Tipología y ciclo de vida de los datos

Práctica 1: ¿Cómo podemos capturar los datos?

Antonio García-Bustamante Usano

Mario García Puebla

agarcia-bustamante@uoc.edu
mariogar@uoc.edu

Universitat Oberta de Catalunya

1. Contexto

El sistema desarrollado en esta práctica obtiene información de la página web de Fotocasa, https://www.fotocasa.es/es/. Esta página web proporciona información acerca de las viviendas en venta en todo España, en donde se puede filtrar la búsqueda por diferentes regiones, a partir del nombre de la localidad o su código postal. Al filtrar por una localidad aparecen todas las viviendas en venta en esa zona, con una pequeña descripción de cada que contiene una determinada información tal como su título, precio y características principales.

2. Título

El nombre elegido para el dataset generado en esta práctica es **Viviendas_Madrid_Oeste.**

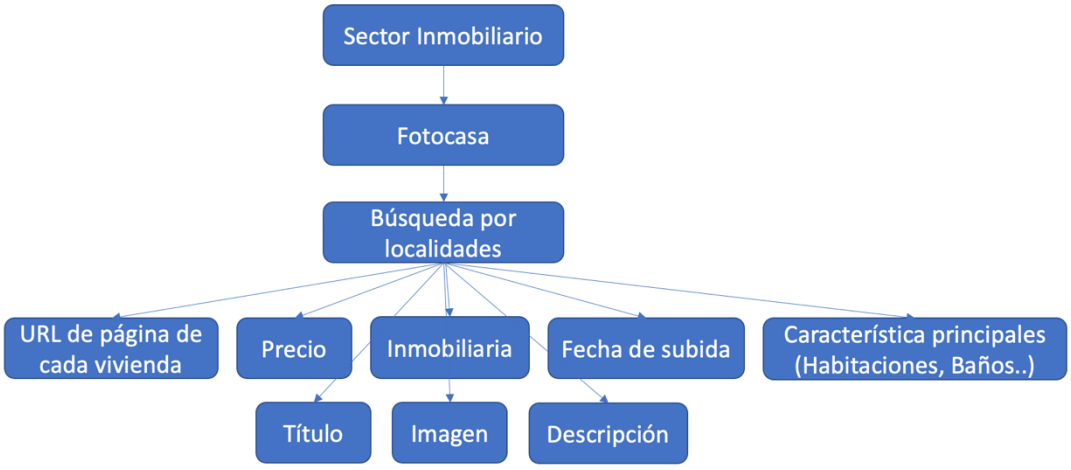
3. Descripción del dataset

Se ha elegido este nombre ya que la información obtenida almacena datos de las viviendas en venta en las localidades que componen la zona de Madrid Oeste:

- Brunete
- Guadarrama
- Becerril de la Sierra
- Boadilla del Monte
- El Boalo
- Cercedilla
- Colmenarejo
- Collado Mediano
- Collado Villalba
- El Escorial
- Fresnedillas de la Oliva
- Galapagar
- Guadarrama
- Hoyo de Manzanares
- Majadahonda
- Los Molinos
- Moralzarzal
- Navacerrada
- Navalagamella
- Pozuelo de Alarcón
- Quijorna
- Robledo de Chavela
- Las Rozas de Madrid
- San Lorenzo de El Escorial
- Santa María de la Alameda
- Torrelodones
- Valdemaqueda
- Valdemorillo
- Villanueva de la Cañada
- Villanueva del Pardillo.

4. Representación gráfica

A continuación, se muestra un diagrama que identifica el dataset.



5. Contenido

Este dataset contiene los siguientes campos:

- URL: enlace a la ficha completa de la vivienda donde se puede ver más información sobre ella que la que aparece en la vista previa.
- Imagen: nlace a la fotografía principal del inmueble.
- Título: título descriptivo de cada vivienda.
- Inmobiliaria: Hace referencia al grupo inmobiliario que administra la venta de cada vivienda.
- Precio: Incluye el precio de cada vivienda.
- Descripción: Larga descripción de la información más específica de cada vivienda.
- Fecha: Incluye el periodo de tiempo que lleva esa vivienda en el portal.
- Propiedades: Incluye las características principales de cada vivienda, como son las habitaciones, los baños, los metros cuadrados y otras características como si hay garaje, piscina, terraza o demás.
- Localidad: Hace referencia a la localidad donde está situada la vivienda.

El periodo de los datos es el más reciente posible, se ha recogido la información actual que había en el momento del *scraping*. En el campo de fecha se puede ver el periodo de tiempo desde la publicación de cada vivienda.

6. Propietario

En cuanto a los propietarios de los datos, se entiende que hay dos principales, Fotocasa y los dueños de cada vivienda.

Fotocasa es el propietario del conjunto de datos completos, en donde los dueños de cada vivienda le proporcionan la información a Fotocasa para que pueda mostrarla y hacer uso de ellos. También hay que tener en cuenta que los datos hacen referencia a los dueños de cada casa que son ellos los que dejan a Fotocasa administrarlos.

Para actuar de acuerdo a principios éticos, no se ha obtenido ninguna información de carácter sensible de cada vivienda, como pudiera ser la dirección, el número de teléfono o nombre de los propietarios. Se ha accedido a información como el título, descripción o características, nada que pueda comprometer a los dueños de las viviendas.

Por otro lado, con este proyecto no se pretende emplear esta información para poder crear un nuevo portal o página con ella, haciendo competencia directa a la web de Fotocasa, sino poder hacer un análisis de las viviendas para luego realizar recomendaciones. Para ello, se ha descargado la URL de cada vivienda donde se da acceso al link de esa vivienda en Fotocasa, por lo que la idea es siempre poder volver a conducir a los usuarios a la página de Fotocasa y no hacer uso de su información que pudiese perjudicarles.

7. Inspiración

Se ha obtenido este dataset porque la idea es poder hacer un análisis de recomendaciones sobre estos datos en la propia página web de Fotocasa.

Como se ha comentado, no se quiere aprovechar la información de Fotocasa de forma individual, sino que se quiere demostrar como de potente podría ser usar esta información en la propia página web de Fotocasa si tuviera integradas una serie de métricas para poder proporcionar más servicios a los usuarios.

Además, en la situación complicada que se vive hoy en día, la posibilidad de poder ayudarnos de herramientas que puedan ayudarnos a elegir las mejores opciones a la hora de adquirir una herramienta.

Con ello, la idea es poder usar estos datos para sacar estadísticas de las viviendas en venta por cada localidad, observando sus precios y pudiendo decir donde se encuentran las casas más asequibles o donde el €/m^2 es más barato. También se puede comparar con datos históricos y observar donde crece más el precio y donde se pierde valor.

Además, el programa está preparado para que su alcance sea mucho mayor. Simplemente agregando nombres de localidades al fichero cfg/localidades.csv se puede extender a un mayor número de poblaciones.

Por último, como líneas futuras, se considera ampliar el proyecto haciendo que el *bot* acceda a la ficha completa de cada inmueble y extraiga información más detallada.

8. Licencia

La licencia que se adecua a este dataset es “Released Under CC0: Public Domain License”. Esto se debe a que en ningún momento se usa información comprometida de los propietarios de la información, ni se hace un uso indebido de ella. Además, se está empleando información que es de acceso público y no presenta ningunas restricciones.

9. Código

El código ha sido incluido en la carpeta source del repositorio según las indicaciones.

Para la ejecución del código, será necesario la instalación de selenium, multiprocessing y BeautifulSoup a través de los siguientes comandos:

- `pip install selenium`
- `pip install multiprocessing`
- `pip install bs4`

En cuanto al proceso seguido por el código, en primer lugar, se crea un multiproceso para poder ejecutar varios procesos de forma paralela y así llevar a cabo un *scraping* mucho más rápido. El usuario puede introducir, modificando una variable, el número de procesos que desea ejecutar simultáneamente, de acorde a la prestaciones de su computador.

Cada proceso, va a llamar a una función que se encarga de crear un *driver* que a su vez accede a la página web de fotocasa, https://www.fotocasa.es/es/, acepta el mensaje de cookies que aparece y posteriormente accede con el usuario creado introduciendo el nombre de usuario y su contraseña. Estos datos no se incluyen dentro del código por privacidad, sino que se encuentran en el fichero /cfg/credentials.csv para que pueda rellenarlos con comodidad.

Una vez accedido con el usuario, se introduce el nombre de la localidad a buscar según el multiproceso en el motor de búsqueda, y se accede a las páginas con las viviendas de cada localidad. Dentro de la búsqueda por localidad, se recorre toda la página mediante un scroll para acceder a toda la información y permitir que se cargue el html completo. A continuación, se obtienen los datos más importantes de cada vivienda gracias al uso de Beautiful Soup.

Después, se comprueba si hay más páginas de información para esa localidad. En caso de que las haya, el *bot* pulsa el botón de siguiente y repite el proceso previo hasta que no haya más. Si no se encuentran, se cierra ese driver para esa localidad.

Por último, se juntan todos los ficheros obtenidos de cada localidad en uno sólo, que será el dataset final, mediante el script join.py.

En este código se han encontrado dos grandes dificultades, una de ellas ha sido poder recorrer varias páginas de información para cada localidad, ya que el sistema para encontrar el botón de siguiente es complejo, sobre todo a la hora de ver si existía o no, debido a la arquitectura de la página. Al final, pudiendo comprobar el enlace al que llevaba el último botón de siguiente página, se podía comprobar si había más páginas o no.

El segundo inconveniente con el que nos encontramos fue crear varios flujos de procesos en paralelo y hacer que, cuando estos hilos acabaran, se ejecutaran los siguientes hasta el final.

Para solucionar esto se crea un nuevo archivo, join.py, que permite que en el main.py terminen de ejecutar todos los hilos guardando bien sus datos por separado y posteriormente guardarlos sin tener que comprobar que haya algún hilo suelto ejecutando que no haya terminado de guardar sus datos.

10. Dataset

En este apartado se adjunta el enlace al dataset publicado en Zenodo.

https://doi.org/10.5281/zenodo.7315303

11. Vídeo

A continuación, se adjunta el link al vídeo explicativo.

https://drive.google.com/drive/u/1/folders/1YmPx1igx_GR7cfhTBaN5coaZ_qYeitvR

12. Contribuciones al trabajo

Contribuciones	Firma
Investigación previa	Mario García, Antonio García-Bustamante
Redacción de las respuestas	Mario García, Antonio García-Bustamante
Desarrollo del código	Mario García, Antonio García-Bustamante
Participación en el vídeo	Mario García, Antonio García-Bustamante