura 39); o bien, realizar más cálculos sobre los nodos en base a sus centralidades.

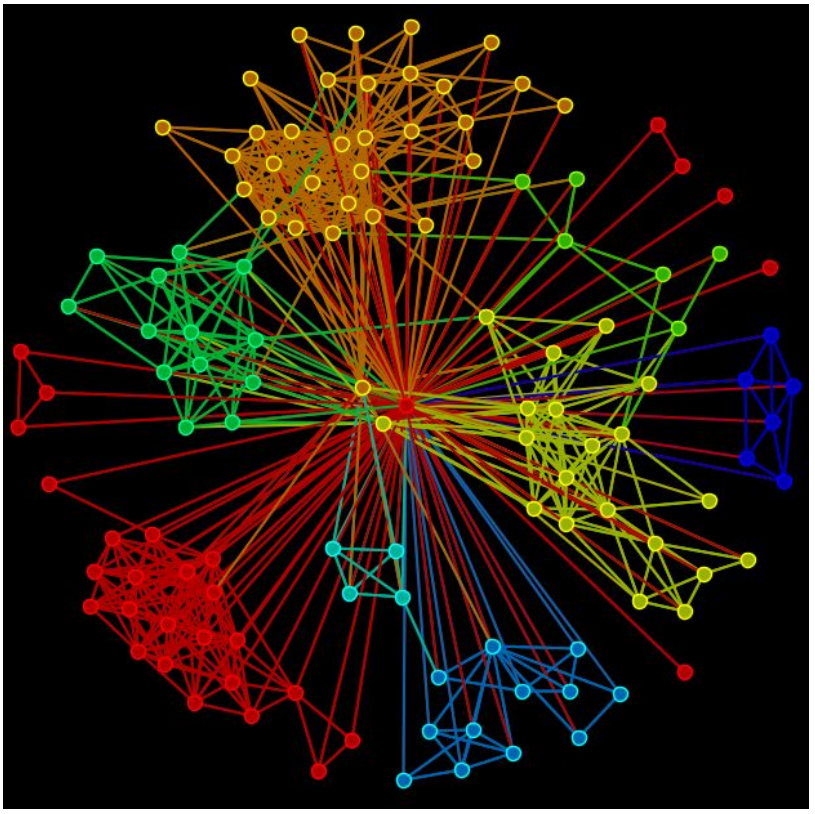

Figura 39. Grafo del entorno de un usuario con vértices agrupados

La imposibilidad de mostrar un grafo con miles de nodos en una visualización interactiva como se ha comentado en el apartado anterior, 7.1 Trabajo realizado, puede dar lugar a otras aproximaciones al problema. Una podría ser la planteada en el capítulo 4. Enfoque: generar una imagen estática generada en el servidor que represente el conjunto del grafo bajo los parámetros que el usuario de la aplicación introduzca por la interfaz. Otra más interesante pero bastante más compleja consiste en aumentar el nivel de detalle según dónde y a qué altura se está enfocando, similar al aumento de resolución que se produce cuando se hace zoom en un servicio de mapas, como podría ser Google Maps.

Con un sistema que permitiera la visualización de grafos mayores, sería muy interesante tener la posibilidad de juntar dos o más grafos de usuarios diferentes en uno único, teniendo varios usuarios de estudio en la misma escena. Aunque en un primer momento pueda parecer que el resultado serían tres grafos con poco que ver entre sí, si se trata de datos extraídos de usuarios similares (por temática, por ejemplo), habría muchos usuarios en común y se descubrirían nuevas relaciones.

Por último, si bien MongoDB es un buen sistema de base de datos para almacenar tuits, hay alternativas mejores para guardar grafos, en este caso usuarios (vértices) y sus relaciones (arcos). Sería interesante estudiar el uso de un sistema de base de datos orientado a grafos puesto que es una alternativa pensada para este tipo de estructuras.

El software desarrollado se encuentra disponible en el siguiente repositorio:

HTTPS: https://NachoDA@bitbucket.org/NachoDA/twitteranalyzer.git

SSH: git@bitbucket.org:NachoDA/twitteranalyzer.git

8 BIBLIOGRAFÍA

- Scott, J. (2017). *Social Networks Analysis (4th ed.)*. London, UK. SAGE Publications Ltd.

- Wasserman, S. & Faust, K. (1994). *Social Network Analysis. Methods and Applications*. New York, NY, USA. Cambridge University Press.

- Parmelee, J. H. & Bichard, S. L. (2012). *Politics and the Twitter Revolution. How Tweets Influence the Relationship between Political Leaders and the Public*. Plymouth, UK. Lexington Books.

- Riquelme, F. & González-Cantergiani, P. (2015). *Measuring user influence on Twitter. A survey*. Universidad de Santiago, Chile. doi: 10.1016/j.ipm.2016.04.003

- Gil, H. & Guilleumas, R. M. (2017). Redes de comunicación del movimiento 15M en Twitter. *Revista Hispana para el Análisis de Redes Sociales*, 28(1). 136-146. doi: https://doi.org/10.5565/rev/redes.670

- Torres, L. C. (2009), Ciberprotestas y consecuencias políticas: Reflexiones sobre el caso de internet necesario en México. *Razón y palabra*, 70. Obtenido de: https://dialnet.unirioja.es/servlet/articulo?codigo=3104448

- M. Cha, H. Haddadi, F. Benevenuto, and K. Gummadi. Measuring user influence in Twitter: The million follower fallacy. *In Proceedings of the 4th International Conference on Weblogs and Social Media (ICWSM'10)*, Washington DC, USA, 2010.

- Nguyen, T. T., Hui P., Maxwell, F. H., Terveen, L., Konstan, J. A. (2014), Exploring the filter bubble: the effect of using recommender systems on content diversity. *WWW '14 Proceedings of the 23rd international conference on World wide web*. 677-686. doi: 10.1145/2566486.2568012