# ESTRUCTURA DE LA PRESENTACIÓN WEB SCRAPING

1. Web scraping
   1. ¿Qué es web scraping?
   2. Uso e historia de Web Scraping
      1. ¿Dónde y cómo se utiliza?
      2. ¿Cómo se llegó al Web Scraping?
   3. ¿Qué herramientas se emplean?
      1. Python (Beautiful Soup y Scrapy)
      2. Selenium
      3. ParseHub
      4. Octoparse
   4. Ejemplos prácticos de Web Scraping
2. Datasets
   1. ¿Qué es dataset?
   2. Uso e historia de Dataset
      1. ¿Dónde y cómo se utiliza?
      2. ¿Cómo se llegó al Dataset?
   3. ¿Qué herramientas se emplean?
   4. Ejemplos prácticos de elaboración de Datasets

Web scraping

Web scraping es una técnica de extracción de datos de páginas web de forma automática. Esta técnica emplea programas o scripts que navegan por páginas web, acceden a su contenido y recogen información que luego se almacena en bases de datos o datasets para ser analizados posteriormente.

El web scraping ha evolucionado conjuntamente con la tecnología debido a la necesidad de recopilar la mayor cantidad de información posible. La evolución del web scraping puede dividirse en varias fases:

Origen: Antes, todos los datos se recopilaban de forma manual, navegando por los sitios web y almacenándolos en hojas de cálculos o bases de datos. Este proceso, además de ser lento, se fue complicando a medida que aparecían sitios web dinámicos y crecía la cantidad de datos en línea.

En 1993, Matthew Gray escribió World Wide Web Wanderer, un programa que recorría la web y recopilaba información de páginas web de forma automatizada.

En 1993, surgió el primer motor de búsqueda basado en un robot web. JumpStation dio el paso a que nuevos web crawlers apareciesen durante esos años, como Infoseek, Altavista y Excite; que basaron sus motores en JumpStation, utilizándolo como núcleo.

Aparición de la Web 2.0: En la década de los 2000, empezaron a alzarse los sitios web de contenido dinámico y personalizado. Este hecho impulsó al desarrollo de técnicas más elaboradas.

En 2004, Beautiful Soup proporcionaba ayuda para analizar el contenido web, permitiendo la extracción de datos directamente de los contenedores de las páginas HTML.

Aparecen navegadores automatizados como Selenium, que permite la ejecución de pruebas de software, permitiendo editar acciones o crearlas desde cero, simulando la experiencia de un usuario real.

Democratización y estándares: Con el paso de los años, el web scraping se ha ido regulando, de tal forma que actualmente sigue ciertos estándares a cumplir, aunque a día de hoy siguen surgiendo ciertos problemas legales con los términos de servicio o la privacidad, y problemas con la carga del servidor. Las herramientas se han mejorado, dando lugar a:

Scrapy, un framework de código abierto escrito en Python que permite la construcción de spiders complejos.

Servicios en la nube como ParseHub y Octoparse que hacen más accesible el scraping debido a que se elimina la necesidad de programar.

Usos comunes: Hoy en día, el web scraping tiene distintos usos que abarcan varios campos, desde la Inteligencia Artificial e investigación hasta el estudio económico del mercado o los medios comunicativos.

Análisis de mercado: Se utiliza para recopilar precios y características de productos que están disponibles en el mercado.

Investigación: Permite obtener datos de sitios académicos y gubernamentales.

Contenido de medios: Permite recopilar información acerca de noticias o entradas a blogs.

IA: Se pueden entrenar modelos a partir de los datos recuperados.

Tal y como se ha mencionado anteriormente, existen muchas herramientas que permiten realizar web scraping tanto si eres programador, ingeniero o si lo haces como disfrute personal. Estas herramientas son:

Python y Selenium: Desde Python, mediante librerías como Beautiful Soup o Scrapy, se puede realizar web scraping.

ParseHub y Octoparse: Existen otras formas de hacer web scraping sin necesidad de hacer código.

MOSTRAR EJEMPLOS DE

Python y Selenium.

ParseHub y Octoparse.

Datasets

Un dataset es un conjunto de datos organizados, normalmente, en tablas que se utiliza para el análisis de datos, el entrenamiento de IA y la toma de decisiones. En un dataset, cada fila representa un registro, y cada columna contiene una variable o atributo del registro descrito. Un dataset se compone de variables, registros y metadatos. Un dataset puede ser estructurado, no estructurado o semiestructurado.

Los datasets han ido evolucionando junto a las TIC y a la ciencia de datos, diferenciándose varias fases:

Inicios en Bases de Datos: Entre los años 60 y 80 surgieron los primeros datasets almacenados en las bases de datos relacionales. Para tratar con estos conjuntos de datos se empleaban herramientas como SQL, facilitando la creación y la manipulación de datasets.

Era de Internet: Entre los años 90 y los 2000, con el nacimiento de Internet, la cantidad de datos que había en la red aumentó drásticamente, proveniente de usuarios, redes sociales y sistemas. Para manejar esta cantidad de datos, surgieron herramientas como Hadoop (maneja datasets grandes y complejos) y UCI Machine Learning Repository (proporciona datasets para uso académico y aprendizaje automático) que facilitaron el tratamiento de la cantidad masiva de datos, conocidos como Big Data.

IA y Big Data: A partir de 2010, los gobiernos y las organizaciones científicas decidieron promover la publicación de datasets con el fin de fomentar la transparencia. El auge de las IAs también ha fomentado el cambio en los datasets, surgiendo herramientas como ImageNet y GPT.

Usos comunes: Los datasets se emplean en:

Ciencia de datos e IA: Los datasets son útiles para el entrenamiento de modelos de IA y la detección de patrones, facilitando la predicción de eventos en ciertos campos como los mercados o los sistemas de salud.

Negocio y Marketing: los datasets ayudan a analizar los clientes de cierto mercado, sesgando sus gustos y optimizando las estrategias publicitarias.

Investigación: Los datasets permiten realizar investigaciones y publicaciones, debido a toda la cantidad de información que contiene.

Los datasets siguen una serie de pasos para llevar a cabo su tratamiento:

1. Recopilación de datos:
2. Web scraping (Beautiful Soup, Scrapy, Selenium)
3. APIs (Postman, Kaggle)
4. Datos sintéticos (DataSynthesizer o Faker)
5. Procesamiento de datos
6. Pandas
7. OpenRefine
8. Almacenamiento de datos
9. MySQL/PostgreSQL
10. AWS

MOSTRAR EJEMPLO A PARTIR DEL DATASET QUE HAYAMOS CREADO EN LA PRIMERA PARTE:

1. Usamos los datos recopilados con el Web Scraping
2. Los tratamos con Pandas
3. Los almacenamos en un CSV o en MySQL/PostgreSQL

## REFERENCIAS

1. <https://www.youtube.com/watch?v=nBYIbwgTR7E&list=PLheIVUbpfWZ17lCcHnoaa1RD59juFR06C>
2. ¿Qúe es web scraping? : <https://www.youtube.com/watch?v=bK3EwIMHm94&list=PLcpNLPfo7LVIZj8mU1ka9DLhMIbarxpFK>
3. <https://www.octoparse.es/blog/como-comenzo-y-sucedera-en-futuro>
4. <https://en.wikipedia.org/wiki/World_Wide_Web_Wanderer>
5. <https://en.wikipedia.org/wiki/JumpStation>
6. <https://www.crummy.com/software/BeautifulSoup/bs4/doc/>
7. <https://sentrio.io/blog/que-es-selenium/>
8. <https://scrapy.org>
9. <https://es.wikipedia.org/wiki/Scrapy>
10. <https://elmundodelosdatos.com/extraccion-datos-sitios-web-productos-zara/>
11. <https://github.com/scrapy/scrapy>
12. <https://www.parsehub.com>
13. <https://www.octoparse.es>
14. <https://aws.amazon.com/es/what-is/hadoop/>
15. <https://archive.ics.uci.edu>
16. <https://www.postman.com>
17. <https://www.kaggle.com/datasets>
18. <https://openrefine.org>
19. <https://faker.readthedocs.io/en/master/>
20. <https://github.com/DataResponsibly/DataSynthesizer>