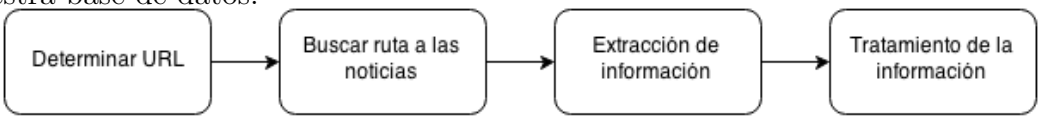
Extracción de la información

Introducción

La extracción de información es uno de los puntos clave del proyecto. Como ya hemos comentado anteriormente extraemos información de varias fuentes, siendo la más importante de ellas la web oficial del ayuntamiento de cada municipio. El caso que vamos a estudiar es el de aquellos ayuntamientos que hacen uso del gestor de contenidos OpenCMS, ya que se puede suponer que la estructura de las páginas serán similares al estar creadas con la misma herramienta. Tras un estudio inicial para determinar la estructura interna de estas páginas hemos determinado siete formas distintas de estructurar los contenidos de la web. Como tenemos varios grupos, el primer paso es determinar a partir de la dirección web la ruta hacia las noticias, una vez queda determinada la ruta se procede a la extracción de las noticias. Para cada grupo se ha creado un script en Python para extraer las noticias según su estructura. Por último se trata la información obtenida para adecuarla nuestra base de datos.



Obtención de ruta hacia la información

Las páginas que hacen uso del gestor de contenido OpenCMS no almacenan la página principal en el directorio raíz del servidor, en la mayoría de casos esta se encuentra bajo la ruta *.../opencms/opencms/X* donde *X* es normalmente el nombre del municipio. El determinar esta parte de la ruta a veces es un proceso simple pues al hacer una petición HTTP al servidor la URL de respuesta ya es la ruta completa por lo que hay una redirección automática y solo hay que seguirla, también puede ocurrir que se dé la redirección automática pero la URL de respuesta no incluya la dirección completa, en este caso lo que se hace es buscar todos los enlaces internos de la página principal haciendo uso de expresiones regulares, ya que incluyen la parte de OpenCMS que buscamos, y partimos los enlaces en tantas partes como caracteres /"tenga el enlace y nos quedamos con las tres primeras partes para reconstruir parcialmente el enlace, estos enlaces parciales se guardan en una lista para que una vez tenemos todos enlaces poder buscar el elemento más común de la lista, que será la parte de la dirección que nos falta. Cuando no se da la redirección automática el servidor devuelve un pequeño código HTML en el que utiliza el atributo *HTTP-EQUIV* para forzar al navegador

a hacer un refresco de la página con la dirección completa, en este caso hacemos, de nuevo, uso de expresiones regulares para obtener la parte extra que añade OpenCMS y construir la dirección completa que nos lleva a la página principal del ayuntamiento que estamos estudiando. Una vez que tenemos la ruta completa, hacemos un repaso por los enlaces internos para determinar la estructura de la página y se compara con los siete grupos que obtuvimos en el análisis inicial intentando acceder a distintas rutas conocidas probando cuál de ellas es correcta y no devuelve ningún código de error al hacer la petición HTTP para obtener la ruta hasta la información que deseemos buscar, en este caso las noticias. Cuando se tiene la ruta a la información que se desea extraer se lanza el script correspondiente para comenzar a extraer información.

Extracción de información

Una vez tenemos la ruta hacia las noticias hay dos casos bien diferenciados, el primero es que las noticias se encuentren ordenadas por categorías, por el contrario en el segundo caso las noticias se encuentran todas en el mismo lugar sin ningún tipo de organización. Si ya se encuentran ordenadas por categorías las almacenamos en nuestra base de datos respetando la categoría que se le da en la web. Si no se le asigna una categoría, las guardamos bajo una categoría especial para posteriormente, durante la fase de tratamiento de la información, asignarles una categoría. En ambos casos el proceso que se sique para extraer las noticias es el siguiente, lo primero es acceder al código HTML donde se encuentra la información, para ello hacemos uso de la librería urllib2 para hacer la petición HTTP y guardar la respuesta, que si no se ha habido ningún problema será el código HTML. Una vez tenemos el código completo hacemos uso de la librería BeutifulSoup, uno de los parseadores de HTML para Python más utilizados. Con esta librería seleccionamos y extraemos el contenido de las etiquetas HTML que contienen la información que buscamos, a veces sacamos dicha información (fechas, texto y enlaces) haciendo uso de expresiones regulares y otras simplemente guardamos el texto que queda dentro de las etiquetas. Como obtenemos el enlace a la noticia completa también accedemos a esta dirección para obtener el texto completo de la noticia y en algunos casos la fecha si no ha sido posible obtenerla en el paso anterior. Una vez se ha extraído toda la información posible se construye el enlace a la siguiente página de noticias y seguir extrayendo información de manera recursiva.

Tratamiento de la información

A la hora de almacenar la información obtenida en concreto los resúmenes y los textos completos de las noticias nos hemos encontrado con determinados caracteres que no son admitidos por el gestor de base de datos. El origen de estos caracteres especiales está en que posiblemente OpenCMS ofrezca formularios para permitir al usuario publicar información y en ese formulario se copie el texto de otros programas que si admiten estos caracteres especiales como editores de texto por lo que el primer paso es eliminar estos caracteres, para ello repasamos todo el texto eliminando caracteres que queden fuera de la codificación UTF-8. Otra situación que nos obliga a hacer tratamiento del texto obtenido son los fallos que contiene el código de las propias páginas web, ya que en muchos casos las etiquetas no están correctamente cerradas lo que hace que parte del código y comentarios se interpreten como texto por la librería BeautifulSoup. Estos fallos en ocasiones provocan una visualización incorrecta de la web. En este apartado la tarea principal es realizar la clasificación automática de aquellas noticias que se encuentren sin clasificar. Para ello buscamos aquellas palabras que tienen relevancia en cada categoría para poder comparar el texto de cada noticia con estos conjuntos de palabras y determinar con la mayor certeza posible a que categoría pertenece.