Tổng quan về scrapy – crawl data

Về cách hoạt động của scrapy, đầu tiên scrapy application sẽ gửi 1 request đến web server. Thông thường thì yêu cầu sẽ được gởi qua giao thức HTTP, HTTPS. Scrapy đợi phản hồi từ máy chủ web. Response bao gồm HTML và các thông tin khác như header, cookies. Khi nhận được phản hồi, Scrapy sẽ thực hiện trích xuất dữ liệu và lưu nó và cơ sở dữ liệu

* Về cấu trúc của scrapy

**Scrapy Engine**

* Scrapy Engine có trách nhiệm kiểm soát luồng dữ liệu giữa tất cả các thành phần của hệ thống và kích hoạt các sự kiện
* **Scheduler**
* Giống như một hàng đợi (queue), scheduler sắp xếp thứ tự các URL cần download
* **Dowloader**
* Thực hiện dowload trang web và cung cấp cho engine
* **Spiders**
* Spiders là class được viết bởi người dùng, chúng có trách nhiệm bóc tách dữ liệu cần thiết và tạo các url mới từ response gửi từ web server về để nạp lại cho scheduler qua engine.
* **Item Pipeline**
* Những dữ liệu được bóc tách từ spiders sẽ đưa tới đây, Item pipeline có nhiệm vụ xử lý chúng và lưu vào cơ sở dữ liệu
* **Các Middlewares**
* Là các thành phần nằm giữa Engine với các thành phần khác, chúng đều có mục địch là giúp người dùng có thể tùy biến, mở rổng khả năng xử lý cho các thành phần.

## 2. Luồng dữ liệu

1. Khi bắt đầu crawl một website, Engine sẽ xác định tên miền và tìm vị trí của spider đó và yêu cầu spider đó tìm các urls đầu tiên để crawl
2. Engine nhận danh sách các urls đầu tiên từ spider, gửi cho Scheduler để sắp xếp
3. Engine yêu cầu danh sách cách urls tiếp theo từ Scheduler
4. Engine nhận danh sách các url tiếp theo từ Scheduler vào gửi đến Dowloader (requests)
5. Downloader nhận request và thực hiện việc tải trang, sau khi tải xong sẽ tạo một response và gửi lại Engine
6. Respone từ Dowloader sẽ được Engine đẩy qua Spiders để xử lý
7. Tại Spiders, khi nhận được response, chúng bóc tách thông tin từ response (tilte, content, author, date publish...) và những url có khả năng để crawl và đẩy lại cho Engine (requests)
8. Ở bước này, Engine nhận được kết quả từ Spiders sẽ thực hiện 2 công việc: đẩy những dữ liệu đã được bóc tách tới Item Pipeline để xử lý và lưu vào Databases, đẩy những url mới (requests) mới về Scheduler và quay về bước 3\

Và tiếp theo trên hình chúng ta có thể thấy cấu trúc cơ bản của 1 spider. ở phần trên đầu, sẽ định nghĩa các url bắt đầu của các trang web để gửi cho engine

Css selector and xpath

Tại sao phải sử dụng css selector và xpath ?

Khi mà response trả về từ web server, chúng ta sẽ nhận được nội dung cơ bản như sau, bao gồm header , body, status. Dữ liệu chính chúng ta trích xuất sẽ nằm ở body và thường nó sẽ ở dạng các file html.

Do đó chúng ta sẽ sử dụng css, xpath để truy vấn đến các phần tử dữ liệu chúng ta muốn sử dụng. ví dụ ở đây, chúng ta muốn lấy thông tin về nước ấn độ trong bảng dữ liệu , ta cần truy cập đến thẻ chứa thông tin của ấn độ. Chúng ta sẽ sử dụng xpath và css selector như sau.

Bên trong spider, chúng ta sẽ lần lượt truy vấn đến các phần tử của nó thông qua câu lệnh xpath. Và cuối cùng là xuất dữ liệu ra màn hình qua hàm yield.

Cuối cùng là phần sử dụng user agent. Đối với các trang web hiện nay, sẽ không gửi dữ liệu trả về khi không nhận được các user agent. User Agent là một phần quan trọng của giao thức HTTP mà trình duyệt gửi đến máy chủ web khi bạn truy cập vào một trang web. User Agent cung cấp cho máy chủ web thông tin cụ thể về môi trường truy cập của người dùng.