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 |  |  |  |
| Lê Tiến Trí |  |  |  |
| Lê Hoàng Thịnh Phước |  |  |  |

1. Mô tả về dữ liệu và 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 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

1. *Mô tả dữ liệu lấy được*

* Mô tả dữ liệu: Nằm trong file jupyter notebook ( phân tích dữ liệu lấy được )
* Mối quan hệ của dữ liệu:

**Diagram

Description automatically generated**

1. Crawl\_Data:

**Quy trình thực hiện bài làm:**

* Bước 1: Thu thập tất cả các thể loại music hiện có trên soundcloud (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, lúc 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

**Một số lưu ý khi chạy file:**

* Đối với bài làm của em mỗi khi chúng ta Restart - Run all thì sẽ cho ra 1 kết quả khác nhau, lí do là vì em làm bài theo hướng universal do đó mỗi khi dùng link để tìm kiếm trên soundcloud thì sẽ thu lại được kết quả khác nhau.
* Những dòng markdown em ghi bên dưới có số liệu kèm theo có thể sẽ khác với lúc thầy chạy bài.
* Vì để chạy hết các cells thì thời gian thực hiện sẽ khá lâu vì vậy để đỡ mất thời gian, em có một thư mục link gồm các link của playlist, user và track em thu thập được trong 1 lần chạy bài từ đầu, thầy có thể sử dụng các file này để đỡ tốn thời gian nếu muốn ạ.

**Dữ liệu thu thập được lưu trữ:**

* **Mối quan hệ:**

**Diagram

Description automatically generated**

* **Thư mục** Crawl\_data**:**
  + **playlist.csv**
  + **user.csv**
  + **track.csv**

1. Kết quả lấy được
2. API\_Data:
3. *Số lượng records trong csv*

|  |  |  |
| --- | --- | --- |
| Tên file | Số lương records | Số lượng thuộc tính |
| User.csv | 1500 | 38 |
| Track.csv | 2825 | 63 |
| Playlist.csv | 2093 | 32 |

1. *Đánh giá dữ liệu*

* Vấn đề bias:
* Dựa vào việc ta lấy dữ liệu của 1500 user đầu tiên đồng thời track và playlist của họ, ta thấy được việc lấy sample như vậy là không bias vì cách lấy dữ liệu đó thuộc dạng Systematic Random và đảm bảo cách user trong đó có độ ưu tiên là như nhau
* Mẫu có những thông tin về thống kê:

Các trường dữ liệu về playback\_count, like\_count,repost\_count,comment\_count

1. Crawl Data:

*a. Số lượng records trong csv:*

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

*Bảng 2.1*

*b. Số lượng records sau khi phân tích và xử lí dữ liệu thu thập được:*

|  |  |  |
| --- | --- | --- |
| Tập dữ liệu | Số lượng records | Số lượng thuộc tính |
| User | 1342 | 32 |
| Track | 2825 | 39 |
| Playlist | 1473 | 24 |

*Bảng 2.2*

*c. Đánh giá dữ liệu lấy được:*

***Đánh giá về cách làm bài:***

- Ở cách làm này của nhóm em, dữ liệu được lấy theo hướng universal (theo genre nhạc) do đó dữ liệu lấy được sẽ có độ đa dạng cao, không bị bias về 1 phía nào nếu như chúng ta thu thập theo 1 thể loại nhạc nhất định, tất cả các genre nhạc đều đóng vai trò như nhau giúp chúng ta có lượng dữ liệu phong phú, không bị nhàm chán.

- Tiếp theo, để đảm bảo sự có sự liên kết giữa các file user, track và playlist, cách làm của nhóm em đảm bảo tốt được tiêu chí này, vì từ những playlist lấy được, chúng em sẽ thu thập những user là chủ của những playlist đó, sau đó với mỗi playlist lấy được chúng em sẽ chọn 2 bài hát đầu tiên để thêm vào file track. Đều này đảm bảo được sự liên kết giữa 3 file dữ liệu với nhau.

***Đánh giá về dữ liệu:***

- Đa dạng và phong phú, có độ phủ lớn và không bị bias về một phía.

- Tuy số cột thuộc tính là rất nhiều nhưng cũng có không ít những cột thuộc tính không có đóng góp nhiều vào dữ liệu. Sự thay đổi số cột được biểu thị trong hai bảng 2.1 và 2.2.

- Độ phức tạp của dữ liệu là rất lớn, đòi hỏi phải làm sạch, xử lí dữ liệu nhiều hơn nữa nếu muốn thực hiện các phương pháp phân tích dữ liệu hiệu quả.

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