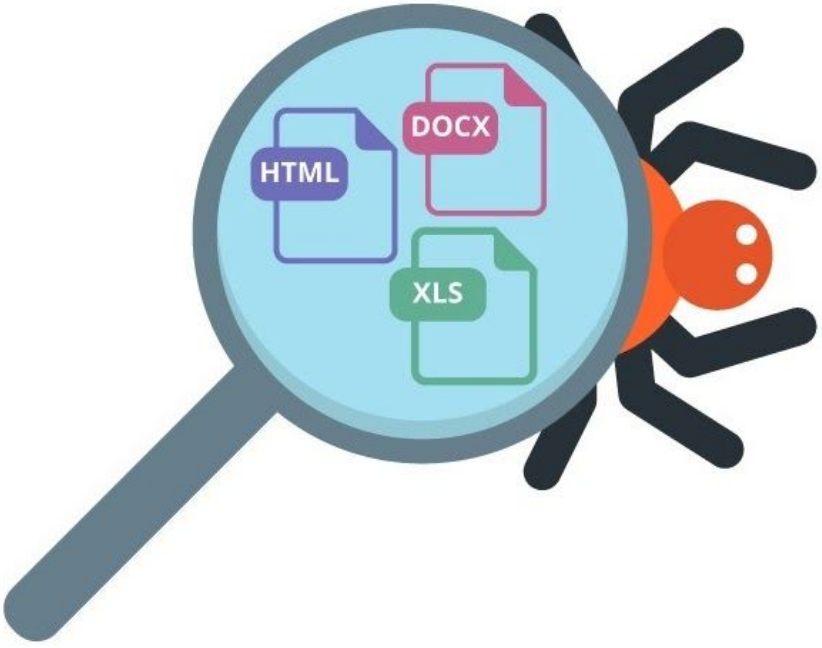
Estado del arte del Web Scraping

Estudio del estado del arte de las principales tecnologías relacionadas con el Web Scraping y la extracción de información

Integrantes:
Julio Arrieta
Joaquin Suarez
Carlos Balbiani



2020/09/30

índice:

1) Definición de Web Scraping:

El Web Scraping se trata de la recolección web o la extracción de datos web, se traduce como “raspado de datos” que se utiliza para extraer datos de sitios web
Está automatizado porque utiliza bots para obtener la información del sitio web.
Es un análisis programático de una página web para descargar información de ella.
El scraping de datos implica localizar los datos y luego extraerlos. No copia y pega, sino que, obtiene directamente los datos de manera precisa. No se limita a la web, los datos se pueden obtener prácticamente desde cualquier lugar donde se almacenan. Puede ser Internet u otra fuente de datos.
Ejemplo de web scraping: El web scraping implicaría obtener la información de una página web específica, por ejemplo, obtener la información de precios de determinados productos de las páginas de Amazon.

2) Algunas de las tecnologías más representativas:

Estas son algunas tecnologías disponibles, aunque existen muchas más, nos centraremos solo en estas, con el fin de no extender demasiado este documento.

Import.io, Hunter.io, Webhose.io, Octoparse, Beautiful Soup, Jsoup, Scrapy.

3) Criterios de evaluación:

Vamos a evaluar las siguientes tecnologías mediante diversos criterios y realizaremos una comparación entre las mismas, para determinar, la tecnologías que más se adapte a nuestras necesidades.

a) Capacidad de crawling personalizado:

El término de crawler viene de la forma en la cual se desplaza una araña. Básicamente es un bot de internet que navega en forma sistemática en la Word Wide Web, generalmente con el propósito de indexar la web. El crawling personalizado implica, que el usuario puede, de forma más o menos sencilla, elaborar un crawler con las características específicas deseadas. No entran aquí servicios de oferta de crawlers predefinidos.

b) Calidad de la Documentación:

Que tan buena, bien explicada y ejemplificada es la documentación de esta tecnología e incluso su comunidad y repositorios públicos gratuitos, esta característica permite reducir la curva de aprendizaje y puesta en marcha de sistemas para la recolección de datos usando cada herramienta. Debido a la importancia de este último criterio, no podemos simplemente observar si dispone de ella o no, debemos considerar fuertemente su calidad.

c) De pago:

La aplicación requiere una habilitación monetaria previa para su uso, sin contar con las demos o simplemente es gratis, esta característica será vital, dado el carácter académico del proyecto.

d) Interfaz gráfica:

No evaluaremos la experiencia de usuario con la interfaz gráfica, sino, simplemente, si dispone de ella o no.
Cuanto más intuitiva sea una herramienta, más limitadas serán sus opciones de uso. Por otro lado, aquellas que no ofrecen ninguna clase de facilidad al usuario no técnico, posibilitan, mayor escalabilidad, mayores casos de uso y mayor potencia.

e) Machine learning:

Algunas tecnologías brindan técnicas de machine learning e incluso Inteligencia Artificial lo cual posibilita el mejoramiento de los algoritmos de búsqueda. Sin Embargo estas técnicas requieren muchos recursos, lo cual encarece el servicio de web scraping, de modo que evaluaremos, a una tecnología simplemente por el hecho de disponer de esta característica o no.

4) Evaluación particular

Contextualización:
Import.io, es una plataforma que facilita la conversión de información semiestructurada en páginas web en datos estructurados, que se pueden utilizar para cualquier cosa, desde la toma de decisiones comerciales hasta la integración con aplicaciones y otras plataformas. Ofrecemos recuperación de datos en tiempo real a través de nuestras API de transmisión y basadas en JSON REST, e integración con muchos lenguajes de programación y herramientas de análisis de datos comunes, es una de las más utilizadas para web scraping.

Evaluación:
Dispone de crawling personalizable, tiene una documentación de muy buena calidad, pero desafortunadamente no es gratuita y no solo eso, es bastante costosa, pero una versión gratuita, pero solo dura 48 horas, lo que la convierte en mala candidata para realizar investigaciones y pruebas de concepto.
Dispone de una interfaz gráfica de usuario amigable y fácil de utilizar.
También permite utilizar algoritmos de Machine Learning para automatizar la recolección de datos.

Contextualización:
Hunter.io,se trata de una herramienta de extracción de datos, pero centrada en correos electrónicos, es decir, un buscador de emails en dominios personalizados. Útil si quieres contactar con alguna compañía, proporciona datos en línea para crear conexiones entre profesionales. Elabora una lista de todas las direcciones de correo electrónico de una empresa o un dominio disponible públicamente en la web y permite a sus usuarios encontrar direcciones de correo electrónico específicas desde el nombre, el apellido y el sitio web de la empresa.

Evaluación:
No dispone de un crawling personalizable, sin embargo tiene muy buena documentación. tiene una versión de pago y otra gratuita pero es limitada. no dispone de una interfaz gráfica y no proporciona herramientas para machine learning.

webhose.io

Contextualización:
Webhose.io, para los perfiles más técnicos, webhose se convierte en una muy buena alternativa. Tienen sus propios scrapeadores de datos y un servicio de estructuración que facilitan el scrapeo de sitios, además posibilita obtener datos estructurados obtenidos de sitios no estructurados conectándose a una API, a la que puedes hacer 1000 peticiones al mes gratuitas. Así, es fácil obtener los datos en formatos estandarizados y estructurados como XML o JSON.

Evaluación:
Al igual que Hunter.io, no dispone de un crawling personalizable, sin embargo tiene muy buena documentación. tiene una versión de pago y otra gratuita pero es limitada, no dispone de una interfaz gráfica y no proporciona herramientas para machine learning.

Contextualización:
Octoparse, es un moderno software de extracción visual de datos web. Tanto a los usuarios experimentados como a los que no tienen experiencia les resultaría fácil, una buena herramienta para gente que busca extraer datos de sitios web sin tener que codificar nada.

Evaluación:
No dispone de un crawling personalizable y su documentación no es muy buena.
Sin embargo es gratuita, y posee una muy amigable interfaz gráfica.
No dispone de herramientas que posibiliten el uso de machine learning.

Contextualización:
Beautiful Soup, una librería de Python lo cual de por sí ofrece muchas ventajas, funciona con su analizador de favoritos, para proporcionar formas idiomáticas de navegar, buscar y modificar el árbol de análisis. Por lo general, ahorra a los programadores horas o días de trabajo, sin embargo el usuario debe encargarse de muchas tareas y estás limitado a sistemas locales, si lo instalas en arquitecturas cloud, pierdes la ventaja de la gratuidad.

Evaluación:
Dispone de un crawling muy personalizable, pero su documentación no es bastante mala. Sin embargo es gratuita, no posee una interfaz gráfica y no dispone de herramientas que posibiliten el uso de machine learning.

Contextualización:
Jsoup es una librería Java que proporciona operaciones para trabajar con HTML. Permite extraer y manipular datos, que podrán ser utilizados convenientemente para nuestras necesidades.
Con Jsoup podemos construir desde parseadores básicos de HTML para analizar y procesar páginas estáticas hasta herramientas de análisis recursivo de sitios completos (crawlers o spiders). No obstante, Jsoup está más pensado para análisis de páginas estáticas que para un crawler complejo. Si lo que queremos es recopilar diferentes tipos de datos de un sitio completo independientemente de sus URLs, puede ser más adecuado utilizar Crawler.

Evaluación:
Dispone de un crawling muy personalizable, con una muy buena documentación.
Es gratuita, no posee una interfaz gráfica y no dispone de herramientas que posibiliten el uso de machine learning.

Contextualización:

Scrapy es una plataforma colaborativa de código libre que corre en Python orientada al web scraping, utilizado en sitios web para rastrear y extraer datos estructurados de sus páginas. Se puede utilizar para una amplia gama de propósitos, desde minería de datos hasta monitoreo y pruebas automatizadas.
Además es la herramienta más ampliamente utilizada para este propósito.

Evaluación:

Al igual que Jsoup dispone de un crawling muy personalizable, con una muy buena documentación. Es gratuita, no posee una interfaz gráfica y no dispone de herramientas que posibiliten el uso de machine learning.

5) Conclusión:

Dado que el proyecto que desarrollaremos es con fines académicos, y no disponemos de un gran presupuesto, las opciones pagas quedan descartadas (import.io), incluso las opciones con versiones gratuitas pero limitadas (Hunter.io y webhouse), dado que disponemos de opciones completamente gratuitas. Hemos descartado las opciones con una mala documentación por motivos obvios (Octoparse y Beautiful Soup). finalmente debimos decidir entre jsoup y Scrapy, pero Scrapy funciona con python y dado que se nos ha impuesto ese lenguaje, esta última opción es la más adecuada a nuestro propósito.

6) Referencias:

Imoprt.io : https://www.import.io/

Hunter.io : https://hunter.io/

Webhose.io : https://webhose.io/

Octoparse : https://www.octoparse.com/

Beautiful Soup : https://www.crummy.com/software/BeautifulSoup/

Jsoup : https://jsoup.org/

Scrapy : .https://scrapy.org/

herramientas de web scraping:
https://papelesdeinteligencia.com/herramientas-de-web-scraping/