Práctica 1: Web Scraping

Las 1.000 Películas Mejor Valoradas en FilmAffinity (2013–2023)

Noviembre 2023

Índice de contenidos

Integrantes del Grupo

- Juan Antonio Tora Cánovas
- Tim Thorp

Enlaces Importantes

- Página de resumen en el sitio web elegido: **FilmAffinity**
- Detalles de la película número 1/1000 del conjunto de datos: **FilmAffinity**
- Repositorio con el código de la práctica: **GitHub**
- Dataset publicado en Zenodo: **Enlace DOI**
- Vídeo de presentación de la práctica: **Google Drive**

1 Contexto

Explicar en qué contexto específico se han recolectado los datos y argumentar por qué el sitio web seleccionado es una fuente pertinente y fiable de esa información. Indicar la dirección del sitio web.

Los datos se han recolectado con el propósito de analizar tendencias y preferencias en la industria cinematográfica a lo largo de la última década. FilmAffinity es un reconocido agregador de reseñas y calificaciones de películas que cuenta con una vasta base de usuarios activos. Las calificaciones en FilmAffinity se generan por millones de usuarios, lo que proporciona una muestra representativa de la recepción de las películas. La dirección web desde donde se han extraído los datos es www.filmaffinity.com.

---

2 Título

Definir un título conciso y que sea descriptivo para el dataset.

Aunque ya se ha presentado en la portada, se incluye aquí también para mantener la enumeración del enunciado: **Las 1.000 Películas Mejor Valoradas en FilmAffinity (2013–2023)**

---

3 Descripción del dataset

Desarrollar una breve descripción del conjunto de datos que se ha extraído. Es necesario que esta descripción sea coherente con el título elegido.

El conjunto de datos extraído para este estudio comprende las 1.000 películas mejor valoradas en FilmAffinity desde el año 2013 hasta 2023. Este periodo de tiempo se seleccionó para capturar las tendencias contemporáneas en la valoración de películas y para reflejar los cambios en las preferencias de la audiencia a lo largo de los últimos diez años.

---

4 Representación gráfica

Dibujar un esquema o diagrama que refleje visualmente el dataset y el proyecto elegido.

La representación gráfica la podemos ver en la Figura 1, a continuación:

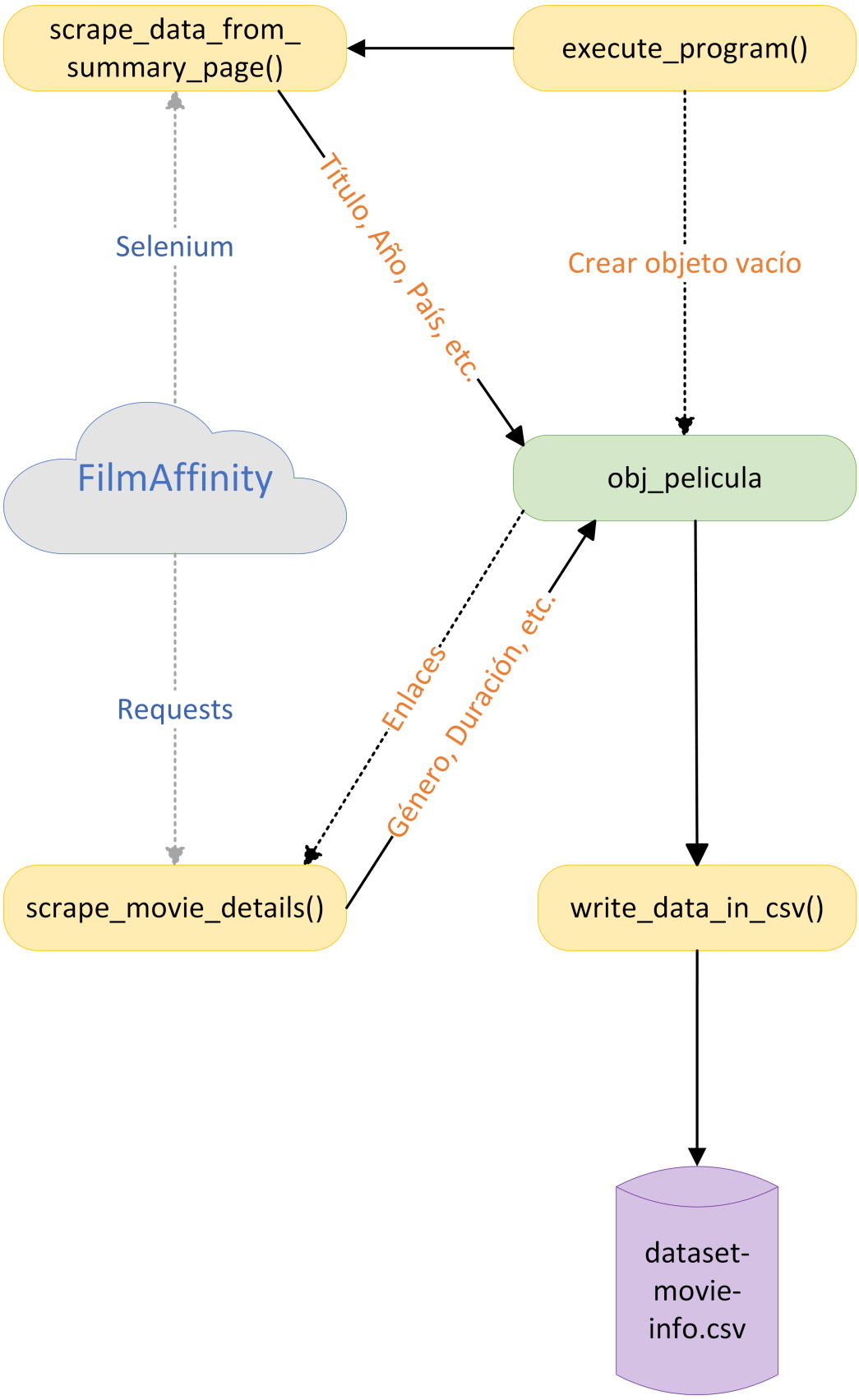

Figure 1: Esquema del proyecto

5 Contenido

Explicar los campos que se incluyen en el dataset y el período de tiempo al que pertenecen los datos.

El dataset abarca el período de 2013 a 2023 y los datos se extrajeron de FilmAffinity en Noviembre de 2023. A continuación, se detallan los campos incluidos:

5.1 Título

El `Título` representa el nombre de la película tal como se lanzó en España. Este campo es de tipo cadena de caracteres.

5.2 Título Original

El `Título Original` refiere al nombre de la película en su país de origen, el cual puede coincidir o no con el nombre en España. Este campo es de tipo cadena de caracteres.

5.3 Año

El campo `Año` indica el año de lanzamiento de la película. Todos los valores son enteros y se encuentran en el rango de 2013 a 2023.

5.4 Duración

La `Duración` se registra en minutos y representa la duración total de la película. Este campo es de tipo entero.

5.5 Género

`Género` describe los géneros cinematográficos de la película, pudiendo incluir múltiples géneros separados por comas. Este campo es de tipo cadena de caracteres.

5.6 País

El campo `País` detalla el país o países de origen de la producción de la película. Este campo es de tipo cadena de caracteres.

5.7 Puntuación Media

La `Puntuación Media` es un valor decimal que refleja la calificación promedio otorgada por los usuarios, en una escala de 1,0 a 10,0, usando la coma como separador decimal.

5.8 Número de Puntuaciones

`Número de Puntuaciones` indica cuántas valoraciones ha recibido la película. Este campo es de tipo entero.

5.9 Director

`Director` lista el nombre del director o directores de la película. En caso de múltiples directores, sus nombres se separan por comas. Este campo es de tipo cadena de caracteres.

5.10 Reparto

El `Reparto` incluye los nombres de los actores principales, separados por comas. Este campo es de tipo cadena de caracteres.

5.11 Sinopsis

La `Sinopsis` ofrece un breve resumen del argumento de la película. Este campo es de tipo cadena de caracteres y usualmente consiste en dos o tres frases.

5.12 Enlace

El `Enlace` es la URL a la página específica de FilmAffinity para la película en cuestión. Este campo es de tipo cadena de caracteres.

---

6 Propietario

Presentar al propietario del conjunto de datos. Es necesario incluir citas de análisis anteriores o, en su defecto, justificar esta búsqueda con análisis similares. Indicar qué pasos se han seguido para actuar de acuerdo con los principios éticos y legales en el contexto del proyecto elegido.

El propietario del conjunto de datos es FilmAffinity, un reconocido sitio web de cine cuyo dominio está registrado bajo la empresa *Arsys Internet, S.L. dba NICLINE.COM*.

6.1 Revisión de Políticas y Términos de Uso

Se llevó a cabo un análisis detallado de los términos de uso y la política de privacidad de FilmAffinity antes de comenzar la extracción de datos. De acuerdo con el archivo `robots.txt` de la plataforma y las condiciones estipuladas en su sitio web, el scraping de datos no está explícitamente prohibido siempre que no se haga con la intención de manipular las puntuaciones o en maneras que violen las políticas de privacidad y protección de datos personales de los usuarios. Se ha realizado la recolección exclusivamente de datos agregados y públicos, evitando cualquier manipulación y excluyendo todo tipo de información personal de los usuarios.

6.2 Limitación de la Frecuencia de Acceso

Para evitar la sobrecarga de los servidores de FilmAffinity (*hal.ns.cloudflare.com* y *jule.ns.cloudflare.com*) y mantener un servicio estable para otros usuarios, las solicitudes de extracción de datos se realizaron con intervalos de 3 segundos entre cada una. Información adicional sobre el propietario y los detalles técnicos pueden ser consultados ejecutando el código 6 del directorio `source/otras_funcionalidades` en nuestro repositorio de Github.

6.3 Análisis Similares

Aunque no se han encontrado estudios previos que analicen específicamente las valoraciones de FilmAffinity, sí existen investigaciones relevantes en plataformas análogas. Por ejemplo, Bristi et al. (2019) implementaron técnicas de machine learning para estimar las calificaciones en IMDb. Del mismo modo, Firmanto y Sarno (2018) y Harish et al. (2019) investigaron el análisis de sentimientos en las reseñas de Rotten Tomatoes e IMDb, respectivamente.

---

7 Inspiración

Explicar por qué puede ser interesante este conjunto de datos y qué preguntas se pretenden responder con ellos. Es necesario comparar con los análisis anteriores o análisis similares presentados en el apartado 6.

El estudio de Bristi et al. (2019) compartía varias variables con nuestro estudio, como `Título`, `Director`, `Género`, `País`, `Año` y `Puntuación Media`. Tal como ellos lo hicieron, nuestro conjunto de datos también podría utilizarse para predecir las valoraciones de las películas basándonos en sus características, añadiendo variables como `Duración` y `Reparto`.

Aunque los estudios de Firmanto y Sarno (2018) y de Harish et al. (2019) se enfocaron en el análisis de sentimientos de las reseñas, con nuestro conjunto de datos podríamos aplicar esta técnica al análisis de las sinopsis para descubrir si existen palabras o temas recurrentes en aquellas películas con altas calificaciones. También es posible identificar grupos de películas con tramas similares mediante análisis de texto de las sinopsis.

Otras preguntas de interés podrían ser:

- ¿Cómo han evolucionado los géneros cinematográficos en popularidad a lo largo de la última década?
- ¿Qué directores tienden a trabajar con frecuencia con los mismos actores?
- ¿Las películas cuyos títulos originales no se traducen directamente al castellano tienen un rendimiento diferente en cuanto a puntuaciones en comparación con aquellas cuyos títulos se han adaptado?

---

8 Licencia

Seleccionar una licencia adecuada para el dataset resultante y justificar el motivo de su elección. Ejemplos de licencias que pueden considerarse:

- *Released Under CC0: Public Domain License.*
- *Released Under CC BY-NC-SA 4.0 License.*
- *Released Under CC BY-SA 4.0 License.*
- *Database released under Open Database License, individual contents under Database Contents License.*
- *Otra (especificar cuál).*

Hemos decidido asignar al conjunto de datos de este proyecto la licencia **CC BY-SA 4.0**. Esta licencia permite que otros remezclen, adapten y construyan sobre este trabajo, incluso con fines comerciales, siempre y cuando otorguen el reconocimiento adecuado y distribuyan sus trabajos derivados bajo una licencia idéntica. Dado que nuestro conjunto de datos se compone de información pública, y que nuestra intención es alentar un análisis y uso abierto, esta licencia fomenta la compartición y la creatividad en el uso del contenido (Creative Commons, 2023).

---

9 Código

Código implementado para la obtención del dataset, preferiblemente en Python o, alternativamente, en R.

- *El código deberá ubicarse en la carpeta `/source` del repositorio.*
- *Se deben indicar las librerías y versiones utilizadas. P. ej., en Python pueden obtenerse mediante el comando `pip3 freeze > requirements.txt`*
- *En la memoria en PDF, se deben comentar los aspectos más relevantes sobre cómo el código realiza el proceso de recolección de datos, qué dificultades presenta el sitio web elegido, y cómo se han resuelto.*

9.1 Lenguaje de programación utilizado

El proyecto se desarrolló utilizando Python.

9.2 Repositorio de GitHub

El código fuente está alojado en un repositorio privado de GitHub, al que únicamente tiene acceso el profesor. Se encuentra en el siguiente enlace, bajo el directorio `/source/`: **GitHub Repository**.

9.2.1 Estructura del Directorio de Código Fuente en `/source/`

- `/source/`: Contiene dos directorios adicionales: uno para el código fuente del programa principal y otro para seis scripts de funcionalidades independientes que complementan la práctica.
- `/source/main`: Aloja el código principal de la práctica PR1, que incluye tres scripts: `main.py`, `librerias_propias.py`, y `dtos.py`.
- `/source/otras_funcionalidades`: Contiene seis scripts independientes de Python utilizados para pruebas de rendimiento, la creación de dos funciones complejas y la extracción de datos del propietario y la tecnología web utilizada por el sitio web.

9.2.2 Contenidos en la Raíz del Directorio de GitHub

- `README.md`: Documento con detalles exhaustivos sobre el contenido del repositorio y las instrucciones para el uso y ejecución del código.
- `/dataset`: Directorio que incluye el conjunto de datos, denominado `dataset_movie_info.csv`.
- `/documentos`: Directorio con tres archivos: `memoria.pdf` y `memoria.Rmd` (la plantilla para el PDF) y `requirements.txt`, que lista todas las librerías de Python instaladas y utilizadas en el proyecto.

9.2.3 Librerías de Python Utilizadas en el Proyecto

Las librerías específicas utilizadas son: `requests`, `pandas`, `selenium`, `time`, `csv`, `os`, `builtwith`, `whois`, y `lxml`, tal y como se especifica en el `README.md`

9.3 Enlace al Sitio Web del Proyecto

Para visitar el sitio web de FilmAffinity, haga clic **aquí**.

9.4 Desafíos y Soluciones en el Proceso de Web Scraping

9.4.1 Dificultades Enfrentadas

- **Problema 1:** Generación dinámica de contenido en el sitio mediante Javascript.
  - **Solución:** Uso de Selenium y LXML tras la renderización ineficaz con request_html.
- **Problema 2:** Alta cantidad de consultas HTTP necesarias.
  - **Solución:** Implementación de LXML por su velocidad superior.
- **Problema 3:** "Datos perdidos" al recolectar el HTML.
  - **Solución:** Uso de rutas XPATH relativas para la identificación de elementos.
- **Problema 4:** Bloqueos por parte del servidor web.
  - **Solución:** Implementación de `time.sleep(3)` entre llamadas HTTP.
- **Problema 5:** Tiempos de ejecución elevados con `request` y `BeautifulSoup`.
  - **Solución:** Reducción del tiempo mediante `selenium` y `lxml`.
- **Problema 6:** Restricciones del archivo `robots.txt` en Amazon.
  - **Solución:** Elección de FilmAffinity por su archivo `robots.txt` permisivo.
- **Problema 7:** Desconocimiento de las normativas de licencias.
  - **Solución:** Selección de datos públicos disponibles sin necesidad de login.
- **Problema 8:** Falta de conocimiento sobre las tecnologías del sitio web.
  - **Solución:** Uso de la librería `builtwith` para identificar tecnologías web.
- **Problema 9:** Necesidad de información sobre el propietario del sitio web.
  - **Solución:** Obtención de datos del propietario mediante la librería `whois`.

9.5 Detalles del Código en `/source/otras_funciones`

En este directorio, encontrarás seis scripts que ofrecen funcionalidades específicas y altamente focalizadas. Aunque los archivos 1, 2, 3 y 6 van más allá del alcance de nuestra práctica principal, han sido esenciales para la elaboración de la memoria y la comprensión de los tiempos de respuesta entre distintas tecnologías.

9.6 Visión General de las Funcionalidades en `/source/main`

El núcleo del programa se encuentra en tres archivos:

- Al ejecutar `main.py`, se inicia el programa, que a su vez llama a las funciones principales alojadas en `librerias_propias.py`.
- `librerias_propias.py` es el script más complejo y extenso, que utiliza `dtos.py` para instanciar el objeto DTO definido dentro de él.

Por lo tanto, `main.py` actúa como el punto de entrada del programa, `librerias_propias.py` contiene la lógica central, y `dtos.py` define la estructura de datos utilizada a través del programa.

9.7 Descripción de Funciones en "/source/main/librerias_propias.py"

9.7.1 execute_program()

Esta es la función principal que dirige el flujo del programa. Administra el tiempo, gestiona el WebDriver con Selenium y LXML, maneja el "user-agent", ejecuta las cuatro funciones principales y, finalmente, escribe la información resultante en la consola al concluir el programa.

9.7.2 return_html_after_scraping_movie_info_from_summary_page(browser, num_clics)

Esta función obtiene el HTML de la página de resumen mediante un proceso de cuatro pasos:

1. Un bucle "for" realiza scrolling y clics para ejecutar el Javascript que carga el contenido dinámico.
2. Hay pausas de 3 segundos entre cada acción para no sobrecargar el sitio web.
3. Se captura y almacena el HTML generado.
4. La variable que contiene el HTML se devuelve al final de la función.

9.7.3 scrape_data_from_summary_page(obj_pelicula)

El objetivo de esta función es extraer datos de la página de resumen. Se sigue esta secuencia:

1. Inicialización de listas vacías.
2. Dos bucles "for" llenan las listas con los datos relevantes.
3. Los datos se asignan al objeto DTO a través de sus métodos "setters".
4. Se devuelve el objeto DTO configurado.

9.7.4 scrape_movie_details(obj_pelicula, tuplas_de_datos_maximas)

Esta función procesa cada enlace obtenido para extraer y acumular datos específicos:

1. Se preparan listas vacías.
2. Un bucle "for" itera tantas veces como enlaces haya.
3. Por cada enlace, se obtiene y raspa el HTML correspondiente.
4. Después de cada iteración, se espera 3 segundos para evitar la sobrecarga del servidor.
5. El objeto DTO se actualiza con los datos recopilados.
6. El objeto DTO se retorna al finalizar la función.

9.7.5 write_data_in_csv(obj_pelicula)

Esta función es responsable de trasladar los datos recabados a un archivo CSV:

1. Se comienza escribiendo la cabecera con los nombres de las columnas.
2. Las filas de datos se añaden sucesivamente hasta concluir o hasta que se maneje una excepción.
3. Al finalizar, se genera el archivo CSV y el control retorna a `execute_program()` para presentar los resultados y cerrar el programa.

---

10 Dataset

Publicar el dataset obtenido en formato CSV en Zenodo, incluyendo una breve descripción del mismo. Obtener y adjuntar el enlace del DOI del dataset (https://doi.org/). El dataset también deberá incluirse en la carpeta `/dataset` *del repositorio. Si existe alguna circunstancia que impida publicar abiertamente el dataset real en Zenodo, se deberá:*

a. *Comentar esta circunstancia y justificar el motivo.*
b. *Generar un dataset simulado y publicarlo en Zenodo, obteniendo el enlace del DOI*
c. *Comunicar al profesor el dataset real de forma privada (p. ej., en el repositorio privado o en una carpeta de Google Drive privada)*

El dataset está disponible en 2 sitios:

1. En el repositorio privado de GitHub, con acceso para el profesor de la asignatura: **Github Repository**
2. En Zenodo, a través del siguiente enlace DOI: **Enlace DOI**

---

11 Video

Realizar un breve vídeo explicativo de la práctica ***(máximo 10 minutos)****, que deberá contar con la participación de los dos integrantes del grupo. En el vídeo se deberá realizar una presentación del proyecto, destacando los puntos más relevantes, tanto de las respuestas a los apartados como del código utilizado para extraer los datos. Indicar el enlace del vídeo (https://drive.google.com/), que deberá ubicarse en el Google Drive de la UOC.*

Para acceder al video de presentación de la práctica, clic aquí: **Google Drive**

Tabla de Contribuciones

Las iniciales representan la confirmación por parte del grupo de que el integrante ha participado en dicho apartado.

Contribuciones	Firma
Investigación previa	JATC, TT
Redacción de las respuestas	JATC, TT
Desarrollo del código	JATC, TT
Participación en el vídeo	JATC, TT

Bibliografía Utilizada

1. Bristi, W. R., Zaman, Z., & Sultana, N. (2019). Predicting IMDb rating of movies by machine learning techniques. *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1-5. IEEE.
2. Creative Commons. (2023). Licencia Creative Commons Atribución-CompartirIgual 4.0 Internacional (CC BY-SA 4.0). Recuperado de https://creativecommons.org/licenses/by-sa/4.0/deed.es
3. Firmanto, A., & Sarno, R. (2018). Prediction of movie sentiment based on reviews and score on Rotten Tomatoes using SentiWordNet. *2018 International Seminar on Application for Technology of Information and Communication*, 202-206. IEEE.
4. Harish, B. S., Kumar, K., & Darshan, H. K. (2019). Sentiment analysis on IMDb movie reviews using hybrid feature extraction method.
5. Lawson, R. (2015). Scraping the Data. En *Web Scraping with Python* (Capítulo 2). Packt Publishing Ltd.
6. Selenium. (2023). Getting started. Recuperado de https://www.selenium.dev/documentation/webdriver/getting_started/
7. Subirats, L., & Calvo, M. (2018). *Web Scraping*. Editorial UOC.