

Análisis Cibernético y Visual de Twitter

Zhangxian Zheng¹, José Luis Alonso Berrocal¹ y Luis Carlos García de Figuerola Paniagua¹

¹Departamento de Informática y Automática - Facultad de Traducción y Documentación, Universidad de Salamanca. Calle de Francisco Vitoria 6-16. 37008 Salamanca, España
{r.carlos, berrocal, figue}@usal.es

Resumen Este trabajo aborda el tema de análisis cibernético y visual de *Twitter*. En él se describe como resolver la necesidad de recoger el perfil, los seguidores y seguidos de un usuario de *Twitter* vía su propia API y desarrollar una aplicación de crawler en *Python* utilizando la librería *Python-Twitter*, con el objetivo de realizar un análisis y visualización de la red que construyen los usuarios de Twitter mediante una herramienta desarrollada.

Keywords: Twitter, Cibermetría, Crawler, Analítica Visual

1. Introducción

En la vida social actual, todo el mundo tiene sus propias relaciones personales; este tipo de relaciones existen entre las personas que constituyen la red social[1] de la vida real. En los últimos años, con la evolución de Internet, han comenzado a aparecer sitios web promocionando las redes de círculos de amigos en línea. El término se empleaba para describir las relaciones en las comunidades virtuales y se hizo popular en 2003 con la llegada de sitios tal como *MySpace*, y continúan avanzando en Internet a pasos agigantados, especialmente dentro de lo que se ha denominado Web 2.0.

Twitter [2] es uno de los sitios más relevantes hoy en día; en la actualidad, tiene más de 200 millones de usuarios activos y se publican más de 400 millones de mensajes de texto cada día [3]. Las informaciones que contiene en su sitio web son muy enriquecidas, significativas e importantes, por lo tanto nos interesa recoger datos desde este sitio web y hacer análisis y visualización para descubrir nuevos conocimientos.

2. Estado del Arte

2.1. Métricas de Análisis de Redes Sociales

Las redes sociales se pueden representar mediante grafos, y se utilizan técnicas de la teoría de grafos para estudiar la estructura de éstas. Sin embargo, existen diferencias en la forma en que se aplican los distintos conceptos de la

teoría de grafos sobre el Análisis de Redes Sociales (ARS). Las propiedades que presentan las redes sociales se pueden representar en base a su forma, distribución y similitud entre los conjuntos de nodos y relaciones que existen en la red.

- **Cohesión:** es una fuerza de atracción entre los individuos, que puede ser medida por el número de intercambios entre dos actores, el promedio de las distancias geodésicas, el tamaño de la red, etc.
- **Densidad:** la densidad de la red nos muestra la alta y baja conectividad de ésta, expresada en porcentaje como el cociente entre el número de relaciones activas entre las posibles. Se mide la proporción de enlaces que existen entre las relaciones posibles de una red en concreto.
- **Diámetro de la red:** el diámetro es la distancia de grafo más larga entre dos nodos cualesquiera en la red, el mayor valor de los geodésicos de la red. Representa el tamaño del grafo y permite saber lo grande que es.
- **Centralidad de Grado:** se define al grado de un nodo como el número de enlaces que posee, es decir, el número de relaciones que tiene el nodo con los otros nodos. Para los grafos dirigidos el grado del nodo suele ser dividido en dos formas: el grado de entrada y el grado de salida.
- **Centralidad de Intermediación:** este tipo de centralidad basada en la frecuencia en la que un nodo aparece en el geodésico entre dos nodos, es la medida de las veces en que se presenta entre un nodo con trayectoria mínima.
- **Centralidad de Cercanía:** la centralidad de cercanía es la distancia media desde un nodo inicial a todos los demás nodos de la red, mide los pasos requeridos para acceder a cada vértice desde un vértice dado.[4]

2.2. *Twitter* y *Twitter API*

El funcionamiento de *Twitter* es muy simple: el usuario envía mensajes de texto plano de corta longitud, con un máximo de 140 caracteres, llamados “tweets”, que se muestran en la página principal del usuario y que pueden ser leídos por cualquiera que tenga acceso a su página. Los usuarios pueden suscribirse a los “tweets” de otros usuarios, a esto se le llama “seguir” y a los usuarios abonados se les llama “seguidores”. A los usuarios a los que se sigue se les llama “seguidos”. [5]

Uno de los mayores éxitos de *Twitter* ha sido proporcionar una API abierta y funcional desde sus inicios, lo que permite que otras personas extiendan las características que *Twitter* ofrece y desarrollen aplicaciones para enviar y compartir nuestros tweets sin estar obligados a utilizar el sitio oficial. Resulta interesante como han surgido las posibilidades de permitir apreciar y analizar los datos del servicio en diferentes formas. Para recoger los perfiles de usuario y obtener los seguidos y los seguidores para crear un grafo de relaciones, son útiles los siguientes métodos del API[6]:

- **GET users/show:** Devuelve una información amplia de un solo usuario determinado, especificado por el ID o el nombre de la pantalla (*screen names*) según el parámetro id requerido.
- **GET followers/ids:** Devuelve un conjunto de IDs de cada usuario que está siguiendo al usuario específico, es decir, todos IDs de los seguidores/*followers* de un usuario específico.
- **GET friends/ids:** Este método es similar al anterior, en lugar de devolver los id de los seguidores/*followers*, devuelve los de los seguidos/*friends*.
- **GET users/lookup:** Este método es casi igual que GET users/show, la única diferencia es que este método devuelve información de hasta 100 usuarios por una petición.

2.3. Web Crawler

Un *Web Crawler* es un sistema capaz de recorrer, de forma planificada, un conjunto de recursos. Uno de los usos más frecuentes de un *Web crawler* es un programa que inspecciona las páginas del *World Wide Web* de forma metódica y automatizada, creando una copia de todas las páginas web visitadas para su procesamiento posterior por un motor de búsqueda que indexa las páginas proporcionando un sistema de búsquedas rápido.[7]

Hay varias formas de hacer un recorrido de *Web Crawler* en la web, aunque básicamente existen tres[8]: recorrido en anchura, recorrido en profundidad y el mejor posible. Las dos primeras son bien conocidas y dependen de que dicha lista se organice como una pila o como una cola. En el primer caso, el entorno inmediato y más cercano a una página es explorado primero, resulta útil cuando se buscan exploraciones exhaustivas de un dominio, aun cuando éstas se limiten a una determinada profundidad solamente.

2.4. Herramientas de Análisis y Visualización

La visualización de las redes sociales sirve como método para descubrir propiedades en las redes, aunque tiene menos peso teórico en el análisis, pero tiene la ventaja de alimentar rápidamente la intuición del investigador [9]. En la actualidad existen diversas herramientas desarrolladas para el análisis y visualización de redes, tales como *Pajek*, *NodeXL* o *UCINET*. *Gephi* [10] es una de las más relevantes.

Gephi es una herramienta de código abierto multiplataforma bajo la licencia GNU GPL 3 y gratuita. Es una de las herramientas más completas para manejar todo tipo de datos y gráficos y con un desempeño bastante considerable, incluso al manejar y procesar datos sumamente grandes, ofreciendo los cálculos de ARS tales como Grado, Diámetro, Centralidad y Modularidad. También está disponible una variedad de disposiciones para la visualización de grafos.

3. Desarrollo de la Aplicación

El primer paso para el desarrollo del trabajo es la realización de un *crawler* de información para *Twitter*. Este *crawler* debe ser capaz de obtener y almacenar información sobre *Twitter* de manera estructurada, continua y además, debe ser tolerante a fallos. El *crawler* de *Twitter* está íntegramente escrito en *Python* [11] y se apoya en la librería *Python Twitter Tools* [12] para realizar la autenticación y las llamadas a *Twitter API* [13].

3.1. Análisis de los Datos a Extraer

Los principales datos que extraeremos desde el *Twitter* de la Usal (Universidad de Salamanca) son los perfiles de todos seguidores de él y un grafo de la red social. Desde la nueva versión 1.1 de *Twitter API*, los resultados devueltos por el servidor de *Twitter* se encuentran únicamente en formato JSON. La estructura de datos es sencilla, cada clave tiene su nombre entre comillas y con su correspondiente valor después del caracter “:”, y los diferentes pares “clave”:valor se separan por comas, como la estructura de datos de diccionario de *Python*, facilitando el tratamiento previo y almacenamiento de los datos recogidos.

Los perfiles de usuarios de *Twitter* contienen las siguientes informaciones que nos pueden interesar extraer para hacer algún análisis estadístico: localización, huso horario, idiomas, la cantidad de tweets, seguidores, seguidos y fecha de creación de cuenta.

3.2. Arquitectura, Diseño y Funciones

Hay tres componentes importantes dentro del *crawler*: Autenticación, *Crawler* y Tratamiento previo. La Autenticación es el componente imprescindible del *crawler*, debido a que desde el retiro del antiguo *Twitter API v1*, cualquier petición enviada al servidor de *Twitter* requiere la autenticación del usuario o la aplicación. El *Crawler* es el componente principal, se encarga de las funciones principales, como mandar peticiones, recuperarse del límite de llamadas al API o error de red, extraer informaciones de interés desde los perfiles, etc. El Tratamiento previo es el componente encargado del pre-proceso y almacenamiento de datos en ficheros para su uso posterior, como importarlo en *Gephi* para hacer análisis y visualización.

El *Twitter Crawler* se compone de varias funciones y una clase de interfaz gráfica. A continuación se explica en detalle cada una de las funciones:

- **Autenticación:** para obtener el acceso completo a toda la *Twitter API* es necesario obtener la autenticación vía *OAuth*. Para ello, necesitamos crear una aplicación en *Twitter Apps* (<https://dev.twitter.com/apps>), y obtenemos el nombre de la aplicación (*App Name*), *Consumer Key* y *Consumer Secret* para el proceso de autenticación.

- **Mandar peticiones robustas [14]:** cuando mandamos peticiones al servidor de Twitter, los resultados que nos devuelve no siempre son los que esperamos, muchas veces el servidor de Twitter nos devuelve código error de HTTP para expresar si ha ocurrido algo. El *Twitter crawler* debe saber como responder a los códigos de error devuelto, por eso utilizamos la función *make_twitter_request* para manejar los errores.
- **Crawl por seguidores o seguidos:** para crear un grafo con los seguidores o seguidos de uno o varios usuarios de *Twitter*, necesitamos hacer al *crawler* caminar desde los primeros usuarios, recoger y seguir por sus seguidores o seguidos, hasta un nivel de profundidad deseado. Es necesario establecer alguna regla al *crawler*. En el *Twitter crawler v1* implementamos el nivel de profundidad de *crawl* y el número mínimo y máximo de seguidores de un usuario para una mejor recogida de datos.
- **Guardar resultados:** es necesario de guardar los resultados obtenemos mediante el *crawl* para usos posteriores. En lugar de utilizar una base de datos, *Python* proporciona el módulo *pickle* para almacenar sus objetos en ficheros, con la ventaja de funcionar bastante rápido y de recuperar la información sin problemas.
- **Exportación en CSV;** para importar datos a una herramienta de análisis o visualización, hay que proporcionar el dato en el formato adecuado. Por ejemplo en el siguiente apartado planteamos utilizar la herramienta *Gephi*, y necesitamos el dato en formato CSV: se escriben dos ids de usuarios separados por una coma, que significa que el usuario anterior está siguiendo al posterior.
- **Recoger perfiles:** aparte de los seguidores y seguidos, también nos interesa recoger el perfil de los usuarios, incluyendo su descripción, idiomas, localización, fecha de creación en *Twitter*, etc. Podemos utilizar el método *GET users/lookup* para lograr hasta 100 perfiles en una petición.
- **Obtener nombres:** El resultado de *crawl* proporciona los seguidores y seguidos mediante los ids de los usuarios. Debido a que necesitamos crear el grafo con las etiquetas en nombres, hacemos la transformación a través del método para recoger perfiles.

4. Análisis Cíbermétrico y Visualización

4.1. Descripción de Datos

- **Perfiles de todos seguidores de @usal:** El twitter de la Universidad de Salamanca tiene 14943 seguidores. Hemos recogido los perfiles de cada uno de ellos, estos datos están almacenado en formato Python pickle.

- **Red social en Twitter a partir de @usal:** Es una red de 3 niveles de profundidad desde el usuario la Universidad de Salamanca. Cada usuario que forma parte de esta red debe tener al menos 5 seguidores o seguidos, y pueden tener hasta 20 seguidores o seguidos en esta red, en resumen, esta red esta constituida por 37684 usuario y 47404 relaciones entre ellos.
- **Perfiles de todos usuarios de la red @usal:** En este dato se incluyen los perfiles de todos usuarios que aparecen en el dato de Red social en *Twitter* a partir de @usal.

4.2. Análisis Estadístico

El twitter de la Universidad de Salamanca tiene aproximadamente 15000 seguidores, que provinientes de diferentes zonas del mundo y tienen diversas características. También ocurre en el grupo de usuarios que aparecen en la red de @usal. Nos interesa a hacer un análisis estadístico sobre ellos para ver como están relacionados los dos grupos de usuarios, qué diferencia y similitud tienen. A continuación detallamos el resultado que obtenemos atendiendo a diferentes aspectos

Idiomas. En el perfil de usuario de *Twitter*, cada usuario tiene su idioma codificado con dos letras de abreviación, por ejemplo “es” significa español, “en” es inglés, etc. Como la Universidad de Salamanca es una universidad española, el resultado es como esperábamos: la mayoría hablan en español, y el inglés ocupa el segundo lugar.

Idiomas	Seguidores @usal	Red @usal
Español	90,24 %	79,00 %
Inglés	6,37 %	16,10 %
Dialectos en España	0,38 %	0,97 %
Idiomas europeos	2,79 %	3,03 %
Idiomas orientales	0,20 %	0,34 %

Localización: Para este dato de perfiles es un poco difícil de hacer un análisis estadístico, porque el contenido de este campo no está normalizado, es libre de editar por los usuarios. Lo que podemos hacer es sacar los términos que aparecen con más frecuencia en las localizaciones de perfiles.

Términos	Frecuencia en Seguidores @usal	Términos	Frecuencia en Seguidores @usal
España	1921	Sevilla	75
Salamanca	3104	Valencia	83
Valladolid	206	Cáceres	100
Ávila	150	Argentina	116
León	164	México	108
Madrid	582	Venezuela	103
Barcelona	159	Colombia	102

Huso horario. A diferencia de la localización, el huso horario en cada perfil de usuario está normalizado. Podemos obtener un resultado sobre esto para generalizar el estado geográfico del grupo de usuarios. Sin embargo, como no todos los usuarios rellenan esta parte del perfil, solo obtenemos las dos terceras partes de ellos.

Huso horario	Seguidores @usal	Red @usal	Huso horario	Seguidores @usal	Red @usal
Amsterdam	8,26 %	5,81 %	Hawaii	1,55 %	1,46 %
Athens	8,71 %	7,13 %	London	1,22 %	2,01 %
Caracas	0,86 %	1,21 %	Madrid	24,99 %	18,59 %
Central Time (US)	1,30 %	3,38 %	Mexico City	0,82 %	1,83 %
Eastern Time (US)	0,95 %	3,96 %	Otros	8,50 %	13,90 %
Greenland	9,26 %	4,59 %	Unknown	33,58 %	36,13 %

Número de Tweets: En el perfil de usuario también viene la información sobre cuántos tweets ha publicado el usuario, en el que podemos observar la actividad de éste.

Número	Seguidores @usal	Red @usal
0	3,14 %	2,42 %
1-10	7,66 %	6,72 %
11-100	18,29 %	17,26 %
101-1.000	36,22 %	32,40 %
1.001-10.000	31,62 %	32,71 %
10.001-100.000	3,04 %	8,29 %
100.000+	0,02 %	0,21 %

Número de seguidores y seguidos: Los números de seguidores o seguidos que tiene un usuario también pueden reflejar su actividad de uso en *Twitter*.

Número de seguidores	Seguidores @usal	Red @usal	Número de seguidores	Seguidores @usal	Red @usal
0	0,63 %	0,54 %	10.001-100.000	0,62 %	7,49 %
1-10	6,72 %	5,70 %	100.001-1.000.000	0,07 %	2,82 %
11-100	37,43 %	25,75 %	1.000.000+	0,01 %	0,66 %
101-1.000	50,08 %	42,13 %	10.000.000+	-	0,06 %
1.001-10.000	4,44 %	14,85 %	-	-	-

Número de seguidos	Seguidores @usal	Red @usal	Número de seguidos	Seguidores @usal	Red @usal
0	0	0,41 %	1.001-10.000	12,07 %	17,68 %
1-10	1,98 %	2,15 %	10.001-100.000	0,18 %	2,38 %
11-100	20,20 %	18,18 %	100.000+	0,01 %	0,33 %
101-1.000	65,56 %	58,86 %	1.000.000+	-	1

Fecha de creación: Por último, el resultado sobre la fecha de creación de los usuarios en Twitter muestra que la mayoría seguidores de @usal se han registrado entre el año 2010 y 2012, y los de la red @usal se registran más uniformemente.

Año	Seguidores @usal	Red @usal	Año	Seguidores @usal	Red @usal
2006	-	0,04 %	2010	17,23 %	15,94 %
2007	0,23 %	0,80 %	2011	38,49 %	25,11 %
2008	0,87 %	2,00 %	2012	26,92 %	24,26 %
2009	7,26 %	10,38 %	2013	9,00 %	21,47 %

4.3. Análisis de Redes Sociales

Para encontrar algunos nodos o patrones más significativos en la red @usal, necesitamos calcular y conseguir las métricas del Análisis de Redes Sociales de esta red. En la herramienta *Gephi* nos facilitan diversos cálculos de este aspecto. A continuación presentamos los resultados obtenidos.

Grado. El grado medio de la red @usal es 1,258. En la siguiente tabla se presentan los 12 primeros nodos que tienen el grado más alto en esta red y sus descripciones. Los usuarios de esta tabla son muy diversos, @usal ocupa el duodécimo puesto en la tabla. Además, se presentan los usuarios que tienen el grado de entrada y salida más alto en esta red.

Tabla 1. Usuarios con el grado más alto

	Nombre	Descripción	Grado
1.	leonactualidad	Diario digital de la localidad de Leon.	67
2.	SalamancArlequinada	Por un baloncesto de calidad en Salamanca	66
3.	Educación INTEF	Instituto Nacional de Tecnologías Educativas	62
4.	Farmacia EL PUENTE	Un lema 365 días de salud Evolucionando	61
5.	Zamora Perfecta	Zamora. Desconocida. Perfecta.	60
6.
..
12	Usal	La Universidad de Salamanca es..	54

Tabla 2. Usuarios con grado de entrada más alto

	Nombre	Descripción	Grado
1.	National Geographic	Desde 1888, hemos estado viajando por Tierra	36
2.	Usal	La Universidad de Salamanca es una..	34
3.	Policía Nacional	Cuerpo Nacional de Policía de España	33
4.	ElinformerdelaUSAL	Envíale tu mensaje al Informer y Él se encargará	32
5.	EL PAIS	Las noticias globales más relevantes y la última	30

Tabla 3. Usuarios con grado de salida más alto

	Nombre	Descripción	Grado
1.	leonactualidad	Diario digital de la localidad de Leon.	44
2.	Farmacia EL PUENTE	Un lema 365 dias de salud Evolucionando	41
3.	SalamancArlequinada	Por un baloncesto de calidad en Salamanca	38
4.	BIOTECHNEWS	Si eres un científico español tus tweets merecen	35
5.	RECIMOVIL	Compra-Venta telefonos de segunda mano:	35

Diámetro y Centralidad: El diámetro de esta red es 18. La distancia media de grafo es 7,84. El usuario que tiene la intermediación más alta en esta red es *@usal*, que posee un valor de intermediación siete veces más alto que el segundo. Se presentan los cinco primeros usuarios con Intermediación más alta en esta red. Se puede observar un punto interesante: como hemos visto, hasta el momento, todos los nodos relevantes en esta red son de organizaciones; sin embargo en esta lista aparecen tres usuarios como individuos, que son dos estudiantes y un delegado.

Tabla 4. Usuarios con Intermediación más alto

	Nombre	Descripción	Intermedia.
1.	Usal	La Universidad de Salamanca es	144 millones
2.	Zamora Perfecta	Zamora. Desconocida. Perfecta.	24 millones
3.	Silvia Ramos Castro	Tecnica Superior en Educación Infantil	23 millones
4.	Susana Junquera Rey	Estudiante de Educación Social y..	22 millones
5.	Rafael Daniel	Delegado de El Economista en..	19 millones

4.4. Visualización de Twitter

En *Gephi* se ofrecen diversas disposiciones para visualización de datos. En nuestro trabajo, elegimos la disposición *ForceAtlas 2* [15] y *OpenOrd* [16] para visualizar este gran tamaño red, que son algoritmos que tienen la capacidad de tratar gran cantidad de nodos de un modo bastante rápido (ver Fig. 1).

Podemos ajustar las visualizaciones para descubrir patrones u obtener una imagen más intuitiva. A continuación aplicamos los resultados de cálculo que obtenemos anteriormente a los diferentes atributos de los nodos (color y tamaño) para poder presentar más informaciones.

En la disposición de *ForceAtlas 2*, aplicamos el grado al tamaño de nodo y una gama de colores rojo-blanco-azul al color de nodos. El nodo que tiene el color más rojo significa que tiene menos grado, y al contrario, el que tiene color más azul, tiene el grado más alto. En la disposición de *OpenOrd*, asignamos al tamaño del nodo su valor de Intermediación y su color con una gama de color de blanco a rojo según su grado de entrada. El nodo con color más rojo tiene el grado de entrada más alto. Como el nodo *@usal* tiene un valor de Intermediación

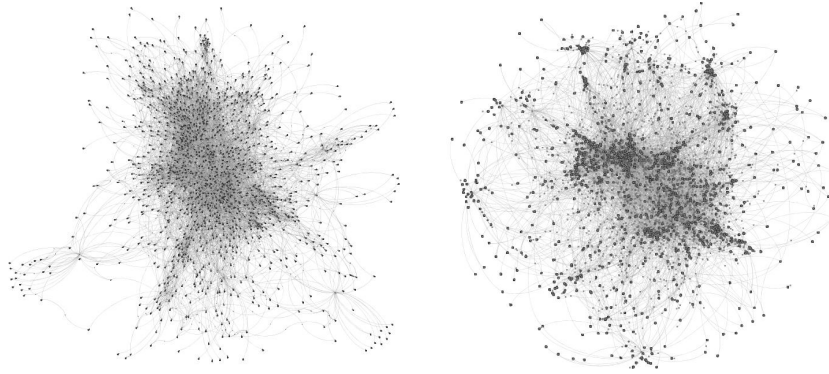


Figura 1. Visualización con ForceAtlas 2 y OpenOrd

mucho más alta que los demás, y su grado de entrada también es alto en la red, es fácil de encontrar este nodo en la figura de visualización (ver Fig. 2).

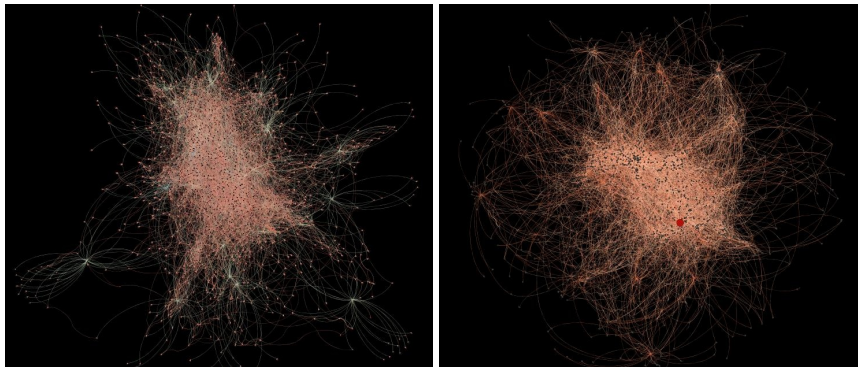


Figura 2. Visualización con grado y centralidad

Por otro lado, podemos utilizar los resultados de Modularidad, el algoritmo de detección de comunidades, tinturar los nodos y aristas con diferente color según su grupo de comunidad, para obtener una nueva representación de visualización (ver Fig. 3).

5. Conclusiones

La realización de este trabajo ha servido para aprender cómo manejar el *Twitter API* y emplearlo en un *crawler*, controlando el comportamiento del

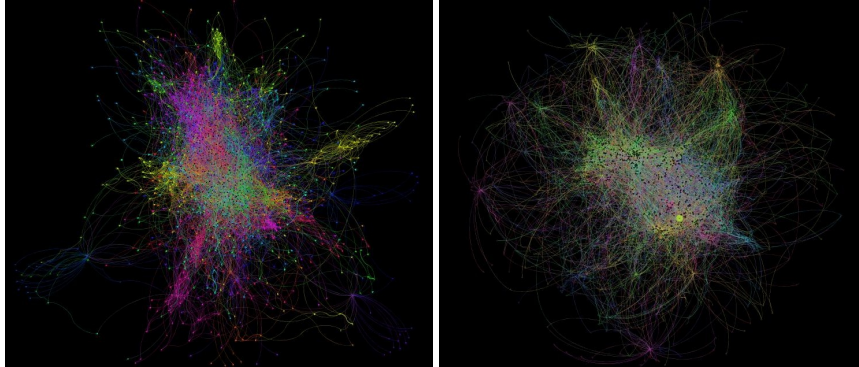


Figura 3. Visualización con Modularidad

crawl para conseguir una mejor recogida de datos. Hemos realizado un análisis y visualización de los datos obtenidos con las herramientas de análisis, hemos hecho una prueba con el Twitter de la Universidad de Salamanca y conseguimos un resultado muy interesante, que significa conseguimos el método para resolver el problema y la necesidad que planteamos al principio del trabajo. Desde este punto de vista, podemos decir hemos alcanzado los objetivos del trabajo.

Sin embargo, no hay que olvidar las deficiencias que existen en este trabajo, hay muchos aspectos que serán susceptibles de mejora, por ejemplo el límite impuesto por *Twitter API* es el mayor problema que nos acompaña durante toda la realización de trabajo, especialmente en el *Twitter API v1.1*, donde que se ha reducido considerablemente el límite. Deberíamos encontrar una manera para resolver esta problemas evitando gastar mucho más tiempo en la espera de recogida de datos que en el proceso más importante de análisis de los datos.

Para el trabajo futuro, podemos desarrollar un *crawler* de tipo distribuido para solucionar el problema de límite de API, podemos mejorar el presente trabajo en diferentes aspecto como una interfaz gráfica más estable, un algoritmo de *crawl* más eficiente o añadir más funciones en el crawler. Además, aparte de *Twitter* existen muchos sitios de redes sociales en Internet, pudiendo aplicar el método que aprendimos en este trabajo a otros sitios dependiendo sus APIs disponibles.

Bibliografía

- [1] Wikipedia, “Red social — wikipedia, la enciclopedia libre,” 2013. [Internet; descargado 1-julio-2013].
- [2] Wikipedia, “Twitter — wikipedia, la enciclopedia libre,” 2013. [Internet; descargado 1-julio-2013].
- [3] TwitterBlog, “Celebrating #twitter7 | twitter blog,” 2013. [Internet; descargado 1-julio-2013].
- [4] Wikipedia, “Centralidad — wikipedia, la enciclopedia libre,” 2013. [Internet; descargado 15-julio-2013].
- [5] J. L. Alonso Berrocal, “Twitter,” 2013. [Internet; descargado 1-julio-2013].
- [6] Dev.twitter.com, “Twitter developers,” 2013. [Internet; descargado 1-julio-2013].
- [7] J. L. Alonso Berrocal, C. García Figuerola Paniagua, and Á. F. Zazo Rodríguez, “Sacarino (sonda automática para la recuperación de información en la web): un robot para recorrer y procesar la web.” *Scire: representación y organización del conocimiento*, vol. 12, no. 1, pp. 211–224, 2006.
- [8] J. L. Alonso Berrocal, C. García Figuerola Paniagua, and Á. F. Zazo Rodríguez, “Diseño de spiders,” tech. rep., Departamento de Informática y Automática, Universidad de Salamanca (España), 2006.
- [9] M. M. Moreno, “Análisis de redes sociales: un tutorial,” 2008.
- [10] Gephi.org, “Gephi, an open source graph visualization and manipulation software,” 2012. [Internet; descargado 1-julio-2013].
- [11] Python.org, “Python programming language – official website,” 2013. [Internet; descargado 1-julio-2013].
- [12] Mike.verdone.ca, “Python twitter tools (command-line client and irc bot),” 2013. [Internet; descargado 1-julio-2013].
- [13] M. A. Russell, *Mining the social web: Analyzing data from Facebook, Twitter, LinkedIn, and other social media sites*. O’Reilly, 2011.
- [14] M. Russell *et al.*, *21 Recipes for Mining Twitter*. O’Reilly, 2011.
- [15] M. Jacomy, S. Heymann, T. Venturini, and M. Bastian, “Forceatlas2, a graph layout algorithm for handy network visualization,” *Paris http://www.medialab.sciences-po.fr/fr/publications-fr*, 2011.
- [16] S. Martin, W. M. Brown, R. Klavans, and K. W. Boyack, “Openord: an open-source toolbox for large graph layout,” in *IS&T/SPIE Electronic Imaging*, pp. 786806–786806, International Society for Optics and Photonics, 2011.