NotweetBot

Basilio Carrero Nevado
Daniel Jiménez Rodríguez

15 de enero de 2015

Índice

1. Introducción

El presente trabajo tiene como objetivo desarrollar un sistema multiagente y así afianzar los conocimientos vistos en la asignatura de Sistemas Multiagentes. Para llevar a cabo dicho desarrollo se hará uso de la plataforma SPADE (del inglés, Smart Python multi-Agent Development Enviroment).

1.1. SPADE

SPADE es una plataforma libre de sistemas Multiagentes basada en la tecnología XMPP/Jabbery desarrollada en Python. Dicho de otro modo, es una plataforma orientada a la construcción desistemas multiagentes que ofrece muchas facilidades al desarrollador. SPADE tiene soporte delestándar FIPA y es la primera plataforma de agentes basada en la tecnología XMPP [1]. XMPP (anteriormente Jabber) es un protocolo abierto basado en XML que se creó para serusado en sistemas de mensajería instantánea. En Enero de 2007 Jabber Software Foundation cambió su nombre por el de XMPP Standards Foundation [2].

1.2. Propuesta

Como propuesta inicial para este trabajo se presentó "EmoBot: Robot para el análisis de la respuesta emocional a las noticias de prensa". Este proyecto pretendía estudiar los estados de Facebook, tomando aquellos estados que contengan alguna emoción (Figura 1) y también un enlace a alguna noticia (un enlace a alguna página web que se encuentre en una lista de posibles páginas definida previamente, no cualquier enlace). De este modo se podría estudiar qué noticias están siendo de interés y qué sentimientos están provocando estas noticias.

Figura 1: Emociones en Facebook

Se encontraron problemas de permisos para poder utilizar las APIs de Facebook. Este hecho unido a que no es posible controlar los casos de sarcasmo o ironía de los sentimientos asociados a ciertas noticias tuvo como consecuencia llevar este mismo planteamiento a la red social de Twitter. El proyecto final sería capaz de recuperar aquellos tweets que contengan un enlace a una página web aceptada (que esté en la lista predefinida de páginas web aceptadas) y extraerñia el título de dicha noticia. El resultado final será la exposición de los títulos de las noticias aceptadas ordenadas por número de repeticiones de la noticia.

Para este trabajo se aceptarán aquellos enlaces que se dirijan a subdominios de: elmundo.es, 20minutos.es, abc.es, eldiario.es, elpais.com, eleconomista.es, as.com, marca.com, mundodeportivo.com, antena3.com, lainformacion.com, publico.es, ultimahora.es, lavanguardia.com y sport.es.

2. Objetivos

Las tareas que debe llevar a cabo el sistema por orden de realización son: recuperar tweets, filtrar aquellos que tengan un enlace a una página web que sea de interés (que esté en la lista predefinida de páginas web aceptadas), dirigirse a dicha página web, extraer el título de la noticia, llevar un contador de las veces que se ha twitteado el enlace y mostrar en una interfaz los títulos de las noticias ordenados por número de veces que se repite (en orden decreciente).

El sistema contará con dos agentes que se llamarán Sender y Receiver que se comunicarán y entre los que se repartirán estas tareas (Figura 2).

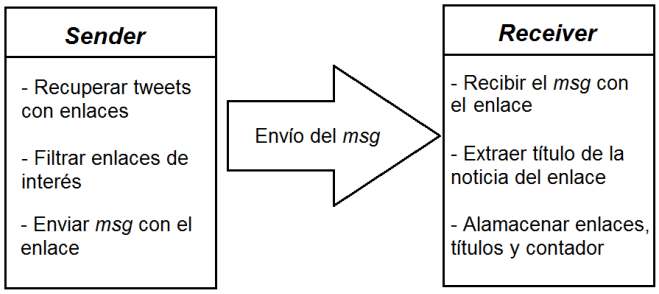

Figura 2: Agentes Sender y Receiver

El agente Sender será el encargado de recuperar tweets. Recuperará tweets que contengan al menos un enlace. Filtrará los enlaces quedándose solamente con aquellos que sean de interés y si encuentra un enlace a las páginas web definidas previamente mandará un mensaje al agente Receiver con dicho enlace.

Al agente Receiver se le delegan otras tareas. Deberá recibir el mensaje del agente Sender con el enlace. El agente extraerá el título de la noticia de la página web a la que lleva dicho enlace. Después añadirá este título a alguna estructura de datos. Esta estructura de datos debe llevar un contador con las veces que se ha encontrado el mismo enlace, por lo que si ya ha aparecido un determinado enlace no necesitará realizar la extracción del título y simplemente aumentará este contador. El agente también es el encargado de mostrar el resultado en una interfaz gráfica.

La exposición del planteamiento es suficientemente explícita en cuanto al funcionamiento el sistema. Pero antes de empezar con el desarrollo se deben traducir y agrupar estas tareas definiendo los comportamientos que tendrán los agentes en el sistema multiagente resultante.

El agente Sender debe contar con un comportamiento cíclico (finalmente esto no será así, ver apartado 3.1. del presente documento). Este comportamiento estará recuperando tweets continuamente. Dentro del comportamiento se realiza el filtrado de los enlaces y los que correspondan se envían al agente Receiver.

El agente Receiver también contará con un comportamiento cíclico. Debe mantenerse a la espera de que el agente Sender le envíe un enlace. Cuando se recibe un mensaje con un enlace este comportamiento extrae el título de la noticia de la web y lo almacena o bien, si ya había recibido el mismo enlace, lo busca entre los almacenados y aumenta su contador.

3. Tecnología

A parte de los módulos y librerías empleadas cabe destacar el uso de la estructura de datos diccionario. El diccionario ha sido la estructura elegida para el almacenamiento de los enlaces encontrados. La clave del diccionario es la propia url correspondiente al enlace de la noticia y el valor es una tupla formada por el número de veces que se ha encontrado la misma noticia y por el título que corresponde a esa noticia.

Para el envío de mensajes entre el agente Sender y el agente Receiver se han empleado los mecanismos de comunicación que ofrece SPADE. Se han utilizado mensajes FIPA-ACL indicando en el agente que envía quién sería el receptor del mensaje con la ayuda del AID (Agent Identifier).

En esta sección se explica la tecnología empleada en el desarrollo del proyecto. También se añadirán comentarios que no definen como tal a la herramienta empleada pero que se creen oportunos para una mejor comprensión de la sección de código donde aparecen (código en el siguiente apartado).

Tweepy

Tweepy es una librería en la que se encuentran recopilados los métodos necesarios para acceder a la API de Twitter. Nos permite recuperar tweets filtrándolos por ciertos tags que se definan y por el idioma [3].

Para recuperar los tweets se genera un "listener" que se mantiene a la espera de recibir un nuevo tweet. Cuando se recibe un tweet se produce una llamada al método "on status" donde se decide qué hacer con el tweet recibido.

Este "listener" se comporta exactamente como deseábamos que fuese el comportamiento cíclico del agente Sender. Por esto, el comportamiento del agente Sender pasa a ser OneShotBehaviour donde se realiza el proceso de autentificación de Twitter y ya el listener posteriormente se encarga de mantener ese comportamiento cíclico deseable.

Twill

La técnica para extraer información de las páginas web, como puede ser el título de una noticia, es conocida como web scraping. Twill automatiza este proceso [4]. En concreto se ha utilizado para guardar el código HTML de la página en un archivo.

BeautifulSoap

Esta librería de Python permite parsear documentos HTML [5]. Esta librería se ha usado para encontrar el título de la noticia en el fichero HTML que anteriormente ha sido generado con Twill. Después de extraer el título de la noticia ya hemos terminado de trabajar con el fichero HTML y por lo tanto lo eliminamos.

os

El módulo os de Python permite acceder a funcionalidades que son dependientes del sistema operativo [6]. En concreto ha sido empleado para eliminar el fichero HTML que se genera previamente con Twill.

PIL

Pyhon Imaging Library, como bien indican las siglas, proporciona capacidades de procesamiento de imágenes y también de gráficos. La principal ventaja es que es compatible con muchos formatos de archivo [8]. En este proyecto se ha utilizado PIL para añadir la imagen que puede observarse en la cabecera de la interfaz gráfica de usuario.

Tkinter

Es considerado un estándar para la interfaz gráfica de usuario en Python. Provee de una potente interfaz orientada a objetos para Tk GUI toolkit [7]. En este proyecto Tkinter ha sido utilizado para el desarrollo de la interfaz de usuario. El resultado final de esta interfaz consta de dos columnas,

en la primera de ellas aparecen las repeticiones de la noticia y en la segunda el título de la noticia. También se le ha añadido una imagen en la cabecera (Figura 3).

Repeticiones	Titulo
25	Una familia escucha el corazón de su hijo fallecido en el pecho del hombre al que salvó la vida - ANTENA 3 TV
15	El último partido de Thierry Henry
11	Dimite el policía de Ferguson responsable de la muerte de Michael Brown - ANTENA 3 TV
8	Suiza reta a la UE con un referéndum para impedir la entrada de inmigrantes | Internacional | EL PAÍS
7	Miles de personas reclaman en toda España “pan, techo y trabajo” | España | EL PAÍS
7	El videojugador profesional puede ganar 40.000 euros anuales
7	El eje de la pobreza vive de la economía sumergida | Economía | EL PAÍS
5	Otra noche de enfrentamientos en Hong Kong deja 28 detenidos y diez heridos | Internacional | EL MUNDO
4	"En Icesi apostamos por el aprendizaje activo y este hace la diferencia": Rector - diario El Pais
4	Obligados a ganar tras la 'décima' del Madrid | barca | sport.es
4	Falsos mitos laborales de las mujeres | Economía | EL PAÍS
4	FIL DE GUADALAJARA 2014: Ricardo Piglia: “La literatura es el escudo de los tímidos” | Babelia | EL PAÍS
4	Nicolás Maduro recorta el presupuesto nacional debido a la caída del precio del petróleo | Venezuela | EL MUNDO
3	Cataluña, España y Podemos, vistos desde lejos | España | EL PAÍS
3	FIL DE GUADALAJARA 2014: Magris reivindica la escritura que da voz a la tragedia y las injusticias | Cultura | EL PAÍS
3	FIL DE GUADALAJARA 2014: No hay Feria del Libro sin García Márquez | Cultura | EL PAÍS
3	Goytisolo: En la tribu del nuevo Cervantes | Cultura | EL PAÍS
3	Muere la defensora de los partos en casa mientras daba a luz en su hogar | Mundo | elmundo.es
3	La desconocida esposa de Cézanne | Cultura | EL PAÍS
3	Dimite el policía blanco que mató al joven negro de Ferguson | Internacional | EL PAÍS
3	Los 25 grandes golazos de Henry | planeta-barca | sport.es
3	México: Desaparición De Los 43 Estudiantes De Iguala - Los dos meses más negros de Peña Nieto - ABC.es
3	La eclosión de la tele-web | Televisión | EL PAÍS
3	Los israelíes de origen palestino, 'ciudadanos de segunda clase' | Internacional | EL MUNDO
3	Mujica Y El Mendigo - El Gobierno uruguayo pide que no se dé dinero a los pobres - ABC.es
3	Suiza vota en referéndum una severa limitación de la inmigración
3	Acusan al Ayuntamiento de Rivas de adjudicar 812.000 euros en contratos al hermano de Tania Sánchez - EcoDiario.es

Figura 3: Resultado final

Pickle

Pickle es un módulo de serialización de Python. Pickle es capaz de transformar un objeto en una cadena de bytes y es capaz de recuperar ese objeto a partir de la cadena de bytes [9]. En este proyecto se utiliza para guardar esa información en un fichero y posteriormente recuperar el objeto diccionario con los títulos.

4. Manual de usuario

Para ejecutar la aplicación es necesario tener instalado Python y SPADE. Python viene instalado por defecto en la mayoría de distribuciones Linux. Para tener instalado y configurado correctamente SPADE se debe abrir una terminal y posteriormente introducir los siguientes comandos:

```
$pip install SPADE
$tar xvzf SPADE -2.1. tar .gz
$cd spade
$python setup .py install
$configure .py myhost . myprovider .com
$runspade .py
```

“.myprovider.com” para configurar SPADE con la dirección IP del equipo donde se ejecute el agente o si se ejecuta localmente es la dirección IP 127.0.0.1.

runspade.py habrá que lanzarlo siempre que se quiera lanzar la aplicación, ya que se requiere que esté ejecutándose SPADE.

Se debe abrir otra terminal para lanzar la aplicación. Antes de ejecutar la aplicación hay que instalar las siguientes librerías si no están ya instaladas:

```
$pip install twill
$pip install beautifulsoup4
$sudo apt - get install python -tk
$apt - get install python - imaging -tk
$pip install tweepy
```

Ya se puede lanzar la aplicación. Una vez colocados en el directorio del proyecto ejecutar el siguiente comando. Es importante recordar que se debe estar ejecutando SPADE.

```
$python MyAgent .py
```

Nota: Se ha preparado el código para una ejecución local, integrándolo todo en un único fichero para así facilitar su ejecución. Se podrían separar el Sender y el Receiver en dos ficheros diferentes y tan solo sería necesario dividir el main para lanzar cada agente por separado, y también funcionaría todo correctamente de la misma manera.

5. Código fuente

El código fuente se puede encontrar en el fichero MyAgent.py. Este fichero se puede abrir con cualquier editor de texto.

6. Conclusión

Se ha propuesto un problema y se ha resuelto. Como resultado del proyecto se obtiene un sistema que cumple su propósito y que tiene aplicaciones reales. Con el sistema resultante se pueden estudiar cuáles son las noticias más populares del momento en Twitter en un marco de páginas web de noticias predefinidas. El sistema es fácilmente escalable en cuanto a la ampliación de este marco, teniendo que a nadir simplemente la url de enlaces deseables a la lista "fuentes".

Con algunas sencillas variaciones se pueden modificar los objetivos del sistema. Algunos ejemplos pueden ser la inclusión sólo de un género de noticias, la búsqueda de noticias similares filtrando con ciertas palabras clave el título de la noticia o la inclusión de una sola url en el marco de webs predefinidas y así estudiar qué noticias de esa página están siendo más relevantes para los usuarios de Twitter.

Habiendo visto la plataforma Jade en la asignatura de Sistemas Multiagentes, la realización del presente trabajo ha servido por un lado para afianzar los conocimientos adquiridos en la asignatura y por otro lado para conocer SPADE, otra plataforma de desarrollo de sistemas multiagentes diferente.

Referencias

[1] SPADE Users Manual http://pythonhosted.org//SPADE/

[2] http://xmpp.org/about-xmpp/history/

[3] http://tweepy.readthedocs.org/en/v3.1.0/

[4] http://twill.idyll.org/

[5] http://www.crummy.com/software/BeautifulSoup/

[6] https://docs.python.org/2/library/os.html

[7] https://docs.python.org/2/library/tkinter.html

[8] http://effbot.org/imagingbook/pil-index.htm

[9] https://docs.python.org/2/library/pickle.html