(12) **BẢN MÔ TẢ GIẢI PHÁP HỮU ÍCH THUỘC BẰNG ĐỘC QUYỀN GIẢI PHÁP HỮU ÍCH**

(19) **Cộng hòa xã hội chủ nghĩa Việt Nam (VN)**
CỤC SỞ HỮU TRÍ TUỆ

(11) **2-0002367**

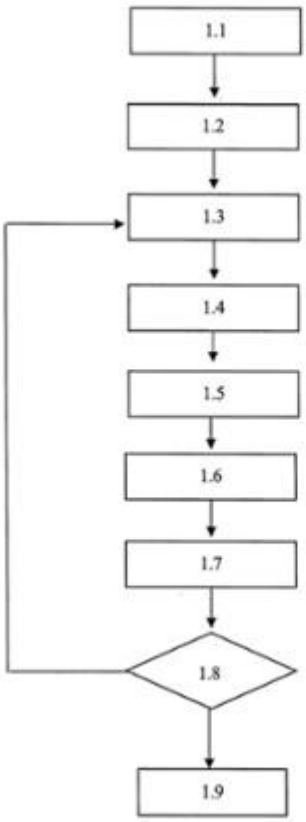
(51)[7] **G06F 16/951**

(13) **Y**

(21) 2-2016-00317
(22) 14/09/2016
(45) 27/07/2020 388
(43) 26/03/2018 360A
(73) Trường Đại học Bách Khoa - Đại học Quốc gia Thành Phố Hồ Chí Minh (VN)
268 Lý Thường Kiệt, phường 14, quận 10, thành phố Hồ Chí Minh
(72) Đặng Trần Khánh (VN); Nguyễn Thanh Tùng (VN); Trương Quang Hải (VN); Lê Thị Bảo Thu (VN).

(54) PHƯƠNG PHÁP THU THẬP DỮ LIỆU TỰ ĐỘNG TỪ CÁC TRANG WEB THƯƠNG MẠI ĐIỆN TỬ

(57) Giải pháp hữu ích này thuộc lĩnh vực Khoa học Máy tính, ứng dụng vào các hệ thống rút trích thông tin tự động từ các trang web thương mại điện tử. Phương pháp được đề xuất trong giải pháp hữu ích bao gồm bảy (7) bước nối tiếp nhau, nhằm mục tiêu giả lập hành vi thông thường của người sử dụng. Phương pháp này đem lại nhiều lợi ích như hạn chế việc thay đổi, giả lập các địa chỉ IP khi bị trang chủ khóa (block). Ngoài ra, phương pháp này còn kết hợp sử dụng nhiều máy khách dưới sự giám sát, điều phối của máy chủ nhằm tối ưu hóa thời gian rút trích, đảm bảo được độ tin cậy. Vì thế, giải pháp hữu ích được đề xuất có ý nghĩa thực tiễn sâu sắc, phục vụ cho mục đích phân tích, hỗ trợ ra quyết định trong các chiến dịch kinh doanh ngày nay.



Lĩnh vực kỹ thuật được đề cập

Giải pháp hữu ích thuộc lĩnh vực Khoa học Máy tính, chuyên ngành Hệ thống thông tin, ứng dụng vào việc rút trích thông tin tự động từ các trang web thương mại điện tử, đề cập đến phương pháp thu thập tự động các loại dữ liệu khác nhau một cách hiệu quả từ các trang web này. Phương pháp được đề xuất kết hợp nhiều phương pháp đơn lẻ bao gồm việc giả lập hành vi của người dùng cũng như giả lập môi trường thu thập thông tin nhằm làm cho việc rút trích thông tin tự động của các trang web thương mại điện tử có tính khả thi và hiệu quả hơn. Mục đích của lĩnh vực kỹ thuật được đề cập là nhằm nâng cao tối đa hiệu qủa khi thu thập dữ liệu tự động trên các trang web thương mại điện tử trong thực tế, qua đó có thể rút trích được đầy đủ, chính xác các thông tin cần thiết.

Tình trạng kỹ thuật của giải pháp hữu ích

Hiện nay, các trang web về thương mại điện tử đang ngày càng xuất hiện nhiều và là một nguồn thông tin to lớn đối với những nhà phân tích dữ liệu. Để có thể thu thập được đầy đủ nguồn thông tin này, cần phải có một phương pháp để rút trích dữ liệu từ các trang web. Do khối lượng dữ liệu là tương đối lớn, và thường xuyên được cập nhật, thay đổi nên việc thu thập thông tin không thể chỉ thực hiện bằng tay, riêng lẻ, mà phải có một hệ thống rút trích tự động bao gồm nhiều máy tính, rút trích dữ liệu đồng thời từ nhiều nguồn khác nhau. Tuy nhiên,việc rút trích tự động này đang gặp nhiều khó khăn vì các trang web thương mại điện tử có thể sẽ áp dụng kỹ thuật ngắt kết nối với những địa chỉ IP (Internet Protocol - giao thức Internet) nào thực hiện việc thu thập dữ liệu. Một vài phương

pháp hữu hiệu đang được sử dụng để ngăn chặn việc thu thập dữ liệu tự động như sau:

- Cấm địa chỉ IP: một cách phổ biến nhất để có thể đánh giá một truy cập có phải là đang rút trích dữ liệu tự động hay không là xem xét tần suất các yêu cầu gửi đến máy chủ. Nếu số lượng các yêu cầu gửi đi từ một địa chỉ IP quá lớn và thường xuyên, địa chỉ IP đó có thể bị chặn. Một yếu tố quan trọng trong phương pháp này là phải tìm ra ranh giới hợp lý của hành vi người dùng thông thường và việc rút trích tự động để không ảnh hưởng đến những khách hàng của trang web.
- Sử dụng Captcha (là hình ảnh chứa một đoạn từ mã khó thấy để kiểm tra tự động nhằm phân biệt máy tính và con người): Đây cũng là một cách rất phổ biến nhằm phân biệt một truy cập đến trang web đang là con người hay là một chương trình máy tính. Phương pháp này tuy gây ra nhiều bất tiện cho người dùng nhưng cũng hữu hiệu để phát hiện các hành vi từ hệ thống rút trích dữ liệu tự động.
- Sử dụng mã JavaScript (là một ngôn ngữ lập trình thông dịch tức là được dịch lúc chạy): Việc lưu trữ các thông tin, dữ liệu qua các thẻ HTML (ngôn ngữ đánh dấu siêu văn bản) thì rất đơn giản, tiện lợi, tuy nhiên chính điều này cũng làm cho dữ liệu dễ dàng bị đánh cắp. Nếu các trang web sử dụng JavaScript để nhận dữ liệu từ máy chủ thì lúc rút trích dữ liệu từ các thẻ HTML sẽ không thấy được dữ liệu nữa.
- Thường xuyên thay đổi, cập nhật cấu trúc trang web: Một trong những phương pháp hữu hiệu nhất để chống lại việc rút trích dữ liệu là thường xuyên thay đổi cấu trúc trang web. Việc thay đổi này bao gồm thay đổi các thẻ HTML và cả cấu trúc phân cấp của trang web. Phương pháp này sẽ

làm rối các hệ thống rút trích thông tin tự động, cho dù họ đã có đầy đủ dữ liệu trong tay. Tuy nhiên, việc áp dụng phương pháp này tốn khá nhiều công sức cũng như thời gian nên thường được thực hiện với tần suất khoảng 1 lần/tháng. Do đó, trong khoảng thời gian này, dữ liệu cũng có thể bị lấy cắp.

- Hạn chế tần số gửi yêu cầu cũng như lưu lượng tải dữ liệu: Phương pháp này làm cho việc thu thập dữ liệu tự động trở nên tốn rất nhiều thời gian.
- Ánh xạ những thông tin quan trọng sang hình ảnh: Những dữ liệu quan trọng như giá cả, địa chỉ email, số điện thoại, thay vì để dưới dạng văn bản thông thường, sẽ được chuyển sang định dạng hình ảnh, gây khó khăn cho việc thu thập thông tin.

Để có thể tiến hành thu thập dữ liệu tự động hiệu quả, một số phương pháp có thể được thực hiện là sử dụng các trình duyệt giả lập như các thư viện Selenium, Mechanize. Đối với việc cấm các IP gửi yêu cầu liên tục, một số phần mềm có thể được sử dụng để che giấu địa chỉ IP thật như BestProxyAndVPN. Tuy nhiên, việc sử dụng các phần mềm này còn rất riêng lẻ, và dễ dàng bị phát hiện bởi các trang web. Do đó, giải pháp hữu ích này nhằm tạo ra một phương thức tổng thể, kết hợp nhiều phương pháp từ việc giả lập môi trường hệ thống, cho đến giả lập các hành vi chi tiết của người sử dụng, giới hạn lại tốc độ rút trích dữ liệu tùy vào khả năng đáp ứng của trang web chủ nhằm có thể làm cho hệ thống rút trích dữ liệu tự động hòa vào đám đông người sử dụng và không thể bị phát hiện. Hay nói một cách khác đi, giải pháp hữu ích này sẽ làm cho hệ thống rút trích dữ liệu có các hành vi tương tự như người dùng thông thường, do đó có thể rút trích đầy đủ các dữ liệu cần thiết.

Ở Việt Nam hiện chưa có bất kỳ kết quả nghiên cứu nào được đăng ký bảo hộ tương tự.

Bản chất kỹ thuật của giải pháp hữu ích

Mục đích của giải pháp hữu ích này là nhằm lấy được đầy đủ, chính xác tất cả các dữ liệu của trang web thương mại điện tử mà không bị phát hiện. Giải pháp hữu ích có thể được áp dụng vào việc rút trích dữ liệu tự động của các trang web thương mại điện tử như amazon.com, lazada.vn, tiki.vn,... Ngoài ra, do dữ liệu từ các trang web là tương đối lớn, việc chỉ rút trích bằng một máy tính là không thực tế. Do đó, giải pháp hữu ích cũng đề cập đến mục đích hỗ trợ nhiều máy tính cùng rút trích thông tin, điều phối nhiệm vụ của các máy tính sao cho thông tin được rút trích không trùng lắp và tối ưu hóa thời gian rút trích dữ liệu. Mục tiêu tổng quát của giải pháp hữu ích là nhằm cung cấp một quy trình rút trích thông tin được thực hiện bởi một máy chủ điều phối, và các máy khách đi thu thập dữ liệu với thời gian thực hiện tối ưu, dữ liệu đầy đủ, chính xác, nhất quán và không bị ngăn chặn bởi các trang web thương mại điện tử.

Để đạt được mục đích trên, giải pháp hữu ích được xây dựng thông qua ba giai đoạn chính như sau:

- Giai đoạn 1: Giả lập môi trường rút trích dữ liệu. Do các trang web hiện nay có khả năng kiểm tra trình duyệt và hệ điều hành của các máy tính có kết nối truy cập đến nó, việc sử dụng các hệ điều hành, trình duyệt lạ có khả năng làm tăng sự nghi ngờ của các trang web thương mại điện tử.
- Giai đoạn 2: Lựa chọn chiến thuật rút trích dữ liệu phù hợp cũng như điều phối, phân chia công việc rút trích dữ liệu cho các máy khách. Tất cả những công việc này được thực hiện bởi máy chủ. Sau khi các máy khách hoàn

thành xong nhiệm vụ rút trích dữ liệu, các thông báo về kết quả công việc, trạng thái, các bất thường sẽ được gửi về máy chủ để phân tích, đánh giá nhằm tạo ra kế hoạch điều phối công việc cho vòng lặp rút trích tiếp theo.

- Giai đoạn 3: Giả lập hành vi của người dùng. Công việc này sẽ được thực hiện bởi các máy khách nhằm mục đích làm cho hành vi rút trích dữ liệu tương đồng với hoạt động và hành vi của khách hàng thông thường.

Nói tóm lại, giải pháp hữu ích này nhằm giả lập hành vi thông thường của khách hàng khi sử dụng trang web thương mại điện tử cho hệ thống rút trích dữ liệu tự động.

Mô tả vắn tắt các hình vẽ

Hình 1: Mô tả quy trình thu thập dữ liệu tự động từ các trang web thương mại điện tử.

Mô tả chi tiết giải pháp hữu ích

Hình 1 mô tả quy trình chi tiết của phương pháp rút trích dữ liệu tự động từ các trang web thương mại điện tử. Cách thức hoạt động của hệ thống trải qua các bước như sau:

- Bước 1.1: Giả lập trình duyệt sử dụng tại các máy khách (mô hình máy khách – máy chủ sẽ được giải thích kỹ hơn ở bước 1.4) để rút trích dữ liệu. Hiện nay, các trang web thương mại điện tử tỏ ra rất thông minh trong việc nhận dạng những yêu cầu bất thường. Để có thể vượt qua được bước kiểm tra này, tất cả các yêu cầu gửi đến trang web đều phải được thực hiện từ các trình duyệt. Theo thống kê của Statcounter Global Stats (https://gs.statcounter.com/), thì đến cuối năm 2015, Chrome chiếm thị phần cao nhất (~52%), kế đến là các trình duyệt IE (~17%), Firefox

(~16%), và Safari (~10%). Do đó, các yêu cầu gửi đến trang web thương mại điện tử phải được giả lập gửi từ các trình duyệt phổ biến này.

- Bước 1.2: Giả lập hệ điều hành. Hiện nay, các truy cập đến từ người dùng bình thường đa phần đến từ các hệ điều hành Windows, MacOS đối với máy tính và Android, iOS đối với điện thoại di động. Tương tự như trình duyệt, hệ điều hành cũng cần được giả lập cho giống với môi trường của người dùng thông thường.
- Bước 1.3: Lựa chọn chiến thuật rút trích dữ liệu. Công việc này được thực hiện bởi máy chủ. Có 3 chiến thuật có thể được lựa chọn để truy cập đến các URL (Uniform Resource Locator – dùng để định vị các tài nguyên trên internet) có trong trang web điện tử: (1) truy cập đến các URL từ các công cụ tìm kiếm như Google, Yahoo hoặc các trang web phổ biến như Facebook; (2) truy cập đến các URL từ danh mục chính từ trang chủ của trang web; và (3) truy cập đến các URL từ các trang hiện hành. Tùy vào tình huống hiện tại, máy chủ sẽ lựa chọn một chiến thuật hợp lý. Thông thường, chiến thuật (1) được sử dụng đầu tiên, sau đó, lần lượt các URL tại trang hiện hành sẽ được truy cập để rút trích thông tin. Các sản phẩm đã được thu thập thông tin sẽ được lưu vào bộ nhớ. Các sản phẩm còn thiếu, sẽ được truy cập cuối cùng nhờ vào chiến thuật (2): hệ thống sẽ đi từ danh mục trang chủ và dò theo từng trang nhằm tìm các sản phẩm còn thiếu để rút trích thông tin.
- Bước 1.4: Điều phối công việc rút trích dữ liệu. Hệ thống thu thập dữ liệu được tổ chức theo mô hình máy chủ - máy khách (một máy chủ có khả năng điều phối một hay nhiều máy khách). Máy chủ có vai trò lưu trữ các thông tin về công việc (danh sách các URL cần được rút trích), trạng thái

của các máy khách, phân tích các thông số trả về từ máy khách,... Máy chủ không thực hiện việc gửi yêu cầu thu thập dữ liệu mà phải thông qua các máy khách. Máy khách có vai trò giả lập các thông số đã được quy định trước bởi máy chủ, thu thập và rút trích dữ liệu cho các URL được chỉ định sẵn. Sau khi nhận được danh sách các URL cần phải rút trích (ví dụ chotot.com, muaban.net), máy chủ sẽ dựa vào trạng thái của các máy khách (có đang thực hiện công việc nào hay đang ở trạng thái chờ, tình trạng tài nguyên trên máy khách như bộ xử lý trung tâm - CPU, bộ nhớ trong - RAM) mà phân phối các URL này để cho các máy khách thực hiện việc rút trích dữ liệu. Việc điều phối đảm bảo tải rút trích các URL được cân bằng giữa các máy khách.

- Bước 1.5: Giả lập hành vi người dùng. Đây là công việc được thực hiện tại máy khách. Các máy khách sau khi nhận được danh sách URL cần phải rút trích sẽ giả lập sao cho giống với hành vi của người dùng đang truy xuất vào những URL đó. Ví dụ như các yêu cầu không thể được gửi liên tục đến trang web thương mại điện tử, mà phải có thời gian chờ. Một vài thông số khác như tần suất gửi yêu cầu đến trang chủ phải phù hợp với kết quả được phân tích, đánh giá ở bước 1.7 (sẽ được trình bày sau) nhằm đảm bảo rằng trang chủ sẽ không phát hiện hành vi bất thường khi rút trích dữ liệu. Ngoài ra, các thao tác của người dùng cũng phải được giả lập một cách đầy đủ như các thao tác nhấp chuột, chạm vào màn hình cảm ứng, kéo màn hình đi lên xuống, điền dữ liệu. Các thao tác này có thể ngẫu nhiên được sinh ra, chèn ngang vào việc gửi các yêu cầu truy xuất URL.

- Bước 1.6: Rút trích thông tin trang web. Sau khi việc giả lập hành vi được thực hiện đầy đủ, các máy khách tiến hành truy cập vào URL, phân tích nội dung HTML được trả về và rút trích các thông tin cần thiết.
- Bước 1.7: Cập nhật thông số, trạng thái của trang web thương mại điện tử. Bước này được thực hiện tại máy chủ. Sau quá trình rút trích dữ liệu tại các máy khách, các thông số về thời gian đáp ứng của trang chủ, số lượng yêu cầu có thể được phục vụ cùng lúc, vị trí của máy khách,... được tổng hợp và gửi cho máy chủ. Máy chủ sẽ phân tích các thông số này, kết hợp với việc xem xét các sản phẩm còn thiếu thông tin, để điều phối công việc cho các máy khách tiếp tục thực hiện việc rút trích dữ liệu cho những vòng lặp sau.
- Bước 1.8: Kiểm tra tín hiệu kết thúc. Bước này được thực hiện tại máy chủ: Sau khi cập nhật các thông số, trạng thái ở bước 1.7, hệ thống sẽ kiểm tra nếu vẫn còn sản phẩm cần được rút trích thì sẽ quay lại bước 1.3, ngược lại sẽ kết thúc ở bước 1.9.
- Bước 1.9: Kết thúc việc rút trích dữ liệu tại trang web.

Cách thức và phương tiện kĩ thuật thực hiện giải pháp hữu ích

- Bước 1.1: Giả lập trình duyệt. Việc giả lập trình duyệt phải đảm bảo sao cho các thông tin cần thiết để gửi một yêu cầu từ máy khách lên máy chủ bằng trình duyệt giả lập giống như thông tin của một trình duyệt thật. Để thực hiện việc này, giải pháp hữu ích kết hợp thư viện Selenium và web driver (www.selenium.dev - công cụ hỗ trợ giả lập nhiều trình duyệt khác nhau bao gồm: Chrome, Firefox, Safari, Opera,…). Giải pháp hữu ích này không sử dụng một trình duyệt giả lập duy nhất mà sử dụng nhiều trình duyệt giả lập trên nhiều máy khác nhau.

- Bước 1.2: Giả lập hệ điều hành. Để tránh việc sử dụng các thông số hệ điều hành giống nhau, mỗi máy khách đều được thiết lập hệ điều hành giả lập khác nhau một cách ngẫu nhiên. Giải pháp hữu ích đã sử dụng BlueStacks để giả lập hệ điều hành Android, VMWare để giả lập các hệ điều hành Windows, Ubuntu và CentOS, và iPadian để giả lập hệ điều hành iOS.
- Bước 1.3: Lựa chọn chiến thuật rút trích dữ liệu. Hiện nay, các trang web đều theo dõi việc truy cập vào trang web của mình từ nguồn nào. Nếu như một máy khách truy cập liên tục vào một trang web từ những nguồn giống nhau, ví dụ như từ trang chủ, từ Facebook,... thì sẽ dễ dàng bị nhận dạng bởi hành vi của mình. Do đó, ở bước 1.7, khi phân tích tham số về nguồn truy cập, các máy khách có thể chọn chiến thuật truy cập từ các công cụ tìm kiếm (cách 1). Ví dụ như để truy cập URL với nguồn từ Google, hệ thống sẽ sử dụng cấu trúc "http://www.google.com/search?=url" – với "url" là đường dẫn cần lấy dữ liệu. Với nguồn từ trang chủ, máy khách đầu tiên sẽ phải truy cập vào trang chủ của trang web, sau đó mới đi theo cấu trúc phân lớp của trang web để truy cập đến nội dung cần lấy (cách 2), hoặc truy cập trực tiếp đường dẫn chứa nội dung cần rút trích (cách 3).
- Bước 1.4: Điều phối công việc. Việc liên lạc giữa máy chủ và máy khách được thực hiện bởi RabbitMQ (www.rabbitmq.com). RabbitMQ là một message broker (truyền tin trung gian) cung cấp phương tiện trung gian để giao tiếp giữa nhiều thành phần (máy chủ và các máy khách) với nhau. Có thể hiểu message broker như là một bưu điện, máy chủ (đóng vai trò là producer – người gửi thông điệp) sẽ gửi thông tin các công việc cần thực hiện đến message broker. Thông tin sẽ đi qua message broker để đến được

máy khách (đóng vai trò là consumer – người nhận thông điệp). Máy chủ và máy khách không liên lạc trực tiếp với nhau mà thông qua message broker.

- Bước 1.5: Giả lập hành vi người dùng. Các thông số thiết lập ban đầu bao gồm thời gian và thao tác sẽ được thực hiện một cách ngẫu nhiên. Cứ sau 2000 lượt yêu cầu, tất cả dữ liệu và thông số về thời gian trả về kết quả từ trang web kể từ lúc gửi yêu cầu, thời gian chờ giữa hai lần gửi yêu cầu, nội dung dữ liệu trả về, cấu hình trình duyệt và hệ điều hành, thao tác nhấp chuột (từ trang chủ, từ các công cụ tìm kiếm, từ các mạng xã hội,…) sẽ được gửi đến máy chủ (bước 1.7) để phân tích và tìm ra thông số phù hợp. Sau khi nhận được thông số từ máy chủ, các máy khách sẽ tiến hành giả lập trình duyệt và hành vi gửi yêu cầu giống như thông tin đã được thiết lập. Ví dụ thời gian để truy cập vào URL sau phải cách thời gian truy cập vào URL trước ít nhất 1 giây, trình duyệt giả lập trên hệ điều hành Linux là Firefox, trình duyệt giả lập trên hệ điều hành Windows là Chrome, thao tác truy cập vào trang nội dung chi tiết phải được thực hiện sau khi có một thao tác nhấp chuột từ trang chủ trước đó,…
- Bước 1.6: Rút trích trang web. Mặc định dữ liệu trả về từ các trang web có định dạng HTML. Để sử dụng các dữ liệu này, cần phải rút trích thành dữ liệu có cấu trúc (có thể lưu dưới dạng bảng), và dữ liệu không có cấu trúc (dữ liệu dưới dạng hình ảnh, văn bản, video). Việc rút trích được thực hiện thông qua hai thư viện chính là Selenium và Jsoup (https://jsoup.org/). Jsoup là một thư viện được sử dụng để phân tích tài liệu HTML. Jsoup cung cấp các phương thức dùng để lấy dữ liệu từ tập tin HTML hoặc URL.

Các dữ liệu được rút trích có thể là giá sản phẩm, ngày tạo sản phẩm, nội dung sản phẩm,…

- Bước 1.7: Phân tích thông số. Các thông số bao gồm thông số hành vi (nên truy cập vào URL chứa dữ liệu trực tiếp hay phải thông qua tìm kiếm google hoặc đi từ trang chủ,…), thông số của hệ điều hành, thông số của trình duyệt, thời gian chờ kể từ lần truy cập trước đó và vị trí của máy khách (địa chỉ IP). Khi nhận được thông tin của các máy khách ở bước 1.5, máy chủ sẽ chia dữ liệu thành hai loại: (1) loại một là các thông số mà có máy khách bị chặn (không lấy được dữ liệu hoặc dữ liệu nhận được có sự bất thường - sự bất thường này có thể là do chủ trang web phát hiện việc thu thập dữ liệu tự động từ các máy khách, thay vì chặn việc thu thập thì trang web lại trả về kết quả giả đã được lập trình sẵn. Do đó cần phải kiểm tra xem dữ liệu trả về có bị giả mạo hay không, cách dễ nhất để thực hiên việc này là thực hiện kiểm tra chéo nội dung rút trích giữa các máy khách để xác nhận nội dung rút trích có trùng nhau hay không); (2) loại hai là các thông số mà máy khách truy cập và lấy dữ liệu bình thường. Sau khi đã phân dữ liệu thành hai loại, máy chủ sẽ phân tích những thông số khác nhau giữa hai loại để tìm ra thông số phù hợp. Các thông số này sẽ phụ thuộc vào từng hệ điều hành, trình duyệt, vị trí của máy khách, hành vi trước đó,… Bộ thông số tối ưu là tập các thông số mà có thời gian nhỏ nhất để truy cập được dữ đồng thời việc truy cập không bị chặn bởi chủ trang web. Mỗi lần cần cập nhật lại thông số, mức thời gian nhỏ nhất cho phép để lấy dữ liệu gấp 1.5 lần thời gian trung bình trên các máy bị phát hiện (bị chặn bởi chủ trang web), đồng thời sẽ có hai bộ thông số được sinh ra: (1) một bộ thông số dùng để tiếp tục lấy dữ liệu – bộ thông số này chính là bộ

thông số trung bình trên các máy không bị phát hiện được cộng trừ với một giá trị ngẫu nhiên đối với dữ liệu số hoặc là giá trị thường xuyên xuất hiện nhất trên các máy không bị phát hiện đối với dữ liệu có định dạng danh mục (ví dụ trên hệ điều hành Linux có thể sử dụng trình duyệt Chrome, Firefox, Opera. Tuy nhiên số lần sử dụng trình duyệt Firefox để gửi yêu cầu thu thập dữ liệu mà không bị phát hiện là nhiều nhất, khi đó các máy khách có sử dụng hệ điều hành Linux sẽ ưu tiên sử dụng trình duyệt Firefox để thu thập dữ liệu); và (2) một bộ thông số dùng để tối ưu thời gian đáp ứng – bằng cách giảm 10% thời gian chờ mỗi lần truy cập để kiểm tra xem các thông số này có bị chặn bởi chủ trang web hay không, nếu không bị chặn sau 2000 lần thử, các thông số này sẽ được sử dụng chính thức trên toàn bộ các máy.

- Bước 1.8: Kiểm tra tín hiệu kết thúc. Nếu số lượng sản phẩm còn lại cần rút trích bằng 0, hệ thống sẽ kết thúc quá trình thu thập ở bước 1.9, ngược lại sẽ quay về bước 1.3.
- Bước 1.9: Kết thúc việc rút trích dữ liệu tại trang web và giải phóng các tài nguyên hệ thống trong quá trình rút trích nếu cần.

Ví dụ thực hiện giải pháp hữu ích

Giải pháp hữu ích này có thể được tích hợp vào các hệ thống rút trích dữ liệu tự động. Cụ thể như, hiện nay, việc rút trích thông tin của các trang thương mại điện tử như amazon.com, lazada.vn, tiki.vn,... đang gặp rất nhiều khó khăn vì các trang web này luôn có các cơ chế giúp phát hiện ra các hành vi rút trích tự động. Giải pháp hữu ích sẽ hỗ trợ cho hệ thống trong việc giả lập các hành vi, nhằm làm cho việc rút trích tự động có thể hòa vào những hành động bình thường của khách hàng trên trang web, do đó sẽ không bị ngăn chặn. Ví dụ, đối với các trang web

thương mại điện tử ở Việt Nam, hệ thống sẽ đi thu thập dữ liệu về hệ điều hành, trình duyệt được sử dụng tại Việt Nam, và giả lập các yêu cầu gửi đến trang chủ từ các trình duyệt, hệ điều hành này. Ngoài ra, hành vi của người dùng cũng được mô phỏng một cách hợp lý. Cụ thể như, đối với một hệ thống rút trích dữ liệu thương mại điện tử thông thường, dữ liệu sẽ được truy xuất theo thứ tự từng sản phẩm một, theo một danh mục nào đó, cho đến khi hết sản phẩm mới thôi để đảm bảo dữ liệu không bị thiếu sót. Tuy nhiên, hành vi này rõ ràng không nhất quán với hành vi duyệt sản phẩm của một khách hàng bình thường, vì người sử dụng rất hiếm khi duyệt hết sản phẩm theo một thứ tự nào đó. Việc rút trích dữ liệu một cách ngây thơ như vậy rất dễ dàng bị phát hiện và ngăn chặn. Do đó, giải pháp hữu ích đề xuất nhiều chiến thuật rút trích dữ liệu, và có một máy chủ sẽ liên tục thu thập, đánh giá dữ liệu về trạng thái trang chủ, để đưa ra quyết định đúng đắn cho việc rút trích dữ liệu. Ngoài ra, xen lẫn giữa những lần gửi yêu cầu đến trang chủ, giải pháp hữu ích cũng mô phỏng các hành vi của người dùng thông thường như nhấp chuột, chạm màn hình, kéo màn hình lên xuống,... Tất cả các phương thức này, làm cho hệ thống rút trích dữ liệu tự động có các hành vi tương tự người dùng bình thường.

Những lợi ích sáng chế đạt được

Việc thực hiện giải pháp hữu ích này đem đến một số lợi ích, hiệu quả như sau:

- Hạn chế được việc phải thay đổi, giả lập các địa chỉ IP khi trang web chủ ngăn chặn việc thu thập dữ liệu tự động. Việc phân chia công việc rút trích dữ liệu cho nhiều máy khách khác nhau làm giảm lưu lượng các yêu cầu gửi đến máy chủ, cũng như lượng dữ liệu phải tải về từ một máy khách, giúp việc thu thập dữ liệu tự động được tiến hành hiệu quả hơn.

- Việc liên tục phân tích các tham số của trang web rút trích như thời gian đáp ứng, số lượng kết nối trung bình, độ trễ,... hỗ trợ cho việc giả lập các thông số môi trường, làm cho hệ thống rút trích dữ liệu có hành vi như người dùng thông thường, do đó việc thu thập dữ liệu tự động sẽ hiệu quả hơn.
- Tối ưu thời gian rút trích dữ liệu khi công việc được phân chia cho nhiều máy khách thực hiện và được quản lý, điều phối tải rút trích một cách hợp lý.
- Hoạt động rút trích dữ liệu tự động được diễn ra liên tục, không bị gián đoạn bởi can thiệp của trang web chủ và qua đó dữ liệu có thể được rút trích một cách đầy đủ, toàn vẹn.

YÊU CẦU BẢO HỘ

1. Phương pháp thu thập dữ liệu tự động từ các trang web thương mại điện tử gồm bảy (7) bước như sau:

(i) giả lập trình duyệt web, việc gửi các yêu cầu đến trang rút trích thông tin phải thông qua các trình duyệt web phổ biến được giả lập;

(ii) giả lập hệ điều hành phổ biến đang được sử dụng;

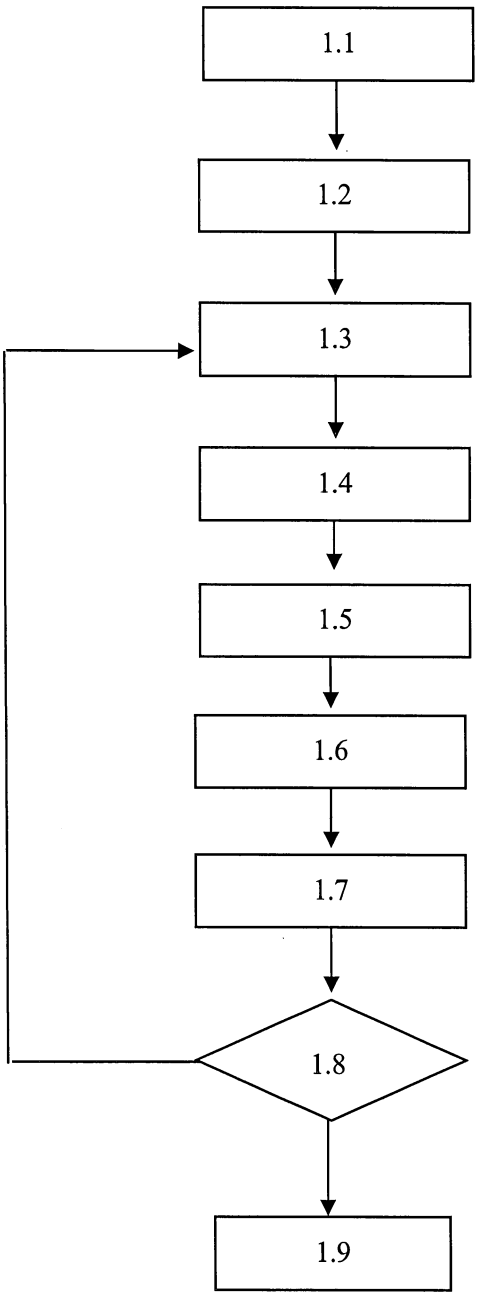
(iii) lựa chọn chiến thuật rút trích dữ liệu phù hợp như rút các trang được dẫn xuất từ công cụ tìm kiếm, từ danh mục chính của trang chủ, hoặc từ các trang hiện hành;

(iv) điều phối công việc rút trích dữ liệu cho các máy khách;

(v) giả lập hành vi người dùng như: nhấp chuột, chạm màn hình, kéo màn hình lên xuống;

(vi) rút trích thông tin trang web;

(vii) cập nhật thông số, trạng thái trang chủ.



Hình 1