Máster Universitario en Sistemas Inteligentes

Trabajo de Final de Máster

Obtención del contexto social de opiniones obtenidas de redes sociales en la plataforma SLOD-BI

Autor:

José Manuel de la Torre Vilariño

Tutor Académico:

Dr. Ismael Sanz Blasco

Fecha de lectura: 29 de Noviembre de 2016
Curso académico 2015/2016

Página Intencionadamente en blanco

Tabla de contenido

Resumen

Las redes sociales a día de hoy producen una enorme cantidad de datos. Centrándonos en Twitter, de la cual utilizaremos su API para recopilar información de su entorno, nos topamos en ella con una red social altamente masiva, con más de 140 millones de usuarios activos publicando unos 400 millones de *tweets* cada día. Esto abre un campo para la minería de datos inmensamente rico para explorar y extraer información valiosa. En este trabajo se implementa, siguiendo las pautas de la API de Twitter, un proceso de extracción de información para hacer uso del modelo SLOD-BI representando esta metainformación recogida mediante grafos. Se propone el análisis de los mismos como una herramienta muy poderosa para abordar y recoger conclusiones de alto valor, así como enfocar dichos análisis a un campo de estudio en particular. Con ello se arrojan líneas de trabajo a seguir y un cúmulo de aspectos de análisis para el campo de la ciencia de los datos. Este trabajo lidia con el mayúsculo tamaño en los datos así como su no estructuración, proponiendo el análisis de los mismos mediante grafos y el estudio de comunidades y subcomunidades. De este modo, se abren nuevas oportunidades para la minería de datos en redes sociales y su posterior análisis mediante herramientas como Knime y Gephi [5].

Palabras clave

Minería de Datos, detección de comunidades, subcomunidades, grafos, redes sociales, geolocalización, modularidad, *tweet*.

Abstract

Social networks today produce an enormous amount of data. Focusing on Twitter, on which will use its API to collect information from their environment, we came into a highly massive social network, with over 140 million active users publishing 400 million *tweets* every day, opening a field for data mining immensely rich to explore and extract valuable information. The project presented is implemented following the guidelines of the Twitter API to make use of the SLOD-BI model by representing this meta information collected through graphs. This work has to deal with problems caused by the huge size of data, and by its unstructured nature. We propose the analysis of the data by graphs and the study of communities and sub-communities, providing new opportunities for data mining social networks and further analysis in tools such as KNIME and Gephi.

1. Introducción

Desde la explosión de la Internet en los 90, diversos han sido los espacios virtuales que se han construido y expandido a través de los años, con una masificación de usuarios que se manifiestan a través de sus servicios. Sitios webs con disímiles finalidades, blogs y las tan aclamadas redes sociales han sido los protagonistas fundamentales, siendo estos últimos los principales actores dentro de este nuevo mundo digital. Este papel protagónico ha sido posible gracias a la llegada de millones de usuarios a las redes sociales, y ha creado con ello una nueva área de investigación y análisis de los datos que empezaron a acumularse sobre sus acciones, comentarios, publicaciones, etc... Tal ha sido este impacto, que a día de hoy si se busca el término "opinión pública", la cual se extrae de los ciudadanos como el propio término indica, es un hecho comprobable que es mayoritariamente extraída de las redes sociales. En estas funciones Twitter ha sido un medio inconmensurable, aun siendo considerablemente joven. Tras haber emergido hace tan solo una década, se ha convertido en el nuevo medio de referencia dentro de los últimos acontecimientos a nivel mundial. Episodios verificables son el atentado a la revista Charlie Hebdo o el más reciente y lamentable acontecimiento en Niza [40]; porque aparte de ser un medio de referencia, se ha convertido en el campo mayoritario de las investigaciones actuales debido a la libertad de expresión que ofrece.

Este servicio de *microblogging*, del cual se ampliará más en capítulos siguientes, permite el uso de su API y brinda la oportunidad de someter los datos extraídos a análisis. Dentro de la plataforma, se definen ciertos aspectos a tener en cuenta en el siguiente trabajo. En primer lugar, detalles técnicos del propio servicio, como el caso de que un usuario puede seguir o no a otros usuarios o ser seguido. Diferentes acercamientos se perciben en otras redes sociales, como Facebook o Instagram en cuanto a la privacidad de usuarios; la relación de seguimiento o ser seguido no requiere de reciprocidad, siendo un seguidor aquel que recibe todos los mensajes, denominados *Tweets,* del usuario al que sigue. Dentro de Twitter, aparte de este estatus de seguimiento, se encuentran otro número de características nombradas por los analistas en el tema, *Social facts,* que se abundarán en mayor detalle al continuar la lectura de este trabajo. Se dedica un mayor interés en el proyecto a las opiniones con respecto a una temática dada que puede manifestar un usuario. Se estudia para un tópico el impacto que tiene o puede alcanzar en el entorno en el que se desenvuelve, denominado en este proyecto como comunidades.

Muchos investigadores han enfocado sus líneas de trabajo en la identificación de características propias para la detección de comunidades online [37]. Grandes esfuerzos en la detección de faccionalismo político, o redes de recomendación, han proporcionado diversas aproximaciones en el comportamiento de grupos sociales del mundo digital, y actualmente más específicos en las redes sociales. Estos trabajos han despertado gran interés, dada la creencia de que estas relaciones en las que se basan los grafos construidos pueden ser extrapoladas y son de algún modo similares en el mundo real, tanto para enfoques comerciales como educativos o políticos. Es por ello que los medios sociales brindan hoy en día un reto inmenso a los investigadores y analistas de datos que intentan descubrir y lograr un adentramiento en el comportamiento humano digital, y la dinamicidad de los grupos online descubiertos. Dichas sociedades componen la base para el análisis de los datos sociales

por parte de los científicos de datos, y son a su vez la base del trabajo que aquí se presenta: extraer de un tema o temas en particular, hechos sociales de relevancia mediante la detección de comunidades online para su posterior integración en SLOD-BI [5].

SLOD-BI, o por sus siglas *Social Linked Open Data for Business Intelligence*, es una infraestructura para la carga y manipulación de datos semánticamente anotados para la Inteligencia de Negocios. Ha sido diseñada e implementada en la Universitat Jaume I [1]. Ella se encarga de proveer nuevas oportunidades para la integración convencional y social de los datos para la Inteligencia de Negocios. En el trabajo propuesto, tanto el enlazar los resultados alcanzados para arribar a nuevos análisis como la integración en SLOD-BI son tareas fundamentales y problemas de obligada solución. Esto, sumado a la inmensidad de los datos disponibles en los medios sociales y su no estructuración, se plantean como las tareas a resolver. En resumen, se propone un análisis exhaustivo de los datos extraídos para, mediante la representación en grafos arribar a un sistema para la detección y estudio de comunidades online en SLOD-BI.

Objetivo General

El objetivo del trabajo es realizar un estudio de caso para extraer hechos sociales a partir de información extraída de la red social Twitter, e incorporarla al repositorio de SLOD-BI para ser analizada junto a otros datos relevantes del dominio.

Los objetivos específicos son:

1. Se estudiarán las características de los principales *hechos sociales*, y se estudiará cómo es posible obtenerlos a través de la metainformación existente en los *tweets*.
2. Se diseñará e implementará un extractor de información que permita extraer la información necesaria de los *tweets*.
3. Se diseñará e implementará un sistema que permita obtener los hechos sociales a partir de la información identificada en el punto anterior y volcarla en SLOD-BI.
4. Se estudiarán e implementarán distintos algoritmos de detección de comunidades.
5. Se validará el resultado mediante su aplicación a un caso de estudio.

Estructura del trabajo

El presente trabajo fin de máster está estructurado en capítulos que describen los diversos bloques temáticos abordados. Dentro de cada uno de ellos, hay una serie de subapartados, que concretan las ideas desarrolladas en los mismos.

En primer lugar de la memoria descrita se encuentra el capítulo introductorio, donde se presenta la motivación para llevar a cabo dicho proyecto, el objeto de estudio así como los objetivos específicos y la estructura del trabajo, apartado actual.

En el segundo capítulo se describe el contexto del proyecto mediante el análisis de las redes sociales para la detección de comunidades, así como de la plataforma de estudio, Twitter y su API. Se describen además algunos aspectos de la minería de datos con respecto a Twitter. Por último, se realizará un breve análisis sobre el trabajo en el que se integra este proyecto: SLOD-BI.

El tercer capítulo describe la planificación inicial, repartida en tareas que abarcan un total de 300 horas y las modificaciones que se realizaron en el curso del desarrollo del presente trabajo fin de máster. Para continuar el informe, en el cuarto capítulo se analizan los aspectos fundamentales y técnicos del proyecto, y se describen además las tecnologías utilizadas y estudiadas en el proyecto. Acto seguido se detalla la extracción de la metainformación, así como la creación de grafos a partir de esta y el posterior análisis sobre estos grafos en la detección de comunidades. Por último se aplicará dicho análisis en un caso de estudio con un nuevo conjunto y su integración a SLOD-BI, documentados en el capítulo 5.

Se describe en el último de los capítulos la consecución de los objetivos mediante las conclusiones generales para el proyecto y se termina este apartado con el trabajo futuro propuesto en la temática. Esta memoria incluye un apartado de referencias bibliográficas donde se muestra la relación de referencias consultadas, en artículos científicos, libros consultados y páginas webs varias, utilizadas en el estudio para la confección de este trabajo fin de máster.

2. Contexto del proyecto

En este capítulo se presenta el contexto del proyecto, se describen las redes sociales y en especial Twitter y su API. A continuación, se realiza un breve análisis sobre el proyecto al que se integra este trabajo: SLOD-BI. Por último se muestran algunas analíticas para grafos en redes sociales.

2.1 Redes Sociales

Las redes sociales o páginas web en la que los internautas interactúan e intercambian información personal tienen en nuestro día a día un papel trascendental. Son cerca de 1700 millones los usuarios en Facebook o 500 millones en Twitter, donde se desarrollan comunidades virtuales e interactivas que se proponen estudiar a fondo en este trabajo. Este tipo de relaciones entre usuarios pueden ser amistosas, familiares, comerciales o de otra índole. Se accede además a servicios que permiten armar grupos según los intereses de los usuarios y se comparten fotografías, videos e información en general.

Pero al hablar de redes sociales, hay dos grandes aspectos que se pueden definir por separado, la topología de la red y la realidad social que estas redes contienen. El primero de ellos, la red o grafo por sí solo, no es más que una estructura matemática con un conjunto de objetos llamados habitualmente nodos o vértices, conectados de forma binaria entre sí por medio de conexiones llamadas aristas o enlaces y, desde un punto de vista matemático, muy interesantes de investigar al disponer de una robusta teoría, la Teoría de Grafos. Esta teoría matemática facilita el análisis y representación de la información, mientras que por su contraparte y el segundo de los aspectos a describir, la realidad social que estas redes reflejan describe como sus nodos pueden representar personas o entidades relacionadas con contextos y las relaciones representarán relaciones sociales existentes entre ellos (membresía, parentesco, amistad, etc.).

A pesar de que intuitivamente las redes sociales se asemejan a los grafos matemáticos, es más habitual que en ellas se trabaje con distintos tipos de relaciones y no solo con un tipo de conexión predefinida. Por ello, en los últimos años se ha hecho necesaria la extensión del concepto de grafo, así como de la teoría asociada, para poder dar cabida a este tipo de redes con características más ricas que las estructuras clásicas. Desde un punto de vista analítico, las diferentes relaciones que se pueden presentar en una red social permiten su uso en áreas de investigación interdisciplinares, ya que posibilitan el reconocimiento de patrones de comportamiento tanto en el ámbito individual (micro) como a nivel de la red global (macro); y proporcionan interesantes interpretaciones en ámbitos tan diversos como el político, social, cultural, económico, educativo, entre otros [9].

En cuestiones de redes sociales el volumen de datos es un acápite muy importante. Sus extensiones en la actualidad sobrepasan con creces a las técnicas para el análisis de grafos propuestas por las capacidades computacionales actuales. Por ello, se proponen metodologías avanzadas tanto en hardware como en la implementación de software para poder hacer frente al crecimiento continuo y de forma exponencial de las redes sociales.

2.2 Twitter

Twitter es un servicio de *microblogging* que nació en 2006 como parte de un proyecto de investigación y desarrollo de la empresa norteamericana Obvious. En la actualidad es un sitio de interacción social masiva con el objetivo de hacer la comunicación más fácilmente expandible, con más de 140 millones de usuarios activos publicando un total de 400 millones de *Tweets* cada día.

En sus inicios Twitter, o la idea preconcebida de lo que es hoy en día Twitter, se utilizaba como herramienta de comunicación interna en su propia empresa fundadora, basada en mensajes SMS en los que los usuarios describían aquello que estaban haciendo en ese mismo momento [14]. Sin embargo, en 2009 Twitter rediseñó la pregunta "¿Qué estás haciendo?" por "¿Qué está pasando?", de forma que los mensajes de la red social pasaron a centrarse en aquello que ocurre a nuestro alrededor en detrimento de aquello que le sucede al propio usuario. Esta reconfiguración de Twitter está vinculada a las teorías constructivistas sobre el rediseño de la tecnología por parte de los usuarios [15].

Su funcionamiento simplemente consiste en que un usuario de esta red puede seguir a otros o ser seguido por otros. A diferencia de Facebook o MySpace la relación de seguimiento o *following* no requiere de reciprocidad. Un usuario puede seguir a quién este desee, y el usuario seguido no tiene que seguir en correspondencia a quien lo sigue. Pero, es propio de Twitter y fundamental de ampliar en ello, la velocidad y facilidad de publicación que ofrecen a quien quiera hacer uso de sus recursos; convirtiéndola en el medio de comunicación más importante actualmente, tanto como para roles en eventos socio-políticos como, por ejemplo, el movimiento de *Occupy Wall Street,* en el cual desempeñó un papel preponderante. Twitter además ha sido de gran ayuda a la hora de publicar reportes de daños o desastres naturales, tanto el Huracán Sandy como los más actuales atentados en la ciudad de Niza. Por tanto y a modo de resumen es sin lugar a dudas la referencia mundial en los sucesos actuales y el nuevo medio que a pesar de su juventud tiene ya el liderazgo en estos ámbitos y se mantiene en crecimiento [4].

2.2.1 *Tweets* y otras características de Twitter

Twitter permite a los usuarios la creación de sus propios contenidos digitales, con un límite de 140 caracteres denominados "*tweets*" o "tuits". Estos suelen tener forma de texto puro y pueden incluir hipertexto: enlaces a blogs, páginas web, imágenes, vídeos o cualquier otro material online que puede ser publicado desde varias fuentes; tanto la página oficial de Twitter, las aplicaciones de móvil de la propia compañía así como terceras partes contenidas en aplicaciones o sitios webs tras autentificación que permiten postear contenido a Twitter también. Por tanto se abren posibilidades muy diversas con respecto a los contenidos [4]. De algún modo, cuando se plantea la utilización de Twitter para determinadas actividades, es habitual el uso de servicios externos referenciados a través de la red, quedando así Twitter como modo de punto de encuentro y de referencia al exterior [17].

Una vez que un usuario crea una cuenta puede seguir a cualquier otro registrado. De esta forma, podremos leer todos los mensajes que escriban los usuarios que seguimos, sin que exista una necesaria reciprocidad, tal como se explicaba en la sección anterior. Los usuarios también tienen control sobre su privacidad y pueden elegir entre hacer sus *tweets* públicos o privados. Además, un usuario de Twitter puede construir un canal instantáneo y personalizado que se ajuste a su gama de intereses, dependiendo de los usuarios a los que siga que pueden despertar un atractivo personal y/o profesional [18].

Twitter es desde su creación la primera herramienta con estas características y está teniendo un fuerte impacto social. De hecho, el número de usuarios sigue creciendo de forma muy significativa, siendo en España la tercera red social por índice de penetración, con un 43% de seguidores según un estudio publicado del Observatorio de Redes Sociales [16]. Diversos estudios vaticinan que en un futuro muy próximo parte de la investigación educativa se centrará en Twitter [19] o como se afirma en [20], "en Twitter las palabras valen mil imágenes. La plataforma se ha convertido en el alambique en el que se destila la cultura de nuestro tiempo."

2.2.2 API y Minería de datos en Twitter

En informática, una Interfaz de programación de aplicaciones (API) requiere de un software que le respalde para llevar a cabo sus operaciones, entradas, salidas y tipos subyacentes de datos dentro de su funcionamiento. Su principal objetivo es definir un conjunto de funcionalidades que son independientes de su respectiva aplicabilidad, lo que permite tanto la definición e implementación de aplicaciones sin comprometer a la API en efecto.

Una API en resumen, es un conjunto de instrucciones programadas y estándares de acceso a una aplicación basada en la Web o Servicio Web. Una compañía de software generalmente da acceso a su API para que de esta forma otros desarrolladores de software puedan desarrollar productos que son impulsadas por su servicio. En adición a un acceso a bases de datos o hardware, una API puede hacerse para aliviar el funcionamiento de componentes gráficos en interfaces de usuario, las tan llamadas "plug-in-API", y la posibilidad además de poder compartir datos entre distintas aplicaciones. En la práctica una API puede ser encontrada en forma de bibliotecas que incluyen especificaciones de rutinas, estructuras de datos, clases y variables, pero en otros casos y enfocándonos en la API que se utiliza, las podemos encontrar mediante servicios REST o SOAP [38] y se convierten en un modo de hacer llamadas remotas de consumidores. Por su parte Twitter basa su *Application Programming interface* (API) en el servicio con arquitectura *Representational State Transfer* (REST). REST hace uso del protocolo HTTP para sus funcionalidades, como requerir escritura de datos, consultas y borrado.

Por su parte OAuth, abreviatura para *Open Authorization*, nos proporciona una forma para que se autorice a una solicitud, en nuestro caso al programa que se construirá para el análisis de comunidades, para acceder a los datos que haya guardado en otra aplicación sin tener que compartir su nombre de usuario y contraseña. Se hace uso en OAuth de una clave secreta que es entregada a la aplicación que quiere acceder y para la cual mediante la utilización de un token secreto se hace entrega del permiso de acceso a los datos. En Twitter se hace uso de OAuth, para el API de uso público [5].

En resumen, a una aplicación adecuada de un cliente se le irán pasando mensajes que contienen la información de los *tweets* y otros eventos ocurridos en vivo. Twitter ofrece para este propósito varios *endpoints* de streaming. Un *endpoint* se refiere a un servicio en tiempo real que entrega contenido directamente a la aplicación cliente que construimos. Cada uno de ellos personalizado para ciertos casos de uso. Este contenido es apto para los siguientes usuarios o temas específicos en la minería de datos:

- Usuarios a los que se les transmite *streams* de un solo usuario, que contienen más o menos todos los datos que corresponden con la opinión de un solo usuario de Twitter.
- *Streams* para sitios webs, lo cual podría catalogarse como la versión multiusuario de corrientes de usuario destinadas a los servidores que deben conectarse a Twitter en nombre de muchos usuarios.

Usando la API de Twitter se pueden responder una serie de preguntas que compondrán la base para el análisis que se realizará en el capítulo 4. Ejemplo de interrogantes formulables son: ¿Cuántos amigos tengo o tiene alguien?, ¿A quién sigo que no me sigue de regreso?, ¿Quiénes son en mi rango de amigos las personas con más y también con menos amigos? o ¿Quiénes son mis amigos mutuos? Y un número grande de cuestiones que pueden ser formuladas según análisis; además de los *tweets* en bruto, según un tema o un autor, lo cual es principalmente el mayor uso hecho por parte del presente trabajo final de máster hacia la API de Twitter. Para dichos propósitos se desarrollarán scripts en Python que serán capaces de extraer mediante las interrogantes y los parámetros antes mencionados la información que nos interese para posteriormente visualizarla y realizar la búsqueda de conjeturas valiosas mediante el análisis y la teoría de grafos.

2.3 SLOD-BI

SLOD-BI, o por sus siglas, *Social Linked Open Data for Business Intelligence,* es una infraestructura para la carga y manipulación de datos de sentimientos en redes sociales semánticamente nueva en el entorno de la inteligencia de negocios. Introduce la integración de la Voz del Cliente (VoC) y la Voz del Mercado (VoM) con el conjunto de factores externos que pueden potencialmente afectar a una entidad. La primera se refiere a las opiniones de los clientes sobre un producto y servicios ofrecidos por una compañía, mientras que la VoM comprende toda la información relacionada con el mercado objetivo que puede afectar a una empresa en sus funciones. Por tanto, SLOD-BI propone la combinación de datos corporativos con datos sociales. Sigue los principios de los datos abiertos y enlazados (LOD), y mediante ellos propone una infraestructura para la publicación de ambos conjuntos de datos sociales, Vom y VoC, y los automáticamente extraídos datos de sentimientos. En términos generales, la infraestructura desarrollada comprende los principales patrones de la inteligencia de negocios para el análisis de datos sociales y corporativos de una forma integrada, y proporciona la funcionalidad requerida para realizar análisis de opiniones masivas en la red. Ofrece a los usuarios la incorporación de dimensiones relacionadas a las opiniones en sus análisis, lo cual se encuentra fuera del alcance de la tradicional inteligencia de negocios [1].

2.3.1 Infraestructura

Los componentes principales y con ello los marcos fundamentales de la infraestructura de SLOD-BI se construyen pensados en primer lugar sobre el respeto a los principios de la Inteligencia de Negocios (BI). Se construye para analizar tanto los datos corporativos como los datos sociales en una forma integradora, y se proporcionan a su vez las funcionalidades requeridas para poder desarrollar análisis de opinión masivas. Ejemplo de ello es la extracción automática de sentimientos en los datos sobre publicaciones en la red, proporcionando nuevas dimensiones relacionadas a la opinión dentro de sus análisis y hasta el momento externas al alcance de la BI tradicional.

En la Figura 1, que se muestra a continuación se visualizan de forma concisa dos capas, que constituyen la distribución principal de dicha infraestructura. Una capa interior o anillo constituye el vocabulario y los conjuntos de datos de la arquitectura diseñada, SLOD-BI, mientras que la estructura externa nos muestra los vocabularios abiertos y enlazados (LOV, *Linked Open Vocabularies*) y los conjuntos de datos que están directamente relacionados a la infraestructura estudiada.

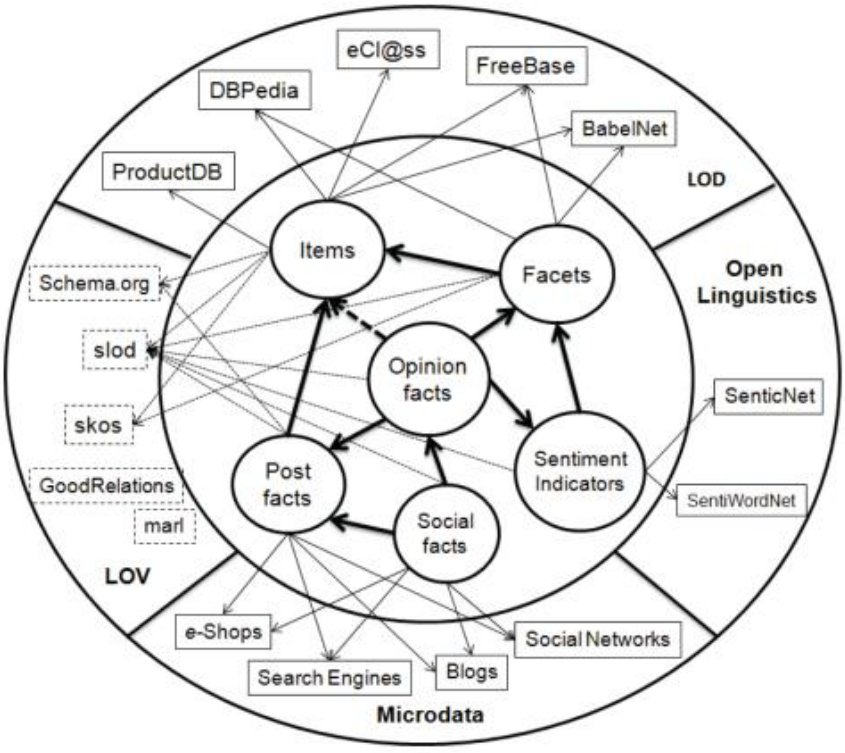


Figura 1. Vista de la Infraestructura de SLOD-BI

Cada componente mostrado en el anillo interno consiste en una serie de conjuntos de datos formados por tripletes RDF. La principal contribución en esta nueva arquitectura son los datos basados en

sentimientos (*Sentimental Indicators, Opinion Facts, Social Facts*, etc.). Todos estos conjuntos de datos aquí mostrados son confeccionados y actualizados independientemente unos de otros y pueden estar ubicados en diferentes servidores. En la Figura 1 se muestran los enlaces entre sus componentes, los cuales se establecen coherentemente y por consiguiente son usados para realizar tareas de análisis en conjunto. Según [1] son nombrados enlaces fuertes, facilitando dentro del anillo interno operaciones de Unión (Join) entre tripletes de estos conjuntos de datos. Por su parte los enlaces que se muestran dibujados con una línea más fina y se localizan uniendo ambos anillos reciben el término de enlaces débiles, ya que estos solo establecen posibles conexiones entre entidades de la infraestructura y las bases de datos externas. Estos conjuntos de datos externos resultan para SLOD-BI extremadamente útiles a la hora de realizar análisis exploratorios, incorporar nuevos datos y ante la posible detección de nuevas dimensiones de análisis para la capa interna de la arquitectura.

2.3.2 Social Facts

Los hechos sociales pueden catalogarse como observaciones realizadas sobre quienes expresan sus opiniones e intercambian sentimientos o estados sobre un tópico en particular. Estos hechos son fácilmente extraíbles de las redes sociales tras un análisis de la estructura que pueden tomar los datos cuando quienes pueden aportar información sobre un tema discuten sobre el mismo [22]. El análisis e implementación de algoritmos propuesto en este trabajo, discutido en el capítulo 4, se enfoca en la detección de comunidades, y el resultante identificador de comunidades como el *Social Fact* específico con que se trabajará. Sobre esta temática se ampliará haciendo énfasis en el conjunto de datos a tratar en Twitter, y el uso e implementación de la extracción de información válida mediante la API de Twitter ya discutida. Sin dejar de mencionar lo dinámicas que pueden ser este tipo de comunidades basadas en un tópico en dependencia de la validez o relevancia en el tiempo del tópico en cuestión. Para ello se añade un atributo de tipo temporal que permitirá su almacenamiento de forma apropiada, y posibilitará en un futuro realizar rastreo de comunidades sobre la detección que en este proyecto se realiza en el apartado 4.4.

2.3.3 Ejemplo de análisis social con SLOD-BI

La plataforma de software, SLOD-BI, se pone a prueba con un caso real. Para ello se lleva sus activos hasta una compañía de alquiler de coches a la cual le interesa mejorar la atención al cliente, en vías de mejorar la calidad de sus servicios mientras logra aminorar sus costes. SLOD-BI para dicho propósito se presenta como la herramienta ideal para investigar en un conjunto de datos gigantesco y ofrecer a la compañía las pautas a seguir y una visión desde el punto de vista del cliente detallada en varios aspectos hacia sus productos y servicios. Para asegurar buenos resultados en la propuesta, la compañía se aseguró un conjunto de objetivos estratégicos y medidas a tener en cuenta en el análisis a realizar;

siendo entre sus tópicos más importantes un análisis de sentimiento detallado sobre sus productos y servicios. Se quiere manejar la opinión de las personas sobre los vehículos que se ofrecen, distribuida en "*comfortability*, *safety*, *driving perception*, *design*, *mechanical issues* and *price"*, para ser explícitos.

SLOD-BI permite crear de forma sencilla un conjunto de gráficos e información relevante a tener en cuenta con el fin de hacer más próspera la compañía en un futuro cercano y a largo plazo mayoritariamente. Uno de los gráficos construidos, es el realizado sobre el estado de sentimientos sobre los problemas mecánicos, ámbito que para una compañía de alquiler de coches es vital y que se muestra en la Figura 2 [1].

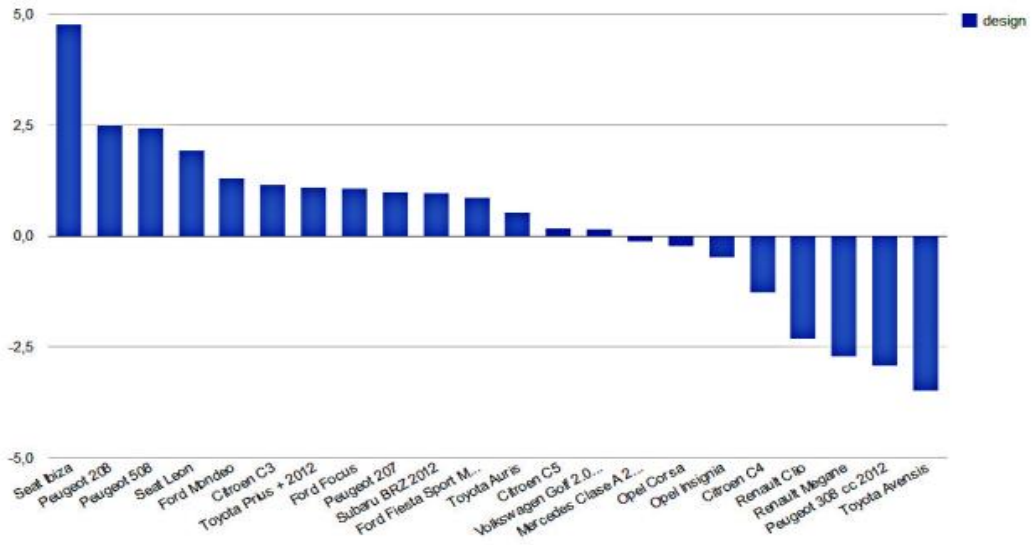


Figura 2: Relación de marcas de coche y opiniones de clientes con respecto a problemas mecánicos.

2.4 Análisis de Grafos para las Redes Sociales

Las interacciones realizadas en las redes sociales generalmente se tratan como grafos, y se usan por consiguiente métricas de grafos para analizar y describir la importancia de ciertos rasgos dentro de la propia red. Se plantean dos representaciones. La primera de ellas consiste en representar a los actores mediante vértices y conectar dos actores cuando estos sostengan una interacción entre ellos. La otra forma de representación son los grafos bipartitos, en donde ambos actores involucrados y la interacción que los une son representados como vértices. La diferencia en esta última aproximación es la adición de un vértice por cada enlace en la red y en consecuencia la duplicación del número de enlaces en el grafo, añadiendo en redes altamente pobladas un considerable número de nodos y aristas que no añaden nueva información a la red. Por tanto en el trabajo que aquí se presenta se utiliza la conexión directa entre actores para el análisis en grafos, y la consiguiente extracción de información y detección de comunidades online.

2.4.1 Comunidades en Redes Sociales

La topología de las redes sociales ha sido y es un campo extensivamente estudiado. En primer lugar, es una premisa la estructuración basada en comunidades que presentan las redes sociales. Este tipo de estructura puede ocurrir debido a razones sociales, políticas o culturales. El análisis de la estructura de la comunidad en las redes sociales se puede utilizar para averiguar los *tweets* influyentes y grupos de usuarios específicos para las marcas, los deportes, las organizaciones políticas y de tecnología. Las comunidades también han sido analizadas para descubrir los eventos de desastre. Por ejemplo, la Figura 3 a) presenta un ejemplo de una red tradicional, la red club de karate de Zachary [8], que ha sido ampliamente utilizada para evaluar la estructura de la comunidad y la detección de redes. La red nos muestra las interacciones sociales entre los individuos en un club de karate en una universidad estadounidense. El club se dividió en dos grupos, como resultado de una disputa entre el administrador del club y el profesor de karate y director. La estructura social real en el gráfico se muestra por los cuadrados y los círculos que representan el grupo de individuos que tomaron partido por el administrador o el profesor de karate respectivamente. Sin embargo, ha habido varias investigaciones y métodos de detección de las comunidades que también han llegado con otros criterios de agrupación aportando resultados significativos, los cuales se toman en consideración en la realización de este trabajo, como son el método de Girvan-Newman o el algoritmo Infomap (ver Figura 3, b) que se describen en el capítulo 4.

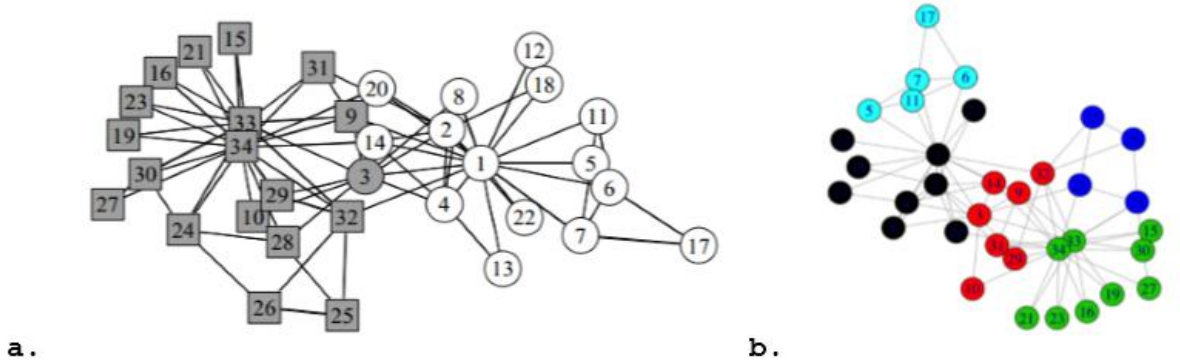


Figura 3. Club de Karate de Zachary.

La mayoría de los algoritmos para el análisis de redes sociales sólo tienen en cuenta las relaciones sociales entre los usuarios para el análisis de clústeres entre los usuarios, e ignoran la gran cantidad de información disponible en las redes sociales actuales. Además de las conexiones sociales, Twitter puede ser utilizado para obtener diferentes tipos de enlaces entre los usuarios como son menciones, similitud entre los *tweets* de usuarios diferentes, *retweets*, *hashtags* o ubicaciones; por tanto se propone enriquecer los métodos tradicionales con esta información.

3. Planificación

En este apartado se presenta la planificación temporal de las tareas que se han de realizar de acuerdo a los objetivos marcados para el desarrollo de este trabajo de fin de máster. En la tabla que se muestra a continuación se muestran las tareas propuestas en la concepción del proyecto.

Tarea	**Horas planificadas**	**Objetivo**
Familiarización con SLOD-BI y los hechos sociales que define.	15	1
Familiarización con el API de Twitter y determinación de cómo obtener los hechos sociales de SLOD-BI a partir de la metainformación disponible sobre los *tweets*.	55	1
Implementación de la extracción de la metainformación de *tweets.*	45	2
Estudio de los métodos para calcular los hechos sociales a partir de la metainformación de los *tweets*.	45	3
Implementación de los métodos elegidos para calcular los hechos sociales a partir de la metainformación de los *tweets*	40	4
Implementación de la carga de datos	20	3
Implementación de un prototipo a partir de un caso de estudio	80	5
Total	300 horas	

Tabla 1. Planificación Inicial para el Trabajo fin de máster

Como se puede ver en la tabla, el tiempo total son 300 horas, tal y como se estipula en la normativa de la asignatura SIU043 Trabajo de fin de máster. Sin embargo ha habido algunas desviaciones respecto a esta planificación inicial. Por una parte en la familiarización con la API de Twitter y la determinación de cómo obtener los hechos sociales de SLOD-BI a partir de la metainformación disponible sobre los *tweets*; fue necesario adicionarle un aprendizaje de las tecnologías que iban a ser usadas en la tarea

que le sucede, la implementación de la extracción de la metainformación de *tweets*, en concreto tecnologías como Anaconda, Python y Tweepy. Además durante la fase de implementación y pruebas se detectaron problemas que supusieron una mayor dedicación para solucionarlos. La planificación real del tiempo así como las nuevas tareas y cambios en las iniciales reflejadas en la Tabla 1, se muestran a continuación en la Tabla 2.

Tarea	**Horas planificadas**	**Objetivo**
Familiarización con SLOD-BI y los hechos sociales que define.	15	1
Familiarización con el API de Twitter y determinación de cómo obtener los hechos sociales de SLOD-BI a partir de la metainformación disponible sobre los *tweets*.	40	1
Aprendizaje de las Tecnologías necesarias para la implementación de la metainformación en Twitter.	35	2
Implementación de la extracción de la metainformación de *tweets.*	40	2
Estudio de los métodos para calcular los hechos sociales a partir de la metainformación de los *tweets*.	35	3
Implementación de los métodos elegidos para calcular los hechos sociales a partir de la metainformación de los *tweets.*	40	4
Análisis sobre los resultados obtenidos de calcular los hechos sociales.	30	3
Implementación de la carga de datos	40	3
Implementación de un prototipo a partir de un caso de estudio	30	5
Implementar la integración a SLOD-BI	15	3
Total	300 horas	

Tabla 2. Planificación real para el trabajo fin de máster

4. Implementación y Análisis

En este capítulo se definen los algoritmos implementados para extraer y construir una topología en forma de grafo sobre la información de Twitter, en específico para el tema Opel Astra. Acto seguido se describe el análisis sobre el grafo construido y las transformaciones realizadas en él, según medidas en Twitter estudiadas. Se introduce a su vez en este capítulo para SLOD-BI el análisis de los usuarios por su geolocalización, para a continuación identificar el método más fiable para la detección de comunidades en nuestro conjunto y los posteriores análisis con los elementos que se disponen tras los anteriores acercamientos.

4.1 Tecnologías Empleadas

Para la implementación de la extracción de la información, así como para la ejecución de los métodos elegidos para calcular los hechos sociales a partir de la metainformación extraída, se utilizaron una serie de tecnologías que abarcan desde la implementación hasta el análisis. A continuación se detallan estas tecnologías y se describen brevemente.

- **Python:** Lenguaje de alto nivel de programación con semántica dinámica, orientado a objetos e interpretado. Contiene estructuras de datos de alto nivel, convirtiéndolo en un lenguaje de programación muy atractivo para el desarrollo rápido de aplicaciones. Su sintaxis es sencilla y hace un gran hincapié en la legibilidad por parte del programador, lo cual reduce la curva de aprendizaje y el costo del mantenimiento de programas desarrollados en el lenguaje. El intérprete de Python y la amplia biblioteca estándar de que disponen están disponibles en forma de código fuente o binario sin cargo para todas las plataformas principales y se pueden distribuir libremente, incrementando el desarrollo de nuevas librerías y funcionalidades para la expansión de sus capacidades [34].

    - **Igraph:** Es una colección de herramientas de análisis sobre redes con énfasis en la eficiencia, portabilidad y facilidad para su importación y uso. Es una biblioteca de código abierto y gratis. Puede ser programada y utilizada en R, Python y C/C++ [30].

    - **SNAP:** Por sus siglas Stanford Network Analysis Platform, es una biblioteca para el análisis y la minería en redes. Fue escrita en C++ y escala con facilidad redes masivas con millones de nodos. Tiene además funcionalidades entre las que se destacan el cálculo de propiedades estructurales, la generación de grafos y la posibilidad de agregar atributos en nodos y aristas de los grafos [35].

    - **Community:** es una biblioteca de Python que implementa el algoritmo de Louvain en C++ y lo expone para Python. Esta a su vez, usa la biblioteca igraph que se comentaba anteriormente, y mejora en algunas aspectos su utilidad, con métodos como: *Modularity* o RBConfiguration.

- **NetworkX:** Es un paquete de software desarrollado en Python que permite la creación, manipulación y el estudio de la estructura, dinamicidad y funcionalidades de redes complejas.

- **Matplotlib:** Es una biblioteca de Python para el dibujado 2D. Produce figuras de calidad de publicaciones científicas en una variedad de formatos y ambientes interactivos. Puede ser usada, como en el caso del trabajo que aquí se presenta, mediante scripts de Python. Entre sus principales funciones se encuentran la de hacer scatterplots, histogramas, plots, entre otros [36].

- **Anaconda:** Es una distribución puntera en la ciencia de datos, gratuita y de código abierto en los lenguajes de programación Python y R. Se encarga del procesamiento de análisis predictivos y científico en datos de tamaño gigantesco.

    - **Spyder:** Es un entorno de desarrollo interactivo de Python que proporciona características similares a Matlab en un software simple y ligero. Proporciona un editor de código fuente con resaltado de sintaxis y características de introspección y análisis de código, contando además con una consola de Python y un editor de matrices Numpy, entre otras características.

- **Knime:** Por sus siglas *Konstanz Information Miner,* es una plataforma de minería de datos que permite el desarrollo de modelos en un ambiente visual. Es además totalmente gratuito. Desarrollado originalmente en el departamento de bioinformática y minería de datos de la Universidad de Constanza, Alemania, bajo la supervisión del profesor Michael Berthold. Desarrollado sobre la plataforma Eclipse y programado, esencialmente, en Java.

- **Gephi:** Es una herramienta de código abierto gratuita, desarrollada en Java para visualizar y analizar todo tipo de gráficos y redes. Es un software puntero en estos asuntos y nos brinda una interfaz lo suficientemente amigable para tener una curva de aprendizaje muy pequeña para el dominio de sus principales funciones [31].

- **JavaScript:** Es un lenguaje de programación interpretado, dinámico y de alto nivel desarrollado por Brendan Eich y originalmente conocido como LiveScript. Fue estandarizado en la especificación de ECMAScript y es en conjunto a HTML y CSS una de las tres tecnologías más difundidas a nivel global.

    - **D3.js *Data Driven Documents*:** Es una biblioteca de JavaScript para la manipulación de documentos basados en datos. Proporciona el nexo adecuado para que los datos cobren vida usando HTML, SVG y CSS. Combina poderosas visualizaciones con las capacidades al máximo de los navegadores modernos.

- **JSON:** Formato ligero para el intercambio de datos. La simplicidad de JSON ha dado lugar a la generalización de su uso, especialmente como alternativa a XML para el intercambio de

información y la integración de fuentes de datos heterogéneas. En las entradas y salidas de los algoritmos generalmente se utiliza JSON, así como para intercambiar información con la API de Twitter y Google.

- **REST:** Estilo de arquitectura software para sistemas hipermedia distribuidos por la World Wide Web. En la actualidad se usa en el sentido más amplio para describir cualquier interfaz entre sistemas que utilice directamente HTTP para obtener datos o indicar la ejecución de operaciones sobre los datos, en cualquier formato (XML, JSON, etc.) sin las abstracciones adicionales de los protocolos basados en patrones de intercambio de mensajes, como por ejemplo SOAP. Los algoritmos desarrollados, así como las API de Twitter y Google trabajan con esta arquitectura. En el apartado 2.2.2 es posible consultar algunos detalles más del estilo de arquitectura software REST y como se encuentra embebido en el trabajo aquí presentado.

- **OAuth:** Es un protocolo abierto para permitir la autorización segura a una API desde aplicaciones de escritorio y aplicaciones web, a través de un método sencillo y globalmente estandarizado. Maneja *handshakes* entre aplicaciones y se utiliza cuando un editor de API quiere saber con quién se está comunicando el sistema. Muchos de los editores de API más grandes (Twitter, Google, Facebook, etc...) han implementado OAuth para manejar escritura de acceso a sus APIs.

- **TagCrowd:** Es una aplicación web para la visualización de frecuencias de palabras. Es capaz de crear de un conjunto de palabras una nube de palabras, que a su vez es una forma muy elegante de comunicar mucha información sobre un texto en una sola imagen [32].

- **JSON-LD:** Por sus siglas denota *JavaScript Object Notation for Linked Data*. Es un formato de serialización y mensajería de datos y en sí, es simplemente JSON con la capacidad de expresar datos estructuradamente y capaz de serializar datos enlazados. Su sintaxis está diseñada para integrarse fácilmente en sistemas desplegados que ya utilizan JSON y logra proporcionar una ruta de actualización sencilla desde JSON a JSON-LD. En términos generales, es una forma de utilizar datos vinculados en entornos de programación basados en la web, de crear servicios webs y de almacenar datos enlazados en motores de almacenamiento basados en JSON [33].

- **API Twitter:** La API de Twitter en resumen, es un conjunto de instrucciones programadas y estándares de acceso a *Tweets*, Usuarios, Entidades y Lugares. Más información detallada al respecto se describe en el contexto del proyecto, epígrafe 2.2.2.

- **API Google:** Es el conjunto de instrucciones programadas por Google que nos permiten la comunicación con los servicios de Google y su integración con otros servicios. Dicha API provee funcionalidades analíticas, de aprendizaje automática e incluso acceso sobre datos de usuarios, cuando el permiso para leer esos datos es concedido. En general es una API muy poderosa debido al poder de la compañía que la maneja y entre sus servicios más destacados se encuentran además API sobre Gmail, Youtube, Google Maps o Translate.

4.2 Descripción del Análisis implementado sobre la metainformación en Twitter.

Se define un grafo no dirigido en el que los nodos son los usuarios extraídos del conjunto de datos proporcionado por el modelo SLOD-BI, para el tema “Opel Astra”. Para el grafo que se construye se definen algunas métricas iniciales. En primer lugar, si un usuario A tiene como amigo a un usuario B entre ellos existirá una arista que unirá a ambos. En segundo lugar se define una máxima de amigos para cada usuario. Se determina que para usuarios con un mayor número de seguidores, exactamente mayores de 10000, se considerarán como máximo 500 amigos. Para los de menor rango, el máximo será de 200 amigos para así hacer al conjunto resultante más fácil de analizar y a su vez propiciar la eliminación de ruido. Se logra evitar con el planteamiento anterior casos de usuarios como celebridades o figuras políticas que son seguidas por decenas de millones de usuarios; por ejemplo recolectar 5 millones de usuarios nos llevaría cerca de 300 días de la forma en que funciona el algoritmo que se describe más adelante en esta sección. Como se logra comprobar en la API de Twitter y en estudios sobre el tema no hay forma de realizar una selección aleatoria no sesgada de los amigos de un usuario, por lo que no hay más alternativa que solamente obtener un subconjunto de estos y guardar como grado original del nodo el número de amigos que realmente tiene ese usuario en la red de Twitter.

Para obtener la lista de usuarios que siguen un usuario o son seguidos por este, se hace uso de la API de Twitter y las consultas GET Friends/ids. Se crea un usuario para la autentificación en la API, nombrado @JoseMan90t, al cual se le adjuntaron varias aplicaciones en el entorno de desarrollo de Twitter y para las cuales se obtuvieron permisos de escritura, lectura y de obtención mensajes directos. Todo ello siguiendo las normativas de dicha entidad, es decir, se permite en un rango de 15 minutos unas 180 consultas a la API y se implementan en los scripts de Python llamadas condicionales que escuchan cuando Twitter mediante la plataforma *Tweepy* nos orienta sobre errores. Varias son las dificultades que se manejan y al ser un programa en tiempo real intercambiando información con los servidores de Twitter se hace necesario corregir problemas que van desde la conectividad, como caída de servicios, de servidores ocupados, o inconsistencia en los datos, como consultas que no pueden ser devueltas o no son encontradas, para conferirle al programa la autonomía suficiente para ejecutarse un período de tiempo prolongado. Un ejemplo del procedimiento a seguir es el caso del error de la “Tasa Límite Excedida”, lo cual hace saltar un fragmento de código que hace al programa detenerse por 15 minutos, para seguir su ejecución tras haber transcurrido este tiempo. Para esta colecta de información en términos generales, se destinó un ordenador solamente a obtener y organizar la información recogida de forma correcta, para posibilitar ser consumida por los posteriores análisis descritos en el capítulo 4, y en un término de 25 días logra recoger la información que se analizará en el epígrafe 4.4.

4.2.1 Extracción de metainformación en Twitter

En la página de Twitter los usuarios que mantienen un vínculo de amistad con una persona se presentan como una lista con *scroll* infinito. Ello significa que a medida que el software que investiga al usuario se acerca al final de la lista, se realiza un nuevo pedido asincrónico al servidor que agranda dinámicamente la lista que se recorre. Para satisfacer tales especificaciones se investigó cual era el método invocado para obtener la lista inicial y cuál era el método invocado asincrónicamente para agrandar esta lista. Para ello se utilizan las funciones definidas en la biblioteca Tweepy, aunque en primer lugar el programa debe hacer uso del protocolo OAuth que ya definimos en el subcapítulo 4.1.

Como paso siguiente se definen dos directorios donde se almacena la información relevante sobre los usuarios que se extraen. Para ello se hace uso de otro elemento importante, el método constructor Cursor. El método Cursor realiza la función de parámetro para cada uno de los requerimientos de la API de Twitter y se encuentra definido como un objeto en Tweepy. Al hacer uso de Cursor no es necesario pasar los parámetros directamente al método que consulta en la API de Twitter. Para ello se utiliza el objeto Cursor y él se encarga de pasar estos parámetros al método cuando este haga el requerimiento.

El programa implementado se compone de tres scripts de Python. El primero de ellos, es el script que se analiza con más detenimiento en el párrafo anterior y que es capaz de tener una autonomía que le permite estar activo indefinidamente y almacenar ordenadamente la información que logra recolectar de la API de Twitter. Este script de Python que se decide nombrar *Main_Program_Collect_Info*.py, produce para cada usuario analizado un fichero con extensión json donde se almacenan el ID del usuario así como el ID de todos los usuarios que componen su lista de amigos, junto a su nombre y el peso resultante del que dispondrá dicho nodo en el grafo resultante. Este primer script además construye un segundo fichero con extensión csv que contiene para cada nombre de usuario sus seguidores de una forma más detallada. Se archivan para los usuarios su nombre, uno “real” proporcionado a Twitter por el propio usuario, así como el ID que ya teníamos recogido. Finalizado hasta este punto, el segundo de los scripts implementado, construye con la información recogida para todos los usuarios un fichero que contiene un modelo tipo red, *Twitter_network*.csv. Dicho fichero se encuentra compuesto por tres columnas: las dos primeras para los nodos que se conectarán, es decir relaciones, y la última para el peso que tendrá el primero de los nodos. Todo ello para pasar en el siguiente paso al último de los scripts construidos, que dibuja un grafo usando las librerías matplotlib y NetworkX, especificadas en la sección 4.1. El grafo que se construye es la base de los análisis a continuación y se muestra en el apartado a continuación, epígrafe 4.3.

4.3 Visualización del conjunto de datos

Se presentan a continuación algunas visualizaciones y análisis realizados sobre los datos extraídos. Las visualizaciones que se construyen conforman una topología circular debido al gran número de aristas. Se agrupan un número elevado de *followers* alrededor de los usuarios extraídos de los hechos sociales de SLOD-BI, y para los que se cuantifica el número de seguidores en el archivoTwitter_Network.csv explicado en el apartado 4.2. Dichas visualizaciones nos muestran a simple vista medidas tales como el impacto de una serie de comunidades en la globalidad de Twitter, o el alcance real de una posible expansión de la información medida por el impacto para un conjunto de nodos. Todo ello se suma a SLOD-BI como herramientas visuales de análisis muy poderosas, se adiciona además la posibilidad de geo-localizar a los usuarios de nuestro conjunto y realizar análisis sobre dichas gráficas. Se construyen en este trabajo mapas reales sobre los cuales son posibles distintos análisis comparativos y analizar semejanzas para la detección de comunidades online.

En primer lugar se analiza el grafo completo que se crea para el conjunto analizado en su totalidad. Este grafo requirió de prestaciones de computador altas, debido a su tamaño, con cerca de medio millón de aristas. Se muestra en primer lugar, para el tema Opel Astra, su impacto para una métrica de uno como propagación de amistad con un usuario. La Figura 4 se muestra como una representación muy aproximada del potencial alcance de un tweet para un tema en específico. Dicha métrica es especialmente importante para cualquier publicista en redes sociales, debido a que es la forma más exacta de medir la audencia, y por tanto el potencial alcance de un tweet a escala global. Se calcula esta métrica como los usuarios que mencionan el tema, tema que puede ser un contenido, otro usuario, una marca, etc., sumado a la cantidad de seguidores que tiene dicho usuario [13].

A continuación se muestra un sub-grafo del grafo completo obtenido en La Figura 4, debido a las posibilidades reales en cuanto a recursos de computador. Se dibujan en rojo los nodos que pertencen al conjunto analizado y se dibuja lo más fiel posible a la realidad la topología que se maneja. Se dibuja además el tamaño de los nodos acorde al número de conexiones reales que presenta.

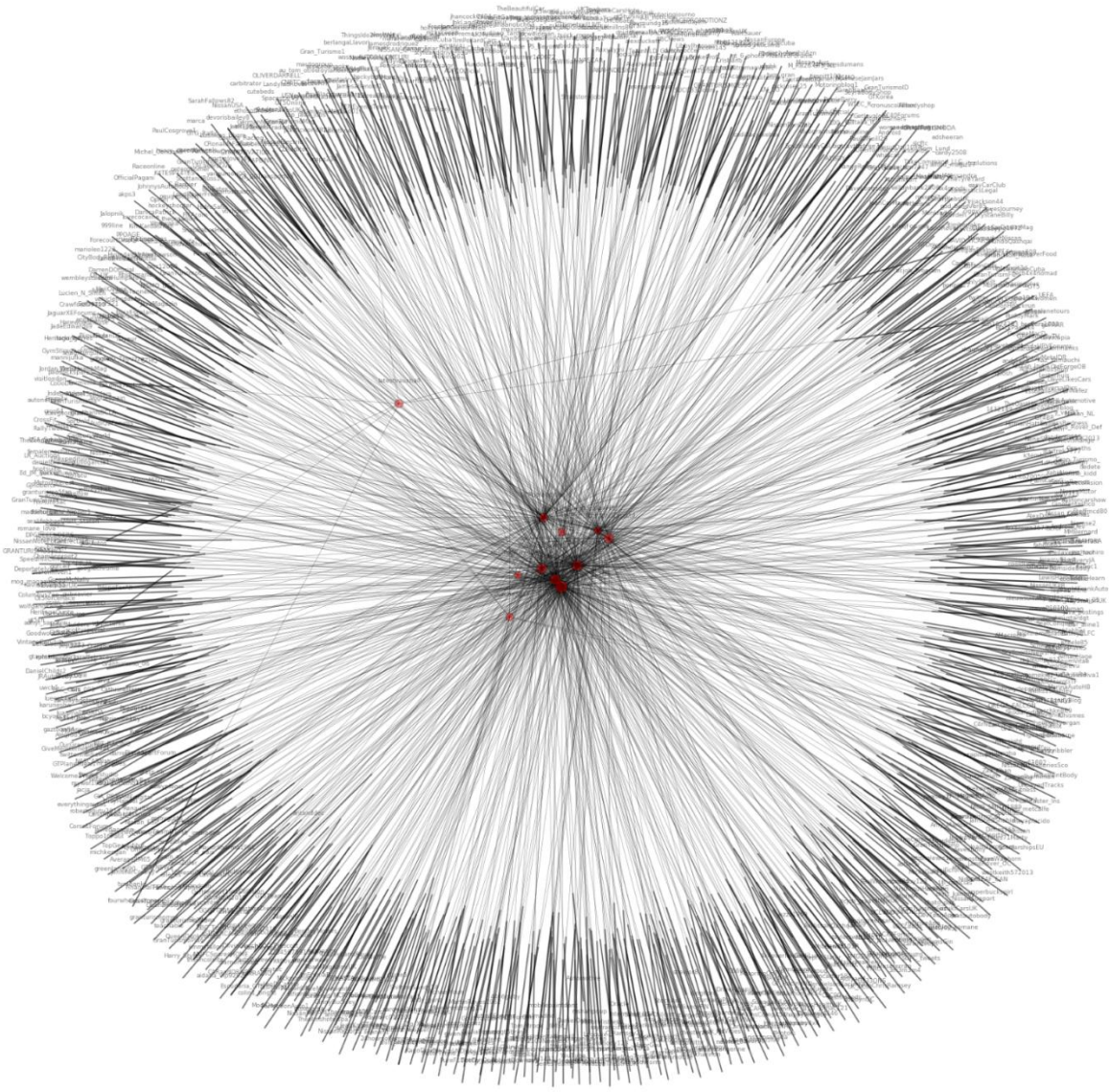


Figura 4. Visualización de un subconjunto de nodos del grafo construido.

La siguiente visualización (Figura 5) que se obtiene del grafo anterior, se enfoca en el uso de la medida *Retweets*. Esta se muestra de gran utilidad para que los usuarios permitan a sus seguidores saber que participan activamente con su marca o tema mediante la re-publicación de su contenido. (Un *retweet*, es un *repost* de un *Tweet* enviado por otro usuario). Estos *Tweets* están marcados con el icono de *retweets* (RT) e incluyen la información del autor, y el nombre del usuario que ha retweetado el contenido. Son una de las herramientas más usadas en Twitter, y puede ser muy útil en la identificación de tendencias web y el contenido que interesa a sus lectores o sus seguidores. Los *retweets* logran además identificar unívocamente *Tweets* que tienen la capacidad de ser virales.
Este grafo aporta una medida muy efectiva de la propagación real de un tema, y por tanto el alcance real para nuestro conjunto de estudio. Dicha métrica será de hecho una referencia en el estudio de los enlaces entre usuarios, ya que logra aportar la cuantía en que un enlace puede ser más fuerte que otro, y por tanto una evidencia importante para la identificación de comunidades [23].

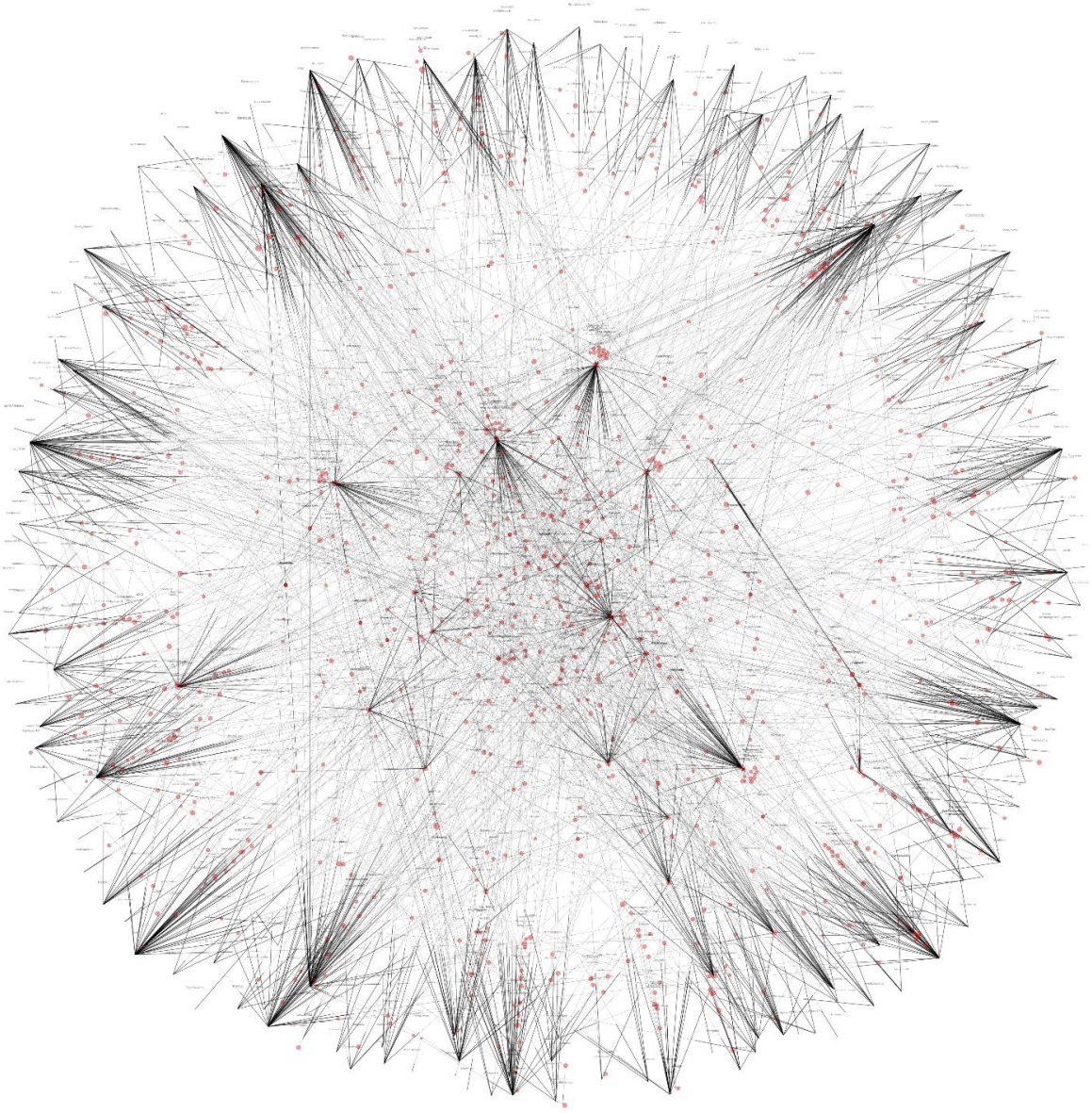


Figura 5: Subgrafo determinado por los *Retweets* (RT) en Twitter.

Se muestra en último lugar, en la Figura 6, un subgrafo obtenido del grafo completo que se ilustra en la Figura 4, el cual será la base para los análisis que prosiguen en este proyecto. Se eliminan en este nuevo subgrafo elementos conectados mediante una sola arista, que solamente añaden ruido al conjunto de muestra y a la hora de ejecutar los algoritmos de detección de comunidades. Además se realiza un filtrado por amigos mutuos, mediante la cual se obtiene el subgrafo que logra contener para todos los nodos, tanto el perfil de *follower* como el de *followed*. La Figura 6 refleja el grafo con las características comentadas anteriormente. Este nuevo grafo es la base para la ejecución de los

algoritmos de detección de comunidades, y los posteriores análisis sobre los resultados obtenidos, documentados a partir del apartado 4.4.

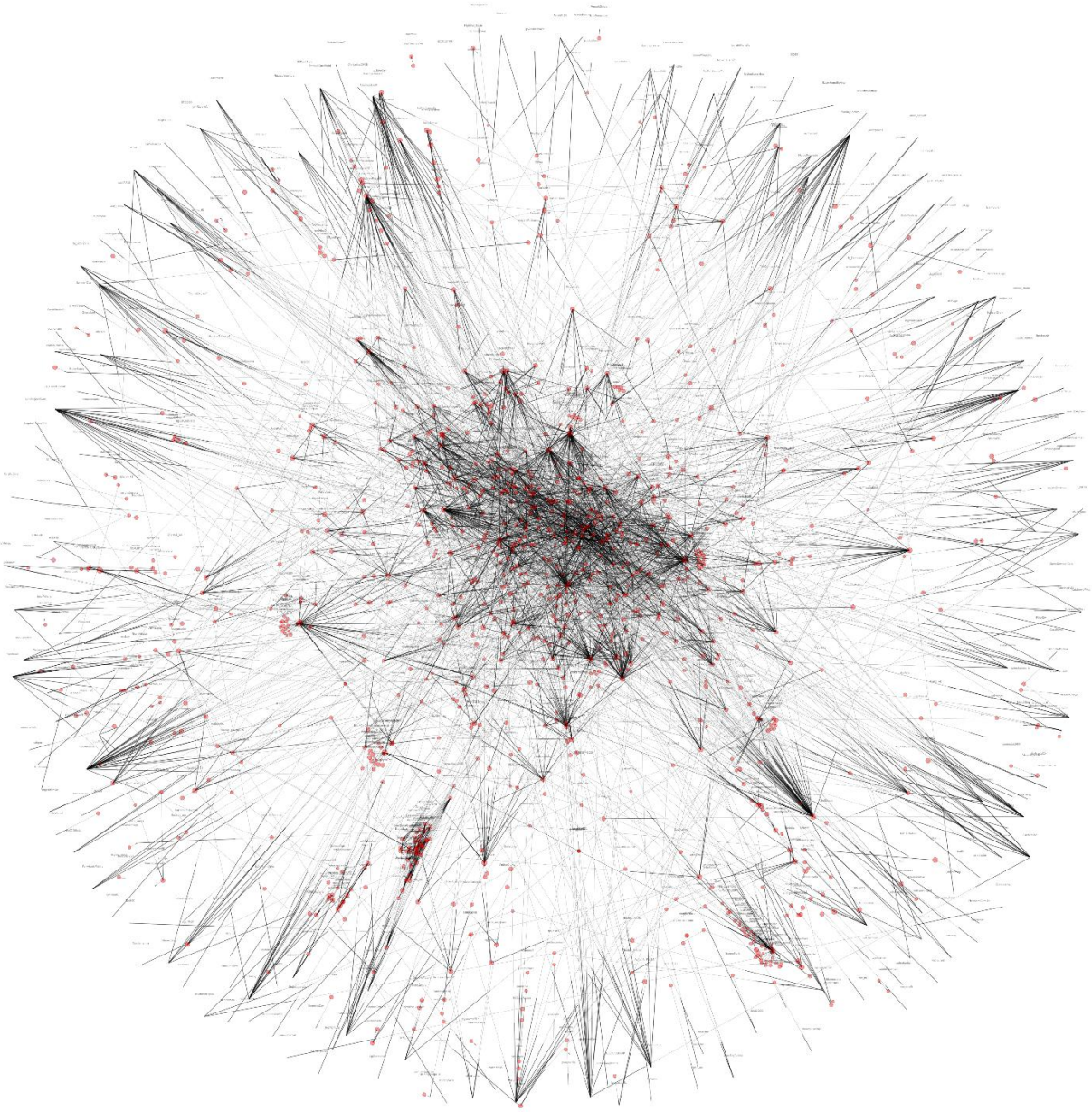


Figura 6: Grafo de amigos mutuos extraído de Twitter.

4.3.1 Visualización de usuarios por su geo-localización

Para lograr el análisis espacial fue necesario realizar algunas consultas extras a la API de Twitter y construir algunos scripts de Python que se explican a continuación. Para lograr la visualización que se pretende en este apartado, se presenta a la API de Google como una herramienta muy útil debido a sus enormes potencialidades. Mediante su API se logra convertir una calle, una ciudad, un parque, etc. en un sistema de coordenadas que puede ser ubicado geográficamente. Por ejemplo, Google logra para el usuario, 'RichmondCarsLtd' que solo presenta en su campo dirección “chesir, hyde”, corregir errores de escritura y devolver en un sistema de coordenadas el punto exacto al que se refiere; se obtiene para el caso particular, “Cheshire Cheese, 407 Stockport Rd, Hyde SK14 5RY, UK”, con latitud y longitud 53.4362193,-2.0756514 respectivamente. En resumen, se muestra la ubicación geográfica a continuación en el visualizador del mundo construido en D3.js de un 28.45% de los usuarios totales del conjunto inicial (ver Figura 7).

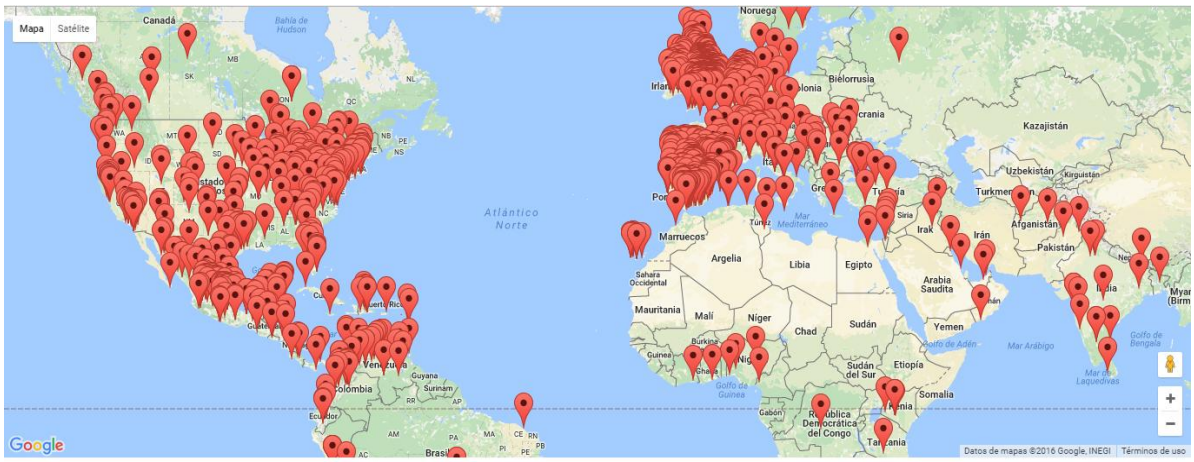


Figura 7. Usuarios geo-localizados mediante GEO-Google.

Para esta nueva parametrización no se amplía por el momento demasiado en su análisis, hasta este punto del proyecto se integran a SLOD-BI los resultados obtenidos. Sin dejar de mencionar que la geolocalización es una métrica muy interesante para integrar en análisis de sentimientos, y en comparaciones que se realizan con los resultados obtenidos de los métodos de detección de comunidades y se encuentran documentadas en el epígrafe 4.6.

Para la visualización que se muestra en la Figura 7 se pueden verificar zonas de densidad de *Tweets* considerablemente altas. Se presenta un ejemplo de como los *Tweets* y su densidad pueden ser usados para determinar ubicaciones específicas relevantes, y determinar el grado de interés que debería mostrar una compañía en una zona determinada. En resumen, esta métrica añadida a SLOD-BI incorpora al análisis de sentimientos en los datos el estudio de posibles zonas de conflicto para algún tipo de problema.

Para ubicar el área donde se desarrolla el análisis, se muestra a continuación para España, de los *Tweets* recogidos, su ubicación geográfica (ver Figura 8). Se compara en la figura la información ya recogida, con la distribución espacial de los concesionarios de la firma Opel para confirmar la estrecha relación en densidad de concesionarios y *Tweets* en el tema. Se muestra en términos generales mediante este breve análisis la extrapolación real que tienen las redes sociales en la actualidad y el valor que pueden llegar a alcanzar análisis en las mismas.

Figura 8: Correlación entre talleres, concesionarios y servicios en general de Opel (https://www.opel.es/tools/concesionarios-opel.html) con geo-localización de usuarios en España para el Opel Astra.

En términos generales el descubrimiento de la información geo-espacial proporciona un conjunto de medidas que poco a poco se deben incorporar a empresas y sobre todo a aquellas de escala global, donde dichos análisis aportan un valor inconmensurable.
Para terminar, se muestra otro ejemplo de los análisis que pueden llegar a realizarse espacialmente. Se toman los usuarios del área de Barcelona y para ellos se construye un mapa de palabras mostrado en la Figura 9, que nos da la posibilidad de observar a simple vista los tópicos de interés para esta área geográfica.

Figura 9: Palabras frecuentes para los usuarios en la zona de Barcelona.

4.4 Detección de Comunidades.

La detección de comunidades describe el proceso de tomar colecciones de objetos como usuarios o *tweets*, con una ubicación aproximada dentro de un radio especificado, y organizarlos mediante su similaridad en grupos. La organización en grupos debe consistir en que objetos similares se agrupen

en un mismo clúster mientras que los diferentes clústeres o grupos se diferenciarán en contener objetos que no contienen similaridad o muy poca hacia otros grupos. El principal problema para el *clustering* de comunidades son las propias definiciones de sus elementos, debido a que ni los conceptos de comunidad o partición se encuentran rigurosamente definidos y requieren un cierto grado de arbitrariedad y sentido común a la hora de aplicarse a un conjunto de datos. Es importante hacer hincapié en que la identificación de los grupos estructurales sólo es posible si los gráficos están lo suficientemente esparcidos, es decir, si el número de aristas (m) es del orden de la cantidad de nodos (n) del grafo ya que para m >> n, la distribución de los aristas entre los nodos es demasiado homogénea para que las comunidades logren tener un sentido. Antes de adentrarse en los algoritmos de agrupamiento para los grafos, es importante discutir el contexto en que se realizan los análisis [6].

Muchos algoritmos han sido desarrollados usando herramientas y técnicas de otras disciplinas como la física, la biología, las ciencias sociales, así como la matemática aplicada en la teoría de grafos, que es la base de los algoritmos que en este proyecto se utilizan. Se destacan en la detección de comunidades los de tipos divisivos, que detectan enlaces entre intercomunidades y eliminan dichos enlaces de la red. Los algoritmos aglomerativos, por su parte, manejan el entorno mediante la fusión de nodos/comunidades similares de forma recursiva y los métodos de optimización, para concluir, que basan su funcionamiento en la maximización de una función objetivo y como se comprobará más adelante, la muy importante modularidad. La modularidad logra medir las particiones resultantes y medir en un rango de -1 a 1 la densidad de enlaces dentro de las comunidades con respecto a los vínculos entre las comunidades, lo cual será la medida más transcendente que se tomará para comparación y según la cual principalmente se medirán los resultados obtenidos en los siguientes subepígrafes.

La noción de comunidad es difícil de definir formalmente. Muchas definiciones se han propuesto en los estudios de redes sociales, pero son demasiado restrictivas o no se pueden calcular de manera eficiente. Los enfoques más recientes sin embargo han llegado a un consenso, y consideran que una partición P = {C1,. . . , Ck} de los vértices de un grafo G = (V, E) ($\forall i, C_i \subseteq V$) representa una buena estructura de la comunidad si la proporción de bordes dentro de los bordes internos Ci es alta en comparación con la proporción de bordes entre ellos, lo cual viene a confirmarnos la indudable certeza en cuanto a comparar la eficiencia mediante la modularidad.

La modularidad, en esencia, es la fracción de los enlaces que caen dentro de grupos dados menos el valor esperado que dicha fracción hubiese recibido si los enlaces se hubiesen distribuido al azar. El valor de la modularidad se encuentra en el intervalo [-.1,1). Existen diferentes métodos para el cálculo de la modularidad. En la versión más común del concepto, la aleatorización de los enlaces se realiza con el fin de preservar el grado de cada vértice. Sea un grafo con N nodos y M enlaces, de tal manera que el grafo se puede dividir en 2 comunidades usando una variable miembro S. Si un nodo i pertenece a la comunidad 1, Si=1 o si i pertenece a la comunidad 2, Si=-1. La matriz de adyacencia de la red estará representada por A, donde Aij =0 significa que no hay ninguna interacción entre los nodos i y j y Aij = 1 significa que hay un enlace entre los dos. También por simplicidad consideramos una red no dirigida, así Aij = Aji. La modularidad Q se define entonces como la fracción de enlaces que caen dentro del grupo 1 o 2, menos el número esperado de enlaces dentro del grupo 1 y 2 para un grafo aleatorio con el mismo grado de distribución como el nodo de red determinado (Fórmula 1).

$$Q = \frac{1}{2m} \sum_{ij} \left[A_{ij} - \frac{k_i * k_j}{2m} \right] \frac{s_i s_j + 1}{2} \quad (1)$$

Es importante tener en cuenta que la Fórmula 1 es válida para la partición en sólo 2 comunidades. Partición jerárquica (es decir, partición en 2 comunidades, a continuación, las 2 sub- comunidades con particiones más en 2 comunidades más pequeñas sub sólo para maximizar Q) es un posible enfoque para identificar múltiples comunidades en una red [5]. Pero puede ser generalizada para la partición de una red en comunidades c de la siguiente forma: (Fórmula 2).

$$Q = \sum_{ij} \left[\frac{A_{ij}}{2m} - \frac{k_i * k_j}{(2m)(2m)} \right] \delta(c_i, c_j) = \sum_{i=1}^{c} (e_{ii} - a_i^2) \quad (2)$$

Eii es la fracción de los enlaces con ambas vértices finales de la misma comunidad i (Fórmula 3):

$$e_{ii} = \sum_{j} \frac{A_{ij}}{2m} \delta(c_i, c_j) \quad (3)$$

y Ai es la fracción de enlaces con al menos un extremo de vértice en la comunidad i (Fórmula 4).

$$a_i = \frac{k_i}{2m} = \sum_{j} e_{ij} \quad (4)$$

Para el proyecto la referencia en cuanto al análisis de los algoritmos implementados y usados es sin lugar a dudas la modularidad por las razones expuestas, aunque la inclusión de un análisis visual de los resultados obtenidos y encontrar otras fuentes para arribar a conclusiones es un propósito también ante la siempre latente falta de confiabilidad en la definición de comunidad. Por otra parte, para cada investigación habitualmente es generado un modelo simple de red, llamado modelo de partición de muestra, donde según los resultados obtenidos sobre dicho modelo se intentan caracterizar los algoritmos desarrollados hasta la fecha [5]. De dichos trabajos se enriquece el aquí redactado, al hacer uso explícito de algunos de los resultados obtenidos en investigaciones previas, razón por la cual se desechan algunos algoritmos que no logran escalar y demostrar un funcionamiento apropiado para redes muy extensas como la propuesta en este trabajo. Algoritmos desechados son: *Markov Cluster Algorithm,* debido a su alta dependencia de un parámetro alfa que es altamente difícil definir a priori para redes sociales y ante su lenta escalabilidad para este tipo de redes*,* o el *Spectral algorithm by Donetti and Muñoz,* debido a que es un algoritmo expresamente construido para redes con pocos vectores propios, mientras que las redes que se analizan muestran características muy distantes a las ideales para este método en especial, entre otros mencionados en [37]. En resumen y tomando en consideración como ya se comentaba una de las pocas ideas compartidas por los analistas del tema, donde se plantea que una comunidad como propiedad principal característica debe tener que sus nodos internos presenten más conexiones entre ellos que con nodos externos, y, por supuesto, de

diferentes comunidades, en otros términos modularidad, se comienza el análisis de los métodos de detección de comunidades que comienza a continuación.

4.4.1 Algoritmo de Girvan y Newman

Es el primero de los algoritmos para la detección de comunidades en grafos propuestos. Girvan y Newman se clasifica dentro de los algoritmos de *clustering* jerárquicos y es a su vez un método divisivo en el cual los enlaces son iterativamente eliminados según el valor de *betweenness* que presenten. Este valor expresa el número de caminos mínimos entre pares de nodos que pasan sobre un mismo enlace. Para lograr hacer agrupamientos sobre un grafo G = (V,E), Newman y Girvan en su algoritmo asignan pesos a las aristas basado en las propiedades estructurales del grafo G. Dicha idea se encuentra sustentada por el concepto *node-betwenness* definido por Freeman [10] para estudios sociológicos. Newman y Girvan [11] definen los pesos como el número de intermediarios de una arista, lo cual es el número de caminos más cortos que conectan a cualquier par de vértices que pasan a través de la arista.

Girvan y Newman como idea central asumen que las aristas con un gran valor *betweenness* son enlaces entre grupos y posibles enlaces entre comunidades, mientras que las aristas de menor valor son aristas que conectan miembros dentro de un clúster. Por tanto, propone separar la red en grupos mediante la separación una a una de las aristas con mayor grado de *betweenness*. Si más de una arista contiene el mayor valor de intermediación o *betweenness* se elige de forma aleatoria una a ser eliminada. La eliminación consiste en el paso que antecede a la re-calculación de los valores de intermediación, lo cual se realiza hasta lograr separar la red en comunidades o clústeres. Su coste computacional se manifiesta de forma polinomial con un O(n*m) y es el algoritmo descrito, el primero de su clase en introducir la métrica de modularidad, que más tarde sería adaptada como consenso en la comunidad científica [26].

Las salidas que se obtienen para el algoritmo no resultan buenos en ninguna de las ejecuciones. Para el conjunto de datos en cuestión no se logra obtener comunidades que resulten a la vista útiles. Se obtiene una modularidad de 0.99, que en teoría es ideal, pero resulta engañosa ya que las comunidades son de solamente pares de vértices. Girvan & Newman descubre en el grafo analizado unas 444 comunidades, que visualmente son muy contrastables con el dendrograma que se construye y que en su gráfica nos ilustra del mal funcionamiento del algoritmo para la muestra (ver Figura 10).

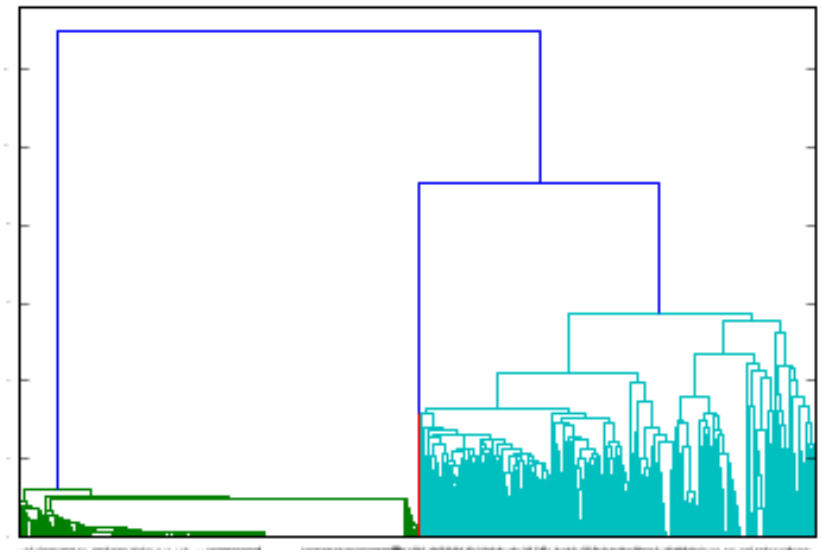


Figura 10: Dendrograma para el algoritmo Girvan-Newman.

En la Figura 10 se propone una posible separación en comunidades entre 3 y 6, pero nunca mediante separar todo el conjunto en pares como máximo número de nodos en una comunidad, lo cual daría lugar a las 444 comunidades. Los resultados alcanzados por dicho algoritmo para la detección de comunidades se encuentran en los ficheros que acompañan a este informe, sin dejar de mencionar que ante tan bajo rendimiento no son integrados a SLOD-BI.

4.4.2 *Fast modularity optimization* o Multinivel por Blondel

El algoritmo de optimización rápida de la modularidad es una técnica que se maneja de forma heurística. Se basa en varios pasos y en la optimización local de Newman-Girvan hacia las cercanías de cada nodo, método abordado en el apartado 4.4.1. *Fast Modularity by Blondel* realiza una partición inicial como su primer paso, y las comunidades que resultan de dicha división son remplazadas por supernodos y se obtiene en consecuencia una red de menor peso. Este proceso descrito se repite iterativamente hasta encontrar que la modularidad no logra incrementarse. Este método ofrece un compromiso justo entre la precisión de la estimación de la máxima modularidad, que es mejor que la entregada por técnicas ávidas, y la complejidad computacional, que es esencialmente lineal en el número de enlaces del grafo.

Es conocida también la técnica descubierta en [24] como método de agregación multinivel, y se decide aplicar al conjunto de datos porque demuestra mejorar en rendimiento a todas las técnicas existentes hasta el momento de su publicación en términos de tiempo computacional, mientras logra presentar una calidad en las comunidades detectadas muy buena. En un segundo acápite el algoritmo multinivel es seleccionado por su desempeño constatado en la red de móviles de Bélgica, donde para un conjunto de 2.6 millones de clientes y tras analizar un grafo de 118 millones de nodos y más de un billón de enlaces, logra mantener una exactitud envidiable para un conjunto de datos tan extenso, siendo multi-

avalado para nuestro conjunto de datos representado mediante grafos. Algunas mejoras se realizan en el rendimiento de dicho algoritmo, mediante el uso de algunas heurísticas simples. Se detiene la primera fase del algoritmo cuando la ganancia de la modularidad es inferior a un umbral dado, o por la eliminación de los nodos de grado 1 (hojas) de la red original y añadiéndolos de nuevo después de que el cálculo de la comunidad se haya completado. Varias de estas mejoras se incluyen en nuestros scripts y los resultados son muy buenos. Se muestran estos resultados en la Figura 11 a continuación, que documenta también el tiempo que consume el algoritmo para el *dataset*, lo cual evidencia la enorme escalabilidad y rapidez con que se manejan estos datos.

```
Number of communities detected: 91
modularity: 0.70200416584
Algorithm Multilevel takes 0.451000 seconds in run
```

Figura 11: Rendimiento y resultados del algoritmo Multinivel

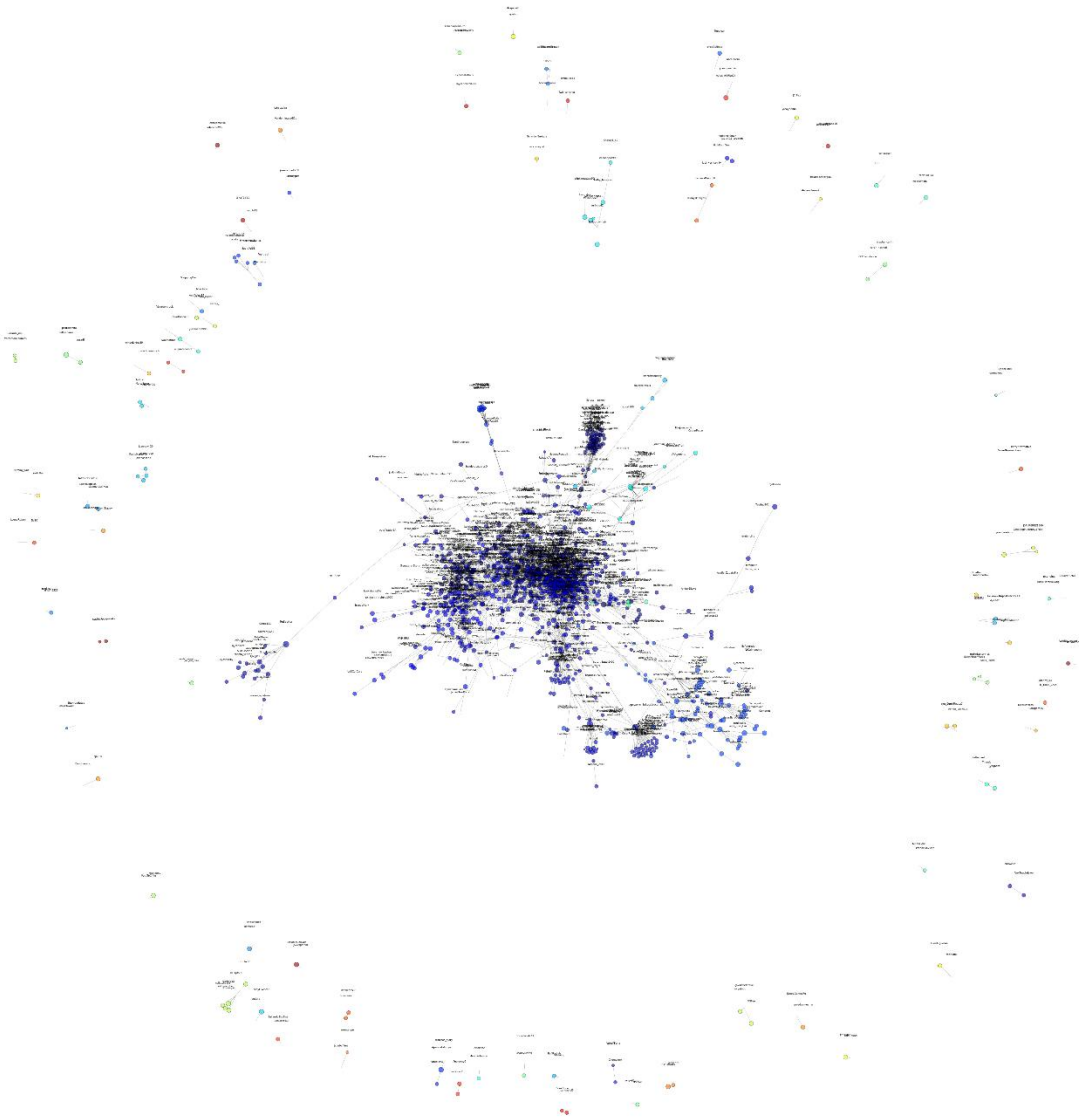


Figura 12: Visualización de las comunidades detectadas por el algoritmo Multinivel.

La distribución de comunidades se presenta en la Figura 12 y se recoge para ellas un valor de modularidad de 0.702, un muy buen resultado, pero que se intenta mejorar mediante el análisis del dendrograma que el algoritmo dibuja. Se parametriza el algoritmo expuesto para el número de clústeres que se detectan mediante el análisis del dendrograma, pero no se consigue mejorar el valor de modularidad en ninguno de los casos del valor ya obtenido. En resumen se divide el grafo en 91 comunidades, en su mayoría agrupaciones aisladas con escasa población, con un conjunto central estrechamente relacionado, que son sin duda las comunidades que se interesan investigar en análisis que más adelante se documentan, epígrafes a partir del 4.5.

Para el algoritmo además se crea un archivo de texto (*Dendrogram Louvain Method*.txt) que contiene el dendrograma que se construye y se utiliza para intentar mejorar la eficiencia en el algoritmo expuesto. El dendrograma por sí mismo describe una distribución del conjunto en dos particiones que no logran confrontar los resultados obtenidos ya que estos coinciden en su análisis de comunidades sobre el grafo y tras pruebas realizadas solo confirman los resultados obtenidos inicialmente. Se establece en resumen el primero de los resultados inalterado y considerado lo suficientemente óptimo para ser tomado en cuenta y comparado con los siguientes métodos que se analizan en este sección 4.4.

4.4.3 La ecuación del mapa o Infomap por Rosvall y Bergstrom

Es un método que reconoce los enlaces de la red como inducidores de movimiento a través de la estructura del grafo y por tanto son quienes dan lugar a inter-dependencias en todo el sistema. Dicho método, presentado por Rosvall y Bergstrom, reconoce que toda red transporta un flujo y pasa a resaltar y simplificar la estructura en el grafo con respecto a este flujo. Utiliza en su implementación lo que los autores definen como la Ecuación del Mapa o *Infomap* [2[]5].

Este algoritmo difiere de los anteriores, sección 4.4.1 y 4.4.2, enfocados en la maximización de la modularidad, debido a que realiza un enfoque prioritario a los patrones del flujo presentes en el grafo, mientras que los algoritmos hasta ahora analizados hacen pasar por alto esta característica. Infomap por tanto aporta resultados dramáticamente diferentes para algunas estructuras de red, principal objetivo de su utilización en el proyecto aquí presentado. En resumen y hasta ahora común en todos los algoritmos documentados, se propone el análisis de la modularidad como valoración inicial de los resultados alcanzados. Se calcula además el tiempo real necesitado en su ejecución lo cual al ser integrado en SLOD_BI y ejecutado para nuevos conjuntos de datos se convierte en una medida relevante del sistema para con el usuario final.

A continuación se muestran distribuidas en colores las comunidades resultantes en Infomap, agrupando en el centro de la gráfica un conjunto de comunidades localmente muy densas y a su alrededor otras pequeñas y dispersas en el grafo (ver Figura 14).

Para nuestro conjunto de datos se obtienen los siguientes resultados, Figura 13:

```
Found a total of: 221 communities
Modularity: 0.426382416487
Algorithm Infomap takes 8.195000 seconds in run
```

Figura 13: Rendimiento y resultados para el algoritmo Infomap

Se recogen un total de 221 comunidades, que varian en diferentes ejecuciones del algoritmo en un rango de 12 agrupaciones, siendo 221 una media para estas ejecuciones y un buen ejemplo para el análisis y la representación mostrada en la Figura 14. Infomap muestra un rendimiento para la medida de modularidad de 0.426, lo cual para un conjunto tan disperso es un muy buen resultado para con las comunidades descubiertas. EL algoritmo en su ejecucion mostrada demora un total de 8.2 seg en ejecución, lo cual para un conjunto de aproximadamente 10.000 *tweets*, y un número tan extenso de conexiones entre los nodos es un resultado aceptable.

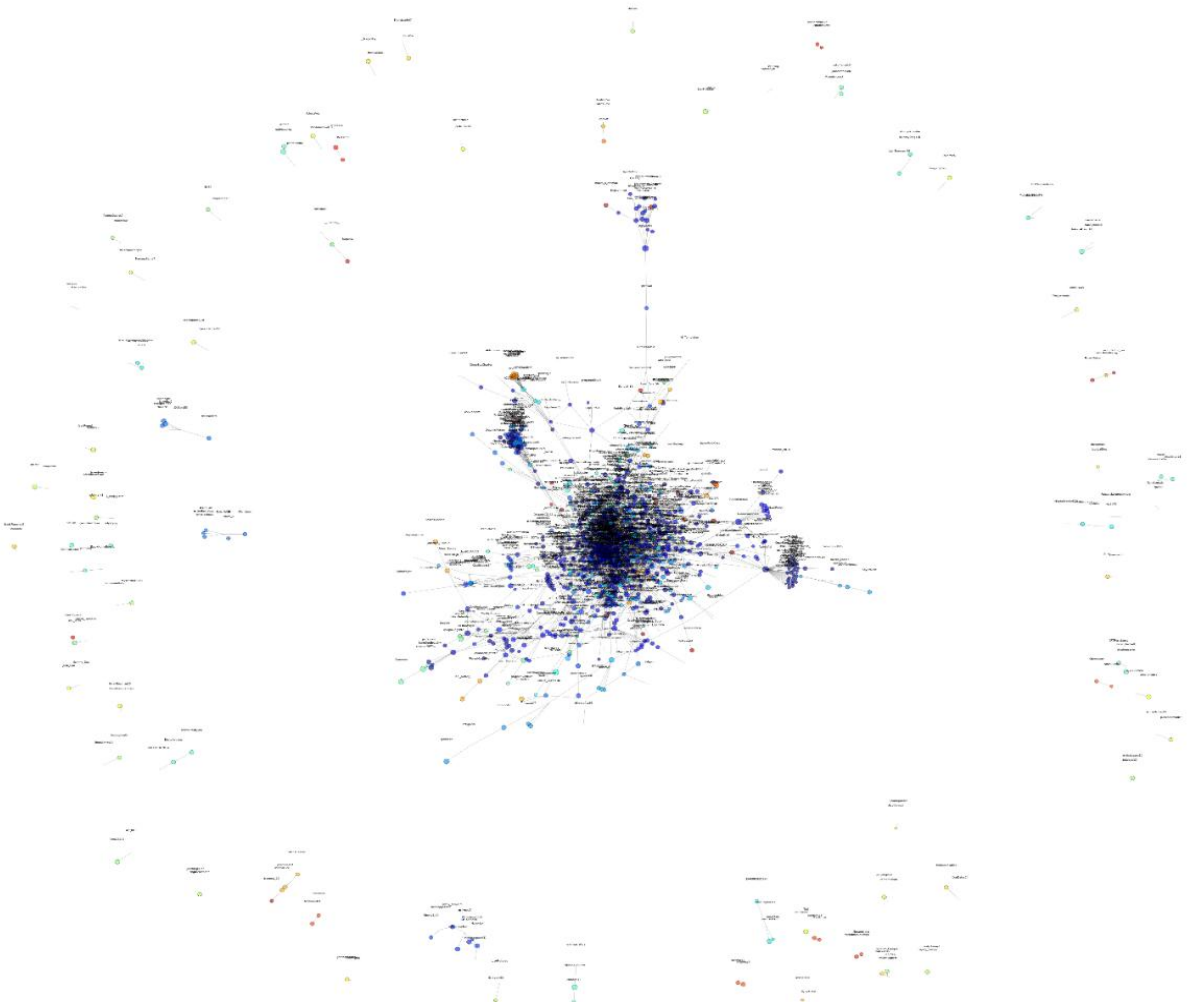


Figura 14: Comunidades detectadas por Infomap.

4.4.4 Algoritmo de Walktrap.

El siguiente método propuesto es el algoritmo Walktrap, que se presenta como uno los más usados para hallar subgrafos densos en grafos dispersos (comunidades), principal interés en el trabajo. Dicho método propone un nuevo enfoque mediante una nueva distancia entre los vértices que cuantifican su similitud estructural utilizando paseos aleatorios [29]. La distancia en que se basa Walktrap tiene varias ventajas. Principalmente captura mucha información sobre la estructura de la comunidad y se puede utilizar en un algoritmo de aglomeración jerárquica eficiente que detecta comunidades en una red.

Es un algoritmo para la detección de comunidades en grafos altamente dispersos que logra trabajar en el peor de los casos para un coste computacional de O (mn^2). En la práctica, las redes complejas del mundo real son escasas (m = O (n)) y la altura del dendrograma es pequeña (H = O (log n)); en este caso, el algoritmo logra ejecutarse en O (n 2 log n). Extensos experimentos realizados muestran que el método ofrece buenos resultados en diversas condiciones (tamaños del grafo, densidades y número de comunidades) y por tanto el enfoque propuesto en teoría muestra tener una clara ventaja en términos de calidad de la partición computarizada y presentar el mejor compromiso de todos los algoritmos estudiados hasta el momento entre la calidad y el tiempo de funcionamiento para redes grandes. No obstante, tiene la limitación de requerir una cantidad bastante grande de memoria [29], lo que hace que la escalabilidad rápida se muestre como un problema relevante para este método en gráficos muy grandes (millones de vértices). Dicho método se presenta principalmente para ser integrado en una herramienta de visualización multi-escala para grandes redes y de relevancia para el cálculo de comunidades de solapamiento (que a menudo se producen en los casos de la vida real), caso totalmente posible para el conjunto de datos que se analiza, al ser una subred de Twitter de carácter real y para la cual el esparcimiento de sus datos y el posible solapamiento en sus comunidades es una característica latente.

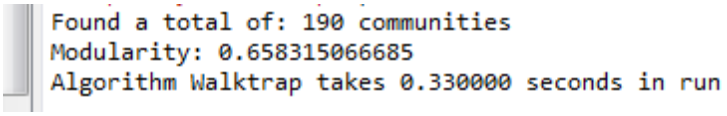


Figura 15: Rendimiento y resultados del algoritmo Walktrap

El algoritmo detecta un número de comunidades bueno y una modularidad excelente para los grupos detectados, con una velocidad para con los datos de un tercio de segundo lo cual es un muy buen resultado (ver Figura 15). Se considera la velocidad en nuestro trabajo como un factor realmente importante en nuestros resultados debido al análisis que tendrán que hacer dichos algoritmos en SLOD-BI sobre grandes conjuntos de datos y para los cuales el desenvolvimiento del algoritmo y la escalabilidad sobre los datos es fundamental.

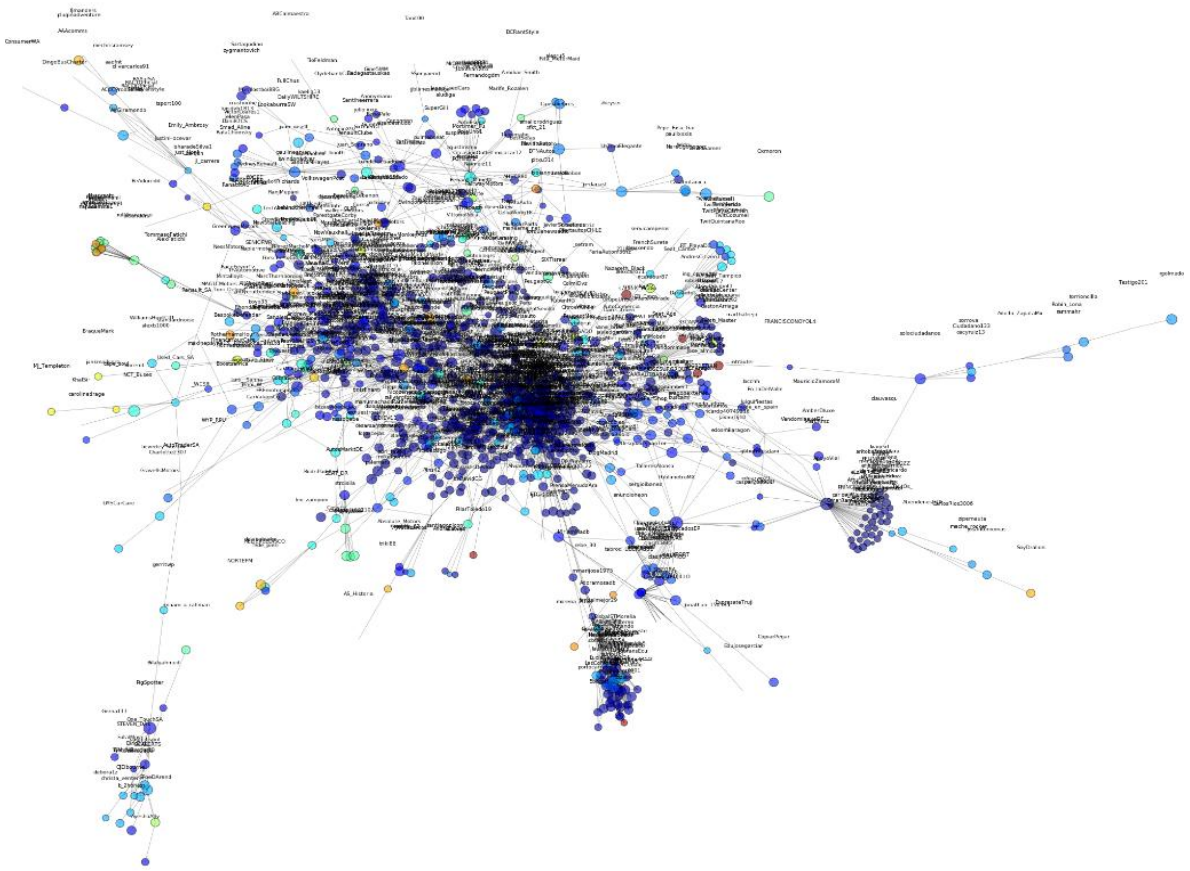


Figura 16: Comunidades detectadas por el algoritmo Walktrap.

En la Figura 16 se muestran dibujados los nodos mediante los colores de las comunidades que los identifican; se obtienen para el algoritmo Walktrap unas 190 comunidades con un conjunto central altamente conectado como en los casos anteriores (ver secciones 4.4.2 y 4.4.3) y el resto dividido en comunidades dispersas con poca presencia de nodos. Se construye para el algoritmo un dendrograma, pero en general este maneja un tipo de particionado muy uniforme: construye dos particiones en todos los casos y se desecha la posibilidad de hacer test a modo de prueba y error tras la experiencia obtenida en 4.4.3. Los resultados obtenidos en el dendrograma en resumen son muy similares a los reflejados en el grafo que se dibuja en la Figura 16, y en resumen solo confirman los resultados ya obtenidos.

En último lugar se propone ante la semejanza en los resultados obtenidos un epígrafe de conclusiones parciales para todos los algoritmos propuestos, apartado 4.4.5. Se continúa el análisis mediante aspectos que aquí se dejan de lado, como la semántica de los nodos y las principales características que manifiestan semánticamente las comunidades detectadas, como nodos centrales o los principales intereses manejados en las comunidades.

4.4.5 Conclusiones Parciales

Para los algoritmos utilizados se construye en cada uno de ellos un grafo distribuido en comunidades y en ellos se aprecia el tipo de topología latente en las redes sociales. Se describen comunidades o subcomunidades que forman agrupamientos de nodos altamente conectados y una serie de comunidades poco pobladas y dispersas. Estos resultados son corroborados con la creación de los dendrogramas. Se resalta tras los análisis realizados el objetivo primario siguiente, el volcado de los mejores resultados a SLOD-BI y hallar las métricas que infieren conocimiento y dan lugar a futuros análisis sobre los datos. Para ello se analizan los resultados de forma global en la tabla 3.

Algoritmos de detección de comunidades	Comunidades detectadas	Modularidad	Tiempo de ejecución(segundos)
Girvan y Newman	444	0.99	1.486
Multinivel	91	0.702	0.451
Infomap	221	0.426	8.19
Walktrap	190	0.658	0.330

Tabla 3 Resultados generales de los algoritmos de detección de comunidades.

Se selecciona el método Multinivel de Blondel por los resultados mostrados en la tabla anterior, que superan de forma clara a los demás algoritmos implementados. Por tanto, los análisis posteriores en el documento y los datos que se integran a SLOD-BI son los que el algoritmo comentado nos ofrece (ver Figura 12).

Se descubre que se establece una relación muy estrecha entre algunos atributos destacados en los métodos de detección, especialmente en el método Multinivel, que nos aporta atributos como Betweenness Centrality y Eigenvector Centrality. Se demuestra que estos atributos delatan en nuestro conjunto a los nodos pertenecientes a entidades como compañías o páginas web dedicadas al marketing de marcas de coches.

Se realiza una visualización de centralidad para los nodos con mayor Betweenness centrality en la Figura 17, para ilustrar lo comentado.

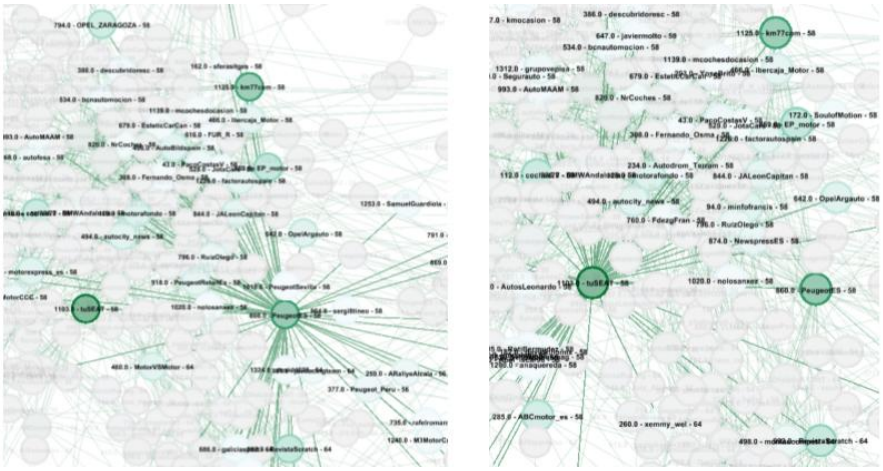


Figura 17: Visualización filtrada en tonos de verde de Betweeness Centrality

Los nodos presentes en la Figura 17 son:

@tuSEAT: Cuenta Oficial de SEAT España en Twitter.

@PeugeotES: Cuenta Oficial de Peugeot España.

Otros nodos con altos valores de centralidad intermedia que no se muestran en la Figura 17 anterior se nombran y especifican a continuación:

@motorpasion: Cuenta que sigue la actualidad del mundo del motor. Salón del automóvil, presentaciones, marcas de coches y análisis varios en el mercado.

@diariomotor: Cuenta oficial de Twitter de la página http://diariomotor.com/.

@km77com: Cuenta oficial de Twitter de la página http://www.km77.com, especializada en información de motor.

@EP_motor: Cuenta oficial de Twitter de la página dedicada a la actualidad del mundo del motor en Europa Press, la agencia privada de prensa española.

@Peugeot: Cuenta oficial de Twitter de la marca francesa de coches.

@TodoMotorVe: Cuenta de Twitter de la revista de autos Todo Motor.

@Opel_Spain: Cuenta Oficial de Twitter de Opel España.

@autopista_es: Cuenta de Twitter para la página web dedicada al automóvil en España.

Se verifica en todos los nodos con altos valores del atributo analizado la característica de pertenecer al perfil de una entidad, cuentas involucradas en escenarios de marketing en la red con objetivos como venta de productos o el mayor impacto de las noticias que difunden. Por tanto, en todas las comunidades detectadas los nodos centrales de las mismas son los principales causantes de la estructura de comunidad a su alrededor. Se demuestra que estas comunidades no son resultado del puro azar, ya son estos nodos quienes manejan el flujo de información en la red y quienes se plantean como objetivos principales mantener a los usuarios ya presentes en su comunidad. Objetivo primario de los nodos centrales en las comunidades es además por todos los medios posibles extender su comunidad para así poder tener un mayor potencial impacto y por consiguiente un incremento en el impacto real que tienen sus *tweets* en la web [7].

Para continuar ampliando en los resultados obtenidos por los algoritmos aplicados se crea un nuevo script de Python que muestra un ejemplo de análisis en el estudio de comunidades online. Este nuevo fichero realiza una búsqueda dentro de la distribución de comunidades y a la comunidad más poblada le halla en sus *tweets* los textos almacenados y sobre ellos se realiza una nube de palabras para permitir observar los temas relevantes y más discutidos en el grupo. La comunidad más poblada descubierta se verifica, ver Figura 18, que realmente se interesa en el tema y discute activamente temas relacionados con coches y en especial con el Opel Astra. Es en resumen una comunidad altamente activa y que se puede marcar como una comunidad factible para realizar campañas de marketing, ya que al ser tan altamente conectada e interesada en la temática con detectar los usuarios más influyentes en ella se inundan todos sus seguidores con la información que nos interesase mostrarles.

Figura 18: Nube de palabras obtenida con los servicios de http://tagcrowd.com/

Se demuestra con un ejemplo muy sencillo en la Figura 18 el tipo de aplicabilidad de este tipo de análisis, sin dejar de mencionar otras posibles aplicaciones relevantes, como el caso de redes metabólicas donde el descubrir comunidades en ella corresponde a las funciones biológicas de las células, por tan solo mencionar un uso más allá de los tópicos de interés para redes sociales.

4.5 Resultados y análisis en Gephi y Knime

Para ampliar los resultados expuestos se proponen dos herramientas de extensivo uso profesional para el análisis de grafos en redes sociales. La primera de ellas, Gephi, es una herramienta *open-source* desarrollada en Java para visualizar y analizar grandes gráficos de red. Sus principales funciones son la exploración y ayudar a entender redes complejas a analistas de datos. Permite la verificación de hipótesis y descubrir patrones ocultos en los grafos. Knime por su parte, o *Konstanz Information Miner* es una plataforma de minería de datos que permite el desarrollo de modelos en un entorno visual. Fue desarrollada originalmente en el departamento de bioinformática y minería de datos de la Universidad de Constanza, Alemania y desarrollada sobre la plataforma Eclipse, esencialmente, en Java [3], más detalles en el apartado 4.1.

Se procede a hacer uso de ambas herramientas. En primer lugar Gephi, además de permitir un visualizado altamente profesional, hace uso del algoritmo Multinivel y permite extraer un conjunto de parámetros locales para los nodos y aristas que con la librería igraph no era posible. Dichos resultados se exportan en dos tablas que se analizan en detalle en Knime (ver Figura 19). En primer lugar se analizan dos de los atributos que se incluirán a SLOD-BI, Betweenness Centrality y Eigenvector centrality, ya que estos demuestran una alta correlación para la caracterización de los nodos y se les atribuyen características muy importantes en un grafo [5]. Eigenvector Centrality almacena con un punto cada enlace que a un nodo llega, pero distingue a su vez que tipo de enlace es el que llega a un nodo, porque logra separar nodos que son más relevantes que otros; e intuitivamente, “Un nodo es importante si esta enlazado con otros nodos importantes”. Por su parte Betweenness Centrality mide el rango en que un vértice se encuentra en el camino de otros vértices. Describe que nodos con altos valores de este atributo suelen tener un impacto considerable en la red, ya que a través de ellos fluye la información que manejan los demás; al igual que son aquellos vértices que al ser eliminados de un

grafo interrumpirán en mayor grado las comunicaciones entre otros vértices porque se encuentran en el mayor número de caminos que toman los mensajes [5].

Para confrontar los resultados hasta ahora expuestos en el proyecto en primer lugar se visualizan las comunidades con respecto al valor de Eigenvector centrality que presentan los nodos, donde es posible comparar la importancia de las comunidades en la red. La comunidad con ID 3, como ejemplo más evidente, se presenta como la mayor comunidad en el grafo en cuanto a relevancia y centralidad en el grafo, se visualiza en la Figura 19 que es la única comunidad que alcanza valores de 1 en su centralidad, lo cual resalta la importancia de este atributo integrado a SLOD-BI.

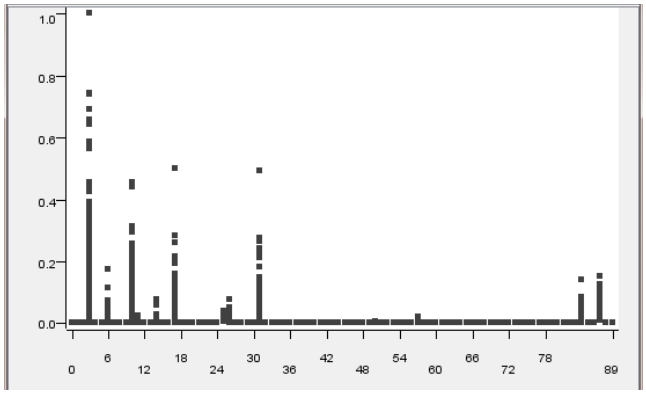
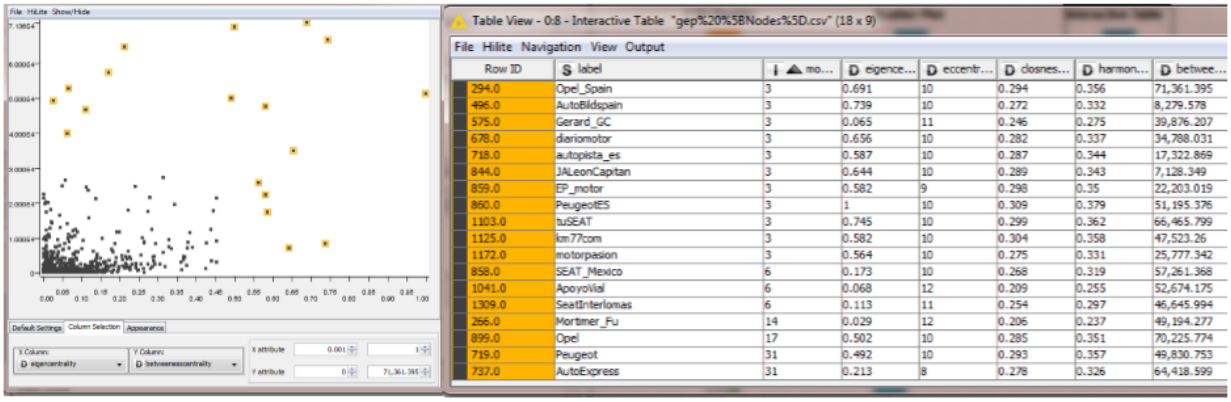


Figura 19: Comunidades y la centralidad detectada en ellas.

A continuación se comprueba para todo el conjunto analizado la relación entre el par de atributos ya citados, Eigenvector centrality y Betweenness Centrality. Para la totalidad del conjunto se demuestra que existirán una mayoría de nodos, que aproximadamente y por análisis semánticos de forma manual tienen una tasa de un 86% aproximadamente de usuarios reales, es decir personas, mientras que el resto pertenecen a páginas webs y compañías. Se logra comprobar que los usuarios resaltados en amarillo y que presentan altos valores de los atributos analizados, pertenecen a empresas que intentan promocionarse a través de Twitter y escuchar a la voz del cliente mediante esta vía. Como ya se ha mencionado, logran atraer una mayoría de usuarios que no sostendrán un papel importante dentro de las comunidades, pero que sí estarán contenidos en la comunidad y por supuesto al alcance de los *tweets* publicados y de la información esparcida por dichas empresas. Por tanto se maneja en la investigación sostenida que el servicio de Twitter, al menos para los temas analizados del coche Opel Astra, sostienen un carácter plenamente comercial y publicitario, siendo los usuarios conectados a estas comunidades los objetivos de estas campañas publicitarias y a la vez ellos mismos la principal vía de expansión hacia otros usuarios. Se propone además la expansión de los usuarios que directamente siguen a la cuenta oficial de la compañía para a su vez contribuir a tener un impacto mayor a medida que expandimos la comunidad globalmente en Twitter. Otros factores de crecimiento y medidas que se utilizan en la vida real no son estudiadas y analizadas en este trabajo, pero el combinar un conjunto de medidas junto a dar significado a los usuarios que componen la comunidad para la expansión de la misma son la base en sí de un esquema de marketing en la red en la actualidad. A continuación se muestran las gráficas comentadas.

a) **b)**



Row ID	label	mo...	eigence...	eccentr...	closnes...	harmon...	betwee...
294.0	Opel_Spain	3	0.691	10	0.294	0.356	71,361.395
496.0	AutoBildspain	3	0.739	10	0.272	0.332	8,279.578
575.0	Gerard_GC	3	0.065	11	0.246	0.275	39,876.207
678.0	diariomotor	3	0.656	10	0.282	0.337	34,788.031
718.0	autopista_es	3	0.587	10	0.287	0.344	17,322.869
844.0	JALeonCapitan	3	0.644	10	0.289	0.343	7,128.349
859.0	EP_motor	3	0.582	9	0.298	0.35	22,203.019
860.0	PeugeotES	3	1	10	0.309	0.379	51,195.376
1103.0	tuSEAT	3	0.745	10	0.299	0.362	66,465.799
1125.0	km77com	3	0.582	10	0.304	0.358	47,523.26
1172.0	motorpasion	3	0.564	10	0.275	0.331	25,777.342
858.0	SEAT_Mexico	6	0.173	10	0.268	0.319	57,261.368
1041.0	ApoyoVial	6	0.068	12	0.209	0.255	52,674.175
1309.0	SeatInterlomas	6	0.113	11	0.254	0.297	46,645.994
266.0	Mortimer_Fu	14	0.029	12	0.206	0.237	49,194.277
899.0	Opel	17	0.502	10	0.285	0.351	70,225.774
719.0	Peugeot	31	0.492	10	0.293	0.357	49,830.753
737.0	AutoExpress	31	0.213	8	0.278	0.326	64,418.599

Figura 20: Relación de usuarios y atributos de centralidad Eigenvector y Betweenness.

En el apartado b de la Figura 20 se logra apreciar los usuarios que tienen al menos uno de los atributos señalados con altos valores y se separan de la mayoría de usuarios cercanos al eje de coordenadas. Muchos de estos usuarios se analizaron en el apartado 4.4.5, y por tanto se puede apoyar en lo ya descrito para el análisis de este conjunto de usuarios.

4.5.1 Caracterización de Comunidades

Un análisis recurrente en investigaciones de detección de comunidades usando Knime y Python es la caracterización de las agrupaciones construidas con respecto a algún atributo o clasificación interesante de abordar para el tema [2]. En nuestro proyecto realizar una caracterización de las comunidades mientras se utilizan las facilidades que nos brinda SLOD-BI era una tarea de obligada ejecución. Para dicha tarea se crean scripts de Python que hacen uso de SLOD-BI mediante llamadas REST y tras clasificar a los usuarios en comunidades, se agrupan dichas comunidades y se les realiza un análisis de sentimiento para determinar de los *tweets* publicados las opiniones que son extraíbles.

Se proponen dividir las comunidades dentro de las siguientes clasificaciones, SuperFans, Positiva, Neutral, Negativa y *Trolls*. La primera de las etiquetas se define para las comunidades que logran alcanzar un valor de sentimiento hacia el tema considerablemente positivo, en concreto un valor mayor al número de nodos que conforman su propia comunidad. Fans o comunidad Positiva para aquellas comunidades que logran tener su índice de análisis de sentimiento en el lado positivo; mayores que la mitad del conjunto de usuarios que contiene pero sin llegar a SuperFans. Por su parte, la clasificación neutral se establece para las comunidades donde el análisis de sentimiento no obtiene resultado alguno o sus análisis terminan siendo 0 o muy cercanos a 0, en concreto mayor o menor que 0 pero sin alcanzar la mitad de los usuarios en el clúster. En último lugar, en el lado opuesto tendremos para las comunidades negativas dos agrupaciones: una para las negativas como exactamente lo opuesto a la comunidad positiva y otra para los clústeres realmente negativos en el mayor grado

posible, clasificados en muy negativas o *Trolls*, exactamente opuestos a SuperFans, y necesarias estas últimas de estudio para el marketing ante la posible presencia de usuarios *trolls* en ellas [41].

Ejemplo de análisis de sentimiento en SLOD_BI:

I find this Opel Astra to be very good and useful, but it is a bit too expensive.

SLOD-BI logra clasificar la expresión de ejemplo como positiva, ya que en ella encuentra dos palabras positivas ("*good*" y "*useful*") y una palabra negativa, ("*expensive*"). Para el análisis de sentimiento SLOD-BI establece distintas clasificaciones en las palabras y da una puntuación específica en dependencia del nivel del sentimiento positivo o negativo, dejando las palabras que no se encuentran en tonalidades de rojo o verde como palabras neutras [21], [1].

Tras la clasificación por parte de SLOD-BI y extraer para cada usuario todos los *tweets* publicados en el conjunto de análisis, se realiza una sumatoria de todos sus *tweets* para la temática "Opel Astra" y todos estos usuarios son agrupados dentro del clúster al cual pertenecen. Se muestran en la

Tabla4 las comunidades con mayor número de nodos y su clasificación en dependencia a los clasificadores expuestos anteriormente.

ID de Comunidad	Tamaño	Buen-Mal_Rating	Clasificación
1	161	0	**Neutral**
3	81	0	**Neutral**
4	167	-9	**Neutral**
5	114	-4	**Neutral**
6	49	-31	**Negativa**
8	99	104	**Súper-Fans**
9	245	4	**Neutral**
17	98	-8	**Neutral**

Tabla 4: Reporte asociado al análisis de sentimiento en las comunidades detectadas.

Un aspecto a tener en cuenta en esta caracterización es que, para comunidades con altos índices de centralidad, se detectan un mayor número de cuentas pertenecientes a empresas, páginas webs o servicios varios. En resumen, para estas comunidades el análisis de sentimiento no aporta una polaridad distinguible, ya que regularmente suelen publicar en la red, promociones o noticias relacionadas resultando generalmente en comunidades neutras.

Se muestra a continuación para la comunidad con ID: 6, con polaridad negativa, la nube de palabras que de ella se construye (ver Figura 21).

Figura 21. Nube de palabras para la comunidad 6.

En la Figura 21 se aprecian palabras que SLOD-BI detecta con una polaridad negativa. Algunas de estas palabras son *died, missing o crime,* por tan solo abordar algunas de las más citadas. Se procede por tanto al análisis más en detalle de los tweets publicados y se detectan en los usuarios de la comunidad, un ambiente hostil hacia la marca Opel Astra. Se publican en esta comunidad noticias de accidentes o desapariciones con coches Opel involucrados, potenciando un ambiente negativo hacia posibles compradores y por tanto una comunidad a tener en cuenta por la marca.

4.6 Comparación entre Geolocalización y Comunidades detectadas en Blondel

Con la información hasta el momento documentada en el proyecto, se cuentan con medidas que fueron agregadas a los nodos en el grafo construido, expuestas en la Figura 21. Se dispone de topologías de grafo resueltas en comunidades en un espacio de tiempo establecido, así como la geo-localización de los usuarios en el momento en que se redactaron los *tweets* almacenados en nuestro conjunto de entrada. Toda esta información recopilada es especialmente rica para abordar análisis, razón por la cual se propone investigar sobre la existencia de similitudes entre la topología de comunidades detectada en el mejor de los casos resueltos y las regiones geográficas desde donde se originaron.

En primer lugar se resuelve dividir el mapa construido en la Figura 7 en regiones de mayor o menor tamaño según la densidad de población existente en nuestro conjunto de datos. Para el continente americano, por ejemplo, se decide dividir el área norte de América del resto del continente y a su vez subdividir a los Estados Unidos entre este y oeste. Se obtiene en resumen en América tres grandes regiones, una para toda América del Sur debido a la baja densidad de *Tweets* en el área, mientras que ante la mayor densidad se divide en dos regiones la parte norte de América, en este y oeste como ya se comentaba. Este tipo de divisionamiento es considerado debido a la posible similitud en los tópicos en áreas comunes, así como la posibilidad de existir un mayor grado de asociación entre los nodos si

estos pertenecen a las mismas áreas geográficas y la verificación y comparación de la geolocalización con la detección de comunidades resuelta por el algoritmo de Blondel.

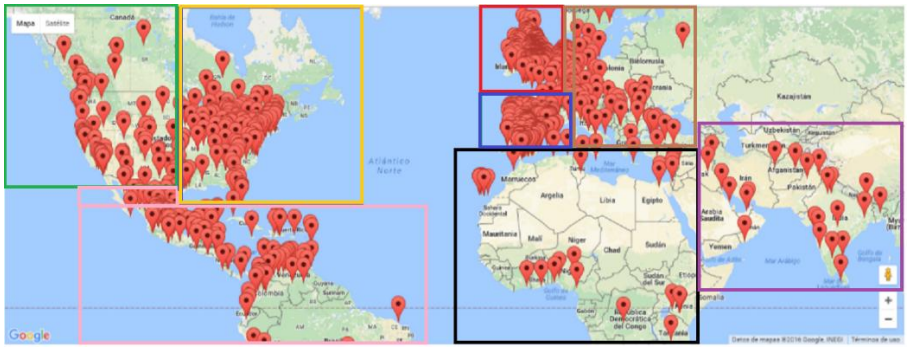


Figura 22. Geolocalización y su divisionamiento en regiones

Algunos aspectos de similaridad se mantienen, como la topología en los grafos para ser capaces de comparar visualmente sin la necesidad de apoyo de algún software externo. Para ello es requerido mencionar que los usuarios geo-localizados, de 1332 usuarios en el grafo inicial, solo son 379 usuarios, debido a que en muchas cuentas de Twitter se suele omitir este tipo de información, reduciendo a un 28.45% nuestro análisis del total de usuarios. En resumen, se logra el principal propósito de comparar las topologías existentes y de analizar el nuevo grafo que se construye y ver la sociedad real que existe entre usuarios con áreas geográficas cercanas (ver Figura 23).

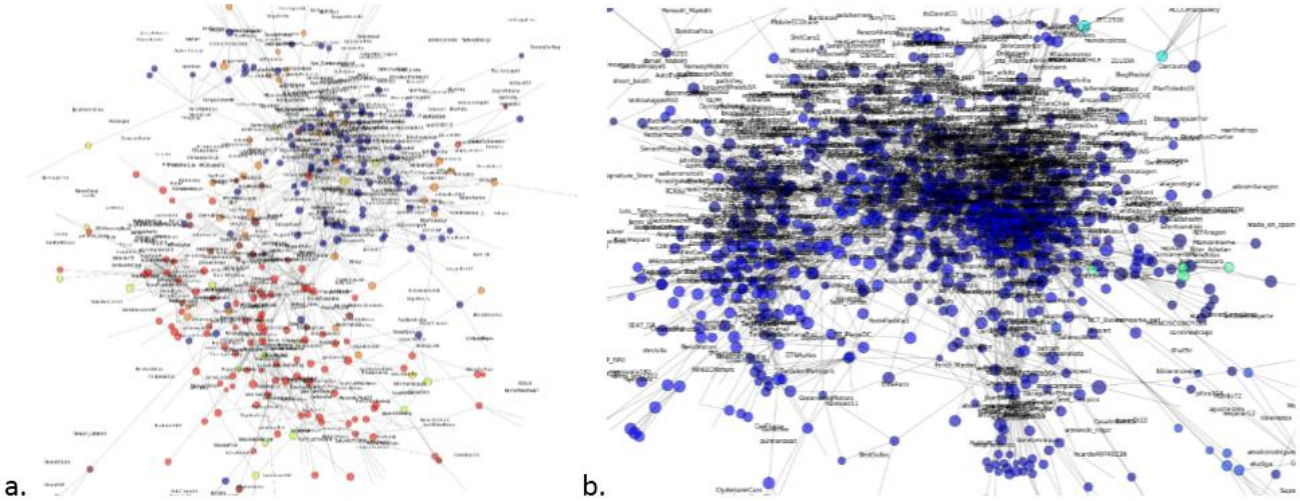


Figura 23: Comparación entre los grafos construidos por el divisionamiento en regiones(a) y el algoritmo Multinivel de Blondel(b) respectivamente.

Para el grafo a, mostrado en la Figura 23, se aprecia una falta de asociación en los nodos. A simple vista se aprecian los nodos de la zona geográfica de Gran Bretaña, en rojo, mientras los azules reflejan Europa continental occidental. Pero existe una gran cantidad de nodos que logran inmiscuirse de otras áreas de geográficas. En términos generales y para abordar la escasa asociación de los nodos en las

comunidades geográficas, se obtiene un valor de modularidad 0.026. Por tanto, se declaran algunos puntos aun así en favor de esta comparación aunque no resultase idealmente productiva, ya que logra advertir sobre la creación de comunidades y que aunque estas no aportan en realidad altos valores de modularidad, si logran mantener un valor positivo de la métrica y por tanto contienen principios de asociación en comunidades. En resumen se amplían las medidas para la creación de comunidades expuestas en [13], que recomienda medidas como los *Retweets*, los *Hashtags* o las menciones y se recomienda por tanto ante la existencia de asociaciones reales en la cercanía geográfica abordar de una forma adecuada dicha medida en futuras investigaciones y planear su consideración dentro de los propios algoritmos de detección de comunidades, para probar si logran mejorar los resultados alcanzables.

5. Pruebas e Integración en SLOD-BI

Después de realizar la implementación y un vasto análisis para el conjunto de datos extraído, se propone un nuevo conjunto de datos y se describen algunas de las pruebas realizadas para comprobar el correcto funcionamiento de los algoritmos de extracción de la información, construcción de la topología del grafo, detección de comunidades, así como algunos de los análisis realizados. Le sigue a ello la posterior integración de dicha información relevante a los datos públicos y enlazados de SLOD-BI para ser tratada en posteriores investigaciones.

5.1 Análisis sobre un caso de estudio en particular

Para este apartado en el proyecto se llega al punto donde se hace necesario aplicar todo el contenido construido a un caso de prueba para comprobar que los objetivos propuestos se han alcanzado. Para ello se somete a prueba los algoritmos construidos para verificar su capacidad de recoger en *streaming Tweets* en tiempo real de un tema específico para construir con ellos grafos. Este primer acercamiento intenta validar los objetivos 1 y 2:

1. Se estudiarán las características de los principales *hechos sociales*, y se estudiará cómo es posible obtenerlos a través de la metainformación existente sobre los *tweets*.

2. Se diseñará e implementará un extractor de información que permita extraer la información necesaria de los *tweets*.

En concreto se ha obtenido la metainformación existente sobre *tweets* a partir de las características de los principales hechos sociales y las comunidades. En primer lugar se procede a capturar la información de todos los usuarios y esta pasa a ser procesada para recuperar de Twitter sus seguidores y a quienes el usuario sigue. Se establece un sistema de archivos que se decide nombrar “following” y “twitter-users” con la información relevante que se necesitará para con ello crear un archivo en forma de relaciones y con ese archivo ser capaces de construir la topología de un grafo social o varios, dependiendo del análisis requerido.

Se construye sobre el proyecto SLOD-BI un script de Pyhton capaz de extraer *Tweets* para un tema dado, se decide en este caso como palabra clave “Peugeot”. Cabe mencionar que para alcanzar un grafo lo suficientemente denso se puede requerir un seguimiento muy largo de un tema. Por ejemplo, en [28] se requirió 150 días de recogida de datos. En nuestro caso se han tomado 500 tweets de SLOD-BI y de ellos se construye un grafo siguiendo las medidas documentadas en el apartado 4.3.

3. Se diseñará e implementará un sistema que permita obtener los hechos sociales a partir de la información identificada en el punto anterior y volcarla en SLOD-BI.

Se aplican para el grafo construido medidas como *retweets*, impacto, influencia de un usuario, amigos mutuos y otras muchas, expuestas en [13]. Para el conjunto representado en la Figura 24**,** se registra

un impacto realmente grande para un grafo tan pequeño, debido a algunas cuentas de Twitter con cientos de miles de seguidores y para las cuales se decide manejar un reducido número de usuarios (como en el conjunto inicial, epígrafe 4.2.1). Por tanto el grafo que se decide analizar solamente contiene las cuentas de usuarios que muestran tanto un papel de *follower* como de *followed by*, para eliminar un importante número de usuarios que solamente aportan ruido en los algoritmos de detección de comunidades. Se implementa un script de Python que utiliza el algoritmo diseñado por Blondel, Multinivel, por las razones expuestas en la Tabla 3 en la sección 4.4.5, cumpliendo con el objetivo 4: Se estudiarán e implementarán distintos algoritmos de detección de comunidades, y se obtienen los resultados que se muestran en la Figura 24.

En resumen el apartado actual aborda el último de los objetivos propuestos:

5. Se validará el resultado mediante su aplicación a un caso de estudio.

Nuestro caso de estudio se centra en el conjunto recopilado para la palabra clave “Peugeot”, para el cual se realizan todos los experimentos realizados para el conjunto inicial en este trabajo. Cabe mencionar que algunos de los experimentos no son factibles de realizar debido a la escasez de información para algunas métricas de Twitter. Estos detalles los veremos más adelante.

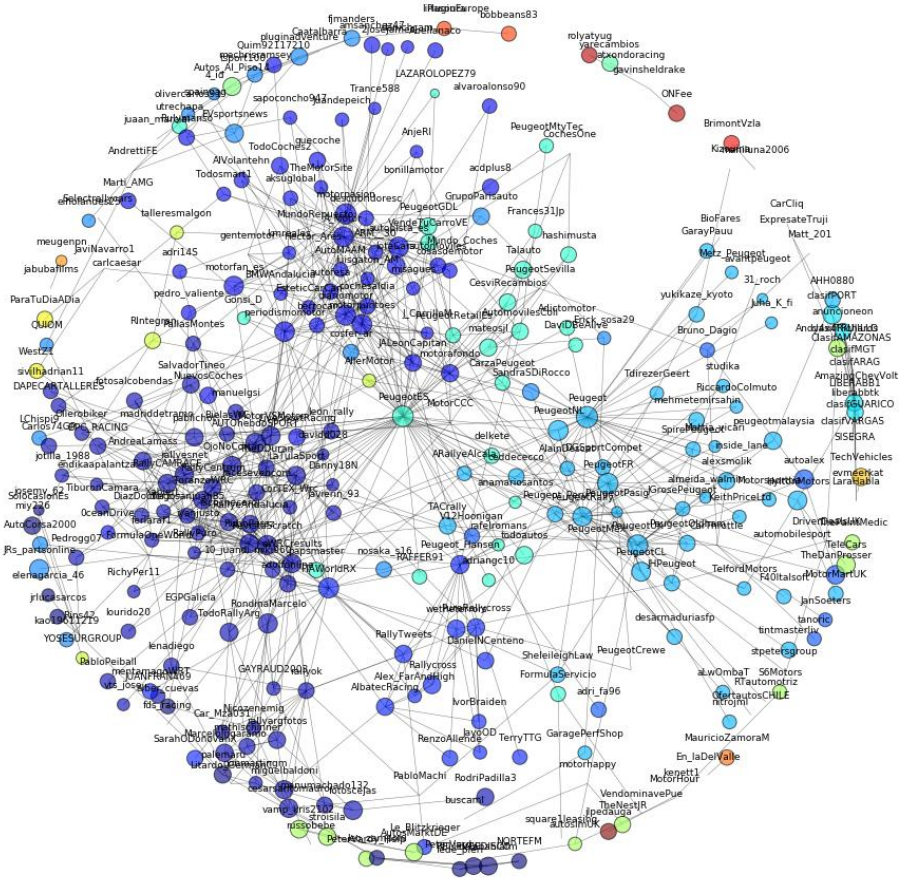


Figura 24: Comunidades detectadas bajo la temática “peugeot”

En la Figura 24 se visualiza la presencia de 3 grandes comunidades, con un total de 19 comunidades detectadas y una modularidad muy buena para el conjunto de 0.602. Esto nos confirma la calidad del algoritmo y el esperado valor que ostentan sus resultados. Para el conjunto se dispone de las herramientas de ubicación espacial, pero debido a la omisión del campo de dirección en la mayoría de los usuarios, se decide no aplicar. Se advierte más factible este métrica en conjuntos mucho mayores, para los cuales es posible obviar pérdidas del atributo.

Como se ha realizado en anteriores análisis se puede elegir una comunidad cualquiera y visualizar el contenido sobre el que publican contenido. En la mayor comunidad detectada para el tema central "peugeot" se decide construir una nube de palabras, para demostrar cuan fehaciente es la comunidad detectada para con el tema global y comprobar uno de las principales paradigmas hasta ahora estipulados, la importancia de la jerarquía de las comunidades en cuanto a su tamaño y por tanto el establecer esta jerarquía para marketing, publicidad o análisis cualesquiera. Los resultados expuestos en una nube de palabras se muestran en la Figura 25.

Figura 25: Visualización de nube de palabras para la mayor comunidad detectada, con ID: 3.

5.2 Integración en SLOD-BI

El presente proyecto por tanto se suma a la iniciativa de los datos abiertos enlazados y fomenta el uso del formato JSON-LD, el cual permite no solamente expresar datos enlazados en forma de JSON si no también añadir semántica a documentos JSON ya existentes. JSON-LD se plantea como objetivo crear JSON semánticamente enriquecido. Entre las ventajas en este nuevo formato se cuentan algunas como la permisibilidad de representar ciclos, ya que su par nativo en su modelo de datos se estructura en un árbol y no es capaz de contener bucles en él. JSON-LD además permite con dos palabras claves (@context y @id) acceder a todas las funcionalidades básicas que ofrece. Dado que JSON-LD es 100% compatible con JSON, los desarrolladores pueden seguir utilizando sus herramientas y bibliotecas; característica especialmente importante para las empresas, ya que permite añadir significado a sus documentos JSON de una manera que no perturba sus operaciones y es transparente para sus clientes. Al mismo tiempo, JSON-LD es lo suficientemente expresivo como para contener todos los principales conceptos RDF.

En términos generales su idea básica es crear una descripción de los datos en la forma de un llamado contexto (@context), que enlaza los datos de manera que objetos y sus propiedades en un documento JSON sean conectados a los conceptos en una ontología. Un contexto puede ser directamente un

documento JSON-LD o un archivo independiente referenciado por diferentes documentos. Estas características, sumadas al hecho de que los antiguos documentos JSON pueden hacer referencia a un contexto mediante una cabecera de enlace HTTP, proporciona una ruta de actualización para la infraestructura existente ya que permite la mayor parte de la funcionalidad sin tener que cambiar el contenido de un documento existente en lo absoluto.

El proyecto construido hasta el momento se encuentra entonces en la necesidad de hacer uso del modelo de datos JSON-LD para publicar los resultados y análisis obtenidos en el repositorio de datos abiertos y enlazados de SLOD-BI. Para la publicación de los mismos se analizan que datos convienen ser publicados y se decide en primer lugar por los usuarios detectados. Se les atribuyen las características en las que ahora se encuentran catalogados, su ID de usuario, un ID de comunidad detectado tras la ejecución de los algoritmos de detección de comunidades, su localización geo-espacial, así como la detección de sí es un bot en la comunidad descubierta. El análisis para la detección de robots en Twitter se implementa mediante el análisis del comportamiento de la cuenta, en la repetición de tweets publicados y en la característica innata de hacer retweets sobre las cuentas que realmente crean este tipo de robots. Se añaden además en el documento JSON-LD algunas métricas, documentadas a lo largo de la memoria, ver epígrafes 4.4.5 y 4.5, como *betweenness centrality* y *eigenvector centrality,* entre otras. Se muestra a continuación un ejemplo del JSON-LD que se construye.


```
{
 "@context": {
  "@context": {
   "slod": "http://krono.act.uji.es/datasets/cars/",
   "foaf": "http://xmlns.com/foaf/0.1/",
   "ical": "http://www.w3.org/2002/12/cal/ical#",
   "sch": "http://schema.org/",
   "latitude": {
    "@id": "http://schema.org/latitude",
    "@type": "xsd:float"
   },
   "slod-onto": "http://krono.act.uji.es/datasets/ontologies/slodonto.rdf#",
   "geo": "http://schema.org/geo",
   "longitude": {
    "@id": "http://schema.org/longitude",
    "@type": "xsd:float"
   }
  },
  "@base": "http://schema.org/"
 },
 "@graph": [
  {
   "a": "foaf:User",
   "sch:name": "Plymouthcars",
   "slod-onto:IdCom": "slod:1",
   "slod-onto:bot": "False",
   "foaf:friends": [
    "slod:User:Aly",
    "slod:User:aju"
```

```
      ],
      "@id": "slod:Plymouthcars",
      "slod-onto:Eigen_Vector_Centrality": "0.054915962967039746",
      "ical:datepick": "2011-04-09T20:00Z",
      "slod-onto:Betwenness_Centrality": "4.160035080158918",
      "geo": {
        "latitude": "10.26543",
        "longitude": "-2.3654"
      },
      "@type": "sch:Person"
    },
    {
      "a": "foaf:User",
      "sch:name": "RafaTwitter",
      "slod-onto:IdCom": "slod:1",
      "slod-onto:bot": "False",
      "foaf:friends": [
        "slod:User:Aly",
        "slod:User:Jose"
      ],
      "@id": "slod:RafaTwitter",
      "slod-onto:Eigen_Vector_Centrality": "0.0542215411546",
      "ical:datepick": "2011-04-09T20:00Z",
      "slod-onto:Betwenness_Centrality": "0.0256554459746",
      "geo": {
        "latitude": "10.26543",
        "longitude": "-2.3654"
      },
      "@type": "sch:Person"
    }
  ]
}
```


En resumen toda la información recopilada finalmente se serializa a N-tripletes RDF y es incorporada en este formato a SLOD-BI. Los N-tripletes se desarrollan específicamente como un formato para almacenar grafos RDF, con la facilidad de poder ser generadas por software. Por tanto son la herramienta ideal para a partir de JSON-LD volcar la información construida al repositorio SLOD-BI [41]. En resumen se adicionan 247773 tripletes generados a partir del fichero JSON-LD. Se muestra un ejemplo del fichero generado a continuación.

<http://krono.act.uji.es/datasets/cars/1001ptsES>
<http://krono.act.uji.es/datasets/ontologies/slodonto.rdf#IdCom> "slod:54" .
<http://krono.act.uji.es/datasets/cars/1001ptsES>
<http://krono.act.uji.es/datasets/ontologies/slodonto.rdf#bot> "False" .
<http://krono.act.uji.es/datasets/cars/1001ptsES> <http://schema.org/geo> _:b1216 .
<http://krono.act.uji.es/datasets/cars/1001ptsES> <http://schema.org/name> "1001ptsES" .
<http://krono.act.uji.es/datasets/cars/1001ptsES> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://schema.org/Person> .

<http://krono.act.uji.es/datasets/cars/1001ptsES> <http://www.w3.org/2002/12/cal/ical#datepick>
"2016-09-01T10:00Z" .
<http://krono.act.uji.es/datasets/cars/1001ptsES> <http://xmlns.com/foaf/0.1/friends>
"slod:User:10CosasQue" .
<http://krono.act.uji.es/datasets/cars/1001ptsES> <http://xmlns.com/foaf/0.1/friends>
"slod:User:1Mavalsan" .

6. Conclusiones y Trabajo Futuro

En este capítulo se presentan las conclusiones del proyecto, así como el trabajo futuro que se podría realizar para mejorar la detección de comunidades online, profundizando en los análisis realizados, así como el estudio sobre el rastreo de este tipo de comunidades online.

6.1 Conclusiones

En este proyecto de fin de máster se ha presentado el desarrollo y análisis de resultado de un sistema de detección de comunidades en la red social Twitter, así como la posterior incorporación al repositorio de SLOD-BI y su puesta en marcha en un caso de estudio concreto.

En el capítulo 2 quedó reflejado que la detección de comunidades en redes sociales es un tema ampliamente abordado en investigaciones recientes, pero aun así es un tema altamente polémico ante la falta de una definición consensuada. En el proyecto que aquí se presenta el estudio de dicho hecho social, la comunidad, y su detección en redes sociales; se presentan además algunos acercamientos en el tema y se resalta el papel de la métrica modularidad en estos aspectos. Se alcanza el primero de los objetivos planteados, ver página 8, para diseñar a continuación la implementación que se realizó sobre los usuarios del conjunto “Opel Astra” en Twitter.

Después de lograr una adecuada extracción de la metainformación en Twitter, representada en gráficos como el representado en la Figura 4, se implementa un sistema para la obtención de hechos sociales, en específico de comunidades. Dicho sistema, tras ser interpretado en términos generales, aporta una característica apreciable sobre las comunidades construidas en Twitter, su inherente estructuración alrededor de nodos centralizados. Los vértices centrales resultan en el conjunto de análisis ser individuos populares, celebridades u organizaciones de cualquier índole, sin descartarse que individuos menos populares en Twitter pueden desempeñar este papel; en estos casos desempeña un rol muy importante los temas abordados por la comunidad. Estos usuarios centrales en redes personales de Twitter se caracterizan por un alto valor de *betweenness centrality*: una medida de análisis de redes sociales que indica como un individuo aparece en el camino más corto entre todos los pares de personas en la red, con un papel crítico en la construcción de la comunidad y control de la información que circula en su comunidad. En resumen después del análisis en la detección de comunidades realizado en el apartado 4.5, se declara el algoritmo Multinivel de Blondel el ideal para la topología que se maneja, ante la maximización de la modularidad y el manejo de métricas como el betweenness centrality y la eigenvector centrality, ambas integradas a SLOD-BI para futuros proyectos.

Twitter no fue originalmente diseñada como una herramienta que respalda el desarrollo de comunidades online, se imaginó simplemente como una herramienta sencilla para compartir actualizaciones en Internet. Por lo tanto, Twitter es un caso ideal para comprender como las personas se integran a las tecnologías de la información y las comunicaciones para formar nuevas conexiones sociales o mantener las existentes. En términos generales, el trabajo de fin de master presentado y los continuos análisis de carácter social en las comunidades detectadas amplían los conceptos introducidos en [12], que demostraron que Twitter está siendo utilizado para la colaboración y las

conversaciones a pesar de ser originalmente concebida como una plataforma de difusión de la información.

Algunos aspectos más allá del estudio de las comunidades son realizados también en este proyecto y se logra introducir el estudio de la geo-localización de los usuarios a SLOD-BI y comparar las métricas obtenidas con esta "nueva" medida espacial. Se completa el proyecto con un caso de estudio sobre la firma Peugeot para validar los resultados hasta el momento comentados, donde se cumple con el último de los objetivos trazados a inicios de este documento, para por último volcar los resultados hacia SLOD-BI mediante la conversión de los resultados al formato JSON-LD para la publicación de los mismos en el entorno de los datos abiertos y enlazados.

6.2 Trabajo futuro

Como trabajo futuro se propone la mejora de los servicios prestados en los algoritmos implementados. Para ellos se expone la creación de un software unificador y el desarrollo de una interfaz lo suficientemente potente para permitir el manejo de los algoritmos de detección de comunidades y controlar los parámetros en los mismos, el tipo de entrada y el manejo de la colecta de metainformación en Twitter. En resumen, para hacer posible en un solo programa la detección online de comunidades y eliminar la dependencia de un desarrollador para cumplir estos propósitos.

Además se recomienda el estudio de las comunidades detectadas en el tiempo, para la detección de como la estructura de las comunidades afecta la formación y evolución de enlaces entre los nodos. Plantear el estudio de lo que hoy se conoce como rastreo de comunidades online, para las agrupaciones descubiertas, ante la notable característica de dinamicidad de las comunidades online. En último lugar se propone oportuno una continuación en el estudio de los metadatos sociales geo-referenciados en la plataforma GIS, mediante el desarrollo de un sistema que puede representarse de alto valor tanto como para análisis empresarial como para análisis científico en la identificación de tendencias.

7. Referencias

[1] Rafael Berlanga Llavori, Lisette García-Moya, Victoria Nebot, María José Aramburu, Ismael Sanz, Dolores María Llidó: SLOD-BI: An Open Data Infrastructure for Enabling Social Business Intelligence. IJDWM 1-28 (2015)

[2] Rosaria Silipo, Phil Winters, Killian Thiel, Tobias Kötter: Creating Usable Customer Intelligence from Social Media Data: Clustering the Social Community. Knime (2012)

[3] Killian Thiel, Tobias Kötter, Dr. Michael Berthold, Dr. Rosaria Silipo, Phil Winters: Creating Usable Customer Intelligence from Social Media Data: Network Analytics meets Text Mining. Knime (2012)

[4] David Ediger, Karl Jiang, Jason Riedy, David A. Bader, Courtney Corley, Rob Farber, William N. Reynolds: Massive Social Network Analysis: Mining Twitter for Social Good. In Proceeding of 39th International Conference on Parallel Processing. (2010)

[5]Shamanth Kumar, Fred Morstatter, Huan Liu: Twitter Data Analytics. Springer (2013)

[6]Domingos, P. and Richardson, M. 2001. Mining the network value of customers. In Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'01).

[7]Cha, M., Haddadi, H., Benevenuto F., and Gummadi, K. P. 2010. Measuring user influence in twitter: The million follower fallacy. In Proceedings of the 4th International AAAI Conference on Weblogs and Social Media (ICWSM).

[8] Zachary, Wayne W. "An Information Flow Model for Conflict and Fission in Small Groups."*Journal of Anthropological Research* 33.4 (1977): 452-73. Web.

[9] Gutiérrez Martín, A. (1997): Educación multimedia y nuevas tecnologías. Madrid: Ediciones de la Torre.

[10] L.C. Freeman, A set of measures of centrality based upon betweenness, Sociometry 40 (1) (1977) 35–41.

[11] M.E.J. Newman, M. Girvan, Mixing patterns and community structure in networks, in: R. Pastor-Satorras, M. Rubi, A. Díaz Guilera (Eds.), Statistical Mechanics of Complex Networks: Proceedings of the XVIII Sitges Conference on Statistical Mechanics, in: Lecture Notes in Physics, vol. 625, SpringerVerlag GmbH, Berlin, Germany, 2003.

[12] Susan C. Herring, Courtenay Honeycutt, "Beyond Microblogging: Conversation and Collaboration via Twitter", *2014 47th Hawaii International Conference on System Sciences*, vol. 00, no. , pp. 1-10, 2009, doi:10.1109/HICSS.2009.602

[13] The Complete Guide to Twitter Analytics (2014) Simply Measured, Inc. Recuperado de http://get.simplymeasured.com/rs/simplymeasured2/images/CompleteGuidetoTwitterAnalyticsSimplyMeasured.pdf Visitado el 10 de noviembre de 2016.

[14] Fandos, M. & Silvestre, R. (2011) Servicios de microblogs en la enseñanza secundaria. Edutec-e, Revista Electrónica de Tecnología Educativa, 38. Recuperado el 10/06/2016 de http://edutec.rediris.es/Revelec2/Revelec38/servicios_microblogs_ensenanza_secundaria.h tml

[15] Yus Ramos, F. (2010). *Ciberpragmática 2.0. Nuevos usos del lenguaje en Internet*. Barcelona, Ariel. Disponible resumen en versión inglesa: http://goo.gl/WJD6FX

[16] The Cocktail Analysis (2013). *5º Oleada de El Observatorio de Redes Sociales.* Extraído el 6 de junio de 2016 desde http://tcanalysis.com/blog/posts/el-70-de-losusuarios-de-redes-sociales-se-muestran-receptivos-a-la-presencia-de-marcas-en-esteentorno

[17] Naso, F., Balbi, M. L.; Di Grazia, N.O.; Peri, J.A (2012). *La importancia de las redes sociales en el ámbito educativo.* VII Congreso de Tecnología en Educación y Educación en Tecnología.

[18] Mollet A., Moran D. y Dunleavy P. (2011). *El uso de Twitter en la investigación universitaria, la enseñanza y el impacto en las investigaciones: una guía para los académicos e investigadores.* Curso de formación "Redes Sociales en Educación". Universidad de León.

[19] Bicen, H., Cavus N. (2012). Twitter usage habits of undergraduate students. *ProcediaSocial and Behavioral Sciences 46,* 335-339.

[20] José Luis Orihuela (2011) Mundo Twitter: una guía para comprender y dominar la plataforma que cambió la red. Barcelona: Alienta, 2011. 266 p. ISBN 978-84-92414-89-5

[21] Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. Found. Trends Inf. Retr., 2(1-2):1–135

[22] Pak, A. and Paroubek, P. 2010. Twitter as a Corpus for Sentiment Analysis and Opinion Mining. In Proceedings of the Seventh conference on International Language Resources and Evaluation (LREC'10), European Language Resources Association (ELRA), (Valletta, Malta, 2010), 1320-1326.

[23] Kanak Biscuitwala Sajid Shariff Ketaki Sheode (2011) AFFILIATE MARKETING: AN OVERVIEW visitado el 20 de julio de 2016. Recuperado desde https://web.stanford.edu/class/msande239/lectures-2011/Lecture%2008%20in-class%20pres1.pdf

[24] Vincent D. Blondel, Jean-Loup Guillaume, Renaud Lambiotte y Etienne Lefebvre, (2008) Fast Unfolding of communities in large networks. Physics.soc-ph.

[25] M. Rosvall y D. Axelsson (2009) The map equation. Physics.soc-ph

[26] A. Clauset, M.E.J. Newman, C. Moore. (2004), Phys Rev E 70, 066111

[27] M.E.J. Newman, M. Girvan. (2004) Phys Rev E 69, 026113

[28] Jorge Merlino (2015) Análisis del grafo de Twitter. Social Networks

[29] Pascal Pons y Matthieu Latapy (2005) Computing communities in large networks Using random walks. Physics.soc-ph.

[30] http://igraph.org/redirect.html Visitado el 10 de septiembre de 2016.

[31] https://gephi.org/ Visitado el 21 de septiembre de 2016.

[32] http://tagcrowd.com/ Visitado el 22 de septiembre de 2016.

[33] http://json-ld.org/ Visitado el 10 de octubre de 2016.

[34] https://docs.Python.org/3/tutorial/index.html Visitado el 1 de septiembre de 2016.

[35] https://snap.stanford.edu/snappy/doc/index.html Visitado el 1 de septiembre de 2016.

[36] http://matplotlib.org/ Visitado el 10 de septiembre de 2016.

[37] Andrea Lancichinetti, Santo Fortunato (2010) Community detection algorithms: a comparative analysis. PACS numbers 89.75.-k 89.75. Hc. Physics.soc-ph.

[38] Cesare Pautasso (2008). REST vs SOAP: Making the Right Architectural Decision. Proceedings of the 1st International SOA SYMPOSIUM, October 7-8, 2008.

[39] http://www.cyberclick.es/numerical-blog/10-a%C3%B1os-de-twitter-10-grandes-sucesos-en-esta-red-social .Visitado el 1 de julio de 2016.

[40] OAuth The big picture. http://pages.apigee.com/rs/apigee/images/oauth-ebook-2012-02.pdf Vistado el 20 de octubre de 2016.

[41] Paraskevas Tsantarliotis, Evaggelia, Panayiotis. (2016) Troll Vulnerability in Online Social Networks. In proceeding of 2016 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining (ASONAM)

[42] Sören Auer, Jens Lehmann, Axel-Cyrille Ngonga Ngomo, Amrapali Zaveri (2011) Introduction to Linked Data and its Lifecycle on the Web. AKSW, Institut für Informatik, Universität Leipzig, Pf 100920, 04009 Leipzig