TRABAJO FIN DE GRADO

Grado en Ingeniería Informática

UNIVERSIDAD DE ALMERIA

ESCUELA SUPERIOR DE INGENIERÍA

“Andotter – Aplicación para el análisis temporal y geográfico de opinión en tendencias de Twitter”

Curso 2016/2017

Alumno/a:
José Luis Navarro Motos

Director/es:
José del Sagrado Martínez

José Luis Navarro Motos
Grado en Ingeniería Informática - Universidad de Almería

Prólogo

Durante los 4 años de carrera son muchas las asignaturas por las que pasamos.

Resulta habitual, durante los primeros años, preguntarse si he elegido la titulación que realmente me gusta tras pasar asignatura y asignatura sin acabar con esa sensación de "esto es lo que realmente me gusta". Pero al final, más tarde que temprano, llega el esperado momento, que en mi caso, no ha sido hasta el cuarto curso de carrera con la asignatura de Inteligencia de Negocio, asignatura con la cuál he descubierto que la rama que más me apasiona y en la que menos esfuerzo me cuesta trabajar es la Minería de Datos. Quizás una de las ramas menos "puramente" informáticas pero no por ello menos importante.

Cuando, durante el transcurso de la asignatura saboree la capacidad que, junto a la Minería de Datos, tenemos para recoger una inmensa cantidad de datos, sin apenas información a priori, y convertirlos en un ente capaz de aportar una información tan potente como para plantearse un cambio en la política de una empresa es cuando descubrí que lo que tenía frente a mi tenía que ser el ámbito al que dedicar mi Trabajo de Fin de Grado.

En ese momento comencé a darle vueltas a la cabeza sin conseguir establecer un tema específico dentro de la Minería de Datos sobre el que profundizar. Descubrí tal cantidad de temas que me abordaron y perdí el rumbo.

Fue el momento de acudir a una persona con experiencia en el campo que me cogiera de la mano y me situara en el inicio del camino que yo realmente buscaba. Y que mejor persona que el profesor que me impartió la asignatura anteriormente citada.

Durante una charla en su despacho, con solo hablar del tema, casi sin querer, se iba formalizando el tema del proyecto. Y ahora que lo tengo en mis manos pienso que no podría haber elegido otro mejor.

Así que desde aquí, no me gustaría pasar al siguiente punto para entrar en materia sin agradecer su ayuda. José, mi más sincera gratitud por la colaboración prestada. He disfrutado y aprendido más de lo que nunca hubiera pensado durante la realización de este proyecto y gran parte de culpa es tuya.

Sin más dilación, comencemos con el proyecto, que, espero guste a los leyentes igual o más que lo ha hecho a mí.

José Luis Navarro Motos
Grado en Ingeniería Informática - Universidad de Almería

Índice general

Capítulo 1

Introducción

1.1 Presentación del problema.

Resulta sencillo darse cuenta que las redes sociales son un campo en pleno auge en la sociedad actual.

Según Facebook: *casi un total de 1.230.000.000 usuarios de la plataforma inician sesión habitualmente en sus cuentas, llegando a pasar un promedio de 17 minutos al día conectados.*

Además, recientes estudios estadísticos revelan que el 72% de los hombres y el 80% de las mujeres poseen un perfil activo en alguna red social.

Redes sociales como twitter suelen utilizarse para expresar opiniones acerca de una determinada temática, como podría ser una noticia sobre un caso de corrupción. Teniendo en cuenta la inmensa cantidad de usuarios que usan las redes sociales, estamos hablando de millones de opiniones diarias.

Puede resultar más que interesante ser capaz de realizar un análisis de todas esas opiniones para obtener información acerca de los intereses de la sociedad en cada momento y ubicación geográfica.

De esto precisamente se encarga la minería de datos aplicada a las redes sociales. Ésta consiste en la extracción no trivial de información que reside de manera implícita en los datos. En otras palabras, la minería de datos prepara, sondea y explora los datos para sacar información oculta de ellos.

Los usos más habituales de la minería de datos, en el campo de las redes sociales, son los relacionados con servicios de publicidad personalizada (no es casualidad que en la red se nos muestren anuncios relacionados con nuestras aficiones o gustos) y con los procesos de contratación de personal en empresas (en la selección de personal, el estudio de los perfiles de las redes sociales puede aportar a las empresas información relevante que no se obtendrá en una simple entrevista de trabajo y puede ser clave para una correcta elección).

Nuestro enfoque no se corresponde con ninguno de los dos usos anteriores. Nuestro interés se centra en estudiar la opinión de la sociedad ante una determinada temática en función de la situación temporal y geográfica. Con el desarrollo de este proyecto se pretende obtener una herramienta que nos permita analizar el sentimiento (positivo, negativo o neutro) y la emoción (alegría, miedo, enfado, etc.) de los usuarios de Twitter ante una determinada tendencia o temática, todo esto desde un análisis temporal y geográfico.

Por ejemplo, podría resultar útil analizar qué opina la sociedad acerca de las próximas elecciones y cómo ésta opinión varía en función del lugar geográfico en que nos encontremos o del momento temporal en el que nos situemos.

1.2 Objetivos del proyecto.

Cómo objetivo de este proyecto se pretende obtener una herramienta con la que, a partir de una temática o tendencia de entrada, obtengamos el sentimiento y emoción que dicha tendencia causa en la sociedad a través de Twitter.

La aplicación desarrollada es interactiva con el usuario y muestra la información de una manera clara y precisa. Realizará 2 tipos de análisis:

- *Análisis de sentimiento basado en el léxico*. Realizamos una ponderación y un recuento de las palabras o expresiones que denotan un sentimiento positivo, neutro y negativo y nos basamos en este recuento para expresar el sentimiento del mensaje completo [5, 6].
- *Análisis mediante clasificador Naive Bayes*. Un clasificador Bayesiano es un clasificador probabilístico fundamentado en el teorema de Bayes. Tendremos un clasificador Naive Bayes entrenado con un conjunto de datos de entrenamiento con el cuál, a partir de un mensaje de entrada, será capaz de clasificar su sentimiento y emoción [5, 6].

Para ambos análisis la aplicación permite al usuario introducir diversos parámetros de entrada que tendrán reflejo en el resultado del mismo.

Junto a esto, la aplicación contiene una sección para la obtención de los actuales trending topics sobre los cuales el usuario puede realizar posteriormente uno de los dos análisis anteriormente citados. Éstos trending topics pueden obtenerse a partir de una localización geográfica concreta.

1.3 Estructura del documento.

El presente documento se inicia con un prólogo.
En el primer capítulo se presenta una introducción con la presentación del problema y los objetivos del proyecto. En el segundo capítulo se detalla el estado del arte con la información obtenida antes del inicio del proyecto sobre el ámbito del mismo. A continuación, en el capítulo 3, se pasa a detallar el análisis del sistema, esto es, una descripción detallada de la solución sin entrar en detalles de implementación, es decir, el qué y no el cómo. Una vez presentado el análisis, en el capítulo 4, se detallan todos los recursos y herramientas empleados para el desarrollo del proyecto. En el capítulo 5 se describe el diseño del sistema, comenzando por la arquitectura del mismo y pasando a detallar cada módulo implementado junto con el diseño de la interfaz de usuario. Seguidamente, en el capítulo 6, se presentan los resultados, detallando 2 ejemplos, uno de cada tipo de análisis junto con una comparación final de los 2. Finalmente, en el capítulo 7 se detallan las conclusiones del proyecto y el trabajo futuro.

Además, en los anexos se incluye el análisis de requisitos del sistema (Anexo A), el diagrama UML de casos de uso para el modelado de los requisitos del sistema (Anexo B), el cronograma asociado a las fases de desarrollo del proyecto (Anexo C) y el enlace al código del proyecto (Anexo D).

Capítulo 2

Estado del arte

Uno de los desafíos del Análisis de Sentimientos es la definición de los objetos de estudio de las opiniones y la subjetividad. Originalmente, la subjetividad fue definida por lingüistas, dentro del que destaca, Randolph Quirk [20]. Quirk define un estado privado como algo que no se encuentra abierto a la observación objetiva o verificación. Estos estados privados incluyen emociones, opiniones y especulaciones, entre otros. La definición misma de este estado privado dificulta el análisis del sentimiento. La subjetividad está a menudo implícita en una conversación, además de ser altamente sensible al contexto, y su expresión a menudo es peculiar de cada persona. Sin embargo, esa subjetividad no implica que no sea verdad [21]. Por ejemplo, la frase "Jennifer ama el chocolate" expresa un sentimiento de Jennifer para con el chocolate, pero esto no significa que no sea verdad. Es así, como de esta misma manera no todas las frases objetivas son verdaderas.

Como campo de investigación, el análisis de sentimientos, está estrechamente relacionado con (o se puede considerar una parte de) la lingüística computacional, procesamiento del lenguaje natural y la minería de textos. Partiendo por el estudio del estado afectivo (psicología) y el juicio (teoría de la evaluación), este campo tiene por objeto responder a las preguntas estudiadas durante mucho tiempo en otras áreas sobre el discurso, utilizando nuevas herramientas proporcionadas por la minería de datos y la lingüística computacional.
Análisis de Sentimientos tiene muchos nombres. A menudo, se conoce como análisis de subjetividad, minería de opinión, y extracción de evaluación, con algunas conexiones con la informática afectiva (reconocimiento computacional y la expresión de la emoción) [22]. Este campo por lo general estudia los elementos subjetivos, definidos como "expresiones lingüísticas de los estados particulares en contexto"[21]. Estas suelen ser palabras sueltas, frases u oraciones. A veces, los documentos enteros son estudiados como una unidad de sentimiento, pero es generalmente aceptado que el sentimiento reside en pequeñas unidades lingüísticas [23]. Tanto el sentimiento, como la opinión a menudo se refieren a la misma idea, en este documento se utilizan los términos indistintamente.
Los sentimientos que aparecen en textos se ven de dos formas, la primera es explícitamente, donde la frase subjetiva directamente expresa la opinión ("Es un hermoso día"), mientras que la segunda es implícita, en donde el texto implica una opinión ("Los audífonos se quebraron en dos días") [24]. La mayoría de los trabajos realizados se han enfocado en el primer tipo de sentimiento, debido a que este es más fácil de analizar.

La polaridad de los sentimientos es una característica particular de los textos. Ésta se hace presente regularmente de forma dicotómica, positivo o negativo, a pesar de que también puede ser vista dentro de un rango. Un documento posee varias frases que demuestran opiniones, las cuales podrían tener una polaridad mixta, que es diferente a que estas no tuviesen polaridad. Yendo más lejos, se debe hacer una distinción entre la polaridad del sentimiento y la fuerza que este tiene.
Otra importante parte del sentimiento es el objetivo, pudiendo ser un objeto, un concepto, una persona o cualquier cosa. La mayoría de los trabajos han sido realizados sobre productos o criticas de películas, donde es fácil identificar el tópico del texto. Pero también es útil poner atención a la característica del objeto del cual el escritor se está refiriendo: "¿es la pantalla de la cámara o la duración de la batería el problema que más detectan los consumidores?" [25]. Debido a la disponibilidad de datos pertenecientes a comentarios de productos, por ello la extracción de características ha sido altamente estudiada en la década pasada [24]. La mención de estas características en los textos también puede ser explicita ("La duración de la batería es muy corta") o implícita ("La cámara es muy grande") [24].

Durante la última década, con el auge de las redes sociales, se han realizado muchos trabajos en el campo de la minería de datos orientados al análisis de sentimiento.

Un estudio destacado sobre el problema de la clasificación de opiniones en positivas o negativas, lo realiza Pang, 2002 [1], utilizando como datos las críticas de películas encontradas en la web. El hecho de que el usuario además de escribir una opinión, pueda evaluar con un número de estrellas la película en cuestión, hace que no sea necesario etiquetar manualmente cada una de las opiniones como positivas o negativas. En su estudio utilizan tres algoritmos ya utilizados anteriormente para tareas como la clasificación de textos por tema: Naive Bayes, maximum entropy (MaxEn) y support vector machines (SVM). La conclusión obtenida es que, a pesar de que la precisión del resultado del uso de métodos de aprendizaje automático supera los estándares producidos manualmente por un humano, éstos no tienen un rendimiento tan bueno como el que se obtiene al tratar el problema de categorización por tema, convirtiendo por tanto el problema de análisis de sentimiento en una tarea más compleja.

Los primeros estudios únicamente consideraban el aprendizaje a partir de ejemplos con una polaridad positiva o negativa, ignorando los ejemplos que muestran un sentimiento neutro. Existen estudios como el de Koppel et al., 2006 [2], en el que se muestra la importancia que tiene el uso de ejemplos neutrales en el proceso de aprendizaje, demostrando una mejor distinción entre polaridad positiva y negativa si se hace uso de éstos.

En cuanto al análisis de sentimiento aplicado a las redes sociales, podemos destacar el artículo en español de Grigori Sidorov, 2013 [3]. Exploran diferentes configuraciones para ver cómo cada una afecta a la precisión de los algoritmos de aprendizaje automático. Experimentan con los algoritmos de Naive Bayes, Decision Tree y SVM, dado que éstos ya han presentado buenos resultados para el idioma inglés. En sus configuraciones tienen en cuenta diferentes tamaños n-gram, la longitud del corpus, el número de clases de sentimientos, corpus balanceado vs. Corpus no balanceado y diferentes dominios para entrenar y testear (teléfonos móviles y política). En sus conclusiones determinan que la mejor configuración corresponde al uso de unigramas como características, un número tan pequeño como se pueda de clases (positivo y negativo), un tamaño de al menos 3000 tweets en el conjunto de entrenamiento (un tamaño superior no incrementa la precisión significativamente), un corpus no balanceado, muestra una ligera mejoría en los resultados y el clasificador con más precisión es el SVM.

Además, concluyen que el hecho de entrenar el sistema con tweets de un dominio diferente al que posteriormente se utilizará, empeora significativamente la precisión de los resultados, llegando a bajar del 85,8% al 28.0% en la prueba realizada con SVM.

En España, es interesante destacar el trabajo de la SEPLN (Sociedad Española para el Procesado del Lenguaje Natural) y el TASS (Taller de Análisis de Sentimientos), gracias al cual, diversos grupos de investigación españoles han presentado algoritmos sobre análisis de sentimiento con Twitter en castellano dando así soporte de recursos en castellano para el análisis de redes sociales, que, en la actualidad, se encuentra en un nivel muy bajo, sobre todo hablando de material open-source.

El TASS es un taller organizado de forma anual por la SEPLN para el análisis de sentimiento y el análisis de reputación online en el lenguaje castellano. El objetivo de este taller es proporcionar un foro para la discusión y la comunicación en las últimas investigaciones y desarrollos en el campo del análisis de opinión en las redes sociales, centrado específicamente en el idioma español, que puede ser visualizado y discutido por las comunidades científicas y empresariales. El principal objetivo es promover la aplicación del estado del arte de algoritmos y técnicas para el análisis de los sentimientos aplicado a las opiniones de textos extraídos de redes sociales (especialmente Twitter).

Un ejemplo del resultado de este taller es el presentado por Saralegi y San Vicente, 2013 [4]. Consiguieron en este taller los mejores resultados en la tarea de análisis de sentimiento a nivel global de tweet. El método de aprendizaje supervisado que presentan usa un clasificador SVM que construyen con la herramienta WEKA. Esta solución incluye un procesamiento basado en conocimiento lingüístico para preparar las variables/características que utilizará el clasificador. Dicho procesamiento incluye lematización y etiquetado POS (part-of-speech tagging) realizado con Freeling, etiquetado de polaridad para el que construyen su propio léxico de polaridad, tratamiento de emoticonos y de negación. Además, para aumentar su precisión realizan un pre-procesamiento del texto de los tweets en el que realizan correcciones ortográficas.

En la actualidad, herramientas que nos permitan realizar un análisis de sentimientos en tweets pueden ser [19]:

- Chatterscope. Es una herramienta gratuita que registra las menciones de una marca en Twitter, califica el sentimiento de cada tweet (positivos, negativos o neutros) y presenta unas estadísticas con el resumen de las menciones. Además, permite programar alertas que podremos recibir cada hora, cada día o cada semana y sacar una foto de lo que se está diciendo a tiempo real. Para empezar a monitorizar, tendremos que registrarnos y dejar funcionar la herramienta unos cinco días para obtener un conjunto de datos interesante para poder realizar una primera evaluación. Un detalle que me ha gustado mucho es la posibilidad de añadir palabras clave al análisis, palabras que serán utilizadas para evaluar los tweets.

- Twitter Sentiment. No necesita registro, se presenta como un buscador en el que introducir el término (marca, producto, etc) del que queremos evaluar los sentimientos de los tweets. Tras la búsqueda, obtendremos una representación gráfica de la evolución temporal de los tweets positivos y negativos, un interesante análisis de la

tendencia y la popularidad; además de un listado de los últimos tweets generados. Las búsquedas realizadas pueden ser almacenadas a través de nuestra cuenta de Gmail.

- Tweetfeel. Es una herramienta muy simple, tras introducir el término de búsqueda, iremos viendo un contador con los tweets con connotaciones positivas, los tweets negativos y un panel en el que irán apareciendo los tweets a tiempo real. Ideal para realizar un análisis rápido o, por ejemplo, para evaluar opiniones en un backchannel.

- Twitrrart. Es otra herramienta con la que podremos pulsar la opinión que se vierte en Twitter sobre un asunto concreto, tanto en términos positivos como negativos. La herramienta tiene programadas una serie de palabras clave que utilizar para asignar el "valor sentimental" del tweet. La interfaz me ha gustado mucho, puesto que nos agrupa los tweets en tres columnas, en base al valor asignado, por lo que se puede ver de un vistazo la "confrontación" de las opiniones.

Capítulo 3

Fases de desarrollo

El desarrollo del proyecto se ha dividido en las siguientes fases:

- Fase de análisis: En primer lugar, antes de comenzar a usar código, es importante definir de forma clara los objetivos y los límites de la aplicación realizando un análisis de requisitos. De esta forma cuando comencemos la implementación tendremos una idea clara de que vamos a hacer.

- Fase de extracción y preprocesamiento de datos: Al final de éste punto tendremos disponibles los datos necesarios para realizar el estudio sobre la tendencia o temática de entrada que hemos introducido previamente.
  Para esto en primer lugar tendremos que obtener, mediante una API de Twitter, todos los tweets que hablen sobre nuestra tendencia y posteriormente, mediante un análisis de sentimiento, transformar esos tweets en datos útiles que nos sirvan posteriormente para realizar el estudio.

- Fase de análisis o clasificación: Una vez que tenemos datos útiles, es hora de realizar el estudio. Para el desarrollo del proyecto realizaremos 2 tipos de estudios, un estudio léxico y un estudio mediante un algoritmo de clasificación.
  Al final de esta fase tendremos que haber conseguido obtener un valor de sentimiento basado en la posición geográfica y el momento temporal para una determinada tendencia.

- Fase de evaluación del algoritmo clasificador. Una vez que tenemos desarrollado el algoritmo clasificador es importante realizar una evaluación del mismo para verificar su nivel de precisión. Será fundamental ajustar los parámetros del algoritmo para adaptarlo a nuestro problema y elevar lo máximo posible la precisión del algoritmo.

- Fase de presentación de los datos: Por último y como paso más importante para el usuario final, es necesario mostrar de la forma más clara y precisa posible estos datos. Para ello se desarrollará una interfaz gráfica en la que mostrar gráficas y elementos de interacción con el usuario para hacer más sencilla la interacción.

En el anexo B se adjunta el cronograma asociado a las fases del proyecto.

Capítulo 4

Análisis

4.1 Descripción detallada de la solución.

Para el análisis de sentimiento en el proyecto se han empleado dos métodos distintos. Por un lado, tenemos el método sencillo, el basado en el léxico. En base a un listado de palabras con sentimiento positivo y otro listado con palabras con sentimiento negativo, realizamos un recuento de las mismas, siendo el resultado de polaridad el que mayor número de palabras contenga en nuestro texto analizado. El otro método para el análisis ha sido el empleo de un algoritmo clasificador Naive Bayes, implementado en la librería Sentiment para R, usada en el proyecto.

Para el análisis basado en el léxico, el recurso utilizado, como se detallará en el capítulo 5, ha sido un corpus de palabras clasificadas según su sentimiento en negativas o positivas.

Como entrada de la aplicación tenemos una página de presentación.

Desde aquí el usuario tiene la posibilidad de coger 3 rumbos distintos:

- Trending topics.

  A partir de 2 parámetros de entrada (país y estado o provincia), el sistema nos muestra un listado con los 50 trending topics más populares en ese país y provincia o estado.

  Éstos 2 parámetros de entrada son seleccionables tanto desde una lista desplegable como desde un mapa interactivo.

- Análisis basado en el léxico.

  Desde esta sección de la aplicación obtendremos un análisis de sentimiento basado en el léxico a partir de una serie de parámetros de entrada.

  El primero de estos parámetros de entrada es la tendencia o temática a analizar. Si anteriormente hemos realizado una búsqueda de trending topics nos permitirá la selección de uno de ellos. En otro caso también podemos introducirla manualmente desde un campo de texto.

  También se introduce mediante un slider el número de tweets que el sistema obtendrá para realizar el análisis. Además del idioma de éstos tweets y el rango de fecha de publicación de los mismos.

  Una vez introducidos los parámetros la aplicación realiza el proceso de análisis y nos muestra los resultados.

  Por un lado obtenemos una nube de palabras con las palabras más repetidas en los tweets analizados.

  En otro apartado obtenemos el resultado del análisis de sentimiento. En él se nos muestra la media y desviación típica del resultado obtenido. Un resultado menor que 0 indica un sentimiento negativo mientras que un resultado mayor que 0 indicará un sentimiento positivo. Cuanto más se aproxime el resultado a 0 más neutro será el sentimiento.

  Este resultado se completa con un histograma y un gráfico de caja o bigotes para facilitar la interpretación del resultado.

  También se nos muestran en otra sección los tweets analizados junto con la nota asignada por el algoritmo.

  Y por último una sección de análisis temporal, en la que se obtiene el sentimiento de dicha tendencia o temática en los últimos 9 días.

- Análisis con algoritmo Naive Bayes.

  Para este segundo análisis tenemos los mismos parámetros de entrada que para el anterior.

  Obtenemos las mismas salidas de nube de palabras y análisis de sentimiento, pero en este caso tenemos una sección nueva. El algoritmo también realizará un análisis de emoción clasificando los tweets dentro de una de las siguientes categorías: alegría, tristeza, enfado, disgusto, sorpresa y miedo.

4.2 Análisis de requisitos.

En el anexo A se incluye el análisis de requisitos completo junto con un diagrama UML de casos de uso modelando los mismos (véase Anexo B).

Capítulo 5

Recursos y herramientas

5.1 RStudio

El lenguaje de desarrollo de la aplicación es R. Se trata de un lenguaje de programación con un enfoque al análisis estadístico. La elección de éste se basa principalmente en que es uno de los más utilizados en el campo de la minería de datos, la investigación biomédica, la bioinformática y las matemáticas financieras aportando grandes ventajas en el análisis de datos y presentación de los mismos mediante gráficos.

Como entorno de desarrollo se opta por RStudio (ver Fig. 1) por ser en primer lugar open source además de ser el más usado ampliamente para el desarrollo en R. Cuenta con una interfaz simple que incluye una consola, un editor de código y herramientas para la depuración y gestión del espacio de trabajo.

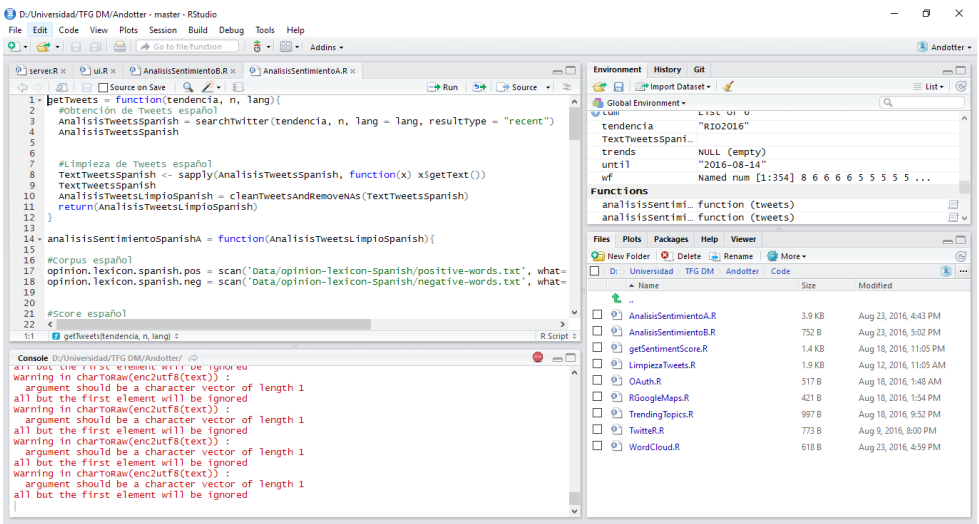


Fig. 1. Interfaz gráfica de RStudio

5.2 Shiny

Para el desarrollo de la interfaz de usuario se buscaba una herramienta simple y capaz de generar una interfaz dinámica y atractiva.

Aprovechando el uso de R en el proyecto se opta por emplear Shiny. Se trata básicamente de una potente herramienta o framework para R que nos permite el desarrollo de interfaces completamente interactivas con un conocimiento muy básico de html.

Los elementos de la interfaz se crean mediante llamadas a funciones de la propia herramienta lo que nos evita tener un gran conocimiento de html o javascript.

Gracias a la programación reactiva que implementa la herramienta obtenemos una interfaz completamente dinámica, actualizándose, por ejemplo, con solo introducir un valor en un campo de texto sin tener que pulsar posteriormente ningún botón.

Con solo instalar Shiny en nuestro proyecto como un paquete más de R y crear los archivos ui.r (contiene el frontend) y server.r (contiene el backend) ya estamos listos para comenzar a desarrollar la interfaz. Contiene la implementación de un servidor local, por lo que, una vez hecho esto con solo pulsar el botón de ejecutar ya tendremos la aplicación en ejecución [8, 11].

5.3 TwitteR

Necesitamos algo que nos permita conectar a Twitter y extraer tweets.

Esta funcionalidad nos la aporta TwitteR. Se trata de una API en R que nos permite conectar a la API de Twitter con nuestras credenciales personales de Developer y extraer tweets y trending topics mediante las funciones determinadas. El resultado de estas funciones lo obtenemos directamente en un data frame (estructura de datos característica de R). Todas las funciones cuentan con diversos parámetros que nos permiten ajustar la consulta a nuestro gusto.

Básicamente se trata de una API que implementa la funcionalidad de la API oficial de Twitter a R, puesto que ésta no tiene soporte para dicho lenguaje.

También se instala como un paquete más de R.

Resulta importante, llegados a este punto, destacar las obligadas limitaciones que posee la aplicación debido a las restricciones de la API de Twitter o API Rate Limits [12]. Todas las restricciones están divididas en ventanas o intervalos de 15 minutos. Esto quiere decir que pasados 15 minutos se renuevan las restricciones. Desde el paquete TwitteR tenemos una función encargada de darnos el estado actual de las restricciones para nuestra conexión. Las restricciones que nos afectan en la aplicación son las siguientes:

- Trending topics: 15 búsquedas por ventana (15 minutos).
- Tweets: 180 búsquedas por ventana.

También es importante mencionar que la API de Twitter restringe la búsqueda de tweets antiguos a solo los 9 días anteriores. Por esto, la capacidad de análisis temporal de la aplicación se verá afectada y quedará restringida a esta ventana temporal.

5.4 Leaflet

Leaflet es un paquete de R (rstudio.github.io/leaflet) que nos permite el uso de mapas interactivos en nuestra aplicación (ver Fig. 2). La librería está desarrollada en Javascript y es una de las librerías open-source más populares para el uso de mapas interactivos usada en paginas como The New York Times o GitHub.
Nos permite seleccionar distintos tipos de mapas y añadir en éstos marcadores mediante el uso de coordenadas. Además cuenta con funciones específicas para capturar eventos en el propio mapa que nos permiten por ejemplo, actualizar una variable de país con el valor del país que el usuario haya pulsado.

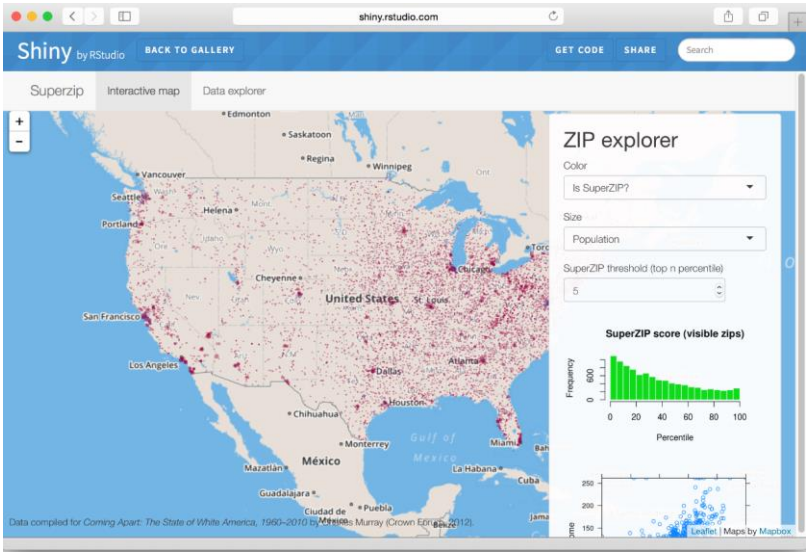


Fig. 2. Ejemplo de mapa interactivo desarrollado con Leaflet

5.5 Sentiment

Sentiment es un paquete de R que implementa un algoritmo de clasificación Naive Bayes para el análisis de sentimiento [18]. De momento solo es válido para textos en inglés.
Cuenta con funciones que nos permiten clasificar la polaridad y la emoción presente en un texto.
Dichas funciones cuentan con parámetros que nos permiten ajustar el clasificador.
El motivo de optar por éste paquete es por ser el más usado en el ámbito del análisis de sentimiento con R y el que mejores resultados obtiene gracias a los corpus que emplea.
El conjunto de entrenamiento usado para el clasificador de polaridad es el de Janyce Wiebe [13] Para el clasificador de emoción el conjunto de entrenamiento usado es el de Carlo Strapparava y Alessandro Valitutti [14] La ausencia de contenido open-source en español para el análisis de sentimiento impide el desarrollo del algoritmo de Naive Bayes para español.

5.6 Shinyapps.io

Uno de los inconvenientes del uso de herramientas framework como shiny es que estamos bastante limitados en el uso de servidores en los que realizar el despliegue de la aplicación.

Se ha optado finalmente por el plan gratuito de shinyapps.io (ver Fig. 3) con la que tenemos una limitación de 5 aplicaciones (nosotros solo necesitamos 1) y 25 horas activas por mes. Ésta es la limitación que más nos perjudica ya que la aplicación no podrá estar online todo el tiempo.

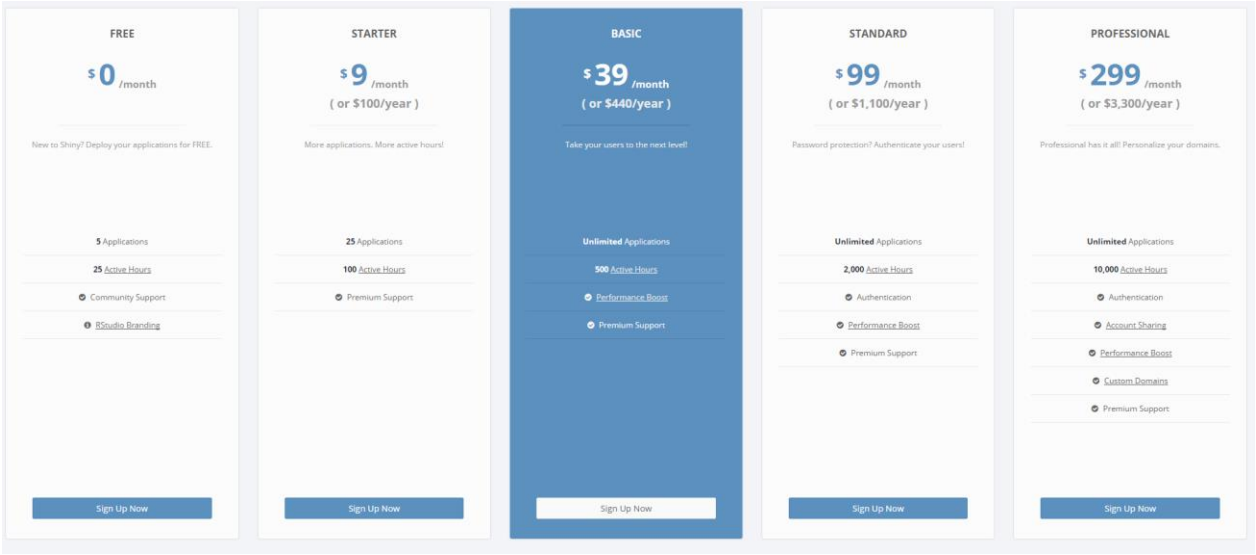


Fig. 3. Precios y restricciones de los planes ofertados por Shinyapps.io

5.7 Léxico de polaridad

Para el análisis de sentimiento basado en el léxico necesitamos de un conjunto de palabras clasificadas según su polaridad. Como éste análisis da soporte tanto a tweets en español como en inglés hemos necesitado de 2 conjuntos, uno para español y otro para inglés.

De los pocos recursos encontrados en español el que mejor resultados ha obtenido ha sido el proporcionado por la Fundación Elhuyar [15]. Éste se creó a partir de diferentes fuentes e incluye alrededor de 5200 palabras clasificadas en positivas y negativas.
El otro conjunto de palabras encontrado en español se trata de un conjunto traducido a partir de otro corpus en inglés. Esto hace que por la traducción muchas palabras cambien su polaridad, lo que finalmente radicaba en unos resultados bastante inexactos. Sin embargo el conjunto de la Fundación Elhuyar ha sido desarrollado manualmente por españoles nativos lo que da mayor robustez y fiabilidad a éste último.

El conjunto empleado consta de varias secciones.
Por una parte tenemos una sección con términos generales que son simplemente palabras tradicionales del castellano clasificadas con su polaridad (ver Fig. 4).
Una muestra de su contenido:

acuchillar	negative
acuerdo	positive
acusación	negative
acusar	negative
adaptarse	positive
adecuadamente	positive
adecuado	positive
adherente	positive
adherido	positive
adhesión	positive
adicción	negative
adicto	negative

Fig. 4. Ejemplo de términos generales del léxico de polaridad

Por otro lado, contamos con una sección de interjecciones (ver Fig. 5). Éstas palabras expresan sentimientos muy vivos por lo que son de gran influencia en el resultado final de clasificación.
Unos ejemplos de éstas interjecciones:

diantres	negative
dios	negative
ejem	negative
eureka	positive
fantástico	positive
fuera	negative
gualá	negative
guay	positive
hombre	negative
hurra	positive
jajaja	positive

Fig. 5. Ejemplo de interjecciones del léxico de polaridad

La siguiente sección se trata de un listado de coloquialismos (ver Fig. 6). Éstos son palabras o expresiones que se dicen de forma familiar o cotidiana. Aunque predominan en el lenguaje oral, también se dan en el lenguaje escrito gracias a las conversaciones electrónicas y los chats, por tanto, será importante tenerlos en cuenta en nuestro análisis.

bufar	negative
buitre	negative
bujarra	negative
cabestro	negative
cabrearse	negative
cabrón	negative
caer_gordo	negative
cagarse	negative
cagueta	negative
calzonazos	negative
canela_fina	positive
cansina	negative
cansino	negative
cante	negative
carajo	negative

Fig. 6. Ejemplo de coloquialismos del léxico de polaridad

Por último, contamos con una sección que nos resulta más que interesante. Ésta se compone de un conjunto de términos empleados generalmente en Twitter (ver Fig. 7).

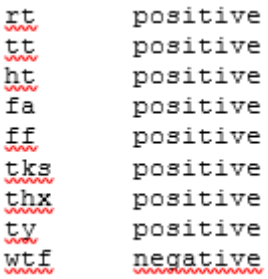


Fig. 7. Ejemplo de términos específicos de Twitter en el léxico de polaridad

El conjunto con palabras en inglés empleado es el elaborado por Minqing Hu y Bing Liu [16] y presentado en el artículo “Mining and Sumarizing Customer Reviews” [17]. Éste cuenta con un conjunto de términos generales en inglés clasificados como positivos o negativos según su polaridad.

Capítulo 6

Diseño

6.1 Arquitectura del sistema.

El sistema se divide en 6 módulos.

En la siguiente figura se puede ver un esquema gráfico general de la arquitectura aquí descrita (ver Fig. 8). En los próximos apartados del capítulo analizaremos en detalle el funcionamiento de cada módulo.

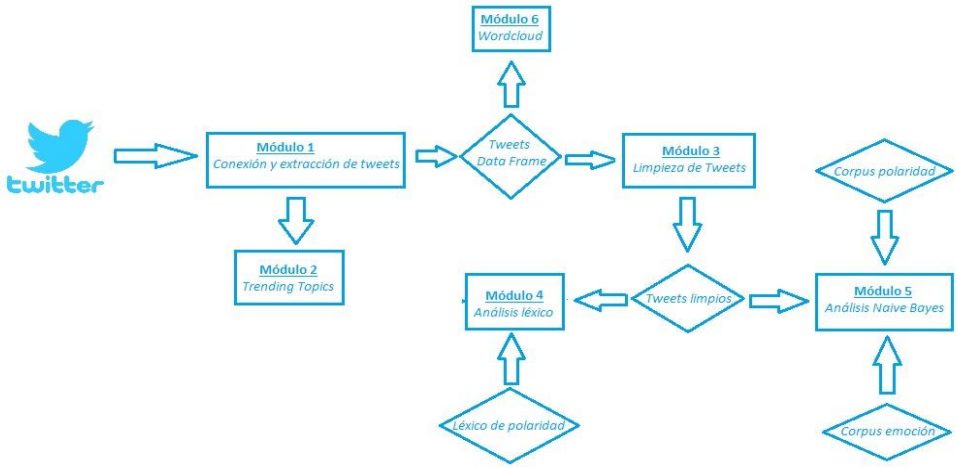


Fig. 8. Esquema general de la arquitectura del sistema

En primer lugar tenemos un módulo encargado de iniciar la actividad de la aplicación realizando la conexión con la API de Twitter y aportando la funcionalidad necesaria para extraer los tweets que posteriormente serán utilizados en los análisis.

Por otro lado tenemos un módulo que implementa la funcionalidad necesaria para extraer los trending topics solicitados por el usuario. Éste lo podríamos catalogar como un submódulo del módulo anterior.

Como tercer módulo tendríamos el encargado de recoger los tweets obtenidos por el primer módulo y realizar un proceso de limpieza sobre ellos para hacer posible su posterior análisis.

En este momento intervienen los módulos que podríamos denominar como núcleo del sistema, los encargados de realizar los análisis. Por un lado tendríamos el módulo encargado de realizar el análisis basado en el léxico que tomaría como entradas los tweets limpios generados por el módulo anterior y los los conjuntos de léxicos de polaridad.

Por otro lado tenemos el módulo encargado de realizar el análisis utilizando el algoritmo de Naive Bayes que, de nuevo, vuelve a usar como entrada los tweets limpios generados por el tercer módulo.

Aunque quizás no posea la envergadura suficiente para clasificarlo como módulo, puesto que es difícil de encajar en uno de los anteriores, añadiremos otro pequeño módulo encargado de generar el wordcloud o nube de palabras con las palabras mas repetidas en los tweets analizados. De nuevo, este módulo vuelve a emplear los tweets generados por el tercer módulo.

6.2 Conexión y extracción de tweets.

Como requisito indispensable antes de realizar cualquier operación en la aplicación es necesaria la conexión con la API de Twitter para la extracción de los datos. Para ello, en primer lugar, es necesario ir a la página de twitter de desarrolladores, registrarnos como desarrollador y crear una app, lo que nos generará unas credenciales para la posterior conexión (ver Fig. 9).

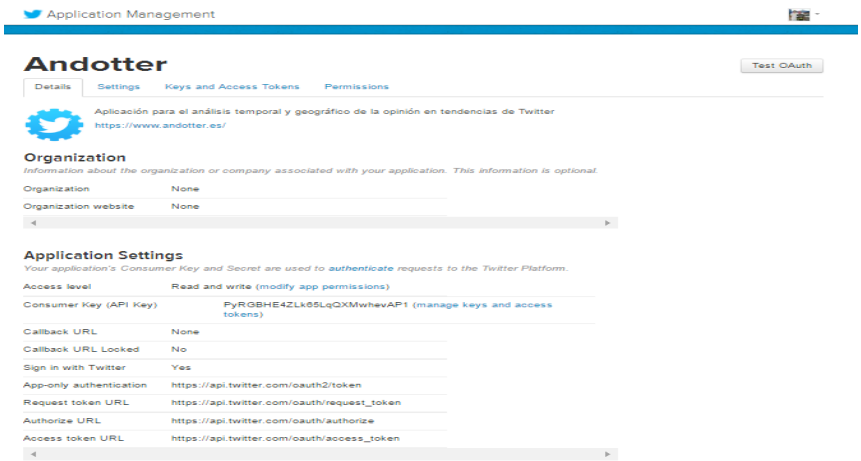


Fig. 9. Ejemplo de aplicación creada en twitter developers

Una vez que tenemos nuestra app creada generaremos los parámetros necesarios para la conexión con la API de Twitter. Éstos son: las api key pública y privada y los tokens público y privado. Con esto estamos listos para realizar la conexión con la función específica del paquete TwitteR.

Una vez que tenemos la conexión realizada con éxito el siguiente paso es la extracción de tweets. Para ello, el paquete TwitteR nos provee de una función que realizará la consulta a la API de Twitter en base a una serie de parámetros introducidos por nosotros:

- Número de tweets a buscar.
- Lenguaje de los tweets.
- Restricción de fecha para los tweets.
- Geolocalización.
- Tweets recientes, populares o una mezcla de ambos.

Con todo esto, una vez realizada la consulta, la función nos devuelve una lista de objetos tipo status. El tipo status es un objeto especial del paquete TwitteR que contiene toda la información correspondiente a un tweet:

- Texto del tweet.
- Nombre del usuario que lo ha publicado.
- Id
- Fecha de publicación.
- Número de retweets y favoritos.

Una vez extraídos los tweets y almacenados en una lista, eliminamos en primer lugar todos los tweets retwiteados con el objetivo de realizar el análisis solo con tweets puros. Una vez eliminados nos quedamos solamente con el texto del tweet que es lo que nos interesa para el análisis y ya tenemos nuestro conjunto de tweets listo para pasar al siguiente módulo.

A continuación se muestra un ejemplo de conjunto de tweets que tendríamos al final de éste módulo en relación al IPhone 7 (ver Fig. 10).

```
[1] "iPhone 7 plateado va a combinar muy bien con los tubos del transporte público."
[2] "Yo soñando con el iPhone 6 y ahora sale el 7 . .."
[3] "Llega el nuevo iPhone 7 sin audífonos: https://t.co/xQtzO5dl1s"
[4] "Ni vendiendo todos los organos de mi cuerpo me alcanza para comprar el iphone 7"
[5] "es joda? iphone 7?\napenas tengo el 6 andate a la concha de tu madre apple"
[6] "algunas ya están diciendo que compraran el iPhone 7 y yo lo que puedo decir es que: Vivo en Venezuela"
[7] "@Unialarga el de moda es el originalísimo \"No he comprado el iPhone 7 y ya perdí sus audífonos\""
[8] "Todos pidiendo el iPhone 7 y yo queriendo por lo menos alguno"
```

Fig. 10. Conjunto de tweets obtenidos tras el primer módulo. Tendencia: IPhone 7

6.3 Trending Topics.

Éste módulo bien podríamos catalogarlo como un submódulo del módulo anteriormente descrito ya que seguimos usando una función del paquete TwitteR para la obtención de los principales trending topics.

En primer lugar, necesitamos saber las localizaciones disponibles en la API de Twitter para la obtención de los trending topics. Tenemos una función encargada de ello que nos devuelve un data frame con los países disponibles, sus provincias o estados y un identificador que usaremos más adelante para extraer los trending topics (ver Fig. 11).

Den Haag	Netherlands	726874
Amsterdam	Netherlands	727232
Rotterdam	Netherlands	733075
Utrecht	Netherlands	734047
Barcelona	Spain	753692
Bilbao	Spain	754542
Las Palmas	Spain	764814
Madrid	Spain	766273
Malaga	Spain	766356
Murcia	Spain	768026
Palma	Spain	769293
Seville	Spain	774508
Valencia	Spain	776688
Zaragoza	Spain	779063
Geneva	Switzerland	782538
Lausanne	Switzerland	783058
Zurich	Switzerland	784794

Fig. 11. Parte de un data frame devuelto por la función que devuelve las localizaciones

Una vez obtenidas las diferentes localizaciones disponibles, se cargan en la interfaz de usuario para que el mismo seleccione de que localización desea buscar los trending topics. Una vez tenemos la localización simplemente llamamos a la función encargada de obtener los trending topics pasándole por parámetro el id de la localización escogida. Una vez realizada la consulta, la función nos devuelve un data frame con el trending topic, su url, un nombre para consultas y un id. Basándonos en el ejemplo anterior, si queremos buscar los trending topics para Madrid, introduciríamos el id 766273.

A continuación se ilustra una muestra del resultado obtenido (ver Fig. 12):

1	#Hipnotizame2	http://twitter.com/search?q=%23Hipnotizame2	%23Hipnotizame2	766273
2	#AppleEvent	http://twitter.com/search?q=%23AppleEvent	%23AppleEvent	766273
3	#ChiringuitoRaul	http://twitter.com/search?q=%23ChiringuitoRaul	%23ChiringuitoRaul	766273
4	#JuegosParalimpicos	http://twitter.com/search?q=%23JuegosParalimpicos	%23JuegosParalimpicos	766273
5	Guardianes de la Galaxia	http://twitter.com/search?q=%22Guardianes+de+la+...	%22Guardianes+de+la+Galaxia%22	766273
6	#STOPCensuraTwitterSpain	http://twitter.com/search?q=%23STOPCensuraTwitt...	%23STOPCensuraTwitterSpain	766273
7	PS4 Pro	http://twitter.com/search?q=%22PS4+Pro%22	%22PS4+Pro%22	766273
8	Shawn	http://twitter.com/search?q=Shawn	Shawn	766273
9	Murray	http://twitter.com/search?q=Murray	Murray	766273
10	El Verdugo	http://twitter.com/search?q=%22El+Verdugo%22	%22El+Verdugo%22	766273

Fig. 12. Ejemplo de Trending Topics obtenidos para Madrid.

Una vez tenemos el conjunto de trending topics nos quedamos solamente con el nombre y la url que será la información que mostremos al usuario.

Por último, detallaremos la función implementada para establecer la localización también desde el mapa interactivo. Como se detalló anteriormente, la funcionalidad del mapa interactivo se ha implementado con el paquete leaflet. Leaflet permite marcar puntos en el mapa mediante la introducción de coordenadas (latitud y longitud), por lo que, para marcar los países obtenidos mediante la función que nos indica que países permiten la obtención de trending topics necesitamos conocer de algún modo su longitud y latitud. Para ello se ha hecho uso de un archivo .csv (ver Fig. 13) con las coordenadas de todos los países. Éste archivo es cargado en un data frame en tiempo de ejecución.

	iso3166	latitude	longitude	nombre
1	AD	42.5000	1.5000	Andorra
2	AE	24.0000	54.0000	United Arab Emirates
3	AF	33.0000	65.0000	Afghanistan
4	AG	17.0500	-61.8000	Antigua and Barbuda
5	AI	18.2500	-63.1667	Anguilla
6	AL	41.0000	20.0000	Albania
7	AM	40.0000	45.0000	Armenia
8	AN	12.2500	-68.7500	Netherlands Antilles
9	AO	-12.5000	18.5000	Angola
10	AQ	-90.0000	0.0000	British Antarctic Territory
11	AQ	-90.0000	0.0000	Ross Dependency
12	AQ	-90.0000	0.0000	Queen Maud Land
13	AQ	-90.0000	0.0000	Peter I Island

Fig. 13. Muestra del data frame con la localización de todos los países.

Una vez que tenemos el data frame con las localizaciones disponibles para la obtención de trending topics simplemente hacemos un merge de los dos data frames y obtenemos un nuevo data frame resultante con el contenido repetido en ambos. De esta forma obtenemos un data frame con solo los países que queremos representar en el mapa y sus coordenadas. Pasando este data frame a la función correspondiente, nos dibuja un marcador en cada país (ver Fig. 14).

```
countryInput = subset(merge, country == countryIn)
countryInput = subset(countryInput, select = c(longitude, latitude, country))
m <- leaflet()
m <- addTiles(m)
m <- addMarkers(m, geos$longitude , geos$latitude, layerId = geos$country)
m <- addCircleMarkers(m, countryInput$longitude , countryInput$latitude)
m <- setView(map = m, lng = countryInput$longitude, lat = countryInput$latitude, zoom = 3)
m
```

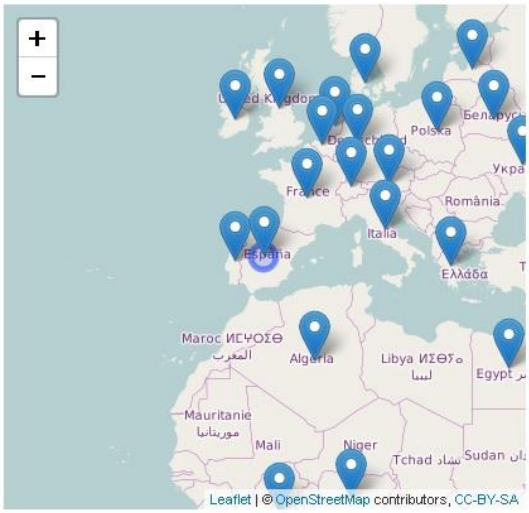


Fig. 14. Muestra de fragmento de código encargado de dibujar el mapa y los marcadores y el mapa resultante.

6.4 Limpieza de tweets.

Los tweets siempre van a contener una gran cantidad de “ruido” (palabras que no aportan información al estudio) que va a entorpecer el trabajo del algoritmo de clasificación e incluso sabotear el resultado final. Podemos encontrarnos tanto signos de puntuación, números, etiquetas, urls y diversos elementos más que no son necesarios para la clasificación pero que si los dejamos pueden hacer que el algoritmo de un resultado equivocado o directamente falle.

A continuación se detalla el proceso de limpieza que se lleva a cabo para cada tweet [9].

- En primer lugar, convertimos todos los caracteres del tweet a minúscula.
- Seguidamente eliminamos cualquier etiqueta html que pueda estar incrustada en el tweet.
- Posteriormente borramos cualquier etiqueta de retweet (RT|via) o cualquier otra procedente de Twitter que pueda haber presente en el tweet.
- El siguiente paso consiste en eliminar los hastags presentes en el tweet.
- Seguidamente eliminamos cualquier mención (@) a algún usuario.
- Posteriormente eliminamos todos los signos de puntuación (comas, puntos, etc.)
- Eliminamos los dígitos.
- Y, por último, eliminamos las tildes. Previamente también se han eliminado todas las tildes del léxico de polaridad. De esta forma evitamos que un usuario haya escrito una palabra sin tilde y el algoritmo no la detecte.

Tras eliminar los tweets repetidos, obtenemos nuestro conjunto de tweets limpio listo para ser analizado por el algoritmo. A continuación, se muestra un ejemplo del conjunto de tweets antes y después de pasar por éste módulo (ver Fig. 15):

Fig. 15. Conjunto de tweets antes y después de pasar por el módulo de limpieza.

6.5 Análisis basado en el léxico.

Llegados a este punto ya tenemos el conjunto de tweets listo para ser analizado. A continuación, vamos a detallar por un lado el funcionamiento del algoritmo clasificador basado en el léxico y en la siguiente sección el de Naive Bayes.

En primer lugar, debemos de saber si el análisis lo estamos realizando sobre tweets en español o tweets en inglés ya que de esto dependerá el léxico de polaridad que el algoritmo cargará para realizar la clasificación.

También tendremos que introducir si queremos realizar el análisis sobre un lugar geográfico específico o no mediante una lista desplegable que contiene todos los países disponibles.

Una vez localizado el idioma, el algoritmo carga el archivo correspondiente a memoria en forma de 2 listas de cadenas de caracteres, una con las palabras clasificadas con una polaridad positiva y otra con las negativas.

Una vez hecho esto pasamos a lo que sería el kernel del módulo, el algoritmo encargado de calcular el valor de sentimiento para cada tweet. En primer lugar, se transforma cada tweet (cadena de caracteres) en una lista de cadenas de caracteres donde cada posición de la lista será una palabra del tweet.

A continuación, comparamos la lista de palabras con la lista de palabras positivas en primer lugar y guardamos el número de ocurrencias. Seguidamente hacemos lo mismo con la lista de palabras negativas. Así tenemos dos valores enteros, uno con el número de palabras positivas en el tweet y otro con el número de palabras negativas.

Por último, calculamos el score final del tweet simplemente restando el número de palabras negativas al número de palabras positivas. Éste sería el valor final de polaridad para este tweet. Si éste es mayor que cero quiere decir que hay más palabras positivas que negativas y, por tanto, denota un sentimiento positivo. Si por el caso contrario, el score es menor que 0, tenemos más palabras negativas que positivas y estamos ante una polaridad negativa. Un valor 0 de score de puede expresar 2 cosas, un sentimiento neutro o que el algoritmo no ha sido capaz de encontrar palabras del tweet en el léxico de polaridad usado y, por tanto, no es capaz de clasificar el tweet. Esto presenta un problema, y es que si añadimos los 0 de este segundo caso al resultado global final de todos los tweets estaremos alterando la media con valores irreales, por tanto, los casos en los que el resultado sea 0 debido a que el algoritmo no ha encontrado palabras negativas ni positivas se eliminan para el cálculo del resultado final.

A continuación se muestra un ejemplo de lo que tendríamos llegados a este punto. Un data frame con el tweet y su score (representado por NA los que el algoritmo no ha podido clasificar)

El conjunto de tweets limpios analizados con la tendencia “Iphone 7” es el siguiente (ver Fig. 16):

```
 [1] "el tema son los muchos usuarios de apple rageando por el jack que luego terminaran comprando el iphone y apple seguira"
 [2] "airpods inalambricos y todo lo nuevo de apple conocelo"
 [3] "apple presento el iphone y la nueva generacion de su reloj inteligente"
 [4] "la camara del iphone plus es la mejor de apple y eso que significa"
 [5] "todo lo que tienes que saber del nuevo iphone esta en"
 [6] "ahora ire por ti iphone 孜猛慎驼邪猛孜猛邪拖邪将孜猛慎驼邪猛孜猛邪柃邪非"
 [7] "diferencias entre el iphone con el iphone s y el samsung galaxy s"
 [8] "acuaticos o por lo menos resistentes al agua como el iphone"
 [9] "que tiene el iphone que no tienen las versiones anteriores de iphone"
[10] "don t blink new iphone"
[11] "htc trollea a apple durante la presentacion del iphone"
[12] "iphone sin conexion de auriculares la ausencia mas polemica"
[13] "iphone  caracteristicas especificaciones con ios de apple …"
[14] "la camara del iphone aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa"
```

Fig. 16. Tweets limpios para ejemplo de análisis basado en léxico

Y este el resultado (ver Fig. 17):

	Tweet_procesado	score
1	el tema son los muchos usuarios de apple rageando ...	NA
2	airpods inalambricos y todo lo nuevo de apple cono...	2
3	apple presento el iphone y la nueva generacion de s...	1
4	la camara del iphone plus es la mejor de apple y eso ...	1
5	todo lo que tienes que saber del nuevo iphone esta en	2
6	ahora ire por ti iphone <U+653C><U+3E64><U+613C...	NA
7	diferencias entre el iphone con el iphone s y el sams...	NA
8	acuaticos o por lo menos resistentes al agua como el...	NA
9	que tiene el iphone que no tienen las versiones ante...	-1
10	don t blink new iphone	1
11	htc trollea a apple durante la presentacion del iphone	NA
12	iphone sin conexion de auriculares la ausencia mas ...	-1
13	iphone <U+F8FF> caracteristicas especificaciones c...	NA
14	la camara del iphone aaaaaaaaaaaaaaaaaaaaaaaaaaa...	NA

Fig. 17. Ejemplo resultado del algoritmo de análisis basado en el léxico

Observamos como el algoritmo, en los 7 tweets clasificados, acierta en la predicción de su polaridad. También podemos observar cómo los tweets 1 y 11 presentan un claro sentimiento negativo aunque sin embargo, el algoritmo no ha sido capaz de clasificarlos. Esto se basa principalmente en el léxico de opinión, ya que no incorpora términos como “trollea” o “rageando” Dado el caso, se podrían ir añadiendo dichos términos para ir mejorando el algoritmo poco a poco.

Finalmente, para acabar el análisis, tenemos que dar un resultado único que indique el sentimiento global hacia la tendencia analizada. Para ello calculamos la media y la desviación típica de los resultados obtenidos. Para el ejemplo anteriormente mostrado obtendríamos el siguiente resultado:

Score: 1
Desviación típica: 1.25

La interpretación del resultado es simple. En general se habla de forma positiva sobre el IPhone 7 (score mayor que cero) y tenemos una desviación típica pequeña, lo que indica que la media no está “contaminada” por valores muy dispersos. Una desviación típica demasiado alta podría hacernos saltar la alarma. Podría ser que algún conjunto de tweets mal clasificados por alguna razón nos esté alterando el resultado final.

El algoritmo también realiza un análisis temporal de los últimos 9 días (en el apartado 4.3 se detalla el motivo de este rango de tiempo tan pequeño). Para ello, simplemente repetimos 9 veces el proceso descrito, obteniendo para cada iteración, los tweets correspondientes solo al día que estamos analizando. Un ejemplo de lo que obtendríamos manteniendo la tendencia de "IPhone 7" es el siguiente (ver Fig. 18):

dias	score
8	1
7	1
6	1
5	1
4	1
3	0
2	1
1	0
31	0

Fig. 18. Análisis temporal para la tendencia "IPhone 7"

Este tipo de análisis, sin contar con la restricción de la API de Twitter, tiene una aplicación más que interesante. En el caso del ejemplo mostrado podríamos, por ejemplo, observar como varía la opinión de los usuarios en torno al IPhone 7 conforme nos acercamos a la fecha de su presentación y después de ella. Ésta sería una forma más que interesante de obtener un feedback acerca del éxito de la presentación.

6.6 Análisis con algoritmo Naive Bayes.

Detallado el funcionamiento del algoritmo basado en el léxico pasamos al de Naive Bayes.

Antes de nada, para colocarnos en situación veamos por encima que es un algoritmo Naive Bayes. El algoritmo comúnmente conocido como Naive Bayes es una de las implementaciones más populares de una red bayesiana. Una red bayesiana es un clasificador estadístico. Éste predice la probabilidad de que, dada una tupla, ésta pertenezca a una clase concreta. Ésta clasificación se basa en el teorema de Bayes.

Para usar un algoritmo Naive Bayes es indispensable entrenarlo. Esto se hace mediante un conjunto de datos de entrenamiento, que, en nuestro caso, es un corpus con una gran cantidad de tweets ya clasificados manualmente. El paquete sentiment [18], ya implementa un algoritmo Naive Bayes entrenado con 2 corpus distintos. Uno para la clasificación de polaridad y otro para la clasificación de emoción. El algoritmo, de momento, solo cuenta con corpus en inglés, por tanto, la clasificación solo podremos realizarla sobre tweets en inglés. Las referencias a los corpus usados por el paquete sentiment se detallan en la sección 4.5 de este documento.

Una vez implementado y entrenado el algoritmo, lo único que nos queda por hacer es seleccionar el tweet, procesarlo con los atributos empleados por el algoritmo y obtener la clasificación. De nuevo, es necesario el proceso de obtención y limpieza de tweets anteriormente descrito. Vamos a representar un ejemplo de lo que obtendríamos con éste algoritmo.
Por no ser repetitivos vamos a cambiar de tendencia. Ésta vez vamos a analizar la tendencia “no mans sky”.

Tras la obtención y limpieza de tweets obtenemos el siguiente conjunto (ver Fig. 19):

```
 [1] "no mans sky"
 [2] "everyone s so salty about no mans sky and yet spore galactic adventures included most of those features everyone complained were missing"
 [3] "no man s sky"
 [4] ""
 [5] "no mans sky ep"
 [6] "i agree but it s cool that most people don t"
 [7] "uc collection is on ps god of war no mans sky street fighter desgea lbp tlou killzone i can keep goin"
 [8] "in case you get bored playing no mans sky on your pc"
 [9] "too bad no mans sky died out in less than a month"
[10] "devin what are you playing no mans sky calls back this is good now because of dabbing"
[11] "deus ez should definitely be infront of madden and no mans hope sky"
[12] "space exploration base building and sweet sweet mountains in"
[13] "think i m going to skip deus ex tonight and chill with some no mans sky"
```

Fig. 19. Conjunto de tweets limpios para la tendencia “no mans sky”

La función encargada de realizar el análisis de polaridad nos devuelve el siguiente data frame.

	POS	NEG	POS/NEG	BEST_FIT
1	1.03127774142571	0.445453222112551	2.31512017476245	positive
2	8.78232285939751	0.445453222112551	19.7154772340574	positive
3	1.03127774142571	0.445453222112551	2.31512017476245	positive
4	1.03127774142571	0.445453222112551	2.31512017476245	positive
5	1.03127774142571	0.445453222112551	2.31512017476245	positive
6	8.78232285939751	0.445453222112551	19.7154772340574	positive
7	1.03127774142571	8.08917567883756	0.127488607290811	negative
8	1.03127774142571	17.1191924966825	0.0602410272345239	negative
9	1.03127774142571	16.4260453161225	0.0627830814769203	negative
10	15.1470736162494	0.445453222112551	34.0037356659242	positive
11	15.8402207968094	8.78232285939751	1.8036481976815	neutral
12	8.08917567883756	8.78232285939751	0.921074732544335	negative
13	8.78232285939751	8.08917567883756	1.08568823426265	neutral

Fig. 20. Resultado análisis de polaridad para tendencia “no mans sky”

Las columnas POS y NEG nos indican el valor de probabilidad calculado por el algoritmo de que la tupla analizada pertenezca a esa clase. La columna POS/NEG indica simplemente la división de los dos valores anteriores. Un valor de 1 indica una polaridad neutra, un valor menor que 0 negativa y mayor que 0 positiva. La columna BEST_FIT nos indica la mejor clasificación en base a lo descrito anteriormente. Del mismo modo obtenemos la clasificación de emoción para los tweets. Para este caso obtenemos el siguiente resultado (ver Fig. 21):

	ANGER	DISGUST	FEAR	JOY	SADNESS	SURPRISE	BEST_FIT
1	1.46871776464786	3.09234031207392	2.06783599555953	7.34083555412328	1.7277074477352	2.78695866252273	joy
2	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
3	1.46871776464786	3.09234031207392	2.06783599555953	7.34083555412328	1.7277074477352	2.78695866252273	joy
4	1.46871776464786	3.09234031207392	2.06783599555953	7.34083555412328	1.7277074477352	2.78695866252273	joy
5	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
6	1.46871776464786	3.09234031207392	2.06783599555953	13.6561935556456	1.7277074477352	2.78695866252273	joy
7	1.46871776464786	3.09234031207392	2.06783599555953	7.34083555412328	1.7277074477352	2.78695866252273	joy
8	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
9	1.46871776464786	3.09234031207392	2.06783599555953	7.34083555412328	1.7277074477352	2.78695866252273	joy
10	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
11	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
12	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
13	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
14	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
15	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	1.7277074477352	2.78695866252273	NA
16	1.46871776464786	3.09234031207392	2.06783599555953	1.02547755260094	7.34083555412328	2.78695866252273	sadness

Fig. 21. Resultado análisis de emoción para tendencia "no mans sky"

De nuevo, los valores de cada columna indican el valor de probabilidad de que la tupla analizada pertenezca a dicha clase. En la columna BEST_FIT obtenemos la mejor clasificación calculada. En los casos en los que el algoritmo no es capaz de clasificar la tupla el valor para esta columna es NA.

Observamos que para este caso la mayoría de los tweets se clasifican con una emoción de alegría. Es importante indicar que esto es solo un ejemplo ilustrativo. Para obtener unos resultados más fiables es importante ampliar en gran medida el número de tweets analizados.

6.7 Nube de palabras (Wordcloud).

Quizás esta parte de la aplicación no tiene la envergadura suficiente como para catalogarla en un módulo aparte, pero con tal de facilitar la comprensión del diseño de la aplicación se ha optado por hacer de ella un módulo independiente.

Una nube de palabras o su término más conocido, wordcloud, es una forma de observar de forma muy simple la tendencia de un texto. Esta es una representación visual de las palabras que conforman un texto, donde cuanto mayor tamaño mayor es la frecuencia de aparición. En nuestro contexto, esto no resulta muy útil para observar a simple vista en que se está haciendo más hincapié dentro la tendencia que estamos analizando.

La creación de un wordcloud en R resulta muy simple gracias al paquete wordcloud que contiene la funcionalidad necesaria para dibujarla. Sin embargo, antes de esto, tenemos que realizar un pequeño proceso.

En cada lenguaje tenemos un conjunto de palabras catalogadas como palabras vacías o stopwords. Esto no son mas que las palabras sin significado como artículos, pronombres, preposiciones, etc. Para generar un wordcloud que resulte realmente útil es muy importante la eliminación de estas stopwords antes. Una vez eliminadas se junta todo en un corpus, posteriormente en una matriz de términos, se ordenan las palabras de mas frecuencia a menos y por último se llama a la función del paquete wordcloud que nos dibuja el gráfico.

A continuación, se muestra el ejemplo de un wordcloud generado a partir de la tendencia “RIO 2016” (ver Fig. 22)

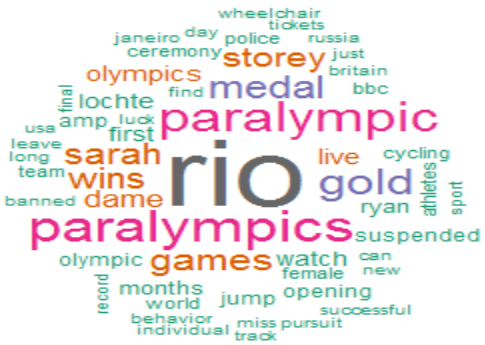


Fig. 22. Wordcloud generado a partir de la tendencia “RIO 2016”

6.8 Diseño de la interfaz de usuario.

Una vez descritos todos los módulos del sistema pasamos a presentar el diseño de la interfaz de usuario. Uno de los objetivos de la aplicación era conseguir una interfaz de usuario amigable y dinámica. Shiny nos ha permitido cumplir con esto gracias a su plantilla interna y su programación reactiva. La creación de interfaces en Shiny se realiza mediante el uso de componentes con un diseño ya programado, por lo que, no necesitamos tocar CSS.

- **Página principal.**

La página principal es simplemente la página de bienvenida de la aplicación en la que se da un poco de información acerca de la misma e información de contacto (ver Fig. 23). Además en la parte superior podemos comprobar la cantidad de búsquedas de tweets y trending topics que quedan para alcanzar el límite permitido por la API.

El menú, situado en la parte izquierda, es dinámico: se puede extender y contraer y su contenido varía dependiendo de la página en la que nos encontremos.

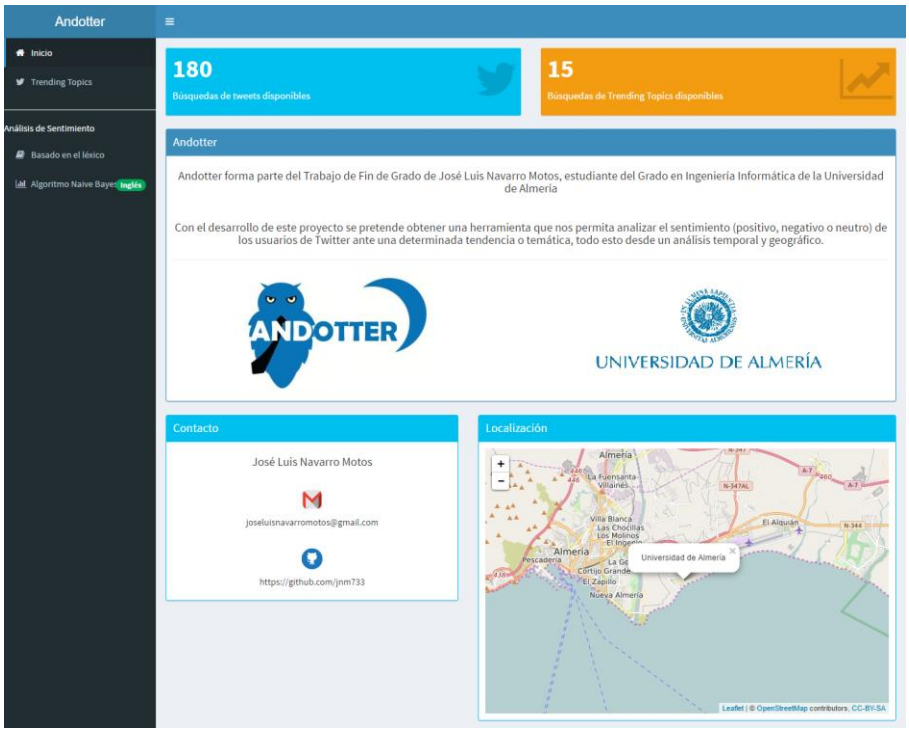


Fig. 23. Página principal

- **Trending topics.**

Esta es la página encargada de contener la funcionalidad correspondiente a la búsqueda de trending topics (ver Fig. 24).

Observamos que en el menú se añade una nueva sección para introducir la localización sobre la que queremos buscar los trending topics. Además contamos con un mapa interactivo en la parte derecha desde el cual también podemos seleccionar la localización simplemente pinchando sobre una de las ubicaciones disponibles.

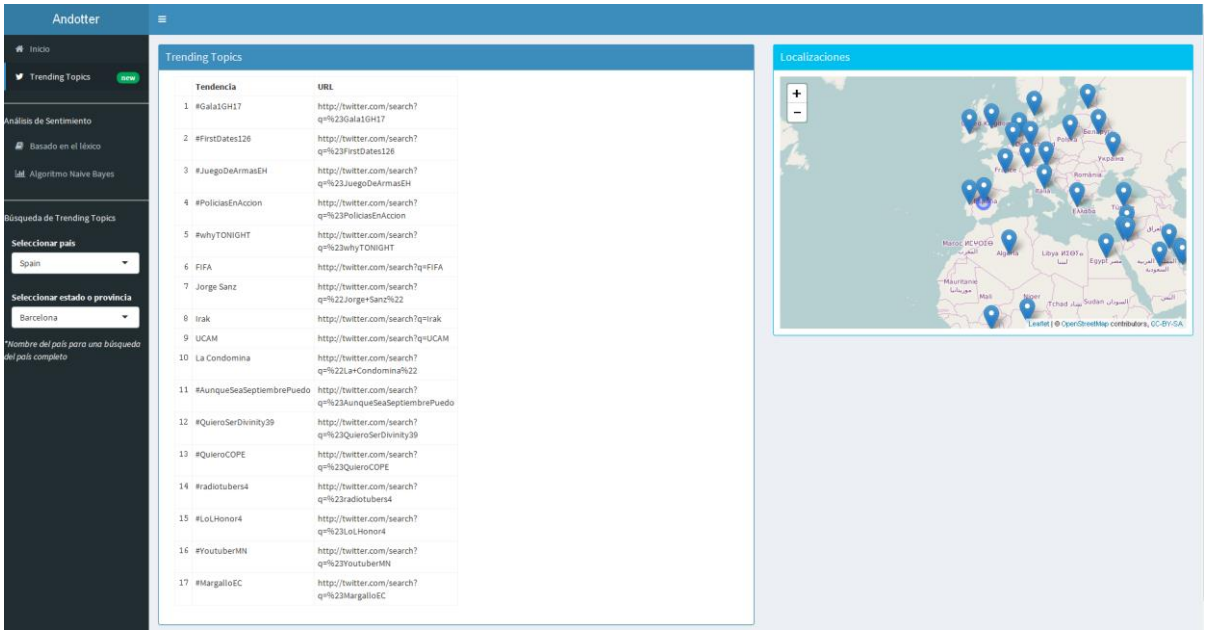


Fig. 24. Página de trending topics.

- **Análisis basado en el léxico.**

A continuación pasamos a detallar la página encargada de ofrecer el análisis basado en el léxico (ver Fig. 25).

En el menú nos aparecen una serie de parámetros de entrada para configurar la consulta. En primer lugar tenemos una lista desplegable para seleccionar un trending topic a partir del cual realizar el análisis (para esto es indispensable haber realizado primero una búsqueda de trending topics). También podemos introducir la tendencia de forma manual desde el siguiente campo de texto. Además podemos seleccionar el número de tweets sobre el que realizar el análisis, el idioma de dichos tweets y el rango de fecha de publicación.

Para la visualización del resultado se muestran un histograma y un gráfico de caja además del wordcloud y un gráfico de línea para el análisis temporal [10]. En otra sección se muestra una vista detalle con la clasificación realizada por el algoritmo para cada tweet.

- *Fig. 25. Ejemplo de análisis basado en el léxico para la tendencia "Benfica"*

- **Análisis con algoritmo Naive Bayes.**

La página encargada de presentar la funcionalidad del análisis con el algoritmo Naive Bayes presenta un aspecto bastante similar a la del análisis basado en el léxico (ver Fig. 26).

De nuevo volvemos a tener el wordcloud en la parte derecha de la ventana y la vista detalle justo debajo. Pero en este caso en lado izquierdo se muestran los resultados de los dos análisis realizados, el de polaridad y el de emoción. Éstos se muestran detallados mediante texto y se acompañan de unos gráficos de barras para una mejor visualización.

En la vista detallada se muestra cada tweet con su clasificación tanto de polaridad como de emoción.

Para esta página se utilizan un nuevo estilo de gráficos generados mediante la librería ggplot2 que dan una mejor visión para estos tipos de histogramas con varias variables distintas.

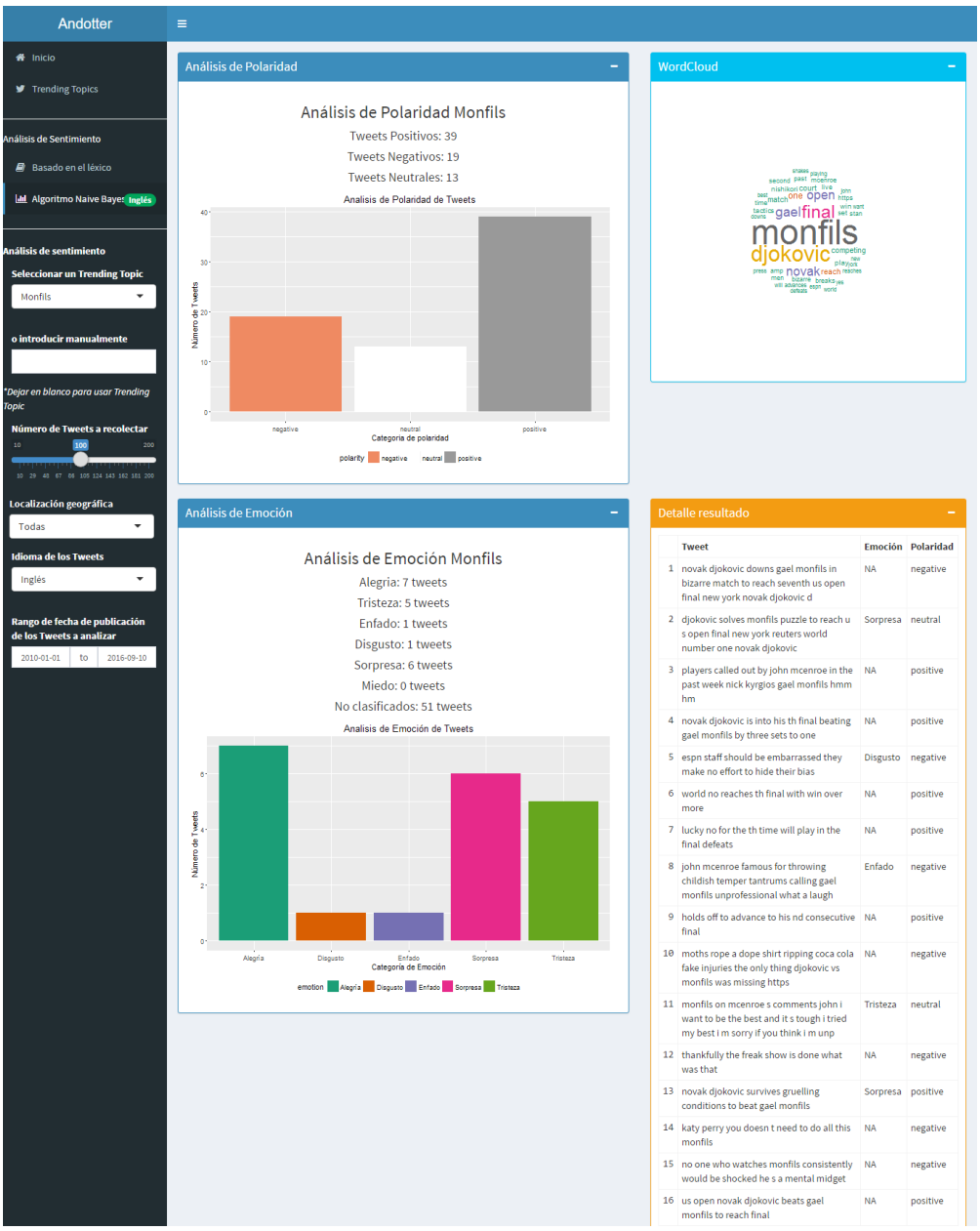


Fig. 26. Ejemplo de análisis con algoritmo Naive Bayes para la tendencia "Monfils

6.9 Deploy con Shinyapps.io

La gran ventaja del uso de un servidor de shinyapps.io es el gran ahorro de tiempo que ganamos con la instalación y configuración del servidor.

Para realizar el deploy de la aplicación desde nuestro proyecto local en R, shinyapps.io se requiere del paquete rsconnect que se instala en nuestra instalación local de R como un paquete mas [11].

Tras crear una cuenta y una aplicación en la web de shiynapps.io tenemos que generar un par de tokens (público y privado) Con estos tokens pasamos a configurar la conexión de la cuenta en nuestro proyecto local mediante una función del paquete rsconnect.

Una vez configurada la conexión, con simplemente llamar a la función deployApp() de rsconnect nuestra aplicación se subirá al servidor y estará completamente accesible desde su url.

La url en la que se encuentra desplegada la aplicación es:

https://andotter.shinyapps.io/Andotter/

Capítulo 7

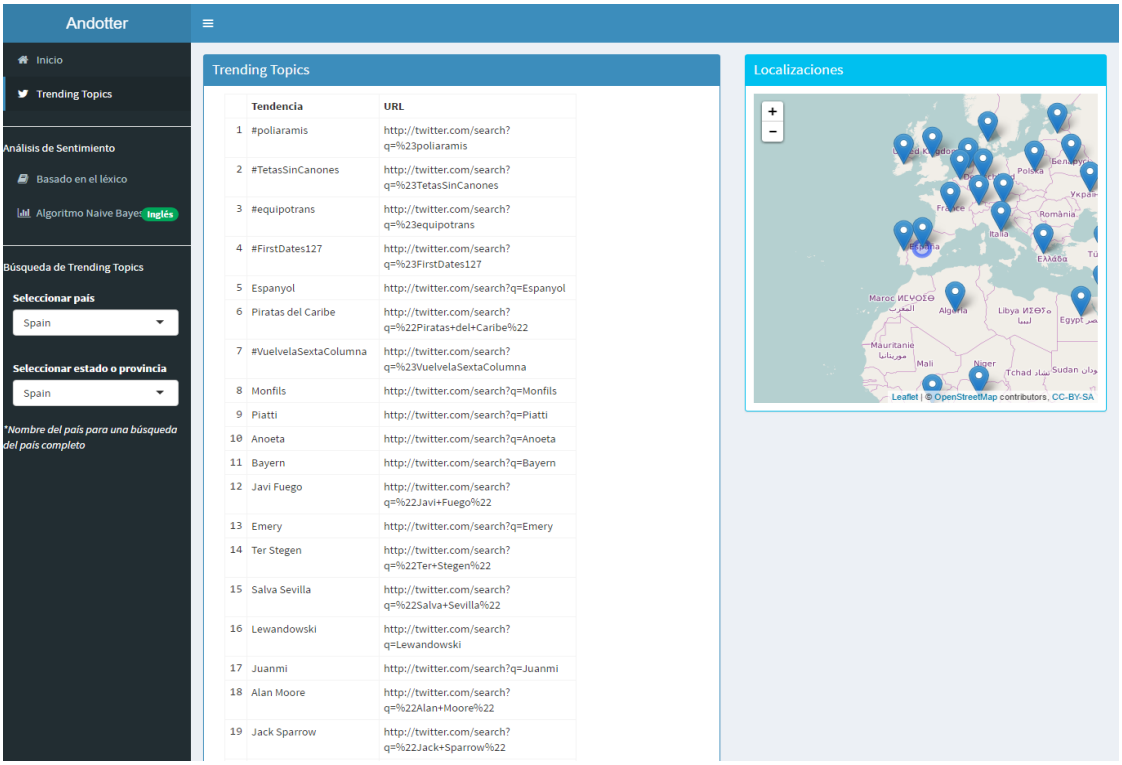
Resultados

En este apartado introduciremos un ejemplo práctico para cada uno de los dos tipos de análisis implementados para finalmente realizar un análisis de los resultados comparando el análisis de los dos clasificadores sobre una misma tendencia.

7.1 Ejemplo de análisis basado en el léxico.

Comenzaremos el capítulo con un ejemplo completo sobre un análisis basado en el léxico a partir del cual podremos obtener una evaluación sobre éste análisis independientemente.

Imaginemos que entramos en la aplicación y no tenemos una tendencia determinada a analizar. Por lo tanto nos iremos a la sección de trending topics para buscar una tendencia que nos interese y realizar el análisis sobre ella (Ver Fig. 27).



	Tendencia	URL
1	#poliaramis	http://twitter.com/search?q=%23poliaramis
2	#TetasSinCanones	http://twitter.com/search?q=%23TetasSinCanones
3	#equipotrans	http://twitter.com/search?q=%23equipotrans
4	#FirstDates127	http://twitter.com/search?q=%23FirstDates127
5	Espanyol	http://twitter.com/search?q=Espanyol
6	Piratas del Caribe	http://twitter.com/search?q=%22Piratas+del+Caribe%22
7	#VuelvelaSextaColumna	http://twitter.com/search?q=%23VuelvelaSextaColumna
8	Monfils	http://twitter.com/search?q=Monfils
9	Piatti	http://twitter.com/search?q=Piatti
10	Anoeta	http://twitter.com/search?q=Anoeta
11	Bayern	http://twitter.com/search?q=Bayern
12	Javi Fuego	http://twitter.com/search?q=%22Javi+Fuego%22
13	Emery	http://twitter.com/search?q=Emery
14	Ter Stegen	http://twitter.com/search?q=%22Ter+Stegen%22
15	Salva Sevilla	http://twitter.com/search?q=%22Salva+Sevilla%22
16	Lewandowski	http://twitter.com/search?q=Lewandowski
17	Juanmi	http://twitter.com/search?q=Juanmi
18	Alan Moore	http://twitter.com/search?q=%22Alan+Moore%22
19	Jack Sparrow	http://twitter.com/search?q=%22Jack+Sparrow%22

Fig. 27. Búsqueda de trending topics en España

Somos seguidores de la liga española de futbol y seguidores del FC Barcelona. Esta mañana hemos leído en el periódico sobre la lesión de Ter Stegen, nos llama la atención verlo en la lista de trending topics y decidimos realizar el análisis sobre dicha tendencia (Ver Fig. 28).

Fig. 28. Resultado de análisis basado en el léxico para la tendencia "Ter Stegen"

Analizando en primer lugar la nube de palabras podemos observar que el tema más mencionado sobre Ter Stegen es su lesión y su correspondiente baja.

Observando el análisis de sentimiento vemos que obtiene una media de -1 y una desviación típica de 1.39, lo que deriva en un sentimiento negativo y una dispersión pequeña de las medidas de polaridad.

Con este caso en concreto resulta muy interesante el resultado del análisis temporal. Observamos como del día 7 al 8 el valor de sentimiento aumenta y el día 9 debido a la publicación de su lesión el valor decae.

7.2 Ejemplo de análisis con algoritmo Naive Bayes.

Imaginemos ahora que estamos interesados en realizar un análisis sobre una tendencia actual presente en Estados Unidos, en concreto, en el estado de Phoenix (Ver Fig 29).

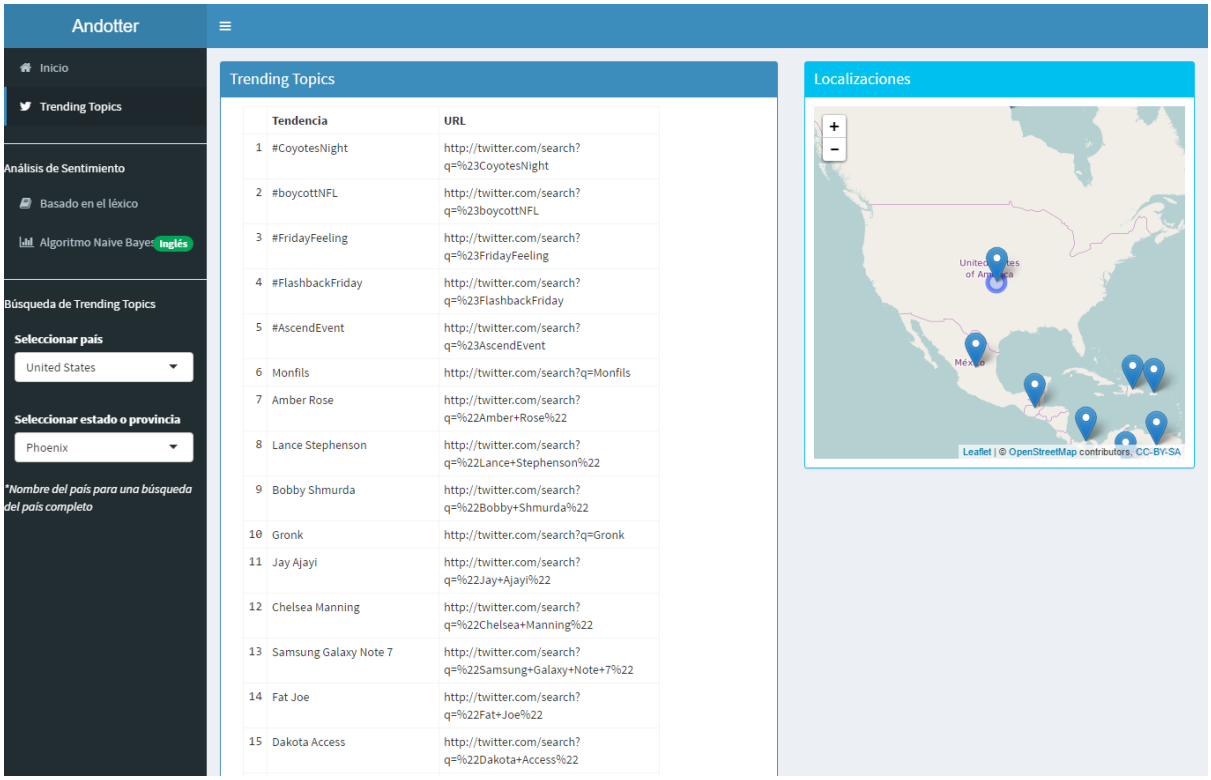
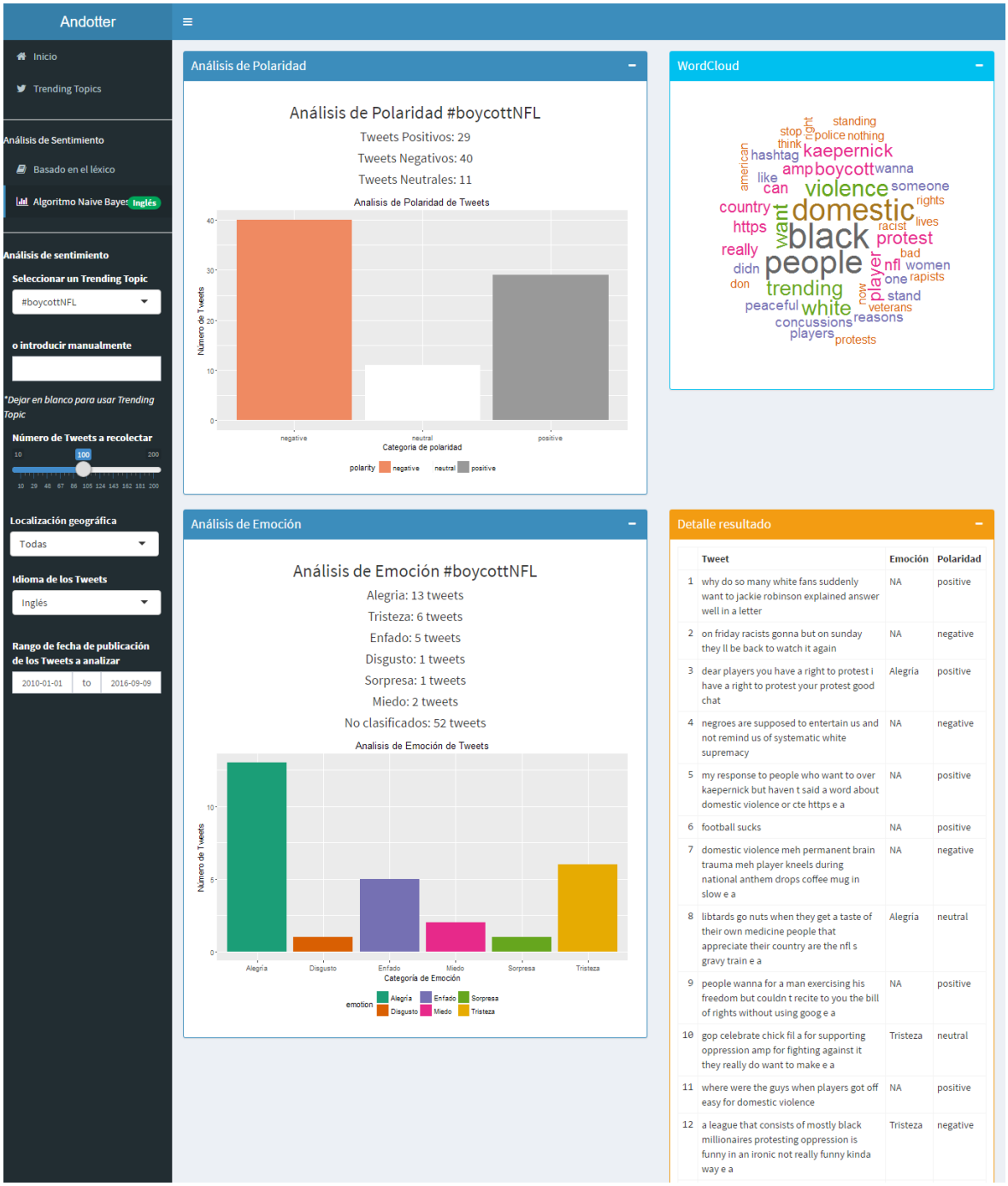


Fig. 29. Búsqueda de trending topics en Phoenix, Estados Unidos

Nos llama la atención el trending topic de "#BoicotNFL" y decidimos hacer un análisis con el algoritmo Naive Bayes sobre ella (Ver Fig. 30).



	Tweet	Emoción	Polaridad
1	why do so many white fans suddenly want to jackie robinson explained answer well in a letter	NA	positive
2	on friday racists gonna but on sunday they ll be back to watch it again	NA	negative
3	dear players you have a right to protest i have a right to protest your protest good chat	Alegria	positive
4	negroes are supposed to entertain us and not remind us of systematic white supremacy	NA	negative
5	my response to people who want to over kaepernick but haven t said a word about domestic violence or cte https e a	NA	positive
6	football sucks	NA	positive
7	domestic violence meh permanent brain trauma meh player kneels during national anthem drops coffee mug in slow e a	NA	negative
8	libtards go nuts when they get a taste of their own medicine people that appreciate their country are the nfl s gravy train e a	Alegria	neutral
9	people wanna for a man exercising his freedom but couldn t recite to you the bill of rights without using goog e a	NA	positive
10	gop celebrate chick fil a for supporting oppression amp for fighting against it they really do want to make e a	Tristeza	neutral
11	where were the guys when players got off easy for domestic violence	NA	positive
12	a league that consists of mostly black millionaires protesting oppression is funny in an ironic not really funny kinda way e a	Tristeza	negative

Fig. 30. Ánalisis con algoritmo Naive Bayes para la tendencia "#BoicotNFL"

Comenzando por el análisis de polaridad podemos observar un claro resultado de sentimiento negativo teniendo en cuenta los 40 tweets clasificados como negativos frente a los 29 positivos. También contamos con 11 tweets clasificados como neutrales, algo que, en este caso concreto, no nos resulta relevante.

Podemos observar en el wordcloud como los términos mas usados en la tendencia tienen que ver con gente negra, violencia, protestas, jugadores, etc. Poniéndonos en situación, este denominado boicot tiene su fundamento en llamar la atención por parte de los jugadores de la NFL sobre la desigualdad racial, y, en particular, la violencia policial contra las minorías.

Analizando esta situación podemos confiar en que el resultado del análisis de polaridad es correcto, ya que, como hemos mencionado anteriormente, hemos obtenido un de sentimiento bastante negativo. Sin embargo, si pasamos al análisis de emoción observamos un resultado de alegría en gran medida obteniendo 13 tweets clasificados con una emoción de alegría entre los 28 totales clasificados. En los siguientes puestos nos encontramos tristeza, con 6 tweets clasificados y enfado con 5. Éstas dos últimas clasificaciones concuerdan mejor con el resultado que se esperaba. Sin embargo, ese resultado de alegría no concuerda demasiado asique vamos a optar por realizar otro análisis sobre una tendencia que a priori sabemos que no se clasifica como alegre y analizaremos el resultado para determinar si es un caso puntual o es una falta de fiabilidad del algoritmo.

Vamos a analizar la tendencia "North Korea nuclear bomb" en relación a los actuales ensayos con bombas nucleares realizados por Korea del Norte, una tendencia que, a priori, resulta claramente negativa.

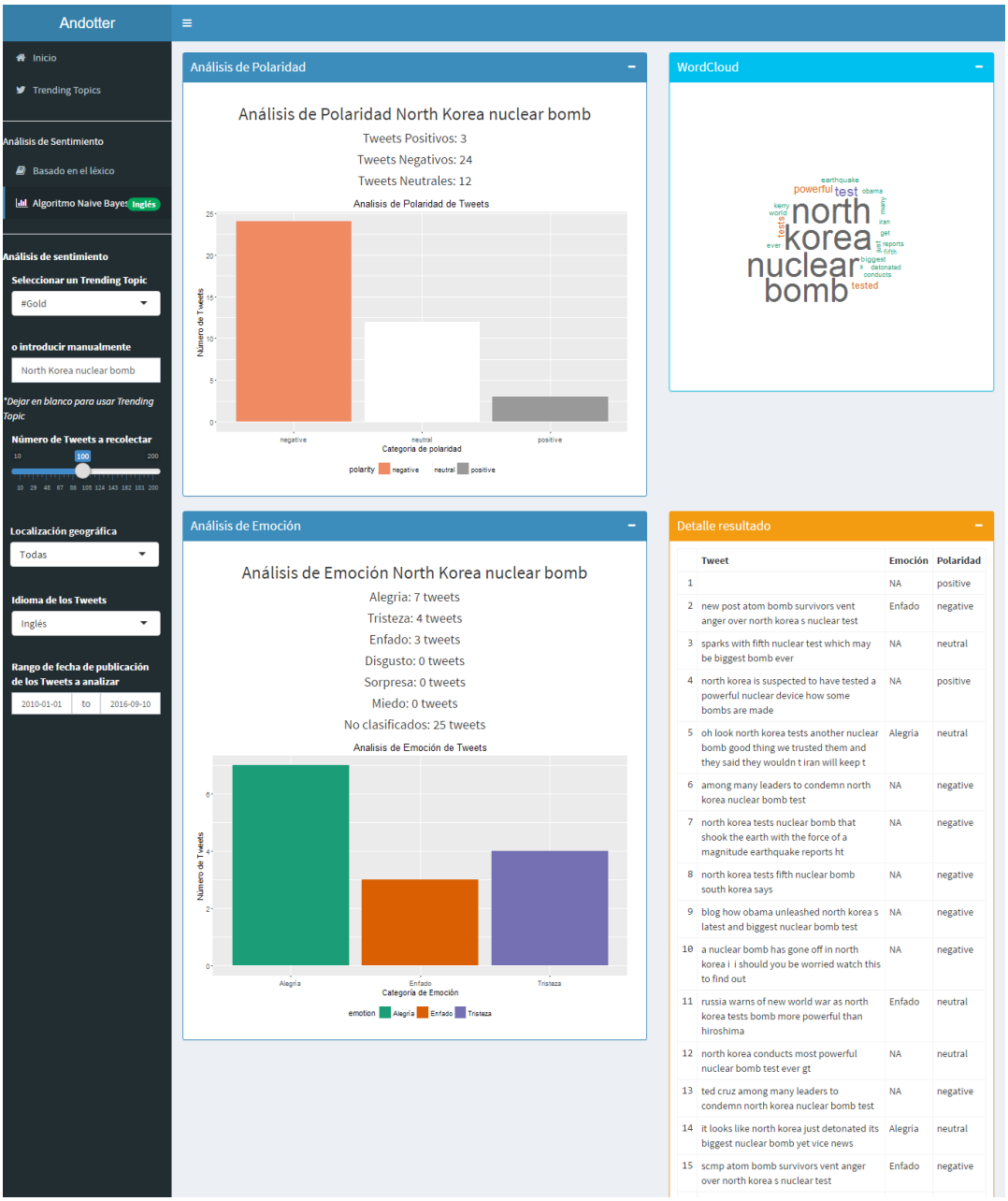


Fig. 31. Ánalisis con algoritmo Naive Bayes para la tendencia "North Korea nuclear bomb"

Analizando el resultado observamos una polaridad claramente negativa como habíamos esperado. Sin embargo, si pasamos al análisis de opinión volvemos a ver una clara tendencia de alegría en el resultado de la clasificación. De nuevo, seguido por tristeza y enfado que serían los resultados más lógicos (Ver Fig. 31).

Tras el análisis de éstos resultados podemos confirmar que el algoritmo encargado del análisis de emoción presenta una tendencia a la clasificación de tweets como alegres, algo que nos va a restar fiabilidad en este análisis.

7.3 Comparación de los 2 tipos de análisis.

Para finalizar el capítulo vamos a realizar una comparación del resultado de los 2 algoritmos implementados.

Para ello vamos a aprovechar el análisis realizado anteriormente con el algoritmo de Naive Bayes sobre la tendencia de "#boycottNFL". Para realizar la comparación vamos a realizar el análisis sobre la misma tendencia con el algoritmo basado en el léxico estableciendo el idioma de los tweets en inglés.

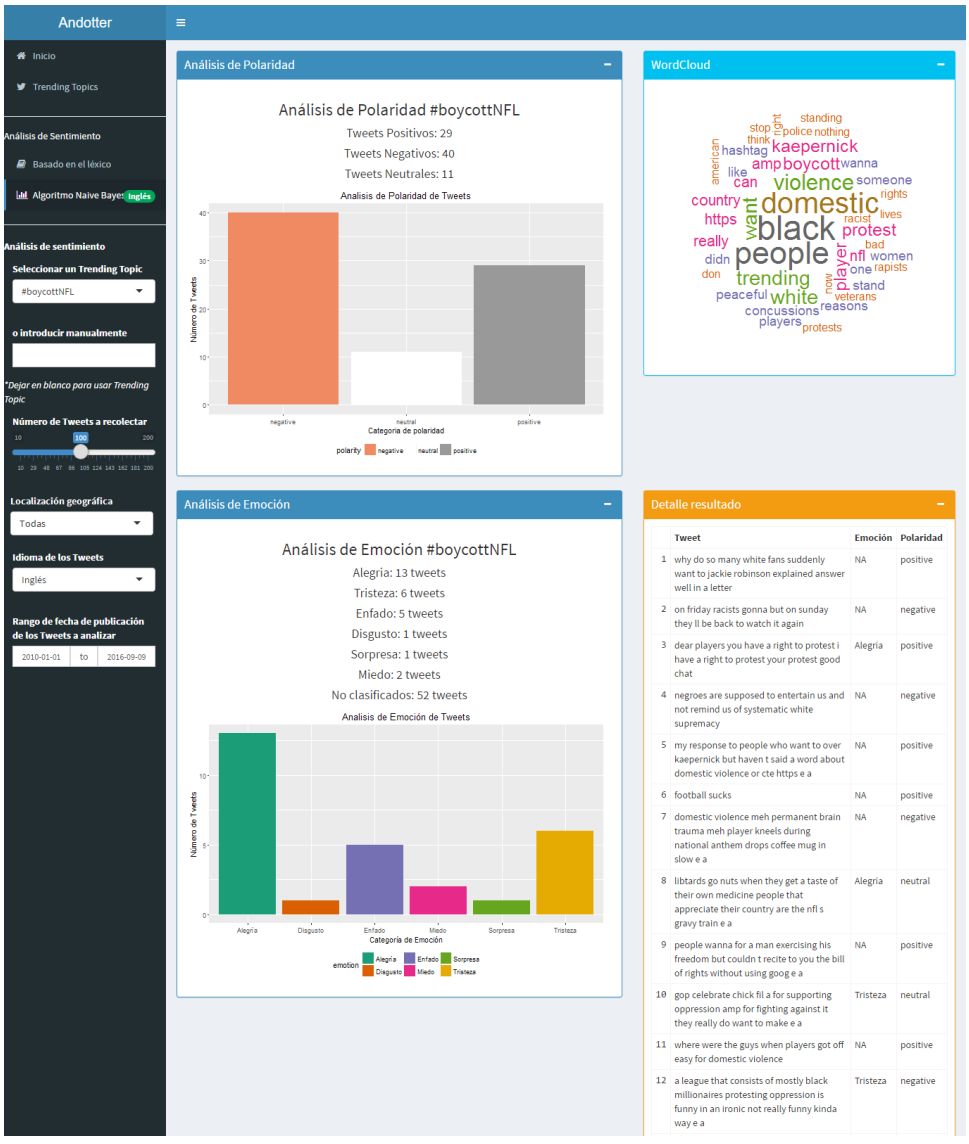


Fig. 32. Análisis con algoritmo Naive Bayes para la tendencia "#BoicotNFL"

Fig. 33. Ánalisis con algoritmo basado en el léxico para la tendencia "#BoicotNFL"

Comparando ambos resultados comprobamos que en ambos obtenemos un resultado de sentimiento negativo. A simple vista quizás pueda parecer que el algoritmo de Naive Bayes (Ver Fig. 32) presente un resultado más negativo pero tras estudiar bastantes análisis con el algoritmo basado en el léxico (Ver Fig. 33) se puede afirmar que un resultado de -1 ya indica un sentimiento negativo importante, pasando esta frontera en muy contadas ocasiones.

Capítulo 8

Conclusiones

8.1 Desarrollo del proyecto.

El proyecto se comenzaba con un nivel muy básico de R y sin ninguna experiencia sobre Shiny ni minería de datos aplicada a Twitter u otra red social.

Cabe destacar que la curva de aprendizaje de Shiny y R ha sido extremadamente alta, llegando a tener soltura en el desarrollo de la interfaz gráfica con Shiny en solo 2 días de aprendizaje.

Sin embargo, no todo ha sido un camino de rosas. El procesado de los datos, en este caso los tweets extraídos de Twitter, ha dado muchos quebraderos de cabeza debido a la codificación de los tweets. Los emoticonos usados en los tweets no se pueden procesar en la codificación usada por las cadenas en R (UTF-8) por lo que es necesario realizar un cambio de codificación y posteriormente resolver los conflictos creados por dicho cambio, como la desaparición de los caracteres con tildes.

Por otra parte, la implementación del mapa interactivo también conllevó una gran parte del tiempo, sobre todo para pensar la forma de obtener las coordenadas de los países habilitados para la consulta de los trending topics y representarlos en el mapa.

Uno de los problemas encontrados durante el transcurso del proyecto es que no podemos controlar los tweets publicados por los usuarios. La mayoría contienen faltas de ortografía, palabras abreviadas, emoticonos, etc. que son irreconocibles por los algoritmos de clasificación.

Otro gran problema en este sentido es la presencia de ironía en los tweets. Resulta muy difícil para un algoritmo detectar la ironía en los mensajes. En muchas ocasiones resulta difícil hasta para nosotros.

8.2 Conclusiones.

Hemos podido comprobar que el resultado obtenido para los análisis de polaridad de ambos algoritmos es fiable, aunque se podría mejorar para algunos casos específicos, como por ejemplo cuando nos enfrentamos a tweets con un gran grado de ironía como ya se ha comentado anteriormente.

Sin embargo, el análisis de emoción nos ha dado unos resultados poco fiables, presentando una clara tendencia hacia la clasificación de los tweets como alegres. El motivo de esta imprecisión quizás radique en un mal diseño del algoritmo o en el corpus de datos utilizado para el entrenamiento del algoritmo.

8.3 Trabajo futuro.

Tras la consecución del proyecto quedan abiertos varios caminos para un trabajo futuro.

En primer lugar, la implementación de un clasificador automático para el análisis de sentimiento en castellano. Para esto se necesitaría de un corpus en español. Durante el final del desarrollo del proyecto se localizó y consiguió acceso a un corpus, el utilizado en las ediciones anuales de TASS con lo que ya tendríamos la base para la implementación del algoritmo.

Por otra parte, resultaría de gran mejora para los resultados de los análisis la mejora del algoritmo de procesado de tweets, añadiendo funcionalidades para intentar detectar algunos casos particulares de ironía.

También resultaría interesante mejorar la aplicación, permitiendo la autentificación de usuarios para guardar los análisis realizados, esta funcionalidad no se ha implementado ahora debido a que el servidor gratuito de shiny server no permite esta funcionalidad, sería necesario pasar a la versión pro de pago.

Anexo A

Análisis de requisitos

RF-01	**Búsqueda de trending topics**	
Actores asociados	ACT-01 Usuario invitado	
Descripción	El usuario puede realizar una búsqueda geográfica de trending topics. Obteniendo una lista con los más populares para esa ubicación.	
Precondición	-	
Secuencia normal	**Paso**	**Acción**
	1	El usuario selecciona la ubicación geográfica mediante una lista desplegable o el mapa interactivo.
	2	El sistema realiza la consulta a la API.
	3	Se devuelve el resultado en una tabla.
Postcondición	El resultado obtenido recarga una lista desplegable que permite la selección de tendencias en los análisis.	
Excepciones	-	
Comentarios	La ubicación se especifica mediante el nombre del país y, si se precisa, el estado o provincia.	

RF-02	**Análisis basado en el léxico**	
Actores asociados	ACT-01 Usuario invitado	
Descripción	El usuario realiza el análisis basado en el léxico sobre una tendencia determinada, obteniendo el resultado de polaridad para dicha tendencia.	
Precondición	Si se desea realizar el análisis sobre un trending topic es necesario realizar la búsqueda de trending topics con anterioridad.	
Secuencia normal	**Paso**	**Acción**
	1	El usuario introduce los parámetros de entrada para el análisis: tendencia, número de tweets, idioma y rango de fecha.
	2	El sistema realiza el análisis.
	3	Se muestra el resultado del análisis de sentimiento con un histograma y un diagrama de caja junto con una vista detalle.
Postcondición	Una vez realizado el análisis se permite generar un análisis temporal para dicha tendencia.	
Excepciones	-	
Comentarios	El resultado de polaridad obtenido es un entero que puede ser negativo, positivo o 0.	

RF-03	**Wordcloud**	
Actores asociados	ACT-01 Usuario invitado	
Descripción	En el resultado de cada análisis se incluye un wordcloud con los términos más mencionados en dicha tendencia.	
Precondición	Para la obtención del wordcloud es necesario realizar un análisis basado en el léxico o con algoritmo Naive Bayes.	
Secuencia normal	**Paso**	**Acción**
	1	El sistema genera y muestra el wordcloud en una sección de la página.
Postcondición	-	
Excepciones	-	
Comentarios	-	

RF-04	**Análisis temporal**	
Actores asociados	ACT-01 Usuario invitado	
Descripción	En los análisis basados en el léxico se permite también la obtención de un análisis temporal de los últimos 9 días.	
Precondición	Para la obtención del análisis temporal es necesario realizar un análisis basado en el léxico.	
Secuencia normal	**Paso**	**Acción**
	1	El usuario despliega la sección específica del análisis temporal.
	2	El sistema realiza el análisis y muestra el resultado en la sección correspondiente.
Postcondición	-	
Excepciones	-	
Comentarios	El resultado del análisis temporal se muestra mediante un gráfico con el valor de sentimiento para cada día analizado.	

RF-05	**Análisis con algoritmo Naive Bayes**	
Actores asociados	ACT-01 Usuario invitado	
Descripción	El usuario realiza el análisis con un algoritmo Naive Bayes sobre una tendencia determinada, obteniendo el resultado de polaridad y emoción para dicha tendencia.	
Precondición	Si se desea realizar el análisis sobre un trending topic es necesario realizar la búsqueda de trending topics con anterioridad.	
Secuencia normal	**Paso**	**Acción**
	1	El usuario introduce los parámetros de entrada para el análisis: tendencia, número de tweets, idioma y rango de fecha.
	2	El sistema realiza el análisis.
	3	Se muestra el resultado del análisis de sentimiento y emoción con dos histogramas junto con una vista detalle.
Postcondición	-	
Excepciones	-	
Comentarios	-	

José Luis Navarro Motos
Grado en Ingeniería Informática - Universidad de Almería

Anexo B

Diagrama de casos de uso

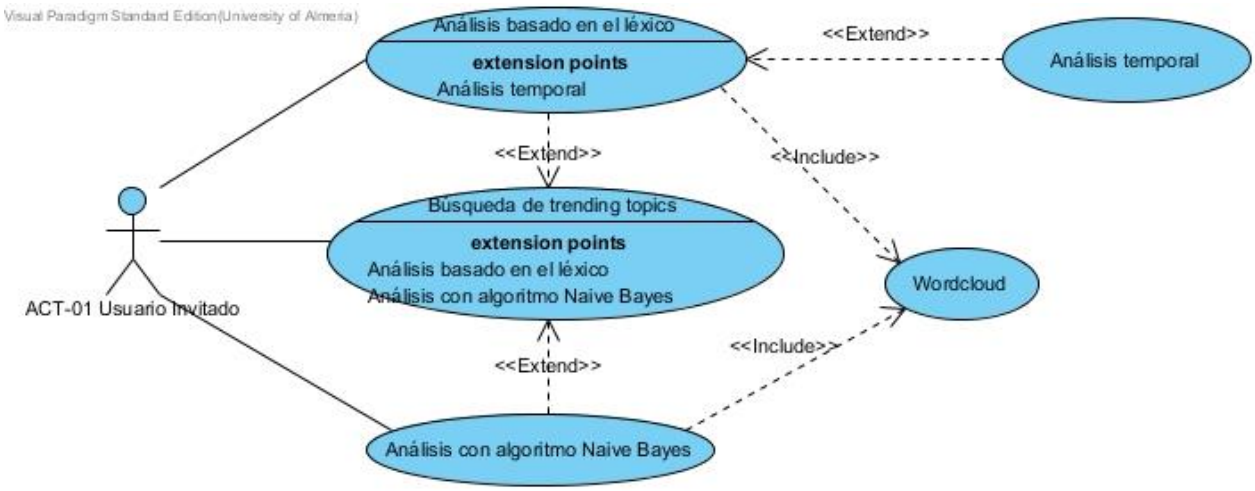


Diagrama UML de casos de uso con el modelado de los requisitos del sistema.

José Luis Navarro Motos
Grado en Ingeniería Informática - Universidad de Almería

Anexo C

Cronograma asociado a las fases del proyecto.

	Mayo	Junio	Julio	Agosto	Septiembre
Análisis					
Estado del arte					
Estudio de herramientas					
Requisitos					
Extracción y preprocesamiento de datos					
Conexión con API					
Obtención de tweets					
Limpieza de tweets					
Análisis o clasificación					
Generación de wordcloud					
Análisis basado en el léxico					
Análisis temporal					
Análisis con algoritmo Naive Bayes					
Evaluación					
Pruebas análisis léxico					
Pruebas análiss Naive Bayes					
Presentación de datos					
Diseño interfaz gráfica					
Implementación interfaz gráfica					

Anexo D

Código en R de la aplicación

Todo el código empleado en el desarrollo de la aplicación se encuentra accesible desde el repositorio público de GitHub:

https://github.com/jnm733/Andotter

Bibliografía

[1] PANG Bo, LEE Lillian & VAITHYANATHAN, sentiment classification using machine learning techniques. En Proceedings of the ACL-02 conference on Empirical methods in natural language processing-Volume 10. Association for Computational Linguistics, 2002.

[2] KOPPEL Moshe & SCHLER Jonathan. The importance of neutral examples for learning sentiment. Computational Intelligence. 22. 100-109. 2006

[3] SIDOROV, Grigori, Empirical study of machine learning based approach for opinion mining in tweets. En Advances in Artificial Intelligence: 11th Mexican International Conference on Artificial Intelligence, MICAI 2012, San Luis PotosÃ, Mexico, October 27 November 4, 2012. Revised Selected Papers, Part I

[4] SARALEGI Xabier & SAN VICENTE Iñaki, XXIX Congreso de la Sociedad Espaola de Procesamiento de lenguaje natural, Workshop on Sentiment Analysis at SEPLN (TASS2013) Páginas 143-150

[5] SHARAN KUMAR RAVINDRAN. Vikram Garg. Mastering Social Media Mining with R. Packt Publishing, 2015

[6] RICHARD HEIMANN, NATHAN DANNEMAN. Social Media Mining with R. Packt Publishing, 2014

[7] REZA ZAFARANI. Social Media Mining. Cambridge University Press, 2014

[8] CHRIS BEELEY. Web Application Development with R Using Shiny, Second Edition. Packt Publishing, 2016

[9] GERGELY DARÓCZI. Mastering Data Analysis with R, Packt Publishing, 2015

[10] ANDREA ISONI. Machine Learning for the Web, Packt Publishing, 2016

[11] HERNÁN G. RESNIZKY. Learning Shiny, Packt Publishing, 2015

[12] TWITTER. Twitter API Rate Limits, https://dev.twitter.com/rest/public/rate-limits

[13] JANYCE WIEBE. Corpus usado en Sentiment para el clasificador de Polaridad, http://mpqa.cs.pitt.edu/#subj_lexicon

[14] CARLO STRAPPARAVA, ALESSANDRO VALITUTTI. Corpus usado en Sentiment para el clasificador de emoción, http://www.cse.unt.edu/~rada/affectivetext/

[15] FUNDACIÓN ELHUYAR. ElhPolar Dictionary, http://komunitatea.elhuyar.eus/ig/files/2013/10/ElhPolar_esV1.lex

[16] MINQING HU, BING LIU. Sentiment Lexicon Corpus, https://www.cs.uic.edu/~liub/FBS/sentiment-analysis.html

[17] MINQING HU, BING LIU. Mining and Summarizing Customer Reviews, https://www.cs.uic.edu/~liub/publications/kdd04-revSummary.pdf

[18] SUBHASREE BOSE. RSentiment; Analyse Sentiment of English Senteces, https://cran.r-project.org/web/packages/RSentiment/index.html

[19] JJVELASCO. Cinco herramientas para analizar los sentimientos de los tweets, https://hipertextual.com/archivo/2010/12/cinco-herramientas-para-analizar-los-sentimientos-de-los-tweets/

[20] QUIRK R., GREENBAUM S, G. L. AND SVARTVIK, J. A comprehensive grammar of the English, Longman, 1985

[21] WIEBE, J. M., WILSON, T., BRUCE, R., BELL, M., AND MARTIN, M. Learning subjective language. Computational Linguistics, Volume 30 Issue 3, September 2004 Pages 277-308

[22] PANG, B. AND LEE, L. Opinion mining and sentiment analysis. Foundations and Trends in Information Retrieval, Volume 2 Issue 1-2, January 2008, Pages 1-135

[23] TURNEY, P. D. AND LITTMAN, M. L. Measuring praise and criticism: Inference of semantic orientation from association. ACM Transactions on Information Systems (TOIS), Volume 21 Issue 4, October 2003, Pages 315-346

[24] LIU, B. Opinion Mining and Sentiment Analysis. En Liu, B. Web Data Mining Exploring Hyperlinks, Contents, and Usage Data, Springer 2011, pages 459-526.

[25] MEJOVA Y. Sentiment Analysis: An Overview, 2010, pages 1-34

UNIVERSIDAD DE ALMERÍA

Resumen/Abstract

Este proyecto se centra en la rama de la informática de la minería de datos, en concreto en el análisis de redes sociales que en este caso es Twitter.

Con el auge en las últimas décadas de las redes sociales resulta interesante ser capaces de realizar análisis sobre ellas en cuanto a una tendencia

Con el desarrollo de este proyecto se pretende obtener una herramienta que nos permita analizar el sentimiento (positivo, negativo o neutro) de los usuarios de Twitter ante una determinada tendencia o temática, todo esto desde un análisis temporal y geográfico.

Palabras clave: minería de datos, redes sociales, twitter, análisis, sentimiento, geográfico, temporal.

This project focuses on the branch of computer data mining, particularly in the social network analysis, Twitter in this case. With the rise in recent decades of social networks it is interesting to be able to perform analysis on them in terms of a particular trend.

With the development of this project, the intention is to obtain a tool that allows us to analyze the sentiment the (positive, negative or neutral) of Twitter users to a particular trend or theme, all from a temporal and geographical analysis.

Keywords: data mining, social networks, Twitter, analysis, sentiment, geographical, temporal.