Escuela Politécnica Superior

AutoTweetly: Generación automática de micro-posts en redes sociales

Grado en Ingeniería Informática

Trabajo Fin de Grado

Autor:
Mohamad Yousef HatHat

Tutor/es:
Elena Lloret Pastor
Jose Manuel Gomez Soriano

Universitat d'Alacant
Universidad de Alicante

1. Justificación y Objetivos

AutoTweetly – Generador automática de micro-posts en redes sociales es un proyecto que nace con la motivación de dar al usuario una herramienta sencilla, intuitiva y fiable para la creación de micro-posts en redes sociales. Dado que el principal objetivo de las redes sociales (en adelante RRSS) es el de comunicar (Díaz-Llairó 2011), y además de realizarlo en una cantidad limitada de caracteres, se le exige al usuario una cierta destreza a la hora de resumir textos.

Las RRSS son servicios prestados a través de Internet que permiten a los usuarios generar un perfil desde el que hacer públicos datos e interactuar con otros usuarios y localizarlos en la Red en función de las características publicadas en sus perfiles. Son sitios o espacios en la red Internet que cuentan con una serie de herramientas tecnológicas muy sencillas de utilizar y permiten la creación de comunidades de personas en las que se establece un intercambio dinámico por diferentes motivos:

- espacios para conocerse,
- intercambiar ideas y reencontrarse con otras personas;
- ofertar productos y realizar negocios;
- compartir e intercambiar información en diferentes medios
- y buscar empleo.

Una posible clasificación[1] de las RRSS es dividirlas por el tipo de servicio que ofrecen:

- Horizontales: buscan proporcionar herramientas para la interrelación en general p. ej. Facebook, Twitter, Google+...
- Verticales que a su vez se dividen en:
  - Por tipo de usuario: dirigidas a un público específico p. ej. Linkedin.
  - Por tipo de actividad: promueven una actividad particular p. ej. YouTube.

[1] http://es.wikipedia.org/wiki/Red_social

Según un estudio realizado por Search Engine Journal[2], Twitter es la red social con mayor crecimiento desde junio de 2012 a marzo de 2013, por ello la cantidad de información que circula es inmensa. Apodada como el "SMS de Internet" (D'Monte 2009) se estima que tiene más de 500 millones de usuarios, generando 65 millones de tweets al día y maneja más de 800 mil peticiones de búsqueda diarias.
Jack Dorsey, fundó Twitter en 2006, y desde entonces hasta ahora, ha tenido una popularidad mundial aplastante respecto a las demás RRSS como Facebook entre otras.
En reglas generales, se mide el éxito de una red social mediante la cantidad de información que hace circular y los usuarios activos que alberga.

El Procesamiento del Lenguaje Natural (PLN), es la disciplina que durante años se ha encargado de procesar y analizar de forma automática toda la información que está disponible en la web. Dentro de esta área de investigación, una de las tareas más significativas es la generación automática de resúmenes, que se usa para extraer la información relevante de un texto.

Por lo tanto, el objetivo de este proyecto final de grado es la creación de una herramienta de generación automática de micro-posts en redes sociales, utilizando técnicas de PLN.
La red social elegida para esta tarea ha sido Twitter por encima de las demás RRSS por la cantidad de información que hace circular. Esta red social tiene la particularidad de que sus micro-posts (en este caso se denominan tweets) tienen un límite establecido de 140 caracteres y por ello la capacidad de extraer la información importante de lo que se quiere compartir es muy importante.

[2] http://www.searchenginejournal.com/growth-social-media-2-0-infographic/77055/

2. Agradecimientos

En primer lugar agradecer a mi familia, en especial mi madre, por todo el tiempo que ha dedicado en levantarme los ánimos, actuar como guía profesional a parte de madre, aguantarme todos los días y saber escucharme cuando más lo he necesitado.

A mi grandísimo amigo y colega Héctor Sempere por estar ahí y tirar conmigo cuando más dura se ha puesto la cuesta en la carrera y por todos esos momentos tanto de sufrimiento como de alegría.

A todos mis compañeros de la carrera en general, sin olvidar los buenos amigos que he tenido el placer y honor de haber conocido, tanto en el plan antiguo como el nuevo y el itinerario.

Especial agradecimiento a mi tutora Elena Lloret, que por desgracia no he podido compartir tanto tiempo como hubiese sido deseado, pero muchas gracias por su paciencia conmigo y por contestar a mis correos por muy tarde que fuese.

Este año, el año que espero que cierre mi etapa como graduado en ingeniería informática, ha sido con diferencia el mejor año de la carrera, en el que he aprendido a solventar errores más enfocados a un ámbito laboral que solventarlos de cara a una examen evaluatorio.

Contento por haber puesto mi grano de arena en la Escuela Politécnica Superior y haber aprendido en mi estancia como estudiante, si Dios quiere, volver para ser docente o volver a estar involucrado en la universidad que me vio crecer como persona y mejorar en todos los aspectos de la vida.

3. Índices

3.1. Índice de contenidos

3.2. Índice de figuras

4. Cuerpo del documento

En este apartado se explica como se ha desarrollado el proyecto, incluyendo una pequeña introducción a las redes sociales de forma global y Twitter en concreto, una sección dedicada a introducir conceptos relevantes de PLN, una breve descripción de los objetivos del proyecto, una explicación detallada de las herramientas y arquitectura utilizadas, un apartado con la metodología de trabajo. La sección finaliza con una serie de ejecuciones del mismo, puntos clave del desarrollo y posibles mejoras futuras.

4.1. Introducción

Aprovechando el auge de las RRSS y la madurez de las técnicas para resumir textos (véase sección 4.2), el proyecto AutoTweetly basa su funcionamiento en generar tweets de forma automática que son el resultado posterior al proceso de resumir el texto que el usuario quiere publicar. Por lo tanto, es una herramienta que permite al usuario que la usa la capacidad de escoger la frase que más se ajusta a sus necesidades en forma de tweet y que el programa le ha podido proporcionar. Es decir, automatiza en cierto aspecto la capacidad necesaria para sintetizar una noticia por ejemplo y sacar una conclusión resumida en 140 caracteres.

Como ya se citó anteriormente, el principal propósito de las RRSS es comunicar (Díaz-Llairó 2011), y debido a eso, mucha información está en constante circulación. El problema viene a la hora de transmitir el mensaje, y es que no siempre se es capaz de publicar contenido adecuado, resumido y conciso. Por eso, mucha información relevante puede quedar en el olvido y no ser transmitida correctamente, de ahí a que en este proyecto se plantea una solución. Además, un estudio (Hahn 2013) ha demostrado que las personas entre 16 y 24 años buscan las noticias en las RRSS antes que en los motores de búsqueda, y las personas que leen noticias online tienden a escanear los titulares de las noticias en vez de leerlas en profundidad (Holmqvist et. al. 2003).

La finalidad de la herramienta desarrollada es la de proporcionar de forma automática al usuario distintas opciones de frases resumidas a partir de un texto para que éste tenga la capacidad de decidir cuál de ellas es la que más se ajusta a sus necesidades. Es decir, el usuario, a partir de un texto de definidas dimensiones, el programa es capaz de obtener las frases más relevantes y devolvérselas para que éste seleccione una y la publique en su red social. Además de la generación de un conjunto de conceptos relevantes transformados posteriormente en hashtags.

Por lo tanto, puede ser de gran ayuda cuando un usuario se encuentra con una noticia por ejemplo de la que quiere seleccionar un fragmento importante pero no sabe que frase extraer para que resuma el texto de forma global.

Debido a que el trabajo gira entorno a las RRSS, es necesario por lo tanto, que la herramienta sea online además de ejecutarse en tiempo real y para ello se ha desarrollado como aplicación web, usando lenguajes de programación web HTML, Javascript y jQuery, mientras que para el lado del servidor (ejecución de scripts externos, análisis de textos y síntesis automática) se ha usado PHP.

4.2. Marco teórico

En esta sección se definirán de forma más detallada los conceptos más importantes para justificar el uso de las diferentes técnicas de generación automática de resúmenes. Debido a la gran cantidad de información que existe en la actualidad y a la imposibilidad de procesar todo ello de manera manual, se ha buscado un campo de investigación, que por un lado combine los conocimientos y teorías lingüísticas y por otro la capacidad de automatización que proporciona la ingeniería informática. Y por tanto, el área de investigación del Procesamiento del Lenguaje Natural (PLN) tiene como objetivo combinar e integrar estos aspectos.

Antes de empezar es conveniente saber más acerca del PLN y como ya se define con anterioridad aunque de forma más concreta es una subdisciplina de la Inteligencia Artificial (IA) que investiga y formula mecanismos computacionalmente efectivos para facilitar la interrelación hombre-maquina, permitiendo una comunicación mucho más fluida y menos rígida que los lenguajes formales (Moreno Boronat et al. 1999).

Tal como se refleja en (Lloret 2009), el PLN puede parecer sencillo, pero su puesta en práctica resulta sumamente compleja. Su dificultad radica por la naturaleza del lenguaje natural y por el contexto socio-cultural en el que se enmarca. Por un lado, se debe tener constancia de las estructuras propias de una lengua concreta y de los fenómenos y propiedades que en dicho lenguaje se producen. La ambigüedad, propiedad inherente en todas las lenguas naturales, o mecanismos de economía lingüística como la elipsis, son dos ejemplos de ello, tratados de forma amplia debido a su complejidad de procesamiento automático. Por otro lado, se debe disponer también de un conocimiento general acerca del mundo para comprender las ideas que se pretenden transmitir a través del lenguaje. Por esta razón, el PLN puede abarcar el tratamiento del lenguaje desde documentos completos, hasta las unidades que forman las palabras, por ejemplo morfemas. Esto da lugar a un amplio abanico de subtareas, que comprende desde aplicaciones más generales, como la recuperación de información, búsqueda de respuestas, extracción de información, generación de resúmenes, atribución de autoría, etc., hasta las aplicaciones intermedias, tales como analizadores morfológicos y sintácticos, reconocedores de entidades, etc. que

constituyen los pilares más importantes permitiendo el desarrollo de aplicaciones generales.

Una de las subtareas que alberga PLN es el de generación de resúmenes automáticos, que se define como la obtención de una versión reducida del documento o documentos fuente, reduciendo su contenido de tal forma que se seleccionen y queden presentes en el resumen los conceptos más importantes de dichos documentos (Spärk Jones, 2007). Para resolver esta tarea, una de las técnicas inicialmente propuestas es la frecuencia de palabras donde Luhn (Luhn, 1958) y Edmudson (Edmundson, 1969) fueron los primeros en usarla.
A día de hoy se sigue utilizando ya que es una técnica sencilla de aplicar que obtiene buenos resultados. Trabajos como los de (Nenkova, Vanderwende y McKeown, 2006) analizan el impacto que la frecuencia de palabras tiene en los resúmenes humanos y se comprobó que éstos están formados por palabras que, a su vez, presentan alta frecuencia en los documentos originales. Además en el trabajo de (Orasan 2009) se demostró que era una técnica competitiva para la generación de resúmenes de forma automática.

En cuanto a la generación de hashtags, es necesario explicar que se obtienen mediante la extracción de las denominadas entidades nombradas (*Named-entity recognition*) y que de forma resumida podría entenderse como una subtarea de la extracción de información que busca localizar y clasificar elementos atómicos de un texto sobre categorías predefinidas como nombres de personas, organización, localizaciones, expresiones de horas, cantidades, valores monetarios, porcentajes, etc. (Mitkov 2003).

4.3. Objetivos

El objetivo principal del proyecto es crear una herramienta que sea capaz de facilitar al usuario la tarea de crear micro-posts de frases resumidas y relevantes de un texto extrayendo estas frases de manera automática.
Como desde un principio, de todas las RRSS se ha decidido una en concreto como es Twitter, es debido a la gran cantidad de información que circula diariamente, su auge como red social actual y la gran cantidad de usuarios registrados y activos que alberga, cabe destacar que esta red social divide sus tweets en dos tipos y que por supuesto pueden juntarse en un único post:

- **Texto plano:** mensaje corto que se comparte en Twitter y tiene un límite máximo de 140 caracteres, esta basado en su significado en ingles tweet que significa "una corta ráfaga de información intrascendente y los sonidos emitidos por los pájaros".

- **Hashtag**: cadena de caracteres formada por una o varias palabras concatenadas y precedidas por una almohadilla (#). Es una etiqueta de metadatos precedida por un carácter especial con el fin de que sea identificada rápidamente por tanto el sistema como el usuario.

En resumen, la aplicación que se propone quiere garantizar que un texto pueda ser resumido mediante diferentes estrategias automáticas y que el usuario tenga la capacidad de elegir la frase más relevante que se adecua a sus necesidades. Además, la frase proporcionada podrá ser complementada con hashtags extraídos de forma automática utilizando herramientas de análisis lingüístico que nos permite extraer conceptos relevantes como son las entidades nombradas.

Dentro de este objetivo general, se pueden distinguir una serie de objetivos concretos pero no por ello menos importantes y son:

- Proporcionar ayuda al usuario a la hora de procesar y extraer información relevante de un texto sobre una temática que quiera publicar.
- Extraer pequeños conjuntos de palabras del texto significativas de los que se pueda obtener una pequeña idea de lo que el usuario quiere transmitir con su tweet.
- Capacidad de elección a la hora de twittear texto, hashtags o los dos mientras respetando el máximo de 140 caracteres establecido por Twitter.
- Capacidad para que el usuario pueda eliminar, modificar o añadir al contenido generado otro de forma parcial o completa si el resultado obtenido por la herramienta no es de su agrado/necesidad.

Cabe mencionar que este proyecto se afronta combinando dos perspectivas, por un lado la perspectiva de investigación dedicada a la extracción y procesamiento de información basada en PLN y la perspectiva ingenieril dedicada a la integración de los resultados obtenidos en una aplicación final ejecutada en tiempo real y online.

4.4. Metodología

Una vez introducida la temática del proyecto y explicados los objetivos de la herramienta, este apartado explica las técnicas utilizadas para obtener el resumen del texto introducido, la generación de hashtags y la API usada para establecer una conexión directa con el perfil de twitter del usuario. Además, se proporciona un breve resumen explicatorio de las herramientas y arquitectura usadas para el desarrollo del proyecto. Por último se finaliza con un desarrollo temporal del proyecto y un pequeño apartado donde se comentan las mejoras de cara a posibles trabajos futuros. La metodología empleada se ha basado en el uso de herramientas PLN que han permitido procesar, analizar, extraer y resumir información textual de manera automática.

4.4.1. Técnicas utilizadas

En esta sección se explican las técnicas usadas de cara a la parte de investigación y estas son dos:

- Frecuencia de palabras para la generación automática de resúmenes en forma de tweet.
- Extracción de entidades nombradas para para la creación de hashtags.

Para la generación de resúmenes, se ha optado por el algoritmo que se basa en calcular la relevancia de una frase en función del contenido. La elección del cálculo de frecuencia de palabras por delante de otros algoritmos es debido a los estudios previamente citados que demuestran su gran eficacia (véase sección 4.2) y porque aunque si existen más técnicas y estrategias que integran mayor conocimiento, el rendimiento no es óptimo para ser integradas en una aplicación que se ejecuta en tiempo real.
El grupo de investigación GPLSI del departamento DLSI (Departamento de Lenguajes y Sistemas Informáticos) tiene a su disposición una plataforma llamada inTime que alberga un generador automático de resúmenes llamado GPSLI COMPENDIUM basado en el algoritmo de frecuencia de palabras y desde el cliente se mandan las peticiones para obtener los resúmenes.

GPLSI COMPENDIUM es un sistema de generación de resúmenes de textos automático capaz de producir resúmenes genéricos e informativos en inglés y que el grupo de investigación GPLSI pone a la disposición del público en el ya citado cliente inTime.

Esta herramienta tiene dos formas de ejecución:

1. Desde terminal usando un script básico escrito en Bash.
2. Desde un proyecto Java usándolo como servicio web WSDL URL.

Se ha optado por la segunda opción ya que la aplicación web está implementada en PHP y éste lenguaje de programación permite la ejecución de programas externos.
GPLSI COMPENDIUM se ha desarrollado con más técnicas aplicables a la hora de resumir textos (entre otras Textual Entailment para la eliminación de la redundancia y Code Quantity Principle para el cálculo de la relevancia) pero requieren más recursos para su ejecución además de la penalización de tiempo que ello conlleva. Eso se debe a que se desarrolló como tesis doctoral y por lo tanto el rendimiento se ha relegado a un segundo plano. No es necesario aplicar técnicas para la supresión de la redundancia en el proyecto ya que se trabaja con textos relativamente cortos y la redundancia de información que se puede encontrar en este tipo de textos es mínima.
Una posible mejora de futuro para el proyecto es la incorporación de una herramienta que pueda tratar más idiomas, incluido el castellano, debido a que solo es posible realizar resúmenes relevantes con el inglés.

Por otro lado, para el análisis lingüístico y la extracción de entidades nombradas para crear hashtags, se ha usado una herramienta opensource de procesamiento de lenguaje conocida como Freeling y que se utiliza en la aplicación para extraer entidades nombradas y con ello la creación de hashtags para poder crear tweets de hashtags o ir acompañando a un resumen formulado por la generación automática de resúmenes.
Destacar que esta herramienta también viene incluida en la plataforma inTime, por lo que su ejecución también se realiza mediante un script en bash y se puede ejecutar como programa externo desde PHP.

Se pueden esquematizar entonces las técnicas utilizadas en esta tabla-resumen:

Características	GPSLI COMPENDIUM	Freeling
Función	generación automática de resúmenes	generación de hashtags
Idioma/s disponible	inglés	inglés y castellano
Técnicas	Frecuencia de palabras (relevancia)	extracción de entidades nombradas
Utilidadad	Crear tweets de texto	Crear tweets de hashtags

4.4.2. Herramientas y arquitectura utilizadas

Una vez explicadas las técnicas utilizadas de base investigadora, en esta sección se explican las diferentes herramientas y arquitecturas usadas para el desarrollo desde un punto de vista ingenieril.

El trabajo consiste en desarrollar la herramienta en torno a la ejecución y presentación de los datos obtenidos por la plataforma inTime, dado que es una aplicación web lo que se quiere desarrollar, la mejor forma y es usar un lenguaje de programación del lado del servidor como es PHP. Se debe a que es un lenguaje intuitivo, permite la ejecución de programas externos y la curva de aprendizaje es bastante rápida.
En cuanto a la seguridad, aunque no sea uno de los objetivos principales del proyecto, se gestionan los posibles errores que podría acarrear el login y autorización de la Twitter App a la hora de interactuar con la API de Twitter.

En cuanto al entorno de desarrollo, se ha llevado a cabo en un editor de textos como puede ser Sublime Text 2 dado que la mayor parte de la implementación no requiere entornos sofisticados. Ayudado de la gran información que circulan por manuales y foros, la

necesidad de usar un entorno de desarrollo o framework (entre los más conocidos CodeIgniter[3] o CakePHP[4]) ha quedado descartada.
Por otra parte, Twitter pone a disposición de los usuarios una serie de APIs distintas para interactuar con los perfiles de usuarios, crear tweets, gestionar seguidores, streaming, etc... Más bien, es una denominada REST API, una API web que funciona por HTTP y se accede a partir de URLs que devuelven contenidos en formatos distintos, como XML, JSON, HTML, etc.

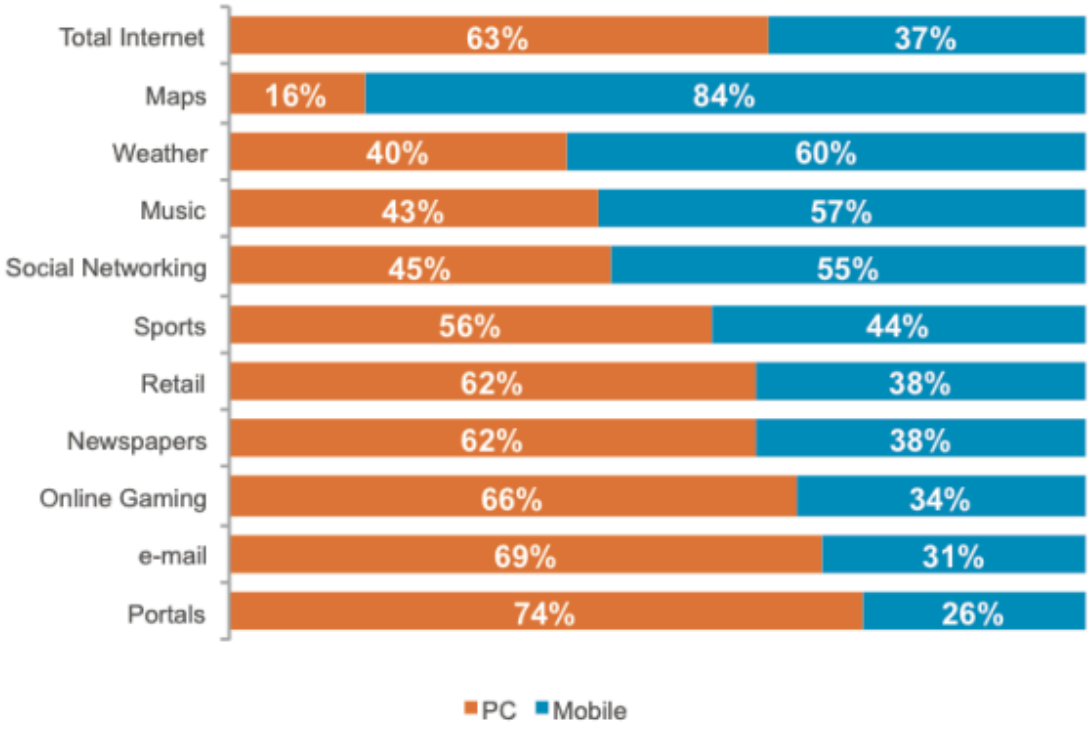
La REST API está basada en OAuth, un protocolo que permite una autenticación segura en un método estándar desde web, móvil y aplicaciones de escritorio. Como las librerías que usan esta API ya están desarrolladas, se ha usado la librería oficial desarrollada para PHP[5].

Para desarrollar la interfaz de usuario se ha utilizado el servidor Apache ya que incluye un módulo de PHP y permite visualizar la implementación de forma local. Finalmente para poder resolver un dominio, se ha utilizado NO-IP, un servicio gratuito de DNS dinámico. NO-IP se utiliza para que cuando el usuario autorice la aplicación desde Twitter, se redireccione a ese dominio. Cabe la posibilidad de alojar el proyecto en algún host de pago o en algún servidor para que se siga usando y además que la comunidad le saque provecho.

Explicadas las herramientas usadas para la parte del servidor, en la parte del cliente la mejor forma de desarrollar una página web y que esté lo más adaptada a todos los navegadores más usados actualmente como Chrome, Firefox y Safari es usar HTML5 apoyado por CSS3. La aplicación por lo tanto, puede funcionar tanto para ordenadores como dispositivos móviles ya que es una interfaz en HTML.

Según un estudio de Marketing Land (Sterling 2013) un 47% del uso de Internet pasa por los dispositivos móviles y que con el tiempo irá creciendo de forma continua y con bastante rapidez, para ello, en el proyecto se han cuidado esos detalles de interfaz para que un usuario con un dispositivo móvil con una resolución que vaya adecuada a una pantalla mayor de 4” pueda observar de forma correcta el contenido.

[3] https://ellislab.com/codeigniter
[4] http://cakephp.org
[5] The First PHP library to support OAuth: https://github.com/jmathai/twitter-async/tree/1185dc839ecee8b0cf4355994977e43e00e08185



Estudio Marketing Land

Se han utilizando lenguajes como Javascript y JQuery para interactuar con el diseño, imágenes y contadores de caracteres entre otros ya que con HTML y CSS no se pueden desarrollar todas esas funcionalidades. Con ello se consigue que el diseño sea más compacto y pueda adaptarse a las diferentes resoluciones o dispositivos que usen la aplicación.

Las funciones que cumplen o en que interviene cada herramienta se explica de forma esquemática en esta tabla-resumen:

Funcionalidades	**PHP**	**Sublime Text 2**	**HTML CSS**	**Javascript JQuery**
Programación lado servidor	X			
Programación lado interfaz			X	X
Interactuar con API Twitter	X		X	
Entorno de desarrollo		X		
Vista adaptada ordenador y Dispositivos móviles			X	
Ejecución de herramientas de PLN	X			

4.4.3. Diario de desarrollo

Antes de empezar a explicar como ha ido evolucionando y desarrollándose la aplicación, es conveniente tener una visión esquemática del proceso.

El desarrollo de este proyecto se ha llevado a cabo en las siguientes fases:

1. Diseño de la aplicación: crear una pequeña interfaz para interactuar y programar el lado del servidor para que tanto interfaz como ejecuciones de las herramientas se adapten.
2. Uso de herramientas PLN: realizar las ejecuciones de las herramientas e integrarlas en el sistema para su correcto funcionamiento en la aplicación.
3. Integración en la aplicación: comprobar que al introducir un texto se procese para las distintas opciones y que devuelva los resultados correctos.
4. Uso de la API de Twitter: integración en el sistema del logueado con Twitter del usuario y comprobar que el tweet generado se publica correctamente.
5. Evaluación y testeado: fase de solucionar errores y ajustar correctamente la interfaz conforme a los dispositivos que usan la aplicación.

Como se ha mencionado anteriormente, el desarrollo giraba en torno a ejecutar los scripts bash del cliente inTime. Entonces, el desarrollo de la parte del servidor ha sido primordial. Primero tuvo lugar la integración de la vista del usuario con la ejecución del servidor de los scripts y por lo tanto de las técnicas tanto de generación de resúmenes como del analizador sintáctico. Sin olvidar la configuración del servidor Apache, el DNS dinámico y cambiar algunos elementos de la configuración de PHP.

Una vez terminada esa fase, en la página de desarrolladores de Twitter (puede registrarse cualquier usuario con cuenta perfil de Twitter) se ha dado de alta una aplicación con la que se establecen los llamados tokens, es decir una cadena de caracteres que actúa de identificador. Esta aplicación que tiene permiso de lectura y escritura en un perfil del usuario que la autorice, permite que el proyecto AutoTweetly pueda establecer un enlace entre lo que se genera en la página y lo que debe de publicarse en el perfil de Twitter del usuario.

En reglas generales, se ha desarrollado un sistema de login autorizado y con las directrices de seguridad del protocolo OAuth de Twitter para que cualquier usuario pueda resumir los textos que quiera usando la herramienta.

Después de que las técnicas fueran correctamente integradas, el desarrollo posterior se postuló en refinar el diseño de la interfaz para el diseño adaptativo entre los posibles y diferentes dispositivos (PC/Mac, móviles, tablets...) que utilizaran la herramienta.
Además se hizo especial hincapié en la detección y eliminación de errores en los que el usuario pudiese incurrir como introducir el texto sin haber seleccionado previamente el idioma, twittear textos mayores de 140 caracteres.

4.4.4. Posibles mejoras

Como la técnica de generación de resúmenes esta desarrollada solo para inglés, el proyecto queda abierto para incluir técnicas más avanzadas, más rápidas y multilingües, es decir, pudiendo llegar a ser en un futuro una aplicación de uso cotidiano para los usuarios de Twitter.

A la hora de seleccionar los hashtags, una posible versión más sofisticada para desarrollar en un futuro es el de un algoritmo que sondee las tendencias (tending topic en inglés) y que con un análisis del texto a resumir consiga una enlazar posibles semejanzas. Con eso se garantizaría que si la noticia por ejemplo que el usuario quiere twittear esté en las tendencias y no se pierda entre los millones de tweets escritos por todo el mundo. Este algoritmo requeriría de un estudio exhaustivo de las tendencias y tener un equipo técnico más sofisticado y también mayor al que se ha dispuesto para el proyecto.

Otra posible mejora, es la de convertir la aplicación web en una aplicación para dispositivos móviles como iOS y Android, se puede programar tanto en HTML5 como en aplicación nativa para cada uno de los SO. Aunque de momento, la aplicación actual permite visualizarse en dispositivos móviles.

Como las páginas de periódicos o revistas importantes usan un botón de compartir noticia en Twitter y que solo twittea el titular de la noticia, se podría ofrecer una funcionalidad que integrase el botón de compartir típico con la herramienta desarrollada para que haga un breve resumen de la noticia y dar al usuario que comparte un abanico a elegir entre tweets usando las técnicas desarrolladas. Con integración se entiende que la aplicación desarrollada pase a ser parte de la página del periódico o revista y no sea una herramienta adicional.

Aunque el abanico de mejoras es infinito, la última propuesta posible es que la herramienta en un futuro permitiese en vez de introducir el texto con el que la herramienta interactúe, exista la opción de introducir una URL de una noticia de un periódico por ejemplo y que el programa automáticamente detecte la noticia y realice las acciones pertinentes.

4.5. Cuerpo del trabajo

Una vez explicadas las técnicas, herramientas usadas y situación historia actual del proyecto, en esta sección se explican las diferentes vistas de la aplicación web, los diferentes apartados de entrada y salida de datos.

Para empezar, al acceder a la página web donde esté alojado el proyecto, aparece la pantalla de bienvenida donde el usuario tiene dos opciones:

1. Loguearse con su cuenta de Twitter para que cualquier texto que resuma pueda ser twitteado automáticamente por la aplicación web.
2. Si no tiene cuenta o no quiere autorizar la aplicación de Twitter, puede probar la demo que hará lo mismo pero sin que la aplicación pueda twittear el resultado final.

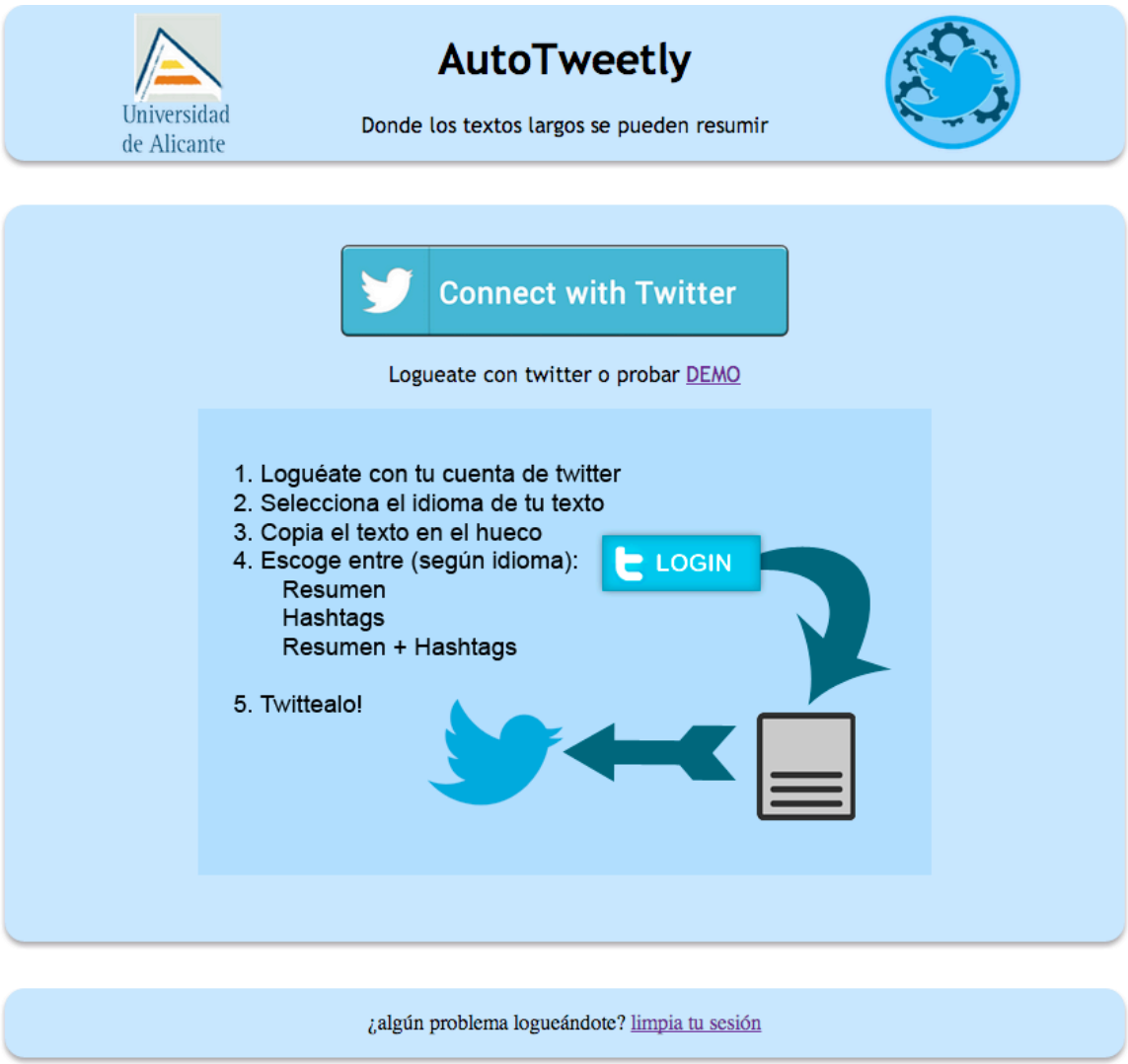


Fig. 1: Página de login

Si el usuario ha decidido loguearse con Twitter, la aplicación le redireccionará a la página oficial de Twitter[6] donde tendrá que aceptar que la aplicación AutoTweetly pueda "publicar tweets por ti" entre otros privilegios que no se utilizan en este proyecto. Una vez logueado con su usuario y autorizada la aplicación, se redireccionará a la página principal de AutoTweetly.

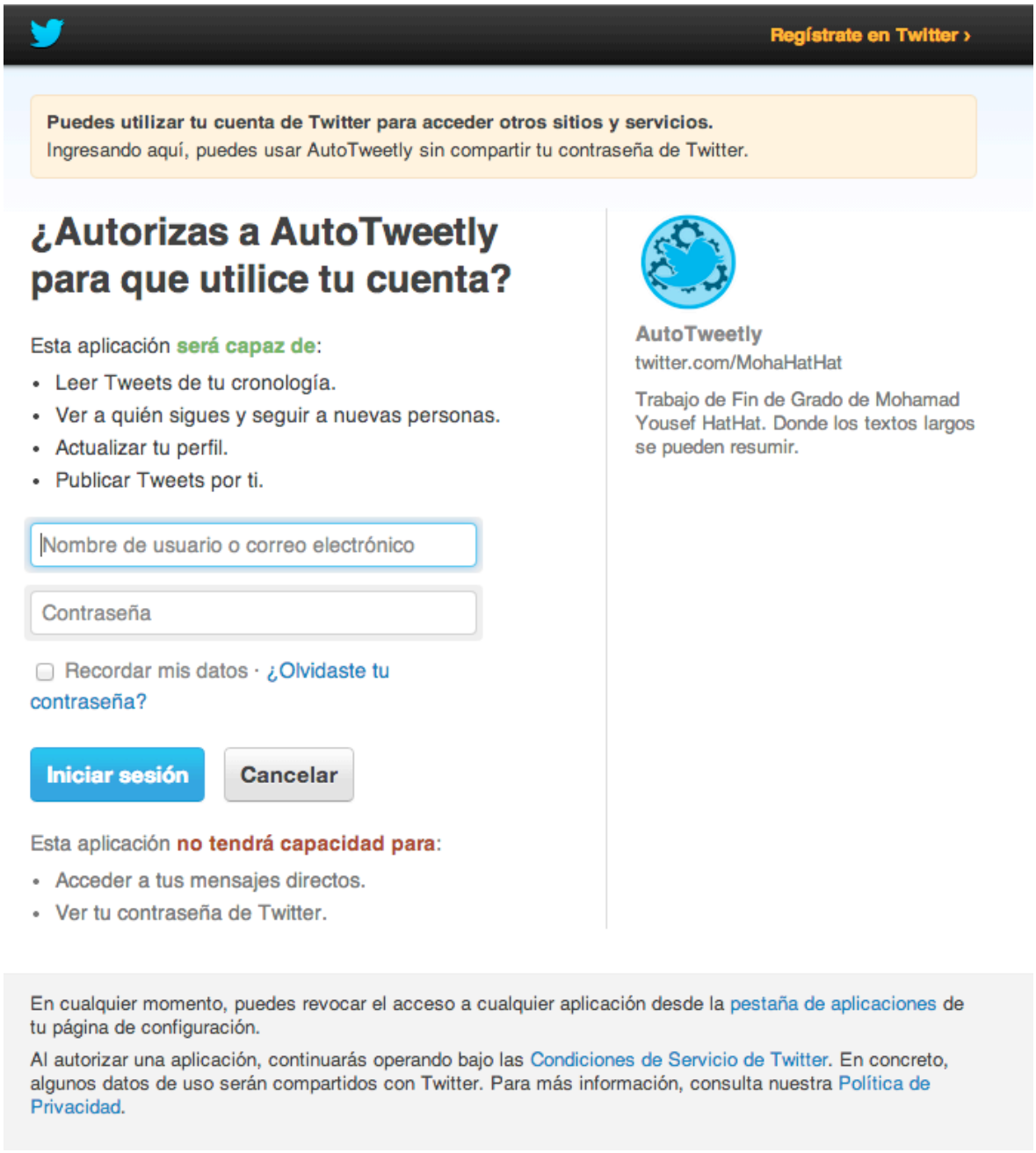


Fig. 2: Página de autorización twitter

[6] api.twitter.com/oauth/authenticate

Una vez terminados los pasos anteriores y estar ya en la página principal de nuevo, el usuario podrá introducir el texto que desee resumir, seleccionar el idioma y de seguido seleccionar la opción que esté disponible para ese idioma:

1. Texto: solo ejecutará el generador de resúmenes.
2. Hashtag: solo ejecutará el analizador lingüístico en busca de entidades nombradas.
3. Texto + Hashtag: ejecutará las dos funciones anteriores de forma conjunta.

Además existe una función para comprobar que el login se ha hecho correctamente y poder cerrar la sesión (solo en la aplicación, no en Twitter). Cabe mencionar que los datos de login de Twitter en la página se guardan en el almacenamiento local (local storage) del navegador y que a la hora de cerrar sesión esos datos se eliminarán.

En la figura siguiente se pueden apreciar tres secciones diferenciadas y son:

1. Encabezado o header: formado por el nombre del proyecto y un pequeño lema, además de dos logos, el de la Universidad de Alicante que redirecciona a la página web de la universidad y el logo diseñado para la aplicación y que redirecciona a la página principal del proyecto (página index).
2. Contenido: pueden llegar a ser tres o cuatro contenedores separados conforme a la funcionalidad que se elija entre ellas. Cabe destacar las secciones:
    a. Texto: donde se introduce el texto a resumir,
    b. Tweet: el resultado de ejecutar el generador automático de resúmenes,
    c. Hashtag: el resultado de ejecutar la extracción de entidades nombradas del texto y
    d. Twittear: en caso de la demo sería resultado y es el contenido que luego se podrá visualizar para su posterior publicación en Twitter en forma de tweet.
3. Pie de página o footer: cuando el usuario se loguea con Twitter aparece su nombre de usuario y una opción para cerrar sesión, en el caso de estar en la demo, es un enlace para volver a la página principal.

Fig. 3: Página principal

Como se citó anteriormente, después de realizar login o seleccionar el modo demo, el usuario deberá de introducir el texto con el que quiera interactuar, seleccionar el idioma y entonces elegir la opción que más le satisfaga. Para probar la aplicación se usará como ejemplo este texto en inglés de una noticia de New York Times[7]:

There's a curious paradox at the heart of mobile apps: Most people don't like to download them. But when they do, they spend nearly all of their time in them.

In fact, people spend much more time using apps than they do web browsers on their devices. Earlier this year, Flurry, the mobile analytics firm, released a report saying that mobile apps accounted for 86 percent of the time the average American consumer spent on their mobile devices, compared with 14 percent for mobile web browsers.

Yet it's murder getting most people to consistently download new apps to their phones. Last month, ComScore, another research firm, put out its own report showing similarly lopsided numbers for mobile browser and app uses.

But it also said that downloading of apps was heavily concentrated in a small segment of the population. Seven percent of smartphone users in the United States account for nearly half of all app downloads in a given month, while over 65 percent of them download zero — that's right, zero — apps per month, ComScore estimated.

Apps can be more attractive because their performance is often snappier than a browser's (anyone who has used the browser and app versions of Google Maps understands this distinction well). Publishers love apps because they can send out notifications alerting their users to fresh content, sales and other new developments.

A start-up called DWNLD is trying to lower the barriers to creating mobile apps for web publishers so that more of them — amateur bloggers included — can enjoy these benefits. The company (pronounced "Download") has a service that will take an ordinary blog or another website and convert it into an app for iOS devices (Android will come later).

[7] http://bits.blogs.nytimes.com/2014/09/03/making-the-leap-from-web-to-mobile-apps-easier/?_php=true&_type=blogs&ref=technology&_r=0

Y este es el resultado cuando se pega el texto en la zona dedicada a ello:

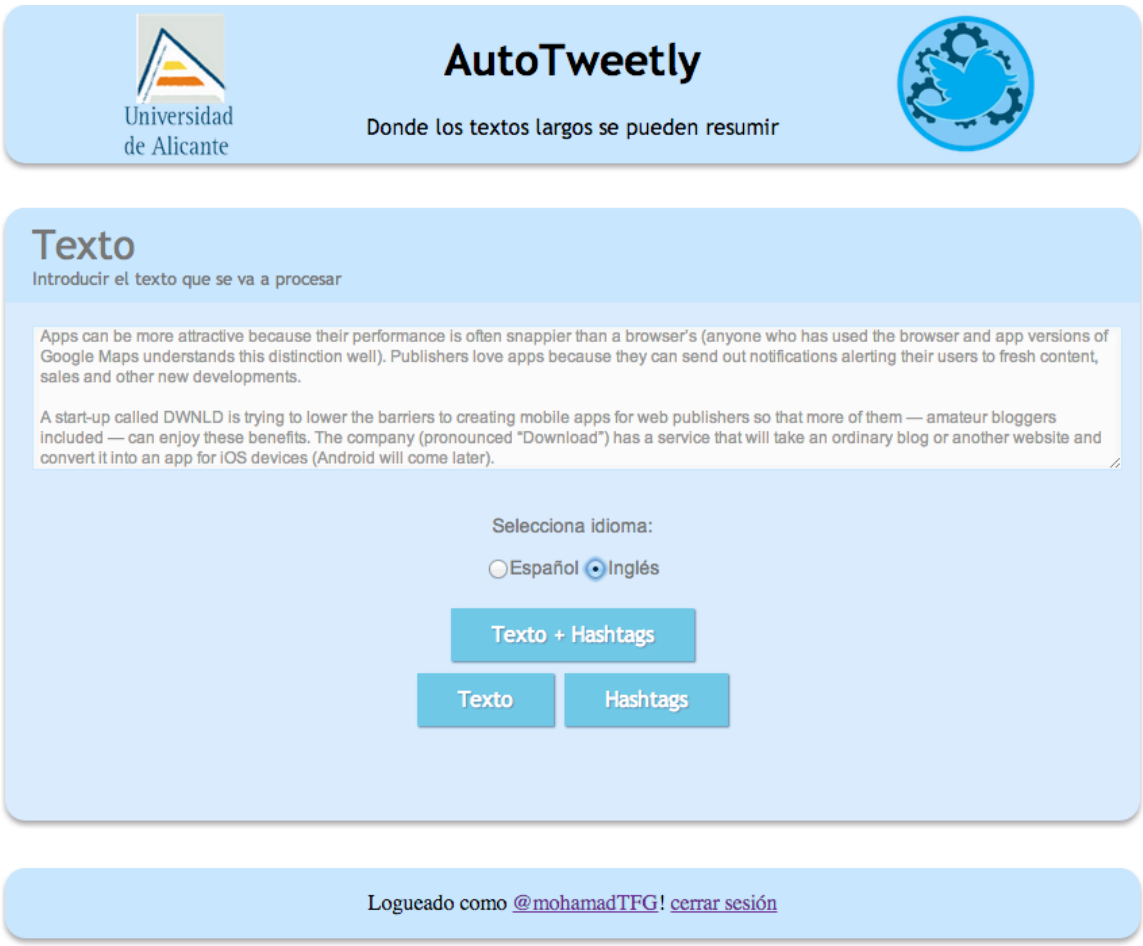


Fig. 4: Página opciones

Se selecciona el idioma inglés y entonces se obtienen tres opciones: crear tweets con texto conjunto con hashtags, únicamente texto y únicamente hashtags.

La ejecución que se lleva a cabo al pulsar el botón, en este caso "Texto y Hashtags", permitirá al programa procesar el texto y obtener como resultado lo siguiente:

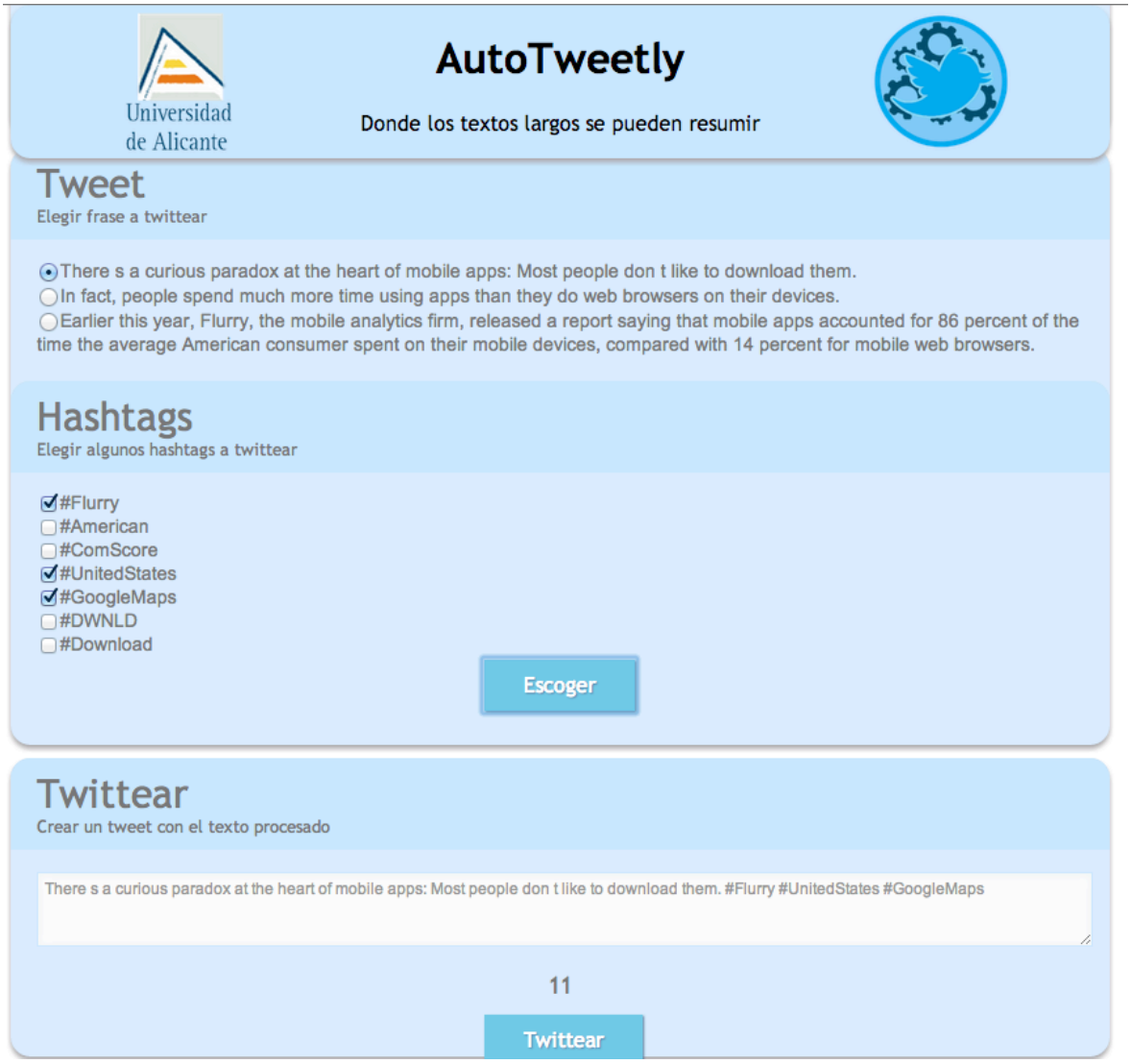


Fig. 5: Texto + Hashtags

Se ve como la aplicación ha generado 3 posibles frases relevantes que son candidatas a ser twitteadas a partir del texto introducido, dando a elegir solo una opción. También ha realizado un análisis de entidades nombradas extrayendo 7 hashtags que el usuario puede incluir en su tweet. Puede proceder el usuario a seleccionar los hashtags que quiera mientras que el contador (en la figura el valor es 11) sea mayor que 0. Una vez la cuenta esté en negativo, no se podrá twittear debido a que se ha sobrepasado el límite de 140 caracteres permitidos.

Cabe mencionar que si se selecciona uno o muchos hashtags, se añadirán al final del tweet elegido. Al pulsar el botton de Twittear el usuario puede comprobar que está publicado:

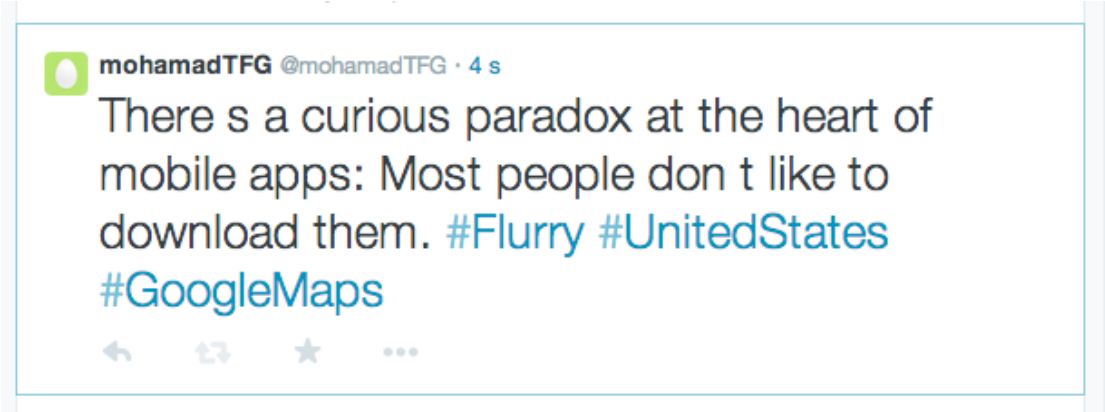


Fig. 6: Página tweet

Las ejecuciones tanto "Texto" como "Hashtag" de forma independiente darían los mismos resultados que la anterior ejecución conjunta.

En caso de que el texto esté en castellano, entonces solo se podrán crear hashtags y por lo tanto la ejecución sería la siguiente, introduciendo este texto de entrada extraído de EL PAÍS[8]:

> *En los primeros ochenta, Gas quiso montar la función en el Romea, recién convertido entonces en Centro Dramático de la Generalitat. Y en 1989, como director del Festival de Tardor de Barcelona, trajo el fabuloso montaje de Bergman, con Jarl Kulle y Bibi Andersson. "Como actor", cuenta, "me la ofrecieron varias veces: primero el rol del hijo pequeño, luego el mayor, y luego el padre, que es el papel que interpreto ahora. ¡El tiempo vuela!". Hará unos meses, Gas estaba a punto de comprar los derechos y montarla, cuando le llamó Alejandro Colubi, el empresario del Marquina: "Me dijo: 'Vamos a hacerte una oferta que te sorprenderá'. Y me sorprendió: el director Juan José Afonso quería contratarnos a Vicky y a mí para protagonizar Largo viaje. Y aquí estamos, con tres estupendos actores jóvenes: Juan Díaz, Alberto Iglesias y Mamen Camacho".*
>
> *Para ser un clásico de su envergadura, la función se ha puesto tan solo cuatro veces en España. En 1960 la estrenó González Vergel, en el Lara. Casi treinta años más tarde volvió a la escena (Español, 1988) dirigida por Narros y Layton. John Strasberg la monta de nuevo en el Albéniz, en 1991. Y Álex Rigola en La Abadía, en 2006.*

[8] http://elpais.com/elpais/2014/09/02/eps/1409660806_952923.html

Y la ejecución sería la siguiente:

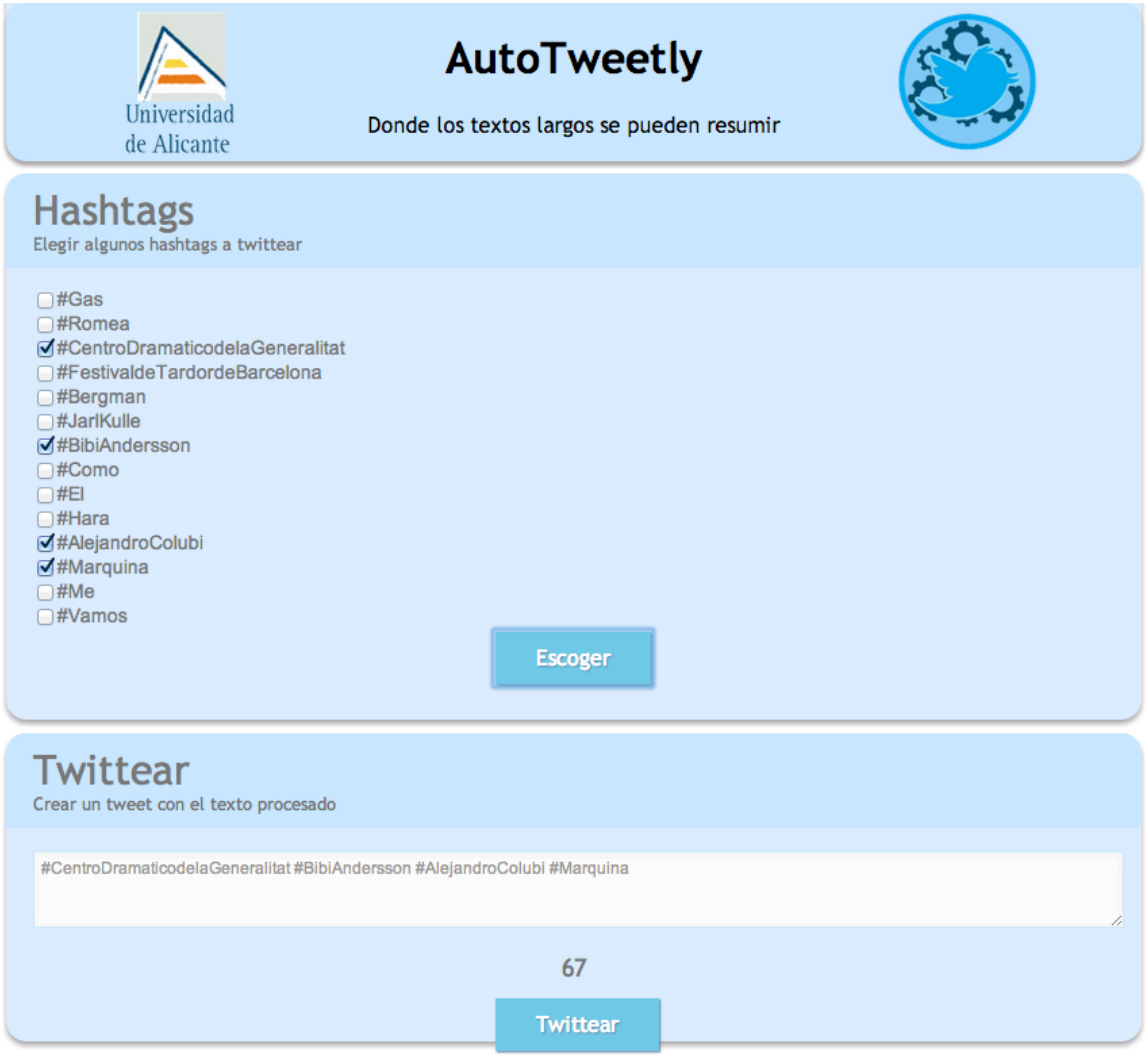


Fig. 7: Página hashtag castellano

El resultado es un tweet generado con 4 hashtags:

Fig. 8: Página tweet 2

En cuanto al diseño visto desde un dispositivo móvil, las pruebas se han realizado desde un iPhone 5 y los resultados se observan a continuación. Aclarar que no se ha realizado ninguna ejecución dado que lo que se quiere apreciar es el diseño de la propia aplicación, la funcionalidad es la misma que en un ordenador:

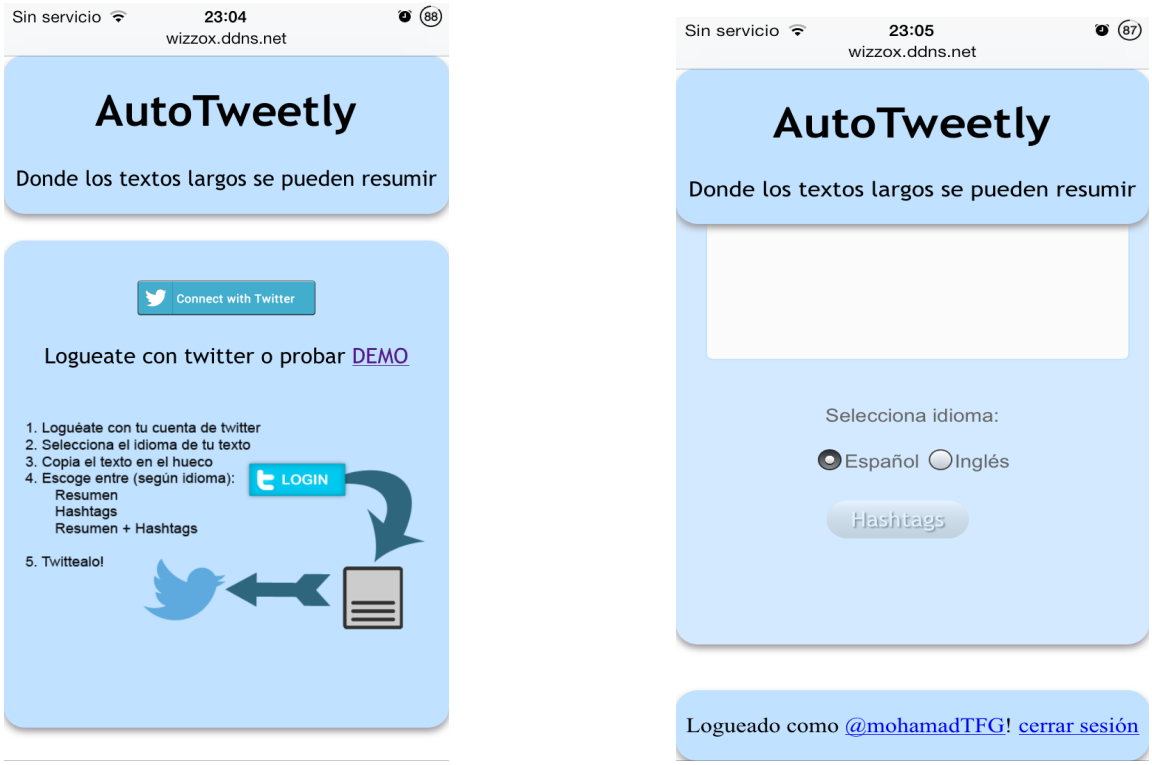
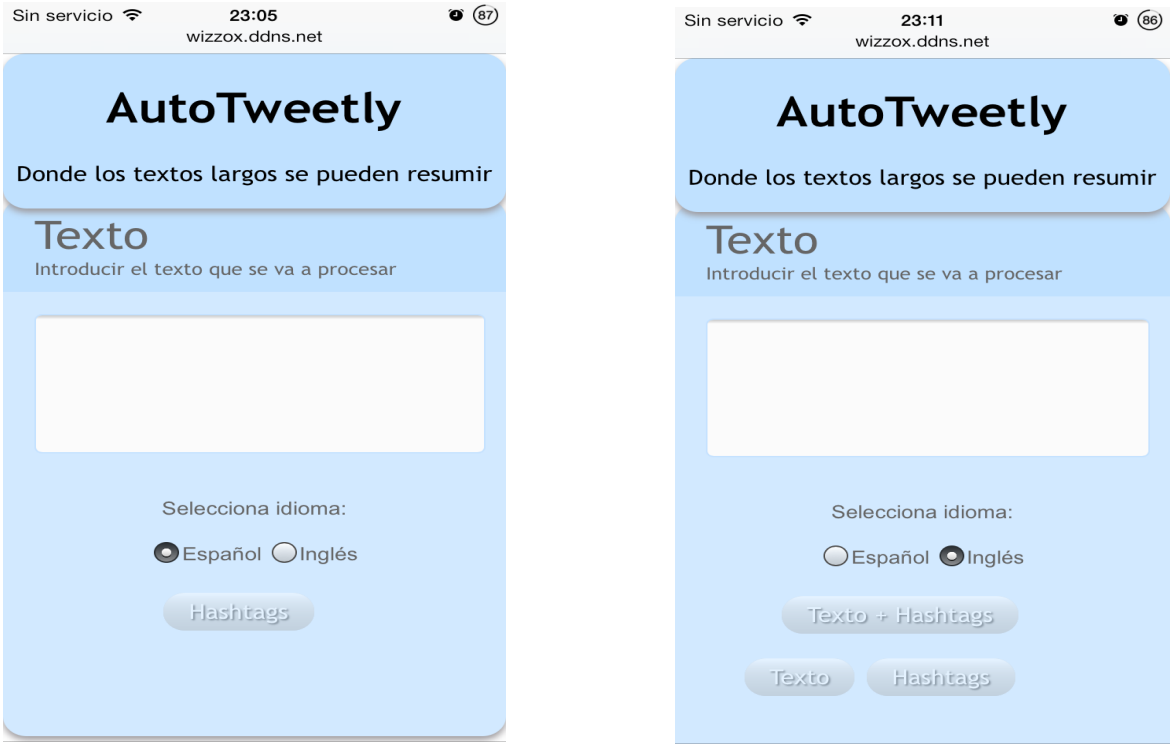


Fig. 9: Pagina dispositivo móvil



5. Conclusiones

Este trabajo de fin de grado ha presentado el desarrollo de una herramienta de generación automática de micro-posts en redes sociales, en este caso Twitter, utilizando técnicas de PLN. El objetivo principal es el de facilitar al usuario la tarea de obtener frases resumidas y relevantes de un texto extrayéndolas de manera automática y se ha alcanzado con éxito.

El PLN es la disciplina que se encarga de analizar y procesar de forma automática toda la información disponible en la web y una de las técnicas más significativas desde el área de investigación es la generación automática de resúmenes. La técnica para obtener resúmenes usada en este proyecto es la frecuencia de palabras debido a la facilidad de su implementación, eficacia y eficiencia de cara a una aplicación web online ejecutada en tiempo real.

Cabe destacar que conseguir que un algoritmo tenga la capacidad casi humana de realizar un resumen y que además sea eficiente, rápido y computable a nivel de servidor es una tarea muy sofisticada y de difícil implementación. Por eso con el tiempo, el proyecto puede llegar a ampliarse bastante y no solo ser usado para crear micro-posts en RRSS, sino para aplicarse en ámbitos educativos (técnicas de estudio, plantillas de resúmenes, etc...) y profesionales (generación de resúmenes automáticos para noticias, publicidad, marketing, etc...).

El desarrollo del proyecto se ha llevado a cabo partiendo de que es una aplicación web y para ello se han usado lenguajes de programación como PHP para el lado del servidor y HTML con Javascript y JQuery para la parte de interfaz de usuario.

El proyecto ha pasado por unas etapas de desarrollo, empezando por el diseño de la aplicación para interactuar servidor e interfaz, pasando por la integración de las técnicas PLN dentro de la aplicación. Posteriormente el desarrollo de un sistema de autenticación de la API de Twitter basado en el protocolo OAuth y finalmente una etapa de evaluación del código y testeado de la aplicación (diseño de interfaz y correcta ejecución de las técnicas).

Como conclusión final, y más en concreto una valoración final del proyecto realizado, he aprendido que el procesamiento de lenguaje natural es un área de investigación muy laboriosa y que sus frutos se deben a años de investigación, métodos de prueba-error y necesita de elevadísimos conocimientos lingüísticos. También he tenido el gusto de poder observar, mientras dedicaba tiempo al proyecto, como en el grupo de investigación del GPLSI perteneciente al DLSI se llevan a cabo dichas tareas de procesamiento.

6. Bibliografía y referencias

Díaz-Llairó, Amparo. El talento está en la red. Lid Editorial Empresarial, 2011 http://www.inqualitas.net/articulos/17114-definicion-historia-y-objetivo-de-una-red-social

D'Monte, Leslie. Swine flu's tweet tweet causes online flutter. Noticia publicada en Business Standard News el 29 de abril de 2009. http://www.business-standard.com/article/technology/swine-flu-s-tweet-tweet-causes-online-flutter-109042900097_1.html

Mitkov, Ruslan. 2003. The Oxford Handbook of Computational Linguistics (Oxford Handbooks in Linguistics S.). Oxford University Press.

Moreno Boronat, Lidia, Manuel Palomar Sanz, Antonio Molina Marco, y Antonio Ferrández Rodríguez. 1999. Introducción al procesamiento del lenguaje natural. Universidad de Alicante.

Luhn, H. P. 1958. The automatic creation of literature abstracts. En Inderjeet Mani and Mark Maybury, editors, Advances in Automatic Text Summarization, pa´ginas 15–22. MIT Press.

Edmundson, H. P. 1969. New methods in automatic extracting. En Inderjeet Mani and Mark Maybury, editors, Advances in Automatic Text Summarization, pa´ginas 23–42. MIT Pres

Elena Lloret, Manuel Palomar: Towards automatic tweet generation: A comparative study from the text summarization perspective in the journalism genre. Expert Syst. Appl. 40(16): 6624-6630 (2013).

Lloret Pastor, Elena. Generación de resúmenes de textos basados en Tecnologías del Lenguaje Humano. Director: Manuel Palomar Sanz. Memoria de suficiencia investigadora. Universidad de Alicante, 2009.

Orasan, Constantin. 2009. Comparative Evaluation of Term-Weighting Methods for Automatic Summarization. Journal of Quantitative Linguistics, 16(1), 67–95.

Tatiana Vodolazova, Elena Lloret, Rafael Muñoz, and Manuel Palomar. A Comparative Study of the Impact of Statistical and Semantic Features in the Framework of Extractive Text Summarization. Proceedings of TSD, volume 7499 of Lecture Notes in Computer Science, page 306-313. Springer, (2012).

Tatiana Vodolazova, Elena Lloret, Rafael Muñoz, Manuel Palomar: The role of statistical and semantic features in single-document extractive summarization. Artif. Intell. Research 2(3): 35-44 (2013).

Elena Lloret and Manuel Palomar: COMPENDIUM: A Text Summarisation Tool for Generating Summaries of Multiple Purposes, Domains, and Genres. Natural Language Engineering (NLE), 19(2): 147-186. ISSN: 1351-3249.

Lluís Padró and Evgeny Stanilovsky. FreeLing 3.0: Towards Wider Multilinguality Proceedings of the Language Resources and Evaluation Conference (LREC 2012) ELRA. Istanbul, Turkey. May, 2012.

K. Holmqvist, J. Holsanova, M. Barthelson, D. Lundqvist. The mind's eye: Cognitive and applied aspects of eye movement research. Elsevier Science, Amsterdam (2003) Ch. Reading or Scanning? A study of newspaper and net paper reading, pp. 657–670.

Página de desarrolladores de twitter: https://dev.twitter.com/

Página de aplicaciones de Twitter: https://apps.twitter.com/

Manual de PHP: http://es1.php.net/manual/es/

Manual de Javascript: http://www.w3schools.com/js/default.asp

Stackoverflow: http://stackoverflow.com

Tutorial para la creación de aplicaciones de Twitter con PHP y OAuth:
http://www.pedroventura.com/desarrollo-web/enviar-tweets-automaticamente-en-php-con-la-api-oauth-de-twitter-y-un-cron-job/

OAuth 1.0a: Introducción e implementación utilizando PHP, PECL OAuth y Twitter:
http://blog.margenn.com/post/2135398085/oauth-introduccion-implementacion-php-pecl-twitter

Usando Twitter como sistema de autenticación en tu sitio:
http://www.maestrosdelweb.com/editorial/twitter-autenticacion-oauth-api-login/

Twitter OAuth Sign in tutorial: Get a user's profile information:
http://webhole.net/2009/09/23/sign-in-with-twitter-example/

Estudio del uso de Internet en los dispositivos móviles:
http://marketingland.com/report-nearly-40-percent-of-internet-time-now-on-mobile-devices-34639