**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**ĐẠI HỌC KINH TẾ TP HỒ CHÍ MINH**

**TRƯỜNG CÔNG NGHỆ VÀ THIẾT KẾ**



**ĐỒ ÁN MÔN HỌC**

**Học phần: Dữ liệu lớn và Ứng dụng**

**ĐỀ TÀI**

**XÂY DỰNG HỆ THỐNG GỢI Ý SẢN PHẨM**

**Nhóm Sinh Viên:**

* Đoàn Vũ Minh Thanh – 31201020910
* Đặng Thị Cẩm Tú – 31201024511

**Chuyên Ngành:** KHOA HỌC DỮ LIỆU

**Khóa: K46**

***TP Hồ Chí Minh, ngày 04 tháng 04 năm 2022***

# MỤC LỤC NỘI DUNG

[MỤC LỤC NỘI DUNG 1](#_Toc131547893)

[I. Giới thiệu đề tài 3](#_Toc131547894)

[II. Cơ sở lý thuyết 5](#_Toc131547895)

[1. Hệ thống gợi ý: [2] 5](#_Toc131547896)

[2. Ngôn ngữ lập trình Python: 6](#_Toc131547897)

[3. Gói thư viện Scikit-Learn: 6](#_Toc131547898)

[4. Ma trận User-Item: 7](#_Toc131547899)

[5. Nearest Neighbors Search: 7](#_Toc131547900)

[6. Độ tương tự Cosine (cosine similarity): 7](#_Toc131547901)

[7. Thuật toán Collaborative Filtering: 8](#_Toc131547902)

[8. Content-based Filtering: [8] 8](#_Toc131547903)

[9. DistilBERT: [9] 8](#_Toc131547904)

[10. Các chỉ số đánh giá mô hình: 9](#_Toc131547905)

[a. Mean Reciprocal Rank (MRR): [10] 10](#_Toc131547906)

[b. Normalized Discounted Cumulative Gain (NDCG): [10] 10](#_Toc131547907)

[c. Precision-Recall Curve: [11] 11](#_Toc131547908)

[III. Đề xuất mô hình: 14](#_Toc131547909)

[1. Mô tả bộ dữ liệu: 14](#_Toc131547910)

[2. Đề xuất mô hình: 15](#_Toc131547911)

[IV. Triển khai và đánh giá mô hình: 17](#_Toc131547912)

[1. Quy trình xây dựng hệ thống gợi ý sản phẩm: 17](#_Toc131547913)

[a. Thiết kế môi trường, cài đặt thư viện: 17](#_Toc131547914)

[b. Nhập dữ liệu 17](#_Toc131547915)

[c. Tiền xử lý dữ liệu: 20](#_Toc131547916)

[d. Xây dựng hệ thống gợi ý: 22](#_Toc131547917)

[2. Phân tích kết quả đánh giá: 30](#_Toc131547918)

[V. Kết luận 32](#_Toc131547919)

[TÀI LIỆU THAM KHẢO 33](#_Toc131547920)

MỤC LỤC HÌNH ẢNH

Hình 1. Công thức tính độ tương tự Cosine 8

Hình 2. Mô hình BERT 9

Hình 3. Công thức tính MRR 10

Hình 4. Công thức tính CG 10

Hình 5. Công thức tính DCG 10

Hình 6. Công thức tính nDCG 11

Hình 7. Công thức tính IDCG 11

Hình 8. Công thức tính precision 11

Hình 9. Công thức tính recall 12

Hình 10. Precision-Recall Curve 13

Hình 11. Công thức tính F1-score 13

Hình 12. Mô hình đề xuất 16

Hình 13. Tạo dataframe tên product\_list tập hợp danh sách mã sku các sản phẩm được sắp xếp theo thứ tự tăng dần và số thứ tự (index) của từng sản phẩm. 21

Hình 14. Mã hóa mã sku sản phẩm 21

Hình 15. Tạo dataframe tên cust tập hợp danh sách ID khách hàng được sắp xếp theo thứ tự tăng dần và số thứ tự (index) của từng ID 22

Hình 16. Mã hóa ID khách hàng 22

Hình 17. Lập ma trận User-Item 23

Hình 18. Sử dụng hàm NearestNeighbors 24

Hình 19. Hàm item\_viewed() và hàm rec\_list() 24

Hình 20. Hàm ask() và hàm print\_list() 25

Hình 21. Kiểm tra hệ thống có thành công không 25

Hình 22. Kết quả thử nghiệm với người dùng có số thứ tự là 24 26

# Giới thiệu đề tài

Sau hơn hai năm chống chọi đại dịch COVID-19, thế giới đã chứng kiến nhiều sự thay đổi lớn, trong đó nổi bật là hành vi, thói quen mua sắm của người tiêu dùng. Phương thức mua sắm trực tuyến qua các trang thương mại điện tử trở nên phổ biến, dần thay thế thói quen mua sắm truyền thống trực tiếp tại cửa hàng. Theo Sách trắng Thương mại điện tử Việt Nam 2022 do Bộ Công thương - Cục Thương mại điện tử và Kinh tế số biên soạn và phát hành [1], cuối năm 2022, tại Việt Nam có khoảng 57 đến 60 triệu người tham gia mua sắm trực tuyến trên các trang thương mại điện tử. Tuy nhiên, khác với hình thức mua sắm truyền thống trực tiếp tại cửa hàng, người tiêu dùng khi mua sắm trực tuyến dành thời gian tham khảo một sản phẩm rất ngắn, chỉ tầm 44 giây đến 1 phút 22 giây. Lý giải cho việc dành ít thời gian tìm hiểu sản phẩm của khách hàng, Sách trắng chỉ ra rằng, các yếu tố khách hàng ưu tiên khi tìm hiểu một sản phẩm trên sàn thương mại điện tử là giá cả, thương hiệu, lượt mua / điểm đánh giá trung bình. Những thông tin này thường xuất hiện ngay trong kết quả tìm kiếm, nên khách hàng có thể dễ dàng đọc và đưa ra quyết định chọn sản phẩm.

Tuy nhiên, nhằm giúp khách hàng thuận tiện hơn trong quá trình tìm kiếm vào làm đẩy giỏ hàng trực tuyến của họ, các trang thương mại điện tử đã phát triển các thuật toán gợi ý sản phẩm. Ví dụ, khi bạn xem xét một đôi giày thể thao Adidas trên Shopee, khi lướt xuống phần “Các sản phẩm cùng thương hiệu” hoặc phần “Có thể bạn cũng thích”, bạn sẽ thấy các mẫu giày thể thao của nhiều dòng khác, nhãn hiệu khác, các phụ kiện khác như vớ, quần áo,... được đề xuất với bạn. Vậy dựa vào đâu để Shopee có thể đưa ra được sản phẩm gợi ý đó? Đó là nhờ vào hệ thống gợi ý (recommendation system) mà Shopee đã xây dựng để khai thác các dữ liệu trong quá khứ từ các hoạt động diễn ra trên trang web của họ.

Trong bài báo cáo này, nhóm sẽ nghiên cứu ý tưởng của hai phương pháp gợi ý, đó là Collaborative Filtering và Content-based Filtering. Sau đó, nhóm sẽ tiến hành xây dựng hệ thống gợi ý bằng hai phương pháp trên một bộ dữ liệu thực tế từ một trang thương mại điện tử, thành lập danh sách gợi ý sản phẩm, và phân tích kết quả gợi ý từ hai phương pháp trên. Cụ thể, cấu trúc bản báo cáo như sau:

* Chương 1: Giới thiệu đề tài
* Chương 2: Cơ sở lý thuyết - các lý thuyết liên quan đến hai thuật toán để xây dựng hệ thống gợi ý sản phẩm
* Chương 3: Kết quả thực nghiệm - mô tả các bước tiến hành xây dựng hệ thống gợi ý, xử lý trên bộ dữ liệu mà chúng tôi đã chọn, và phân tích kết quả nhận được.
* Chương 4: Kết luận - nhận xét về độ hiệu quả mà hệ thống gợi ý chúng tôi xây dựng cũng như đề xuất thêm các hướng phát triển từ kết quả thực nghiệm.

# Đề xuất mô hình:

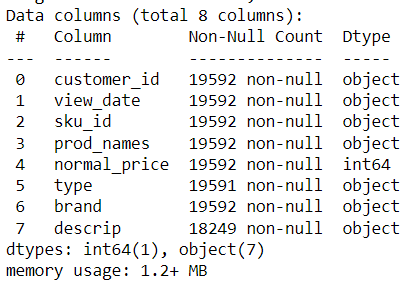
## Mô tả bộ dữ liệu:

Nhóm thực hiện một nghiên cứu về tương tác của người dùng trên trang bán lẻ trực tuyến Tops Market - một thuộc tập đoàn bán lẻ Central Retail.

Dữ liệu được ghi nhận được ghi nhận từ ngày 1/6/2022 - 30/6/2022, bao gồm thông tin người dùng tương tác tác với sản phẩm, trong trường hợp này nhóm sẽ chỉ lấy những tương tác có click vào sản phẩm để xem.

Với mỗi sản phẩm trong bộ dữ liệu sẽ cung cấp thông tin chi tiết về sản phẩm đó, phân loại sản phẩm và thương hiệu của sản phẩm đó là gì. Những thông tin này sẽ giúp nhóm nhiểu rõ hơn về đặc điểm của sản phẩm để có thể đưa ra nhưng gợi ý chính xác hơn cho người dùng.

Về bộ dữ liệu, gồm có 8 giá trị:

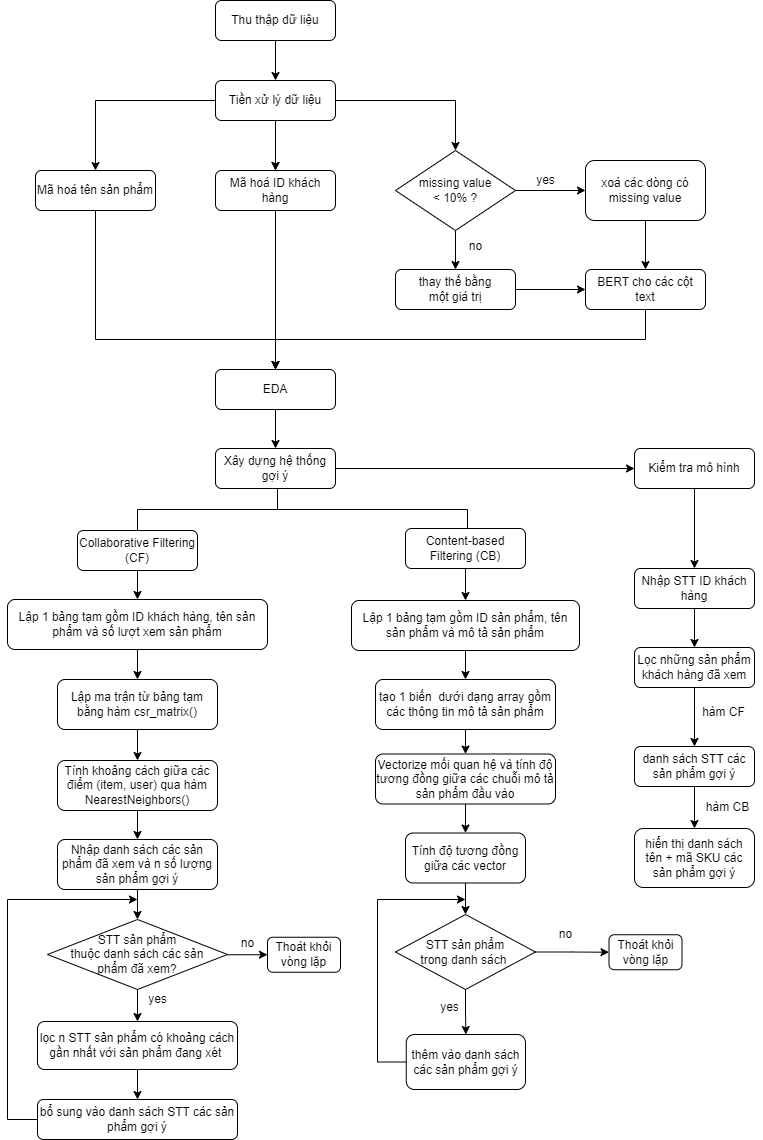


Trong đó:

* Customer\_id: mã định danh khách hàng
* view \_date: ngày xem sản phẩm đó của customer
* Sku\_id: mã sản phẩm
* Prod\_names: tên của các sản phẩm
* Normal\_price: giá sản phẩm
* Type: phân loại nhóm sản phẩm
* Descrip: mô tả sản phẩm

Những thông tin này sẽ giúp nhóm có 1 cái nhìn tổng thể về tập khách hàng và giúp đưa ra những giải pháp phù hợp để cải thiện trải nghiệm người dùng và tăng doanh số bán hàng.

## Đề xuất mô hình:

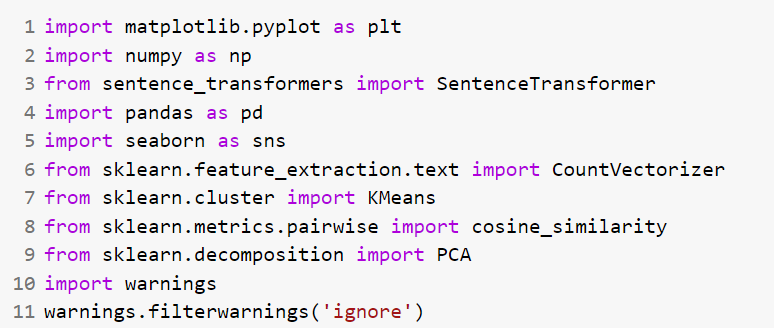


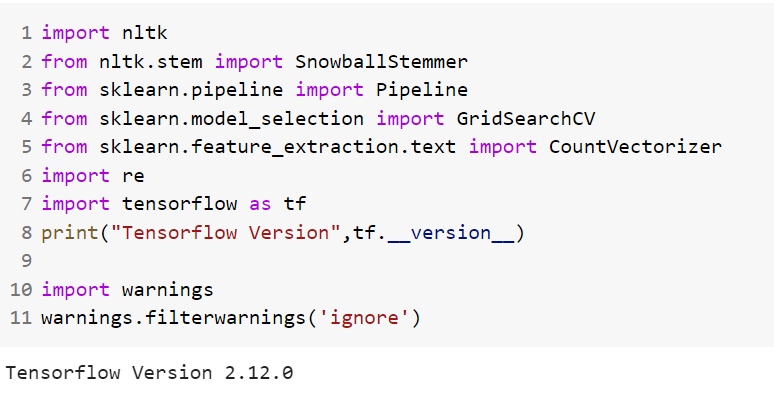
Hình 12. Mô hình đề xuất

# Triển khai và đánh giá mô hình:

## Quy trình xây dựng hệ thống gợi ý sản phẩm:

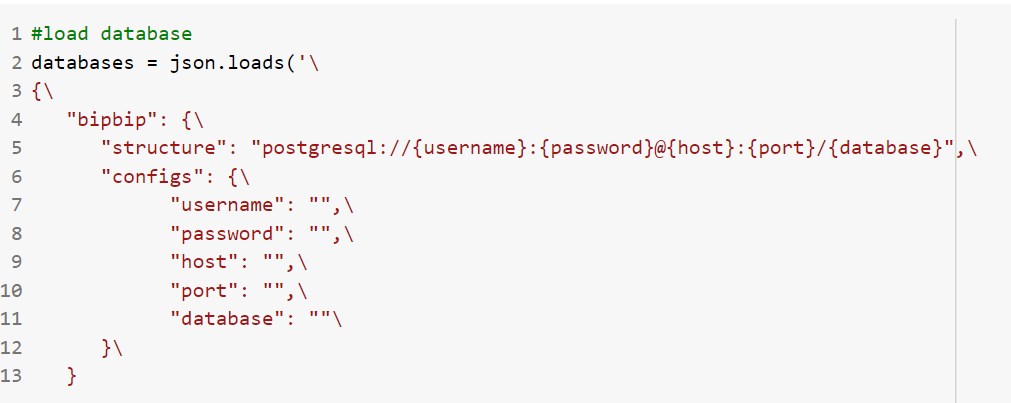
### Thiết kế môi trường, cài đặt thư viện:



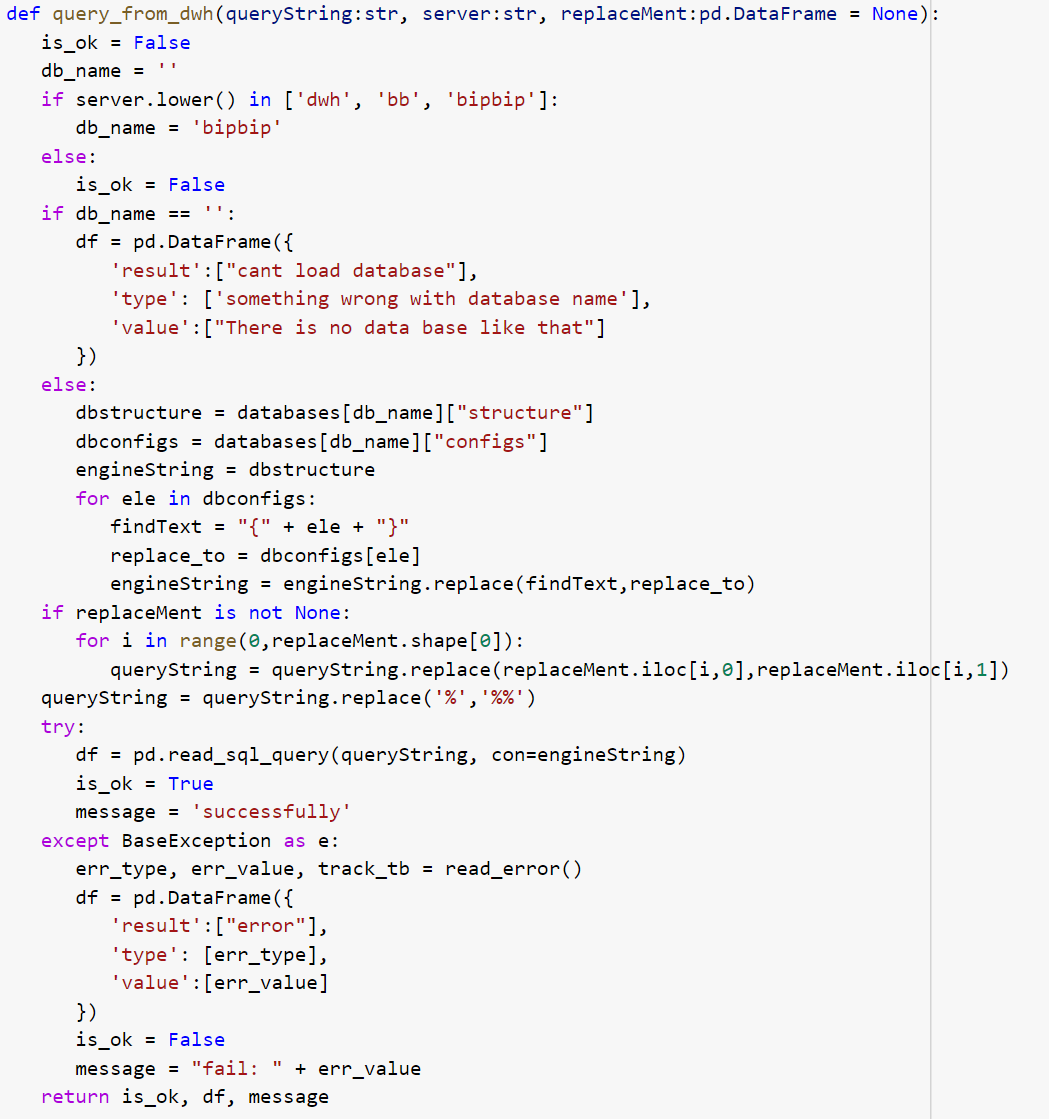


### Nhập dữ liệu

Bộ dữ liệu được lấy trực tiếp từ hệ thống data warehouse dwh của doanh nghiệp thông qua cổng SQLAlchemy, và các credential connection string để có thể đọc, ghi và lấy dữ liệu và tài khoản được cấp quyền dùng để truy cập vào



Nhóm thực hiện gọi một function để có thể query dữ liệu trực tiếp trên python mà không cần phải tải dữ liệu đã được query trên database rồi upload lên notebook để dùng.



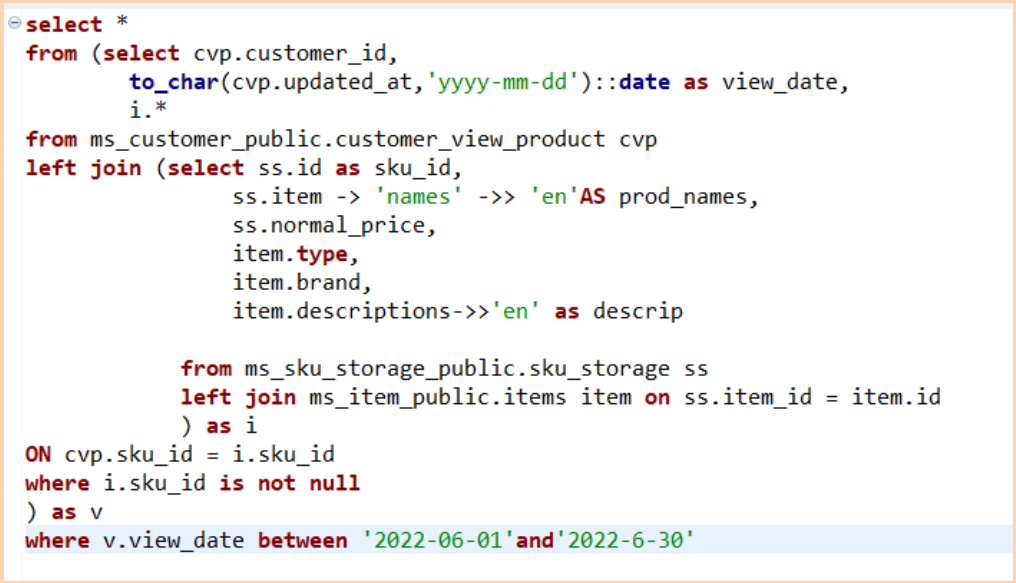
Với instruction cho function trên:

1. Cung cấp một đoạn truy vấn SQL hợp lệ trong tham số `queryString`
2. Chỉ định tên máy chủ dwh để kết nối trong tham số `server`. Các máy chỉ được hỗ trợ là ‘dwh’,’bb’,’bipbip’
3. (Nếu cần) Cung cấp một DataFrame chứa các cặp chuỗi để thay thế trong chuỗi truy vấn trong tham số `replaceMent`

Nó có thể được sử dụng để thay thế các substitute value và chuỗi truy vấn

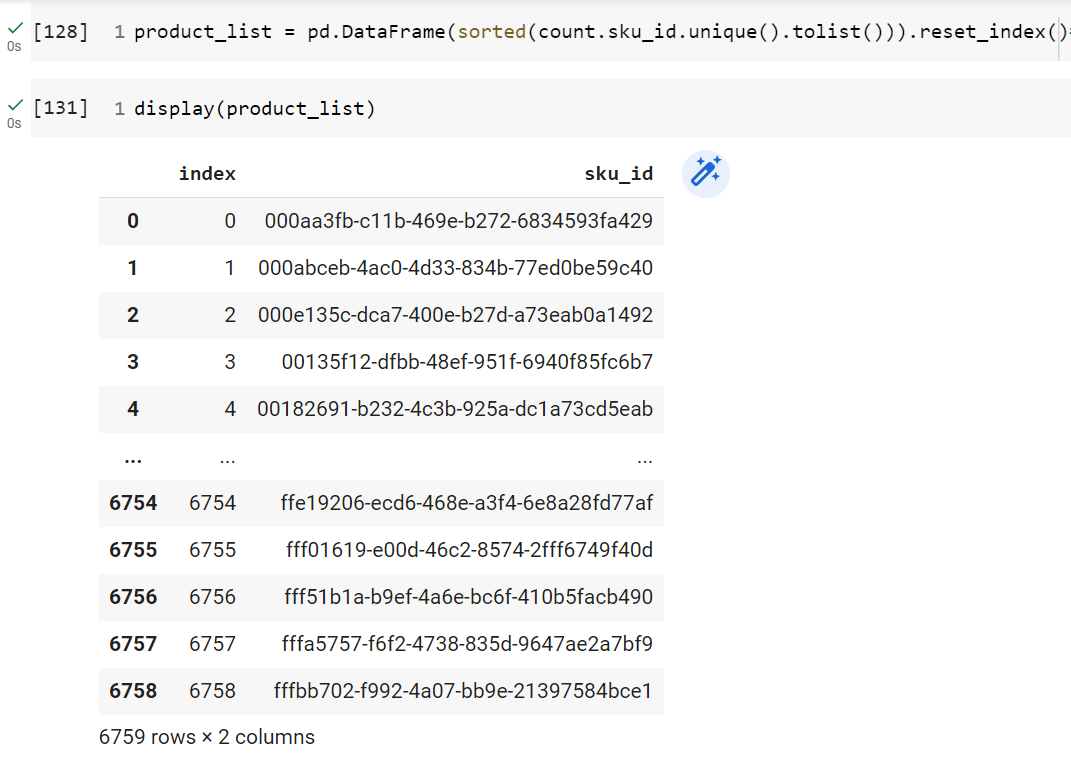
1. Thực thi hàm và kiểm tra các giá trị trả về. Nếu `is\_ok` = True, truy vấn được thực hiện thành công và kết quả truy vấn sẽ được chứa trong Dataframe. Ngược lại, `is\_ok`= False, thông báo lỗi và thông báo này được chứa trong tham số `message`

Và dưới đây là câu truy vấn viết bằng PostgreSQL để lấy dữ liệu



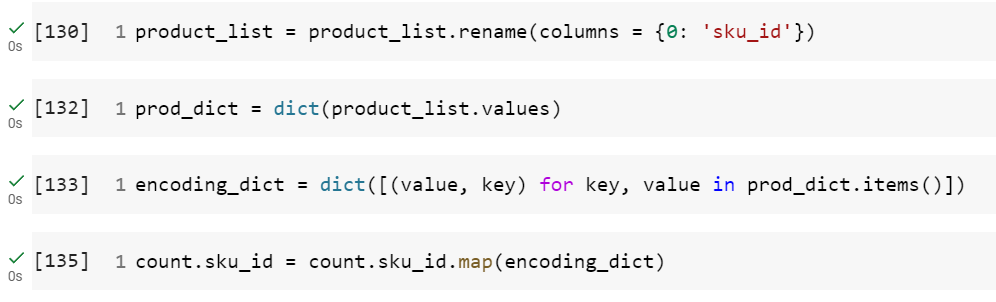
### Tiền xử lý dữ liệu:

Vì cấu trúc của ID của khách hàng và mã lưu kho của sản phẩm (sku\_id) phức tạp, là sự kết hợp của số và kí tự nên để thuận tiện trong việc xử lý dữ liệu khi chạy các thuật toán gợi ý, nhóm sẽ mã hoá 2 biến này. Đầu tiên, nhóm sẽ lập 1 bảng chứa thông tin cần mã hoá, sắp xếp theo chiều tăng dần và đánh STT, sau đó nhóm sẽ thay thế giá trị hiện tại trong cột cần mã hoá bằng STT đã đánh dấu trong bảng thông tin.



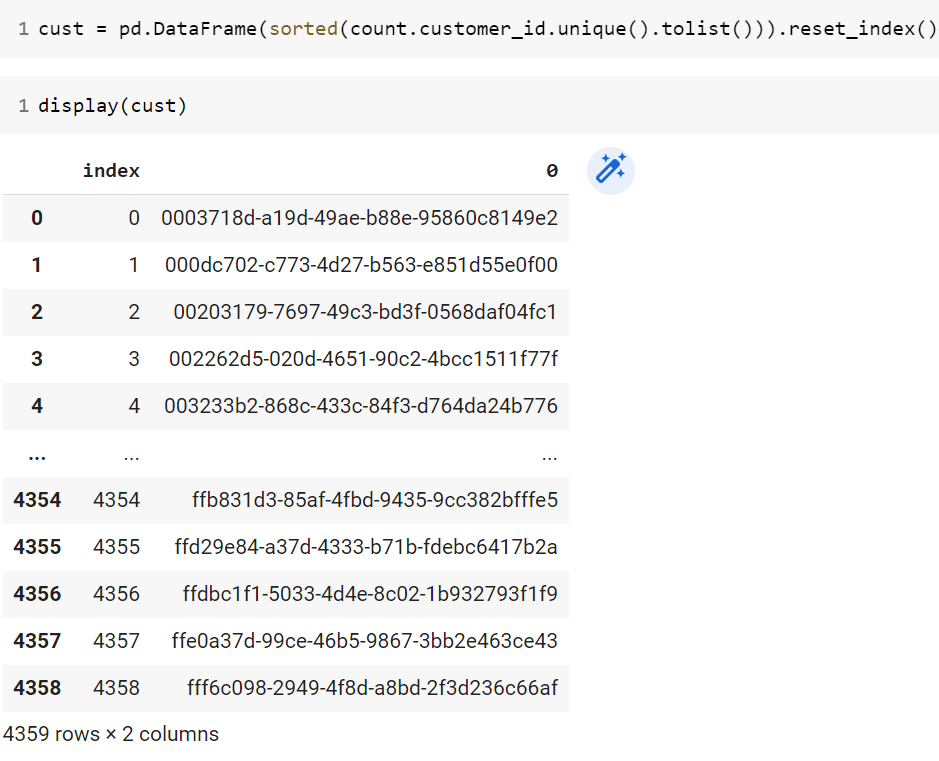
Hình 13. Tạo dataframe tên product\_list tập hợp danh sách mã sku các sản phẩm được sắp xếp theo thứ tự tăng dần và số thứ tự (index) của từng sản phẩm.

Sau khi sửa tên cột tên sản phẩm trong bảng product\_list, nhóm lập 1 từ điển (dictionary) với khóa (key) là số thứ tự của sản phẩm và giá trị (value) là mã sku sản phẩm tương ứng. Tuy nhiên, vì mục tiêu của nhóm là mã hóa từ mã sku sang số thứ tự nên nhóm phải hoán đổi khóa và giá trị trong từ điển. Cuối cùng, nhóm thực hiện mã hóa cột mã sku sản phẩm sku\_id qua hàm map() với tham số là từ điển mã hóa encoding\_dict.

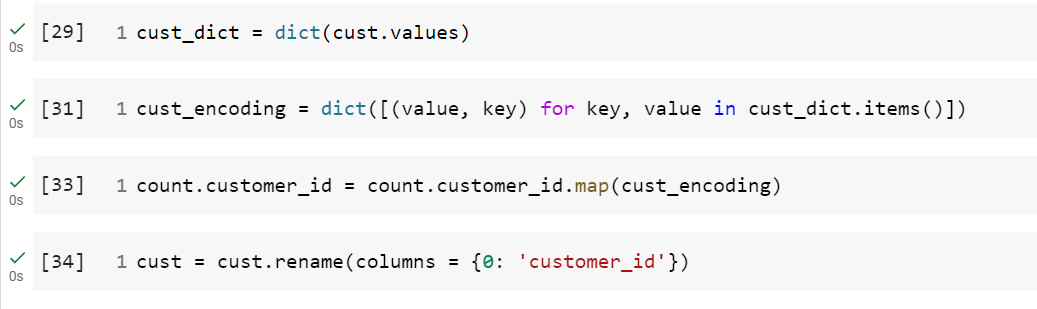


Hình 14. Mã hóa mã sku sản phẩm

Quy trình mã hóa ID khách hàng cũng diễn ra tương tự



Hình 15. Tạo dataframe tên cust tập hợp danh sách ID khách hàng được sắp xếp theo thứ tự tăng dần và số thứ tự (index) của từng ID

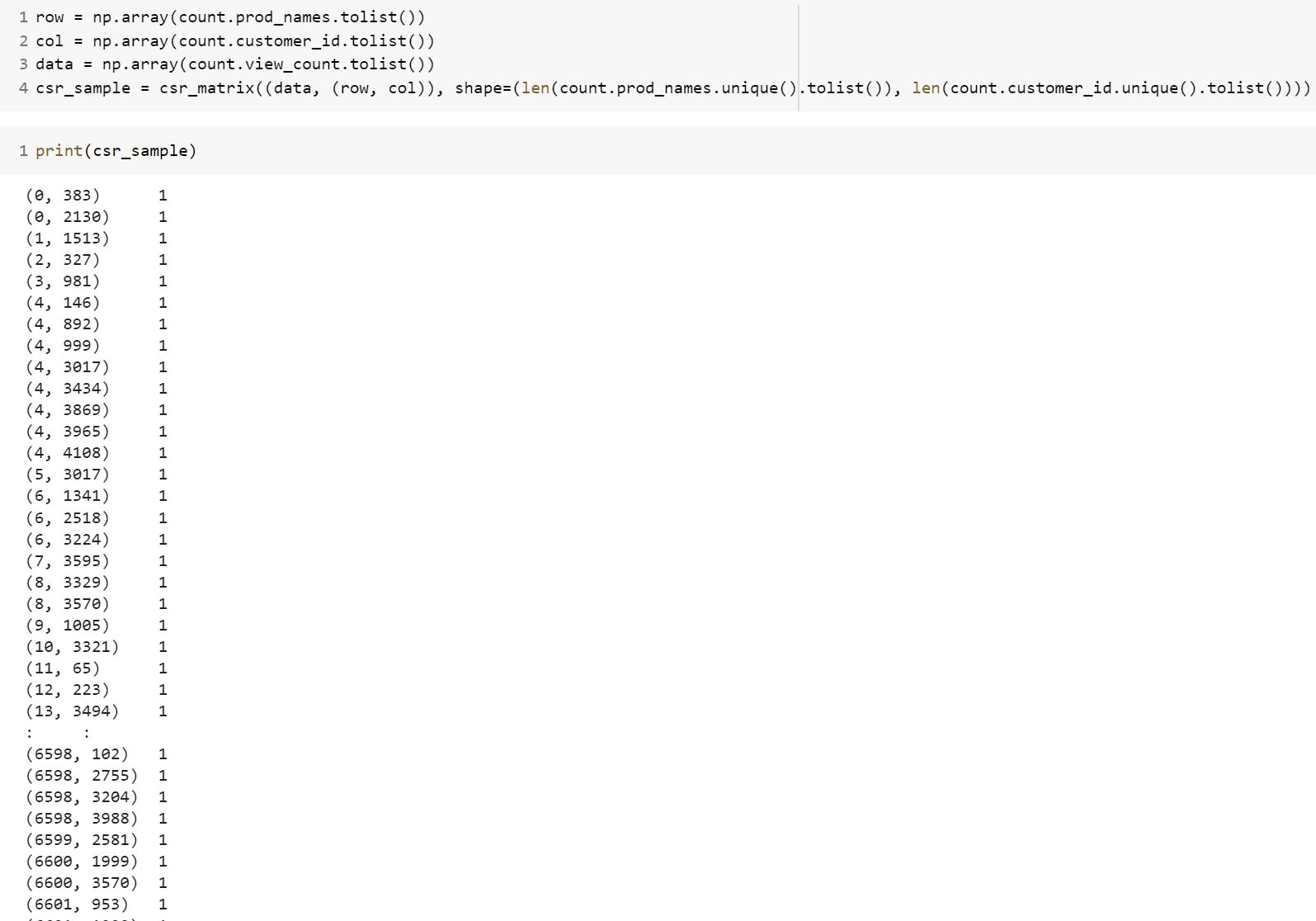


Hình 16. Mã hóa ID khách hàng

### Xây dựng hệ thống gợi ý:

#### Collaborative Filtering

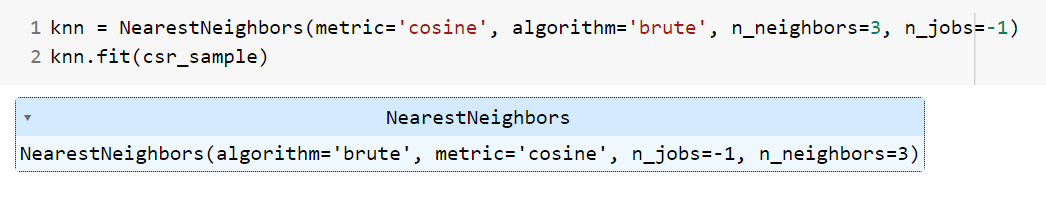
Bước đầu tiên khi xây dựng thuật toán gợi ý theo phương pháp Collaborative Filtering là thành lập ma trận User-Item. Sau khi lập bảng dữ liệu dataframe count gồm ba biến: (1) mã khách hàng - customer\_id, (2) mã sku sản phẩm (đã mã hóa) - sku\_id, (3) lượt xem sản phẩm của khách hàng - view\_count, nhóm sử dụng hàm csr\_matrix() để chuyển ba thành phần dữ liệu trên thành ma trận với danh sách tên cột là mã khách hàng, danh sách tên dòng là mã sku sản phẩm, và dữ liệu trong ma trận là lượt xem sản phẩm tương ứng với mỗi cặp (mã sku sản phẩm - mã khách hàng).



Hình 17. Lập ma trận User-Item

Nhóm sử dụng hàm NearestNeighbors để tính khoảng cách giữa các điểm (mã sku sản phẩm - mã khách hàng) trong ma trận crs\_sample. Nhóm điều chỉnh hai tham số trong hàm là metric và algorithm:

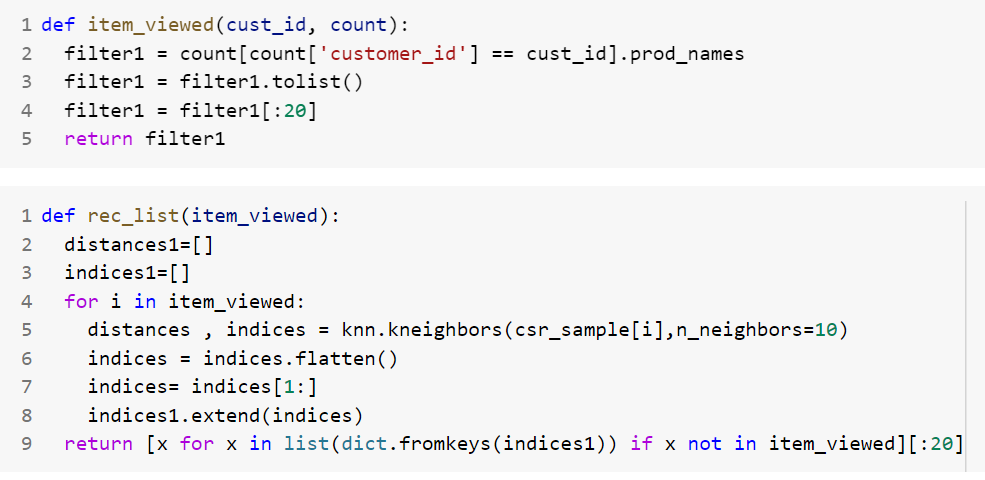
* metric = cosine: chỉ số đo khoảng cách là độ tương tự Cosine
* algorithm = brute: phương pháp tìm điểm lân cận gần nhất là vét cạn



Hình 18. Sử dụng hàm NearestNeighbors

Nhóm xây dựng hàm item\_viewed() với hai tham số đầu vào là mã khách hàng và danh sách chứa lịch sử xem sản phẩm của khách hàng. Kết quả của hàm sẽ trả về danh sách mã sku của các sản phẩm đã xem của khách hàng đang xem xét.

Hàm rec\_list() với tham số đầu vào là kết quả đầu ra của hàm item\_viewed() thực hiện nhiệm vụ: với mỗi mã sku của sản phẩm, hàm kneighbors sẽ tìm ra tối đa 10 sản phẩm có độ tương tự cao nhất với sản phẩm đang xét. Tuy nhiên, trong danh sách trả về sẽ có chứa chính sản phẩm đang xét, thế nên để loại bỏ sự trùng lặp này, hàm trả về sẽ lấy từ phần tử thứ 2 trong danh sách mà kneighbors trả về. Kết quả cuối cùng của hàm rec\_list() là 20 mã sku\_id đầu tiên.



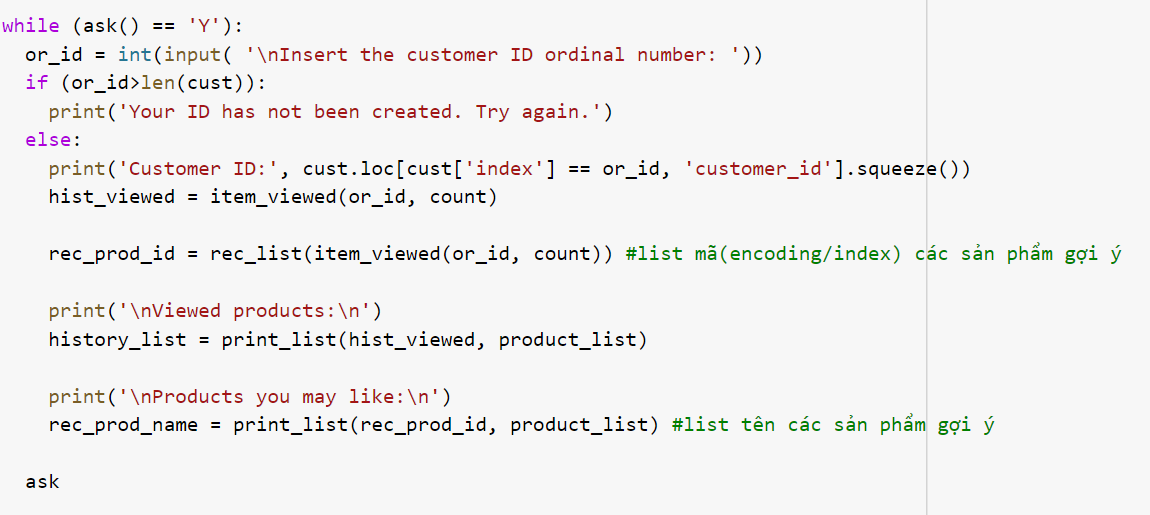
Hình 19. Hàm item\_viewed() và hàm rec\_list()

Nhóm xây dựng hàm hỏi ask() để thu thập yêu cầu đưa danh sách gợi ý của người dùng. Hàm in danh sách print\_list() thực hiện nhiệm vụ in tên sản phẩm từ danh sách đầu vào mã sku.



Hình 20. Hàm ask() và hàm print\_list()

Để kiểm tra thuật toán có thành công hay không, nhóm thực hiện vòng lặp while() với tham số đầu vào là yêu cầu của người dùng qua hàm ask(). Khi người dùng đồng ý nhận danh sách gợi ý (Y - yes), hệ thống ghi nhận số thứ tự ID của người dùng. Nếu số thứ tự nhập vào lớn hơn số lượng khách hàng hiện tại của hệ thống, có nghĩa ID của khách hàng này không tồn tại, và hệ thống sẽ hiển thị thông báo và yêu cầu khách hàng nhập lại số thứ tự ID. Ngược lại, hệ thống sẽ hiển thị ID đầy đủ của khách hàng, sau đó sử dụng hàm item\_viewed() để thu thập các sản phẩm mà khách hàng hiện tại đã từng xem qua. Hàm rec\_list sau đó sẽ hiển thị các sản phẩm gợi ý. Khi đã in xong danh sách gợi ý, hệ thống sẽ hỏi lại người dùng có muốn nhận danh sách gợi ý sản phẩm không, nếu người dùng không còn nhu cầu thì hệ thống sẽ ngừng vòng lặp.



Hình 21. Kiểm tra hệ thống có thành công không

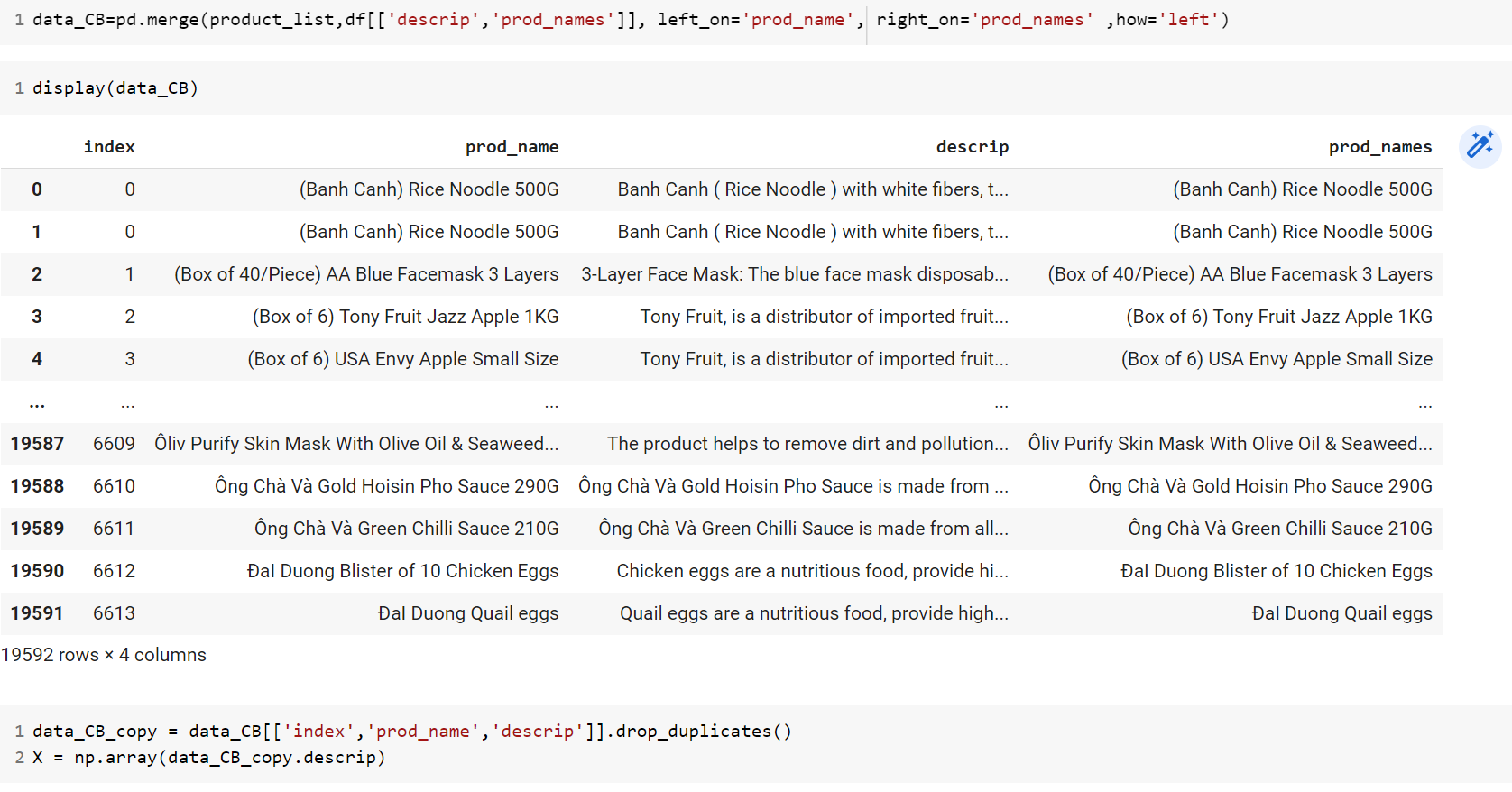


Hình 22. Kết quả thử nghiệm với người dùng có số thứ tự là 24

#### Content-based Filtering

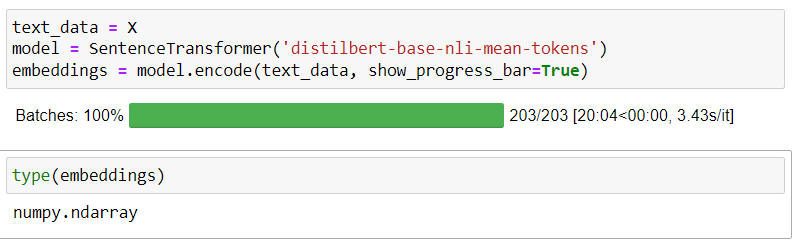
Đầu tiên nhóm thực hiện lập bảng dữ liệu dataframe count gồm ba biến: (1) tên sản phẩm - prod\_name, (2) mã sản phẩm (đã mã hóa) - index, (3) mô tả sản phẩm - descrip, trong đó biến descrip được lấy từ dataframe gốc merge vào datafarme product\_list với key connect chung giữa 2 bảng bằng tên sản phẩm.

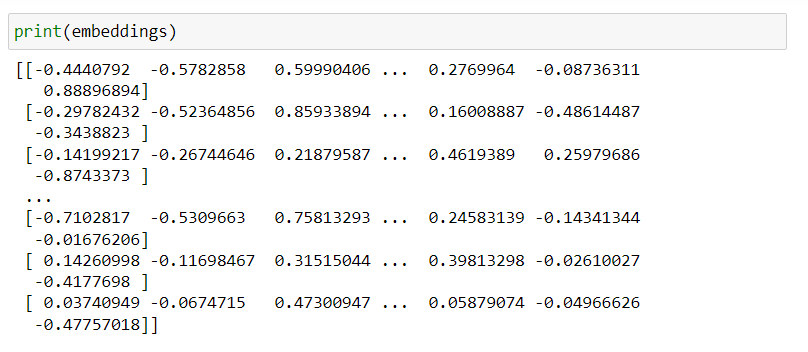
Sau khi đã merge 2 bảng lại với nhau, nhóm thực hiện một số tiền xử lý cơ bản: loại bỏ những giá trị bị trùng lắp và những giá trị bị null có trong biên descrip





Tiếp đó nhóm khi xây dựng thuật toán gợi theo phương pháp Content-based Filtering dựa trên thông tin sản phẩm để đưa ra gợi ý là tạo ra các vector biểu diễn cho các mô tả sản phẩm bằng mô hình DistilBert. 

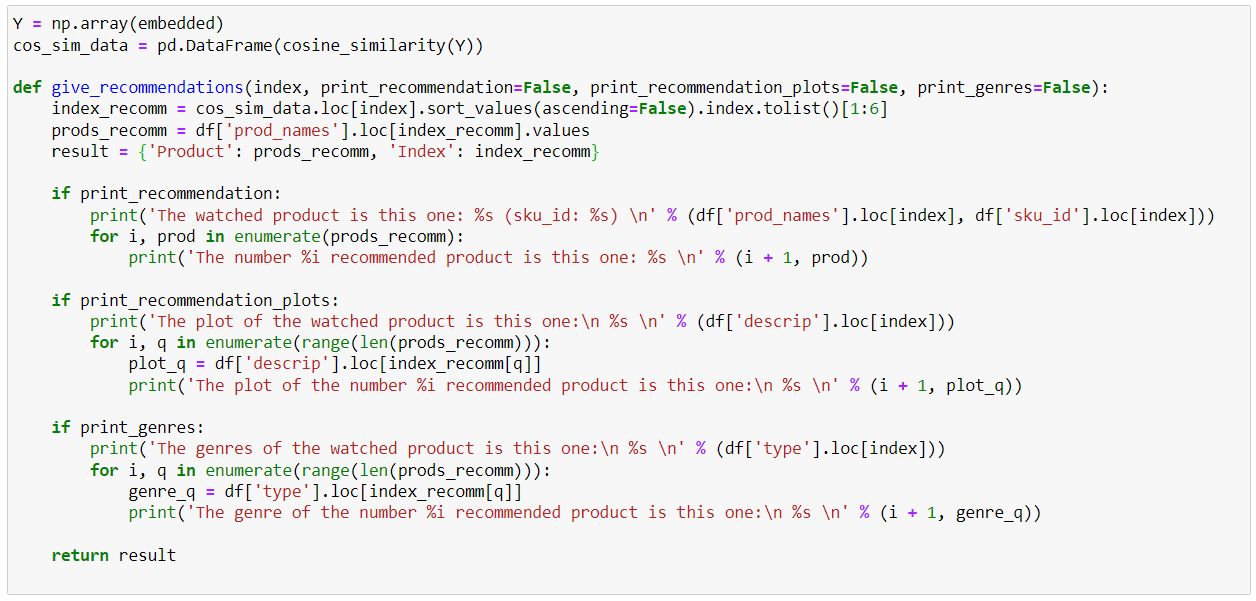


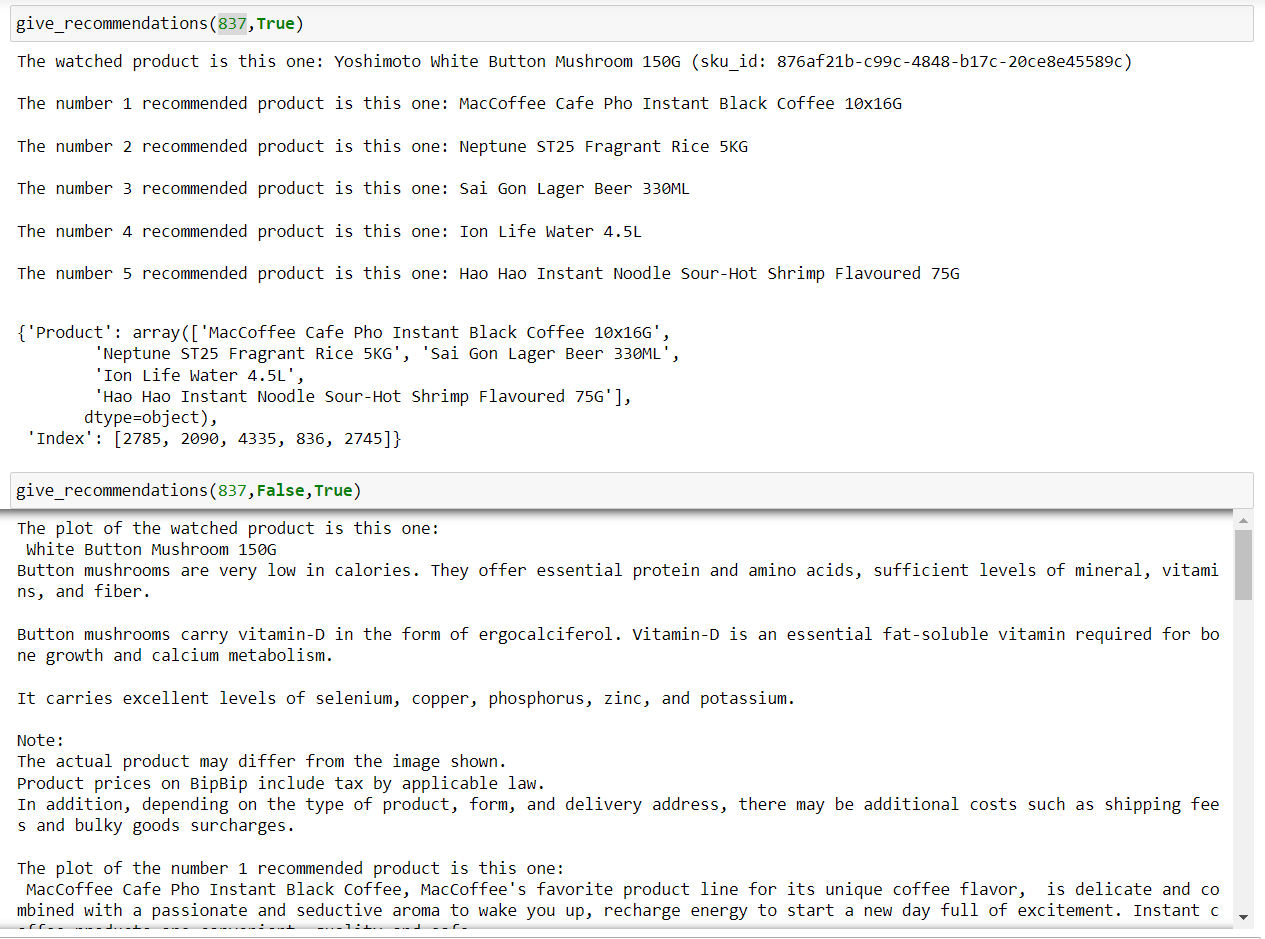


Và tiếp đó là tính độ tương quan giữa các token với nhau bằng Cosine Similarity để có thể đưa ra gợi ý sản phẩm một cách tốt nhât.

Sau khi tính toán độ tương quan giữa các giá trị, nhóm tạo 1 hàm `give\_recomendation`, với index là index của product trong tập dữ liệu. Sau đó, hàm tìm kiếm top 5 sản phẩm có độ tương đồng cosine cao nhất với sản phẩm đã được xem (hoặc sản phẩn được dùng để có thể lấy ra những sản phẩm gợi ý) và lấy lấy tên của chúng trong cột `prod\_name`. Kết quả trả về sẽ hàm một dictionary có 2 key là product và index (của sản phẩm)

Hàm có thể được thiết lập để inra các thông tin khác nhau tuy chọn bằng cách thêm các biến đầu vào là `print\_recommendation`, `print\_recommendation\_plots`, và `print\_genres` là True hoặc False



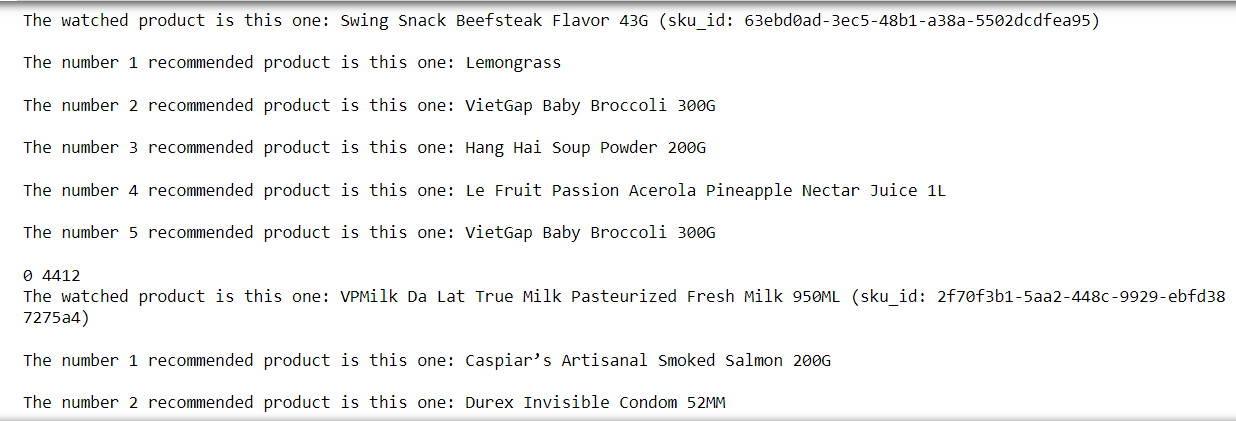
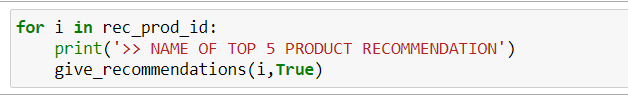


## Phân tích kết quả đánh giá:

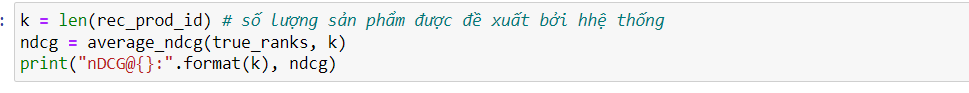
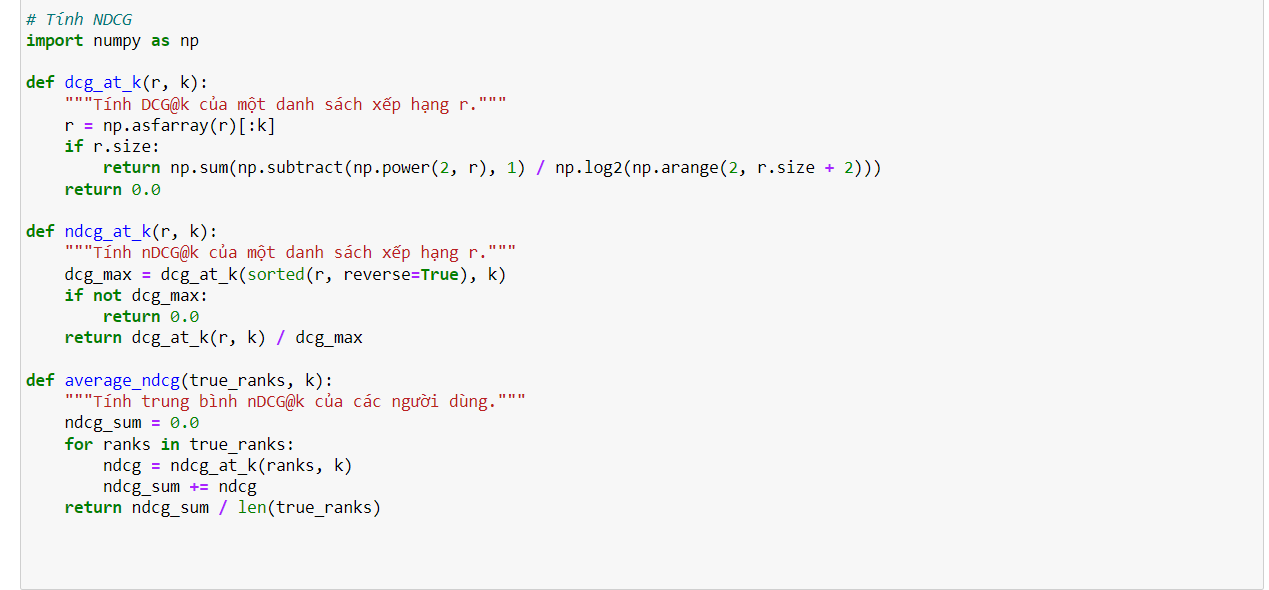
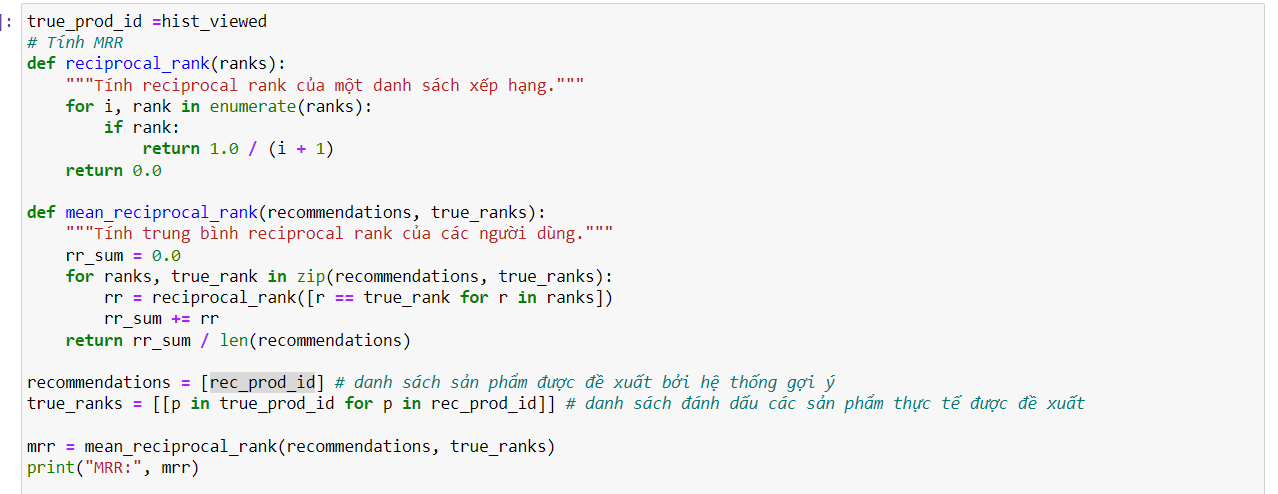
Collaborative Filtering sử dụng thông tin về hành vi của người dùng và sản phẩm để tìm ra các mối liên hệ giữa chúng, từ đó đưa ra các gợi ý cho người dùng. Tuy nhiên, nó không thể đưa ra gợi ý cho các sản phẩm mới chưa có lịch sử tương tác hoặc cho các người dùng mới chưa có dữ liệu lịch sử.

Content-Based Filtering, trái ngược với Collaborative Filtering, tập trung vào thông tin về các đặc trưng của sản phẩm để đưa ra gợi ý. Với Content-Based Filtering, các sản phẩm mới có thể được đưa ra gợi ý ngay khi chúng được thêm vào hệ thống. Tuy nhiên, Content-Based Filtering có thể không đưa ra được gợi ý chính xác nếu các sản phẩm có các đặc trưng giống nhau nhưng lại không phù hợp với sở thích của người dùng.

Kết hợp hai mô hình Collaborative Filtering và Content-Based Filtering có thể giúp tăng độ chính xác và độ phủ của hệ thống gợi ý.



Và để đánh giá kết quả kết hợp của 2 mô hình thì nhóm sử dụng phương pháp Ranking Metric để đánh giá. Từ đó có thể đưa ra những gợi ý tốt nhất cho người dùng



# Kết luận

Nhóm đã thành công thu thập dữ liệu từ một trang bán lẻ trực tuyến Tops Market. Từ bộ dữ liệu này, nhóm xây dựng thành công 2 thuật toán gợi ý sản phẩm là Collaborative Filtering và Content-based Filtering. Cụ thể, với lịch sử xem sản phẩm của nhiều người dùng khác nhau, Collaborative Filtering đã đưa ra danh sách gợi ý các sản phẩm mà những người dùng khác có cùng lịch sử xem hàng xem thêm các sản phẩm này. Content-based Filtering mở rộng, đa dạng hóa danh sách gợi ý của Collaborative Filtering bằng cách gợi ý thêm các sản phẩm có sự tương đồng cao về đặc điểm (nhãn hàng, thương hiệu,...) cho từng sản phẩm gợi ý. Như vậy, khách hàng vừa có thể khám phá thêm những sản phẩm mới do chính những người dùng chung trang mua sắm gợi ý, đồng thời có nhiều sự lựa chọn hơn với những sản phẩm gợi ý có cùng đặc điểm.

Thực tế, để đánh giá hệ thống có thực sự đưa ra gợi ý một cách khoa học và có hiệu quả không, chúng ta cần xem xét nhiều khía cạnh. Về mặt lý thuyết, các chỉ số Mean Reciprocal Rank (MRR), Normalized Discounted Cumulative Gain (NDCG) và Precision-Recall Curve là những lựa chọn hàng đầu. Ngoài ra, các dữ liệu thu thập thực tế trang web như xác suất khách hàng nhấn chọn xem các danh sách trong phần gợi ý là bao nhiêu, khách hàng đánh giá độ hữu ích của danh sách ở mức độ nào,... cần được nghiên cứu và bổ sung thêm, để kết quả đánh giá hệ thống khách quan hơn.

Một điểm cần phát triển hơn đó là việc xử lý với bộ dữ liệu có kích thước lớn hơn, phức tạp hơn. Hiện tại việc sử dụng BERT để xử lý dữ liệu văn bản đang tốn khá nhiều thời gian (trung bình 50-60 phút/lần xử lý). Để tối ưu hóa hệ thống, nhóm cần tìm thêm các phương pháp tiền xử lý dữ liệu tiết kiệm thời gian hơn.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | B. C. t. V. Nam, "Sách Trăng Thương mại điện tử Việt Nam 2022," no. Thương mại điện tử, 2022. |
| [2] | H. Đinh, "Viblo," [Online]. Available: https://viblo.asia/p/tong-quan-ve-recommender-system-recommender-system-co-ban-phan-1-924lJGBb5PM. |
| [3] | Anonymous, "Python (Ngôn ngữ lập trình)," [Online]. Available: https://vi.wikipedia.org/wiki/Python\_(ng%C3%B4n\_ng%E1%BB%AF\_l%E1%BA%ADp\_tr%C3%ACnh) . |
| [4] | K. Ngọc, "Tìm hiểu về Python (phần 1)," [Online]. Available: https://viblo.asia/p/tim-hieu-ve-python-co-ban-1-LzD5djkeZjY . |
| [5] | Anonymous, "What is Scikit-Learn in Python?," [Online]. Available: https://www.activestate.com/resources/quick-reads/what-is-scikit-learn-in-python/#:~:text=Scikit%2Dlearn%20is%20an%20open,categorizing%20data%20based%20on%20patterns.. |
| [6] | Anonymous, "Nearest Neighbors," [Online]. Available: https://scikit-learn.org/stable/modules/neighbors.html#neighbors . |
| [7] | H. Đinh, "Xây dựng Collaborative Filtering RS [Recommender System cơ bản - Phần 3]," [Online]. Available: https://viblo.asia/p/xay-dung-collaborative-filtering-rs-recommender-system-co-ban-phan-3-Az45bMqolxY . |
| [8] | P. Paialunga, "Hands-on Content Based Recommender System using Python," [Online]. Available: https://towardsdatascience.com/hands-on-content-based-recommender-system-using-python-1d643bf314e4. |
| [9] | Victor Sanh, Lysandre Debut, Julien Chaumond, Thomas Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter," 2019. [Online]. Available: https://arxiv.org/abs/1910.01108. |
| [10] | T. Đức, "Đánh giá các mô hình học máy," [Online]. Available: https://viblo.asia/p/danh-gia-cac-mo-hinh-hoc-may-RnB5pp4D5PG. |
| [11] | D. Khanh, "Bài 46 - Đánh giá mô hình phân loại trong ML," [Online]. Available: https://phamdinhkhanh.github.io/2020/08/13/ModelMetric.html. |