TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHÊ THÔNG TIN

BÁO CÁO

ĐỒ ÁN 1 – THU THẬP DỮ LIỆU



Giáo viên hướng dẫn: Lê Ngọc Thành

Sinh viên thực hiện

Lê Tấn Đạt - 19127353

Lê Hoàng Thịnh Phước – 19127518

Lê Tiến Trí - 19127593

*Ngày 12, tháng 11 năm 2021*

1. Phân công công việc

|  |  |  |  |
| --- | --- | --- | --- |
| Tên Thành viên | Công việc | Hoàn thành | Đánh giá(%) |
| Lê Tấn Đạt | . Học và nguyên cứu về selenium  . Tiến hành code về Crawl HTML  . Viết Report | . Hoàn thành code  . Đóng góp vào báo cáo |  |
| Lê Tiến Trí | . Nguyên cứu về API trên SoundCloud  . Tiến hành code về API  . Viết Report | . Hoàn thành code  . Đóng góp vào báo cáo |  |
| Lê Hoàng Thịnh Phước | . Phân tích dữ liệu trên soundClound (hỗ trợ cho phần crawl html)  . Phân tích dữ liệu của HTML và API sau khi lấy được  . Viết Report | . Phân tích dữ liệu  . Đóng góp vào báo cáo |  |

1. Quá trình lấy dữ liệu trên SoundClound
2. API\_Data:
3. *Quy trình lấy dữ liệu*

* Ta khai thác dữ liệu dựa vào API do SoundClound cung cấp, ở đây nhóm sử dụng 3 API để có thể thu thập dữ liệu của *users, tracks, playlists*
* Users: <https://api-v2.soundcloud.com/users/><entity\_id>?client\_id=<client\_id>
* Tracks: <https://api-v2.soundcloud.com/users/><entity\_id>/tracks?client\_id=< client\_id>
* Playlists: <https://api-v2.soundcloud.com/users/><entity\_id>/tracks?client\_id=< client\_id>
* Vấn đề: Trang SoundClound không còn cho đăng kí ClientID, ta chỉ có thể lấy ClientID bằng việc inspect trình duyệt SoundClound và tìm kiếm ClientID ở trong đó

+ Sử dụng selenium để mở google chrome và mở SoundClound để có thể inspect lấy html về. Ta thấy được ClientID được lưu trữ ở thẻ div với id = “g\_id\_intermediate\_iframe” để lấy dãy kí tự string là ClientID.

Graphical user interface, text, application, email

Description automatically generated

* Tiếp theo ta sẽ sử dụng requests các API để có thể lấy dữ liệu về và đưa vào 3 file user.csv, track.csv và playlist.csv

+ Ta sẽ tạo 1 header kèm với requests để có thể tránh lỗi 403 do requests quá nhiều lần. Đồng thời cũng thiết lập thời gian sleep time để chương trình sleep

+ Khi ta lấy được dữ liệu về ta sẽ tiến hành xóa 1 số thuộc tính không cần thiết như:

* user trong track và playlist thay vào đó ta chỉ cần lưu userID
* Xử lí thuộc tính tracks ta sẽ thay thế bằng các ID của cách track đó
* Xử lí dữ liệu lấy được:

+ Xóa các thuộc tính trong file 3 csv nếu dữ liệu thiếu trong đó trên 40%

1. *Đánh giá quá trình chạy trên jupyter notebook*

* *A picture containing text

  Description automatically generated*Hiên tại dữ liệu được khai thác từ API khá lâu mất khoảng 30’ để có thể lấy được dữ liệu trong 3 file csv theo tiêu chí

( 1 ví dụ trong những lần chạy )

1. Parse\_HTML:
2. *Quy trình lấy dữ liệu*

* Bước 1: Thu thập các thể lọai music hiện có trên soundclound (30 thể loại music)
* Bước 2: Tạo ra các link tìm kiếm từ các thể loại nhạc thu thập được để tìm kiếm các playlist liên quan thuộc thể loại nhạc đó. (30 link tìm kiếm)
* Bước 3: Với mỗi link ở B2, lấy khoảng 50 link playlist của thể loại đó và lưu tất cả chúng vào biến playlist\_link có kiểu dữ liệu là set (tránh trùng lặp playlist)
* Bước 4: Với mỗi playlist\_link, lấy ra tên user của người tạo ra playlist và lưu vào biến user\_link có kiểu dữ liệu là set (tránh trùng lặp user)
* Bước 5: Với mỗi playlist\_link, lấy ra link của 2 track đầu tiên và lưu vào biến track\_link có kiểu dữ liệu là set (tránh trùng lặp track)
* Sau khi hoàn thành bước 5 này, chúng ta có 3 biến là playlist\_link ,  user\_link  và user\_link chứa đường link tới các trang cần thu thập dữ liệu.
* Bước 6: Thu thập dữ liệu playlist bằng cách truy cập vào các link trong playlist\_link và lưu vào file
* Bước 7: Thu thập dữ liệu user bằng cách truy cập vào các link trong user\_link và lưu vào file
* Bước 8: Thu thập dữ liệu track bằng cách truy cập vào các link trong track\_link và lưu vào file

1. *Đánh giá quá trình chạy trên notebook*
2. Kết quả lấy được
3. API\_Data:
4. *Số lượng records trong csv*

*Trước khi xử lí dữ liệu*

|  |  |  |
| --- | --- | --- |
| Tên file | Số lương records | Số lượng thuộc tính |
| User.csv | 1500 | 38 |
| Track.csv | 4535 | 63 |
| Playlist.csv | 2093 | 32 |

*Sau khi xử lí dữ liệu*

|  |  |  |
| --- | --- | --- |
| Tên file | Số lương records | Số lượng thuộc tính |
| User.csv | 1500 | 28 |
| Track.csv | 4535 | 37 |
| Playlist.csv | 2093 | 20 |

1. *Mối quan hệ của dữ liệu*

Diagram

Description automatically generated

1. *Đánh giá dữ liệu*

* Vấn đề bias: Dựa vào việc ta lấy dữ liệu của 1500 users đầu tiền đồng thời track và playlist của user đó, thì ta thấy được việc lấy sample như vậy là không bias vì cách lấy dữ liệu và quy trình đó thuộc dạng Systematic Random và đồng thời nó vẫn đảm bảo cho các users, tracks và playlist trong đó có độ ưu tiên là như nhau.

1. Parse HTML
2. *Số lượng records trong csv*

*Trước khi xử lí dữ liệu*

|  |  |  |
| --- | --- | --- |
| Tên file | Số lương records | Số lượng thuộc tính |
| User.csv | 1342 | 39 |
| Track.csv | 2825 | 67 |
| Playlist.csv | 1473 | 33 |

*Sau khi xử lí dữ liệu*

|  |  |  |
| --- | --- | --- |
| Tên file | Số lương records | Số lượng thuộc tính |
| User.csv | 1342 | 32 |
| Track.csv | 2825 | 39 |
| Playlist.csv | 1473 | 24 |

1. *Mối quan hệ của dữ liệu*

Diagram

Description automatically generated

1. *Đánh giá dữ liệu lấy được:*

* Dữ liệu được lấy theo hướng universal ( theo các genre nhạc ) do đó dữ liệu lấy được sẽ có độ đa dạng cao , không bị bias về 1 phía , tất cả các genre nhạc đều đóng vai trò như nhau giúp chúng ta có thể có lượng dữ liệu phong phú và các dữ liệu ấy có độ ưu tiên như nhau
* Dữ liệu có sự liên kết giữa các file user, track và playlist.
* Độ phức tạp của dữ liệu cũng khá lớn, đòi hỏi dữ liệu phải làm sạch, xử lí nhiều hơn nữa để có thể đưa ra được dạng dữ liệu sach nhất có thể để có thể áp dụng cho các quá trình sau

1. Tài liệu tham khảo

* <https://developers.soundcloud.com/docs/api/explorer/open-api>
* <https://stackoverflow.com/questions/54753238/soundcloud-application-registration-form-is-closed>
* <https://nguyennamphong.com/phuong-phap-chon-mau/>
* <https://stackoverflow.com/questions/59387309/multithreading-in-python-selenium>
* <https://docs.python.org/3/library/concurrent.futures.html>
* https://www.toptal.com/python/beginners-guide-to-concurrency-and-parallelism-in-python