Universitat Oberta de Catalunya

uoc.edu

Tipología y ciclo de vida de los datos

PRA 1: Web scraping

Abril 2022

Alicia Contreras Garrudo - acontrerasga
Daniel Garcia Sousa - dgarciasou

1. Contexto

Este documento forma parte de la **PRA1** de la asignatura Tipología y Ciclo de Vida de los Datos, en la que se utilizan técnicas de web scrapping en Python con las librerías de BeatifulSoup y Selenium para la obtención de datos de productos de distintas webs de ecommerce. Se han seleccionado las webs de Amazon y El Corte Ingles (ECI) para generar un dataset a partir de una búsqueda.

Esta práctica ha sido realizada por **Alicia Contreras** y **Daniel García**.

2. Título

El título escogido para nuestro dataset es: ***ecommerce_products_dataset***

3. Descripción del dataset

Los datos que se extraen de los resultados de la búsqueda de Amazon y El Corte Ingles son:

- **product**: Término de búsqueda. Se corresponde con el producto buscado. Tipo: String
- **name**: Nombre del producto. Tipo: String
- **brand**: Marca del producto. Tipo: String
- **price**: Precio del producto. Tipo: Float
- **discount_percent**: Porcentaje de descuento aplicado. Tipo: Float
- **rating**: Valoración del producto sobre 5. Tipo: Float
- **n_coments**: Número de comentarios de usuarios. Tipo: Integer
- **image**: Url de la imagen principal del producto. Tipo: String
- **express_delivery**: Si el producto tiene opción de envío express. Tipo: Boolean
- **ecommerce**: Código de la tienda a la que pertenece el producto. Tipo: String

4. Representación gráfica

A continuación, se incluye un diagrama que identifica visualmente el proyecto elegido y el flujo de obtención de datos.

5. Contenido

El proyecto se compone de 3 scripts dentro del directorio "src":

- **sailor.py**: funciones encargadas de interactuar con los portales de eCommerce para introducir los criterios de búsquedas y navegar entre las distintas páginas de resultados.
- **pricescraper.py**: funciones encargadas de parsear datos HTML para obtener la información de los productos para ser almacenadas posteriormente en los ficheros correspondentes.
- **main.py**: script principal que lee los términos de búsqueda de un fichero TXT y ejecuta el script sailor.py para cada uno de ellos.

El código permite al usuario generar un dataset de productos a partir de un término de búsqueda.

Para ejecutar el script es necesario instalar las siguientes bibliotecas:

```
pip install selenium
pip install requests
pip install beautifulsoup4
pip install webdriver_manager
pip install lxml
```

Seguidamente, procederemos a ejecutar el script de la siguiente manera:

```
python main.py PARAMETRO_1 PARAMETRO_2
```

Donde:

- **PARAMETRO_1**: Ruta a un fichero txt con los términos de búsqueda a realizar, una por línea del fichero. Un ejemplo de fichero es "data/products_list.txt".

Ejemplo del contenido del fichero txt:

```
sudadera negra
pantalón vaquero
zapatillas
hdmi
altavoz
anillo dorado
cinturón
```

- **PARAMETRO_2**: Número de paginas de las que obtener los resultados de la búsqueda, tanto en Amazon como en El Corte Ingles.

Utilizando como ejemplo el siguiente comando, se generará un dataset de los resultados de las búsquedas definidas en el fichero en las 4 primeras páginas de cada uno de los portales de ecommerce especificados anteriormente.

```
python main.py data/products_list.txt 4
```

Tras la ejecución del comando previo, se originará un dataset formado por los campos definidos en el apartado 3, que contienen características de los productos. No podremos saber de antemano el número de registros que contendrá el dataset, ya que depende en gran medida de lo específica que sea la búsqueda y del número de productos que respondan a esa búsqueda. Para este caso concreto se generarán aproximadamente un número ***n*** de productos para cada término de búsqueda de la lista recibida como input y multiplicado por 2, ya que se extraerán tanto datos de Amazon como de El Corte Inglés.

6. Agradecimientos

Agradecemos a los propietarios del conjunto de datos, que serían las plataformas de ecommerce en las que hemos basado la extracción de nuestros datos.

- https://www.amazon.es/
- https://www.elcorteingles.es/

7. Inspiración

El principal motivo del desarrollo de esta herramienta es facilitar una búsqueda preliminar de productos online utilizando técnicas de ciencia de datos en una tarea que se suele hacer manualmente. A continuación, se describen otras motivaciones para la utilización de esta herramienta:

- Realizar un análisis comparativo de precios de productos con respecto a distintas webs de e-commerce
- Realizar un estudio del posicionamiento de productos y del funcionamiento de los buscadores en webs de e-commerce
- Obtención de datasets de imágenes para entrenamiento de algoritmos de clasificación, categorizados con la clase referente a la palabra de búsqueda utilizada por el usuario

También es destacable la oportunidad de utilizar estas herramientas en datos reales. Además, es interesante destacar el hecho de que la gran mayoría de webs que ofrecen servicios para hacer web scraping y obtener información de productos de Amazon son de pago. De El Corte Inglés no conseguimos encontrar ninguna que hiciese algo similar. Es por eso que también pensamos que sería interesante contribuir con datos de productos que pudiesen otenerse de manera gratuíta.

8. Licencia

9. Código

El código puede obtenerse desde el siguiente repositorio de Github:
https://github.com/aliciacg7/ecommerceWebScraper

10. Dataset

El dataset se ha publicado en la web de Zenodo. A continuación, se incluye el enlace y el DOI del dataset.

- Enlace a Zenodo
- Enlace al dataset publicado
- DOI

11. Contribuciones

Finalmente se incluye la tabla de contribuciones revisada por todos los miembros del equipo.

CONTRIBUCIONES	FIRMA
INVESTIGACIÓN PREVIA	DGS, ACG
REDACCIÓN DE LAS RESPUESTAS	DGS, ACG
DESARROLLO DEL CÓDIGO	DGS, ACG