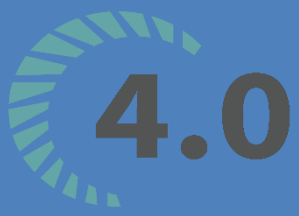
KHOA CÔNG NGHỆ THÔNG TIN

ĐẠI HỌC KHOA HỌC TỰ NHIÊN, ĐẠI HỌC QUỐC GIA TP HCM

MÔN ỨNG DỤNG DỮ LIỆU LỚN
BÁO CÁO ĐỒ ÁN THỰC HÀNH



Giảng viên:

1. Cô Nguyễn Ngọc Thảo
2. Thầy Bùi Duy Đăng

Nhóm LTS:

1. 20120055 – Nguyễn Thế Đạt
2. 20120084 – Nguyễn Văn Hiếu
3. 20120085 – Trần Xuân Hòa
4. 20120113 – Lê Nguyên Khang

ỨNG DỤNG DỮ LIỆU LỚN

HỌC KỲ I – NĂM HỌC 2023-2024

Mục lục

1. Tổng hợp quá trình thực hiện đồ án

1.1 Tổng quan về đồ án

Đồ án thuộc môn học Ứng dụng dữ liệu lớn, áp dụng các kiến thức, kỹ thuật về Khoa học dữ liệu để giải quyết một vấn đề trong thế giới thực.

1.1 Quy trình thực hiện đồ án

Nhóm thực hiện đồ án theo các bước:

1. Xác định đề tài và thu thập dataset
2. Tiền xử lí dữ liệu
3. Khám phá dữ liệu
4. Phân tích dữ liệu
5. Xây dựng Model
6. Rút ra ý nghĩa, kết luận

1.2 Những khó khăn khi thực hiện đồ án

Khó khăn trong việc chọn lọc dataset:

Dataset ban đầu nhóm chọn là Job Dataset, bộ dữ liệu về các tin tức tuyển dụng, tập dữ liệu cung cấp số lượng lớn danh sách công việc trên nhiều lĩnh vực khác nhau. Tuy nhiên sau buổi báo cáo, trình bày với GVHD thì nhóm phát hiện dataset bị trùng với một nhóm khác và nhóm không tìm thấy được insights từ bộ dữ liệu.

Vì vậy nhóm đã gấp rút tìm kiếm và thu thập bộ dataset khác, mặc dù thời gian còn khá ít.

1.3 Những kiến thức học được

- Cách sử dụng các công cụ, thư viện để phân tích, biểu diễn dữ liệu

- Cách áp dụng những kiến thức, kỹ thuật đã được học trong môn học

- Cách lấy (crawl) data

1.4 Phân công công việc

Quy trình	Công việc	Phân công
Thu thập dữ liệu	Sử dụng API lấy dữ liệu từ web	Khang
Tiền xử lý dữ liệu	Kiểm tra dòng, cột	Khang
	Xứ lý duplicated theo từng dòng	Hiếu

	Xử lý missing, sửa lỗi chính tả	Hòa
	Kiểm tra kiểu dữ liệu, thêm các cột mới	Đạt
Khám phá dữ liệu	-Cột author_name theo quantity_sold -Cột author_name theo rating	Hiếu
	-Cột category theo quantity_sold -Cột category theo review_count	Đạt
	-Cột price theo số khách hàng -Cột discount_rate theo số khách hàng -Cột cộ dữ liệu theo thời gian	Hòa
Phân tích dữ liệu	Phân tích đặc điểm khách hàng để đưa ra nhóm khách hàng cùng các cách xử lý: + Khách hàng nào có khả năng rời bỏ? + Khách hàng nào có thể chi tiêu cao hơn? + Khách hàng có khả năng tăng tần suất mua?	Khang
Mô hình	Tạo Target dựa vào insight để đánh giá khách hàng tiềm năng. Chuẩn bị dữ liệu đưa vào model. Model Decision Tree	Hòa
	Model Random Forest	Đạt
	Model Gradient Boosting	Hiếu
	Model MultinomialNB	Khang
Báo cáo	Các thành viên viết nội dung cho các phần được phân công phía trên	Tất cả thành viên, Hiếu(kiểm tra lại định dạng và đề mục, …)
	Hướng phát triển	Hòa
Slide		Hòa

1.5 Tự đánh giá

Step	Project Evaluation (maximum)	Self-Evaluation
Step 1	2.5	2.5
Step 2	3	2.5
Step 3	3	2.5
Step 4	5	4
Step 5	3	3
Tổng	**16.5**	**14.5**

2. Nội dung báo cáo

2.1 Xác định đề tài và thu thập dataset

2.1.1 Về đề tài và các yêu cầu

Đề tài: Áp dụng các quy trình, kỹ thuật Ứng dụng khoa học dữ liệu để giải quyết một vấn đề nhỏ trong thế giới thực.

Các bước cần thực hiện:

1. Thu thập dataset
2. Tiền xử lí
3. Lấy insights từ dataset
4. Xây dựng Model
5. Rút ra được các quyết định, ý nghĩa

Yêu cầu:

1. Sự dụng ít nhất 2 kỹ thuật về Ứng dữ dữ liệu lớn (lưu trữ, xử lí, trực quan)
2. Bước thu thập dữ liệu, nếu dataset được lấy từ Kaggle, kích thước dataset ít nhất là 100MB (đối với chữ), nếu dataset có chứa hình ảnh thì số lượng hình ảnh phải đủ lớn.

2.1.2 Về dataset

Dữ liệu được thu thập về từ Tiki, mô tả về tất cả các sách trong mục nhà sách Tiki.

Gồm hai bộ dữ liệu: **products** và **reviews**.

- **products**: gồm các thông tin về sản phẩm sách trong nhà sách tiki. Dữ liệu có 2030 dòng và 13 cột.
- **reviews:** gồm các đánh giá về sản phẩm được đề cập trong products. Dữ liệu có 605259 dòng và 10 cột.

Mục tiêu:

- Phân tích nhu cầu (giải quyết vấn đề gì, nhu cầu giá là bao nhiêu), hành vi (thói quen mua sắm, sở thích).
- Xem xét các yếu tố ảnh hưởng đến sự hài lòng của khách hàng (chất lượng sản phẩm, giá cả...)
- Tìm ra những khách hàng có khả năng cao sẽ tiếp tục mua sản phẩm dựa trên dữ liệu về sản phẩm và đánh giá của khách hàng. Sau đó xây dựng mô hình dự đoán khách hàng tiềm năng.

2.1.3 Phương pháp thu thập dataset

Dùng API được cung cấp bởi tiki và tiến hành cào các thông tin về sách của nhà sách tiki.

Đầu tiên thu thập các thông tin chung về product(sách) cho tập product.csv:

```
response = requests.get('https://tiki.vn/api/personalish/v1/blocks/listings?'
```

Kế đến ứng về mỗi product dùng API để lấy thêm các thông tin bổ sung về seller và author cho tập product.csv:

```
response = requests.get('https://tiki.vn/api/v2/products/{}?'
```

Cuối cùng cào các thông tin về review của tất cả các review của từng sản phẩm cho tập review.csv:

```
response = requests.get('https://tiki.vn/api/v2/reviews?'
```

2.2 Tiền xử lí dữ liệu

2.2.1 Mô tả các cột dữ liệu

Dữ liệu được thu thập về từ **Tiki**, mô tả về tất cả các sách trong mục **Nhà sách Tiki**. Gồm hai bộ dữ liệu: **products** và **reviews**

products gồm các thông tin về sản phẩm sách trong Nhà sách Tiki. Dữ liệu bao gồm 2030 dòng và 13 cột. Trong đó:

STT	Cột	Ý nghĩa
1	id	Mã sản phẩm
2	name	Tên sản phẩm
3	price	Giá sản phẩm
4	original_price	Giá niêm yết
5	discount_rate	Phần trăm giảm giá
6	quantity_sold	Số lượng đã bán
7	rating_average	Đánh giá trung bình
8	review_count	Số lượng đánh giá
9	seller_id	Mã người bán
10	seller_name	Tên người bán
11	author_name	Tên tác giả
12	category	Phân loại
13	spid	Mã sản phẩm

reviews gồm các đánh giá về sản phẩm được đề cập trong products. Dữ liệu có 605259 dòng và 10 cột. Trong đó:

STT	Cột	Ý nghĩa
1	id	Mã đánh giá
2	product_id	Mã sản phẩm
3	rating	Đánh giá
4	content	Nội dung đánh giá
5	author_name	Tên tác giả
6	title	Tiêu đề
7	created_at	Thời gian đánh giá
8	customer_id	Mã khách hàng
9	customer_name	Tên khách hàng
10	thank_count	Số lượt thích đánh giá
11	seller_product_id	Mã sản phẩm của người bán

2.2.2 Data Cleaning – Làm sạch dữ liệu

2.2.2.1 Xử lí trùng lặp

products

- Xóa cột **spid** vì nó trùng với cột id

```
products = products.drop(['spid'], axis=1)
```

```
products.duplicated().sum()
```

```
21
```

Qua kiểm tra thì có 21 dòng trùng trong tập products. Các dòng này bị trùng trong quá trình thu thập dữ liệu nên ta sẽ xóa đi.

reviews

- Xóa cột **seller_product_id** vì nó trùng với cột **product_id**

```
reviews = reviews.drop(['seller_product_id'], axis=1)
```

```
reviews.duplicated().sum()
```

```
14774
```

Các dòng này bị trùng trong quá trình thu thập dữ liệu và nó chỉ chiếm khoảng 2,4% dữ liệu nên ta sẽ xóa đi.

- Tuy nhiên qua quan sát nhóm thấy ở tập dữ liệu **reviews** trùng nhau ở cột **id** mà nó chỉ khác nhau ở 1 cột. Điều này có thể là do lỗi trong quá trình thu thập dữ liệu nên nhóm sẽ xóa đi các dòng trùng nhau ở tập reviews.

```
reviews[reviews['id'].duplicated(keep=False)].sort_values(by='id')
```

	id	created_at	rating	title	content	thank_count	customer_name	customer_id	product_id
561250	14779771	1643853610	5	Cực kì hài lòng	NaN	3	Nguyễn Long	12536047	146223395
561252	14779771	1643853610	5	Cực kì hài lòng	NaN	2	Nguyễn Long	12536047	146223395
529064	16392632	1653012913	5	Cực kì hài lòng	NaN	0	Trần Dương Minh Quang	7603930	67991600
529065	16392632	1653012913	5	Cực kì hài lòng	NaN	0	Khách Hàng	7603930	67991600

Xóa 2 hàng có **index** là 561252 và 529065 vì chúng bị trùng với các dòng khác.

2.2.2.2 Xử lí Missing values

products

```
products.isnull().sum()
```

```
id                  0
name                0
price               0
original_price      0
discount_rate       0
quantity_sold       0
rating_average      0
review_count        0
seller_id           0
category            0
seller_name         2
author_name       374
dtype: int64
```

Cột **seller_name** bị thiếu có thể là do các sản phẩm này không có tên người bán hoặc do lỗi trong quá trình thu thập dữ liệu.

Qua kiểm tra thì nhóm phát hiện 1 giá trị bị thiếu ở cột **seller_name** có giá trị **seller_id** của 1 người bán. Nhóm sẽ thay thế giá trị này bằng tên người bán tương ứng. Còn giá trị thiếu còn lại thì không trùng **seller_id** nên nhóm sẽ thay thế bằng **seller_id** tướng ứng.

Qua kiểm tra các giá trị thiếu của cột **author_name** là do các sản phẩm không có tên tác giả. Ta sẽ thay thế các giá trị này bằng Unknown.

reviews

```
reviews.isnull().sum()
```

```
id                0
created_at        0
rating            0
title             5
content      392619
thank_count       0
customer_name  6879
customer_id       0
product_id        0
dtype: int64
```

Ở cột **title** bị thiếu có thể là do người dùng không muốn để lại bình luận. Qua quan sát thì các giá trị này điều có **rating** là 5. Ngoài ra thì các có dòng có **rating** là 5 thì **title** không bị thiếu đa phần là "Cực kỳ hài lòng" còn lại là "Hài lòng", ... Nên nhóm sẽ thay thế các giá trị thiếu bằng "Cực kỳ hài lòng".

```
reviews[((reviews['rating']==5) )]
```

	id	created_at	rating	title	content	thank_count	customer_name	customer_id	product_id
0	12559756	1633960872	5	Cực kì hài lòng	Có những người bước đến, họ lấp đầy hạnh phúc ...	389	Vân Anh	22051463	74021317
1	16979365	1657206097	5	Cực kì hài lòng	Thấy nhiều bạn chê tiki gói hàng quá, may sao ...	14	Phan Thảo Yến Nhím	27791831	74021317
2	14069617	1640442831	5	Cực kì hài lòng	Bìa cực xinh, tiki giao hàng nhanh, sách không...	26	Trần Thị Trang	17748750	74021317
3	12322259	1632568671	5	Cực kì hài lòng	Nội dung của sách thì không phải bàn đến rồi, ...	16	Khánh Ly	10149686	74021317
4	18368714	1670438913	5	Cực kì hài lòng	Một cuốn sách rất đáng đọc về tình yêu thương ...	1	Phương Linh	28545286	74021317
...	...	...	...	...	...	...	...	...	...
605249	9561654	1620393883	5	Cực kì hài lòng	NaN	0	HaThuy Vu	10750423	75996803
605250	9554276	1620370994	5	Cực kì hài lòng	NaN	0	Luu Quynh Huong	14854794	75996803
605251	9022442	1618119132	5	Cực kì hài lòng	NaN	0	vo nhan	6698905	75996803
605252	8726727	1616919902	5	Cực kì hài lòng	NaN	0	Xuân Xuân	5668673	75996803
605253	7517182	1612414399	5	Cực kì hài lòng	NaN	0	Tô Nhung	11878	75996803

518401 rows × 9 columns

Ở cột **customer_name** nhóm sẽ thay thế các giá trị thiếu bằng những **customer_name** có giá tương ứng dựa vào **customer_id**. Tuy nhiên cũng có một số **customer_id** không có **customer_name** tương ứng. Nhóm sẽ thay thế các giá trị này bằng giá trị của **customer_id**.

Ở cột **content** bị thiếu có thể là do người dùng không muốn để lại bình luận. Ta sẽ thay thế các giá trị này bằng "No comment".

2.2.2.3 Xử lý Data Type

products

```
id                              {<class 'int'>}
name                            {<class 'str'>}
price                           {<class 'int'>}
original_price                  {<class 'int'>}
discount_rate                   {<class 'int'>}
quantity_sold                   {<class 'int'>}
rating_average                {<class 'float'>}
review_count                    {<class 'int'>}
seller_id                       {<class 'int'>}
category                        {<class 'str'>}
seller_name       {<class 'str'>, <class 'int'>}
author_name                     {<class 'str'>}
dtype: object
```

Chuyển kiểu dữ liệu cột **seller_name** sang kiểu **str** và đổi tên cột **id** --> thành **product_id**.

Chuyển dữ liệu của các cột categorical sang viết thường, xóa khoảng trắng ở đầu và cuối, xoá các dấu câu, các kí tự đặc biệt và xóa khoảng trắng dư thừa.

Ngoài ra nhóm còn phát hiện các lỗi chính tả khác trong cột **author_name**. Nhóm sẽ thay thế các giá trị này bằng các giá trị đúng.

reviews

```
Data columns (total 9 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   id             590483 non-null  int64
 1   created_at     590483 non-null  int64
 2   rating         590483 non-null  int64
 3   title          590483 non-null  object
 4   content        590483 non-null  object
 5   thank_count    590483 non-null  int64
 6   customer_name  590483 non-null  object
 7   customer_id    590483 non-null  int64
 8   product_id     590483 non-null  int64
dtypes: int64(6), object(3)
```

Chuyển kiểu dữ liệu cột **customer_name** sang kiểu **str**.

Chuyển kiểu dữ liệu cột **created_at** sang kiểu **datetime** và thêm các cột **year**, **month**, **day**, **weekend**, **hour** để phân tích dữ liệu.

2.2.2.4 Outliers

Qua kiểm tra thì dữ liệu không có Outlier.

2.2.2.5 Save data

```
print(products.shape)
print(reviews.shape)
```

```
(2009, 12)
(590483, 14)
```

Sau quá trình xử lý dữ liệu, **products** còn 2009 dòng và 12 cột. **reviews** còn 590483 dòng và 14 cột.

```
df.to_csv('dataset/merged_data.csv', index=False)
```

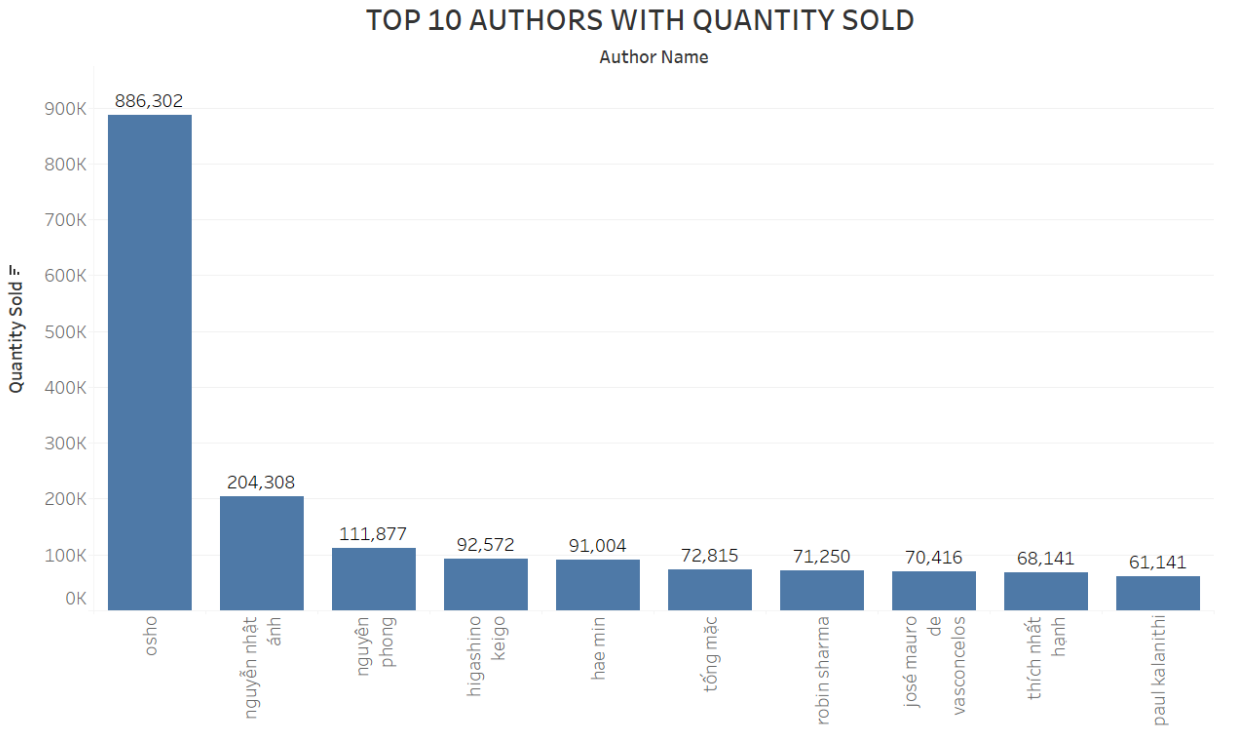
Sau khi xử lý thì tập dữ liệu có kích thước: 183MB.

2.3 Khám phá dữ liệu

Sử dụng Tableau để trực quan hóa.

Dựa vào các biểu đồ trực quan để phân tích sở thích, hành vi của khách hàng.

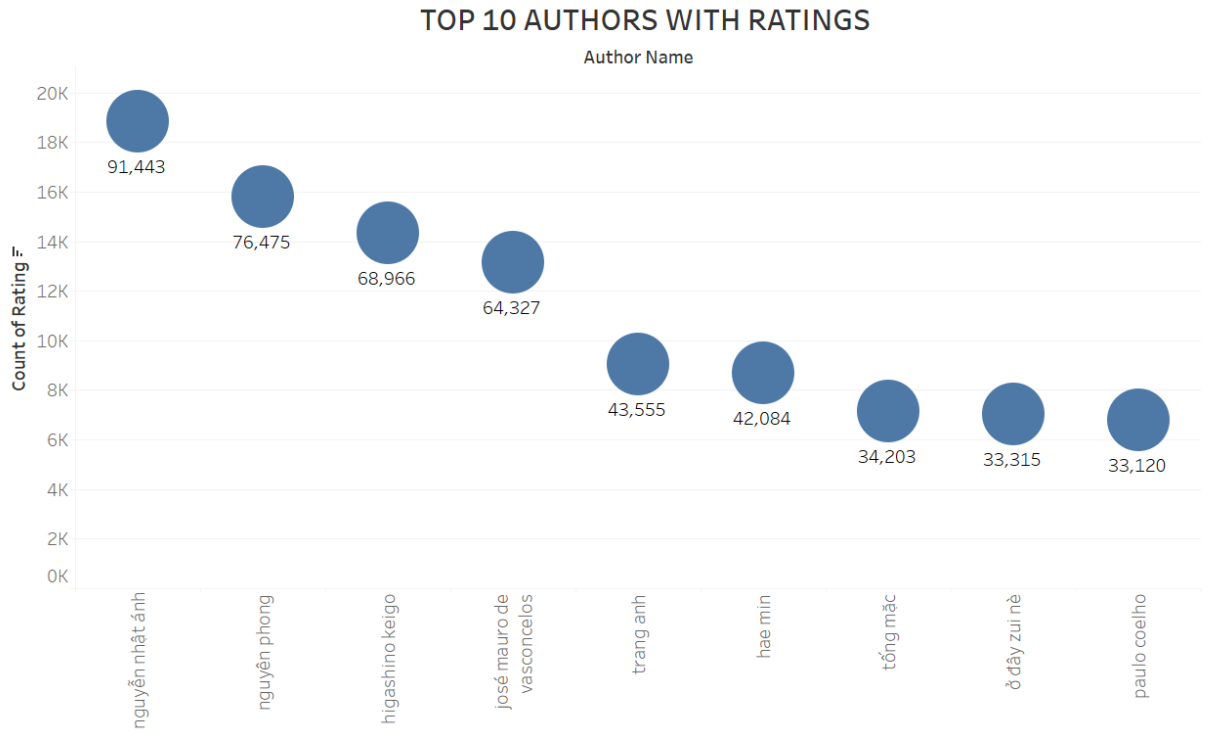
2.3.1 Cột author_name theo quantity_sold



Thông qua biểu đồ các tác giả có số lượng bán sách nhiều nhất ta có thể thấy được tuy thuộc top 10 những tác giả Osho thậm chí có số lượng bán hơn 9 người còn lại cộng lại. Có sự tăng vọt này nguyên nhân nằm ở năm 2021 thời điểm có thể do dịch mà khách hàng có xu hướng mua sách của Osho chủ yếu nói về tìm kiếm sự tự do và hạnh phúc được ưa chuộng

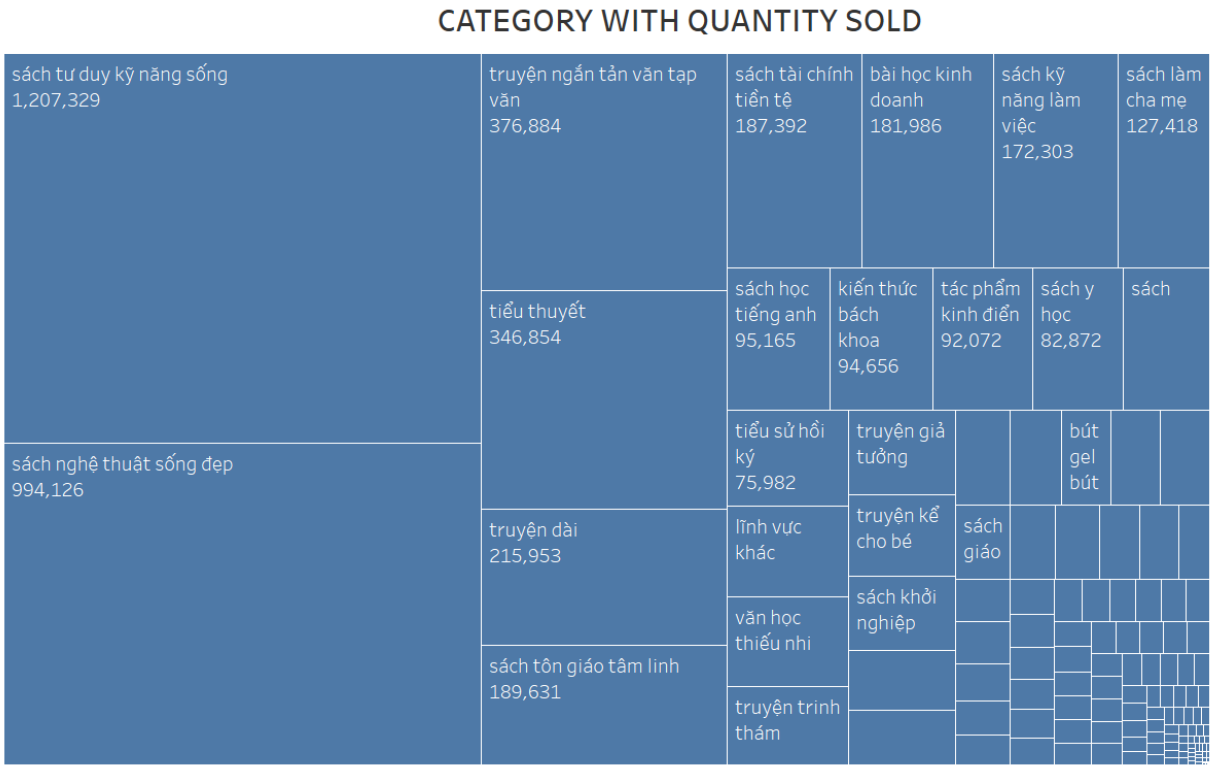
Dẫn đến sách của tác giả bán ra tăng đột ngột thậm chí bằng 9 người trong top 10 cộng lại. Và xu hướng mua hàng của khách hàng cũng tập trung vào top 10 các tác giả.

2.3.2 Cột author_name theo rating



Những tác giả có lượt đánh giá cao thì hầu như đều thuộc những tác giả có số lượng bán cao. Ở đây không có Osho vì sách của Osho tuy có số lượng bán nhiều nhưng hầu như chỉ tập trung vào 1 vài sản phẩm nên số lượng rating không nhiều bằng các tác giả khác.

2.3.3 Cột category theo quantity_sold

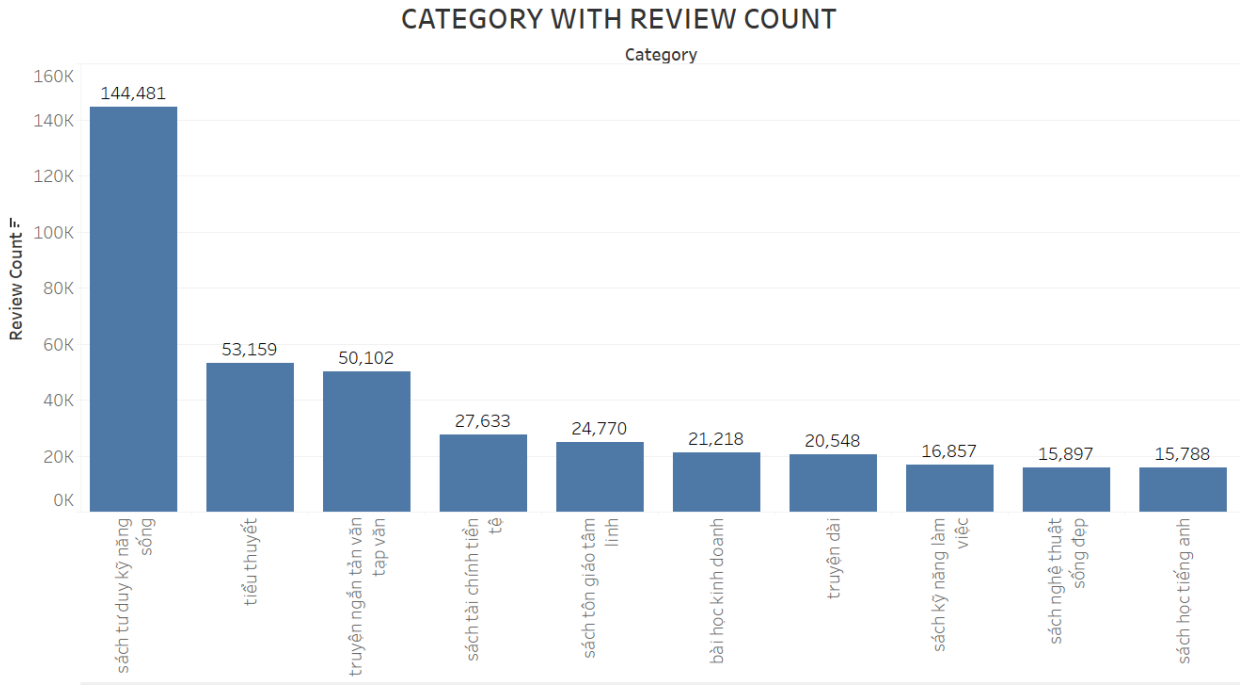


Dựa vào biểu đồ trên ta thấy được những thể loại sách top 10 chiếm gần như 50% số lượng bán.

→Do đó ta có thể thấy được sở thích hay xu hướng của khách hàng đang tập trung vào việc phát triển kỹ năng sống, nâng cao bản than hay tập trung vào các vấn đề thực tế như tài chính, kinh doanh.

→Có xu hướng tìm đọc những tác giả nổi tiếng, có uy tín trong các lĩnh vực của mình. (Osho chiếm hơn **80%** nghệ thuật sống đẹp, Nguyễn Nhật Ánh chiếm hơn **50%** truyện dài).

2.3.4 Cột category theo review_count



Hầu hết các thể loại sách bán chạy đều có số lượng review cao tuy nhiên ở đây **sách học tiếng anh** không nằm trong top 10 bán chạy vẫn có lượng review khá cao. Cho thấy khách hàng cũng có nhu cầu quan tâm đến việc học tiếng Anh khá là lớn.

2.3.5 Cột price theo số khách hàng

Nhóm sản phẩm có giá dưới 200.000 có lượng khách hàng rất lớn. Chiếm tới gần 90% lượng khách hàng mua sản phẩm. à Khách hàng có xu hướng mua hàng với giá trong khoản dưới 200.000.

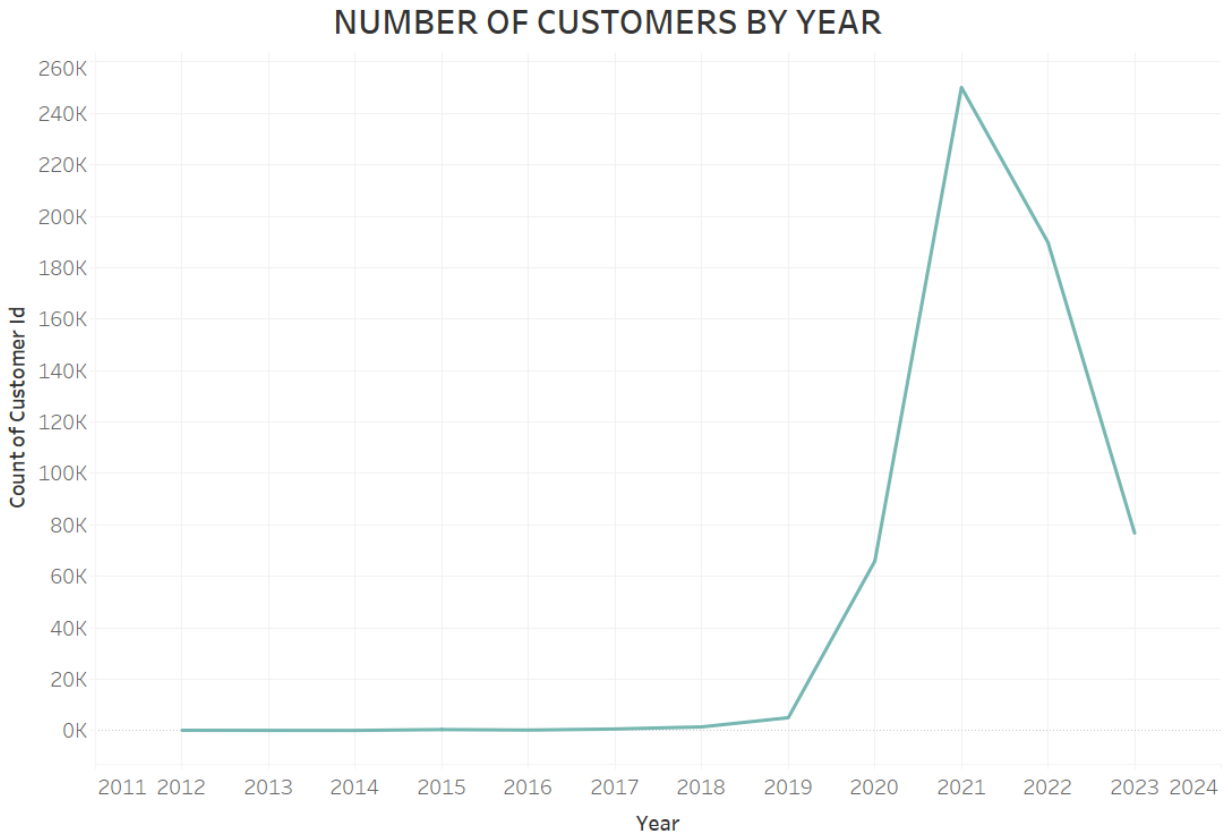
Khách hàng cũng có mua sản phẩm từ 200.000 đến 700.000 tuy nhiên ở mức thấp hơn chiến khoảng 9% lượng khách hàng. Còn ở mức lớn hơn 700.000 thì có số lượng mua rất thấp chủ yếu khoảng 1% có thể là do mua combo sách nên giá khá cao. Hoặc là những sản phẩm này khá ít nên số lượng khách hàng cũng ít.

2.3.6 Cột discount_rate theo số khách hàng

Lượng khách hàng tập trung đông ở các sản phẩm giảm giả từ 28% à 37% thậm chí chiếm đến 79% khách hàng. Khách hàng mua lúc không giảm giá cũng khá cao có thể là do lúc đó có nhu cầu nhưng sản phẩm chưa có đợt giảm giá. Các mức giảm giá nhỏ hơn 28% khá thấp có thể là do khách hàng chưa muốn mua với giá giảm đó. Còn các mức giảm giá từ 38% cũng có ít khách hàng có thể là do số sản phẩm được giảm giá cao ít nên khách hàng cũng ít.

2.3.7 Phân bố của dữ liệu giữa các cột với thời gian

2.3.7.1 Số lượng khách hàng theo thời gian

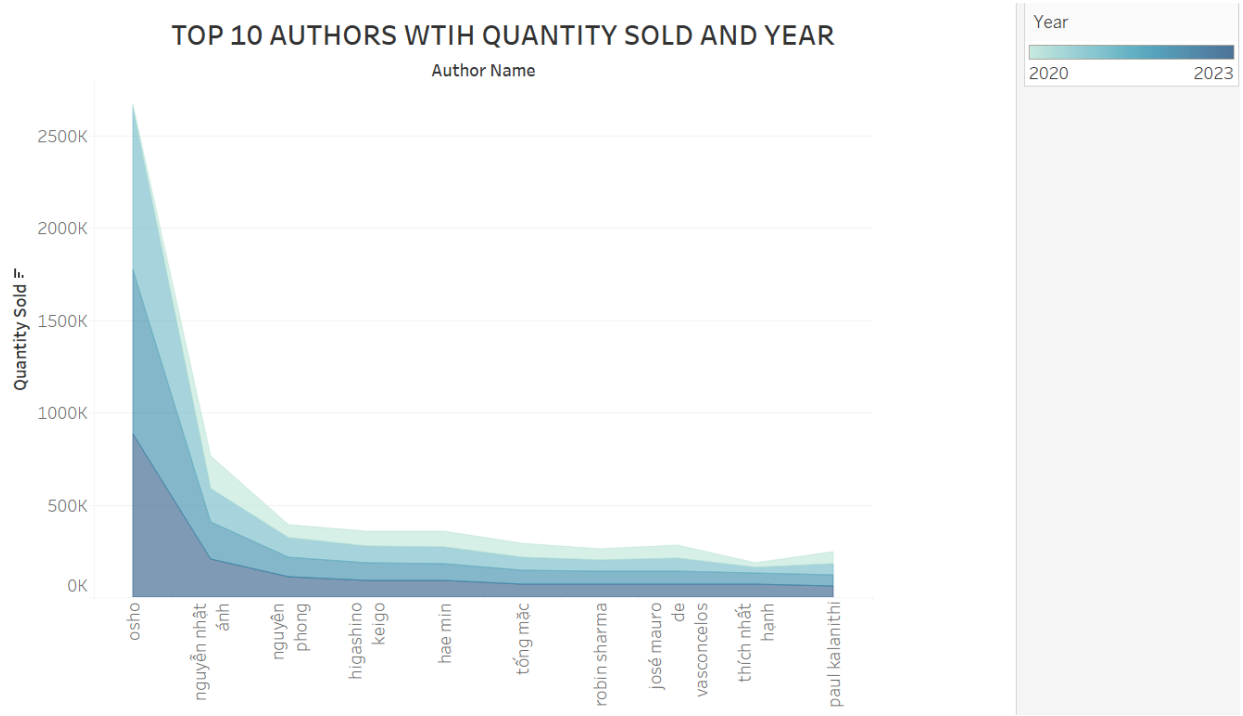


Dựa vào biểu đồ ta có thể thấy được là bắt đầu từ năm 2020 đến nay là lúc có lượng khách hàng nhiều nhất. Điều này có thể là do từ năm 2012 đến năm 2019 việc mua bán hàng online chưa thực sự phát triển thường tập trung ở 1 vài thành phố lớn có thể giao hàng nhanh chóng. Kể từ năm 2020 thì thương mại điện tử đã phát triển lại có những ứng dụng hỗ trợ giao hàng nhanh đến các nơi nên số lượng khách hàng tăng vọt và cũng có thể là thời điểm này mức sống của khách hàng cũng phát triển.

Đặc biệt ở năm 2021 thì số lượng khách hàng đột ngột tăng mạnh gần gấp 4 lần năm 2020 điều này có thể là do năm 2021 có dịch bệnh mọi người ở nhà có nhiều thời gian rảnh để đọc sách nên số lượng khách hàng tăng nhanh.

Đến năm 2022, 2023 thì lượng khách hàng giảm bớt nhanh chóng điều này có thể là do hết dịch mọi người ít thời gian đọc sách hoặc là do kể từ năm 2022 các ứng dụng thương mại điện tử phát triển mạnh mẽ nổi bật nhất là TikTok, …

2.3.7.2 Tác giả, số lượng bán theo năm



Ở những bước phân tích trên ta có được số lượng khách hàng cùng số sản phẩm được bán đều tập trung ở sau năm 2020 nên ta chỉ xem top tác giả them các năm này. Qua biểu đồ có thể thấy: ở năm 2020, tác giả Nguyễn Nhật Ánh vẫn là tá giả đứng đầu về số lượng bán. Tuy nhiên khi qua năm 2021 thì là tác giả Osho, điều nay cho thấy năm 2021 có dịch bệnh bùng phát nên các khách hàng hướng tới cuộc sống tự do, hạnh phúc mang tiêu chí của tác giả Osho.

2.4 Phân tích dữ liệu

2.4.1. Mục tiêu:

Từ tập dữ liệu có sẵn phân loại khách hàng của tiki thành các phân khúc khác nhau để thấy được hành vi đặc trưng của từng nhóm khách hàng. Từ đó, tùy vào từng phân khúc khách hàng mà đưa ra các chiến lực phù hợp.

Dữ liệu được dùng là tập review được cào tính từ 1/1/2020 - 14/12/2023.

2.4.2. Hướng giải quyết:

Áp dụng mô hình phân tích RFM (Recency, Frequency, Monetary): đây là mô hình phân tích và phân khúc khách hàng dựa vào đặc trưng hành vi trong dữ liệu giao dịch trong quá khứ.

RFM sẽ tính điểm của mỗi khách hàng dựa trên 3 yếu tố:

- *Recency*: khoảng cách giữa ngày mua hàng gần nhất với hiện tại. Khách hàng có recency càng cao thì khả năng khách hàng đó rời bỏ cửa hàng càng cao.
- *Frequency*: Tổng số lần mua hàng của một khách hàng. Frequency càng cao thì khả năng khách hàng đó sẽ trở thành khách hàng trung thành của cửa hàng càng cao.
- *Monetary*: Tổng tiền khách hàng đã chi cho việc mua hàng tại cửa hàng. Monetary càng cao thì khả năng tiếp cận up-sell hay cross-sell càng cao.

2.4.5. Phân khúc khách hàng:

Dựa vào chuỗi RFM ta định nghĩa các khúc khách hàng:

- **Champion**: Là những khách hàng mới giao dịch, có mua hàng thường xuyên, chi tiêu nhiều.
- **Loyal Customer**: Là những khách hàng có giao dịch gần đây, chi tiêu mức khá và có giao dịch khá thường xuyên.
- **Recent Customer**: Là những khách hàng có mua hàng gần đây với giá trị giỏ hàng thấp và không thương xuyên.
- **Customer Needing Attention**: Là những khách hàng có tầng suất mua hàng và giá trị giỏ hàng ở mức khá nhưng gần đây không quay lại.
- **Can’t Lose Them**: Là khách hàng lâu không quay lại, từng mua hàng thường xuyên với giá trị giỏ hàng lớn.
- **Lost**: Là những khách hàng mua với tầng xuất thưa, giá trị giỏ hàng thấp và gần đây không mua hàng.

Chuỗi RFM ứng với từng khúc khách hàng như sau:

- **Champion**: 555,554,544,545,455,445,454.
- **Recent Customer**: 512, 511, 422, 421, 412, 411, 311, 525, 524, 523, 522, 521, 515, 514, 513, 425, 424, 413, 414, 415, 315, 314, 313.

- **Loyal Customer**: 543, 444, 435, 355, 354, 345, 344, 335, 553, 551, 552, 541, 542, 533, 532, 531, 452, 451, 442, 441, 431, 453, 433, 432, 423, 353, 352, 351, 342, 341, 333, 323.
- **Customer Needing Attention**: 535, 534, 443, 434, 343, 334, 325, 324, 331, 321, 312, 221, 213, 255, 254, 245, 244, 253, 252, 243, 242, 235, 234, 225, 224, 153, 152, 145, 143, 142, 135, 134, 133, 125, 124.
- **Can't Lose Them**: 155,154,144,214,215,115,114,113.
- **Lost**: 331, 321, 312, 221, 213, 255, 254, 245, 244, 253, 252, 243, 242, 235, 234, 225, 224, 153, 152, 145, 143, 142, 135, 134, 133, 125, 124, 155, 154, 144, 214, 215, 115, 114, 113, 332, 322, 231, 241, 251, 233, 232, 223, 222, 132, 123, 122, 212, 211, 111, 112, 121, 131, 141, 151.

2.4.4. Thực hiện:

Tính giá trị Recency dựa theo created_at:

Recency

```
today = datetime.datetime.now()
df_RFM = merged_data.groupby('customer_id').created_at.max().reset_index()
df_RFM.created_at = pd.to_datetime(df_RFM.created_at)
df_RFM['Recency'] = (today - df_RFM['created_at']).dt.days
```

Tính giá trị Frequency dựa theo các review của khách hàng:

Frequency

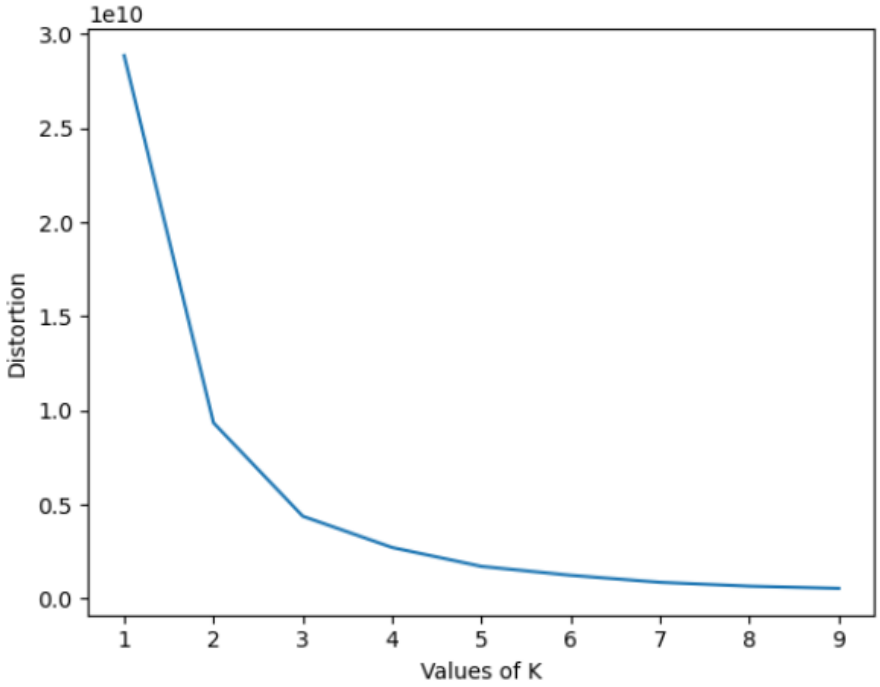
```
df_frequency = merged_data.groupby('customer_id').created_at.count().reset_index()
df_frequency.columns = ['customer_id', 'Frequency']
df_RFM = pd.merge(df_frequency, df_RFM, on = 'customer_id')
```

Tính giá trị Monetary dựa theo price sản phẩm:

Monetary

```
df_monetary = merged_data.groupby('customer_id').price.sum().reset_index()
df_monetary.columns = ['customer_id', 'Monetary']
df_RFM = pd.merge(df_monetary, df_RFM, on = 'customer_id')
```

Chia các giá trị theo các khoảng và đánh điểm cho các khoảng đó tăng dần theo độ tốt của từng giá trị. Để các khoảng được chia phản ánh đúng tập các giá trị trong từng khoảng ta dùng phương pháp Elbow quan sát biểu đồ hàm biến dạng khi điều chỉnh số lượng cụm của Kmeans cho giá trị Recency:



Dựa theo biểu đồ ta thấy có sự gãy khúc ở 2, 3 và 5. Để có thể phân khúc khách hàng rõ ràng chi tiết ta chọn K = 5. Để có sự đồng nhất, ở Frequency, Monetary cũng nhận giá trị K = 5.
Tiến hành chia khoảng và đánh điểm cho giá trị Recency, khi Recency càng lớn thì nhận giá trị là 1 và càng nhỏ thì tăng dần thành 5. Ngược lại với Frequency, Monetary.
Các điểm sau khi được đánh sẽ được tiến hành nối lại thành một chuỗi theo thứ tự RFM.
Ví dụ:

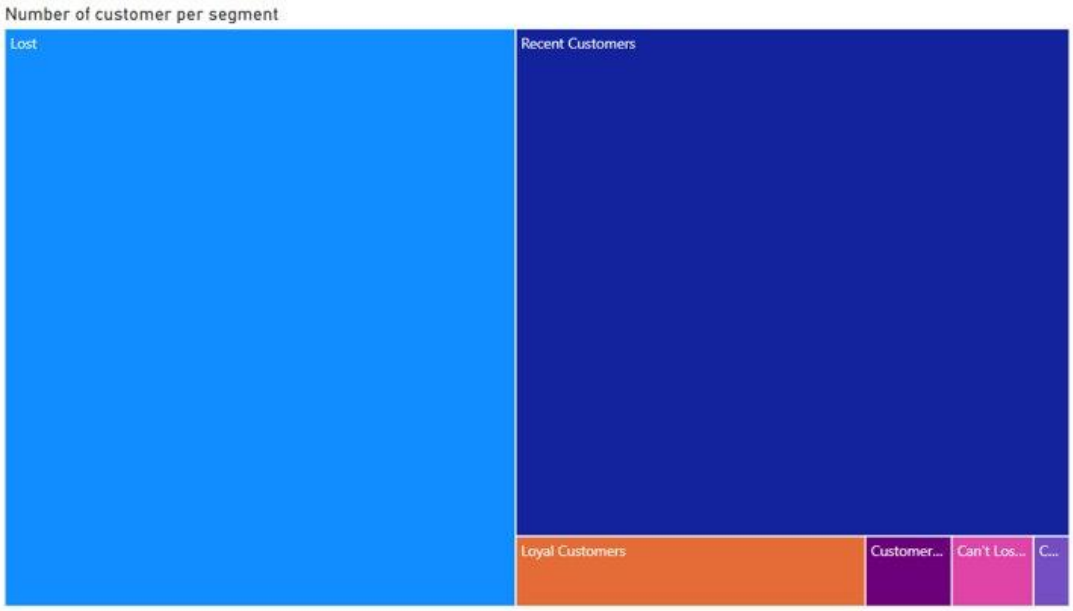
555: nói lên khách hàng này có phát sinh mua hàng gần đây, với tầng suất lớn và giá trị giỏ hàng lớn.

253: nói lên khách hàng đã khá lâu không mua hàng, từng mua hàng tầng suất thương xuyên, và giá trị giỏ hàng ở mức khá.

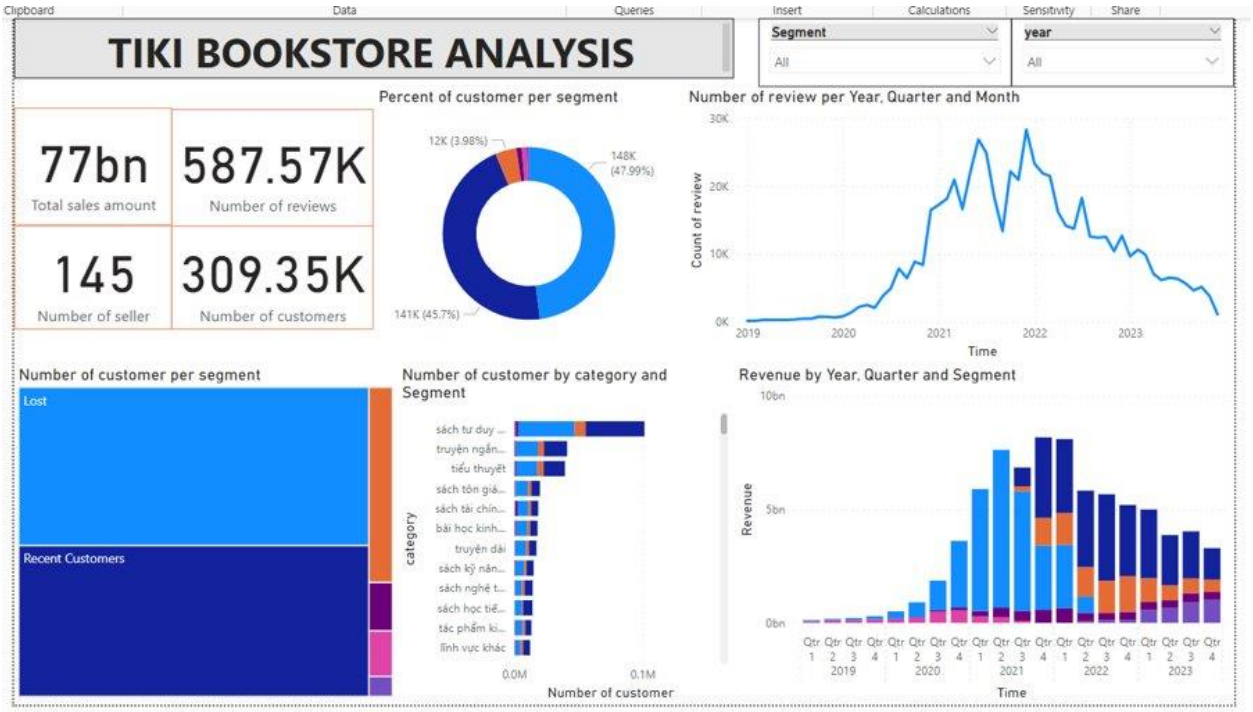
Map các điểm tính được với phân khúc khách hàng tương ứng.

2.4.4. Trực quan:

Dùng treemap thể hiện phân khúc khách hàng của tiki:



Dùng dashboard để có thêm nhiêu thông tin về các phân khúc khách hàng:



Có thể xem dashboard này [tại đây](#).

Biểu đồ đường: cho ta thấy nhìn chung tiki có lượng khách hàng phát triển mạnh năm 2021- đầu 2022 do dịch. Tuy nhiên sau đó đã liên tục giảm, có thể là do sự xuất hiện và phát triển của tiki đã làm cho thị trường sàn thương mại điện tử nói chung của tiki bị thu hẹp hay nhà sách tiki nói riêng.

Treemap: cho thấy lượng khách hàng chiếm đa số là khách hàng mà tiki đã đánh mất (Lost) và khách hàng mua hàng gần đây không thường xuyên với giá trị giỏ hàng thấp (Recent Customers).

Biểu đồ cột dọc: cho ta biết được doanh thu theo quý, năm của khách hàng đến từ nhưng khúc khách hàng nào.
Các thành phần còn lại cho ta biết thêm thông tin về những phân khúc khách hàng.

2.4.6. Hướng xử lý theo từng phân khúc:

Lost, Can't Lose Them: cần kích mua sớm nhất có thể vì khả năng cao họ sẽ nhanh chóng quên đi tiki và trở thành khách hàng của cửa hàng khác. Ta có thể dùng email, quảng cáo về sản phẩm cùng discount để kích mua họ lại, đặc biệt với những khách hàng từng có giá trị giỏ hàng lớn có thể gọi điện thoại và cá nhân hóa chăm sóc họ.

Recent Customers: cần làm họ hài lòng từ những đơn hàng đầu tiên để tăng khả năng quay lại phát sinh đơn hàng. Có thể áp dụng quy trình chăm sóc khách hàng, lấy ý kiến về sự hài long cũng như không hài lòng của họ, tặng kèm các voucher.

Loyal Customers: Cần nâng giá trị giỏ hàng của nhóm khách hàng này. Đưa ra các khuyến mãi đi kèm các ngưỡng chi tiêu khi học phát sinh đơn hàng.

Customer Needing Attention: Tìm nguyên nhân khiến họ ít mua gần đây và khắt phục nó, đề xuất chương trình khuyến mãi kịch thích việc mua hàng của họ trở lại.

Champion: Cần giữ chân nhóm khách hàng này bằng mọi giá. Các chuyến lược được đưa ra như: dành thẻ VIP cho họ, tích điểm theo những chi tiêu của họ, đề suất các combo sản phẩm liên quan để thể loại mà học quan tâm.

2.5 Xây dựng Model

2.5.1. Chuẩn bị dữ liệu:

Dựa theo các phân tích ở trên ta có được các điều kiện của khách hàng tiềm năng.

Theo thể loại:

Top 10 thể loại có có lượng bán nhiều nhất đều liên tục nằm trong top 10 thể loại có lượng bán nhiều nhất từ năm 2021 đến năm 2013. Và tổng lượng chiếm đến 75% lượng bán của tất cả các thể loại. Điều này cho thấy rằng các thể loại này có thể được coi là những thể loại phổ biến nhất trong thời gian dài. Nên nhóm chọn các khách hàng có sở thích mua các sản phẩm thuộc các thể loại này thuộc nhóm khách hàng tiềm năng.

Theo tác giả:

Tương tự thì top 10 tác giả có lượng bán nhiều nhất đều liên tục nằm trong top 10 tác giả có lượng bán nhiều nhất từ năm 2021 đến năm 2013. Và tổng lượng chiếm đến 51% lượng bán của tất cả các tác giả. Điều này cho thấy rằng các tác giả này có thể được coi là những tác giả phổ biến nhất trong thời gian dài. Nên nhóm chọn các khách hàng có sở thích mua các sản phẩm của các tác giả này thuộc nhóm khách hàng tiềm năng.

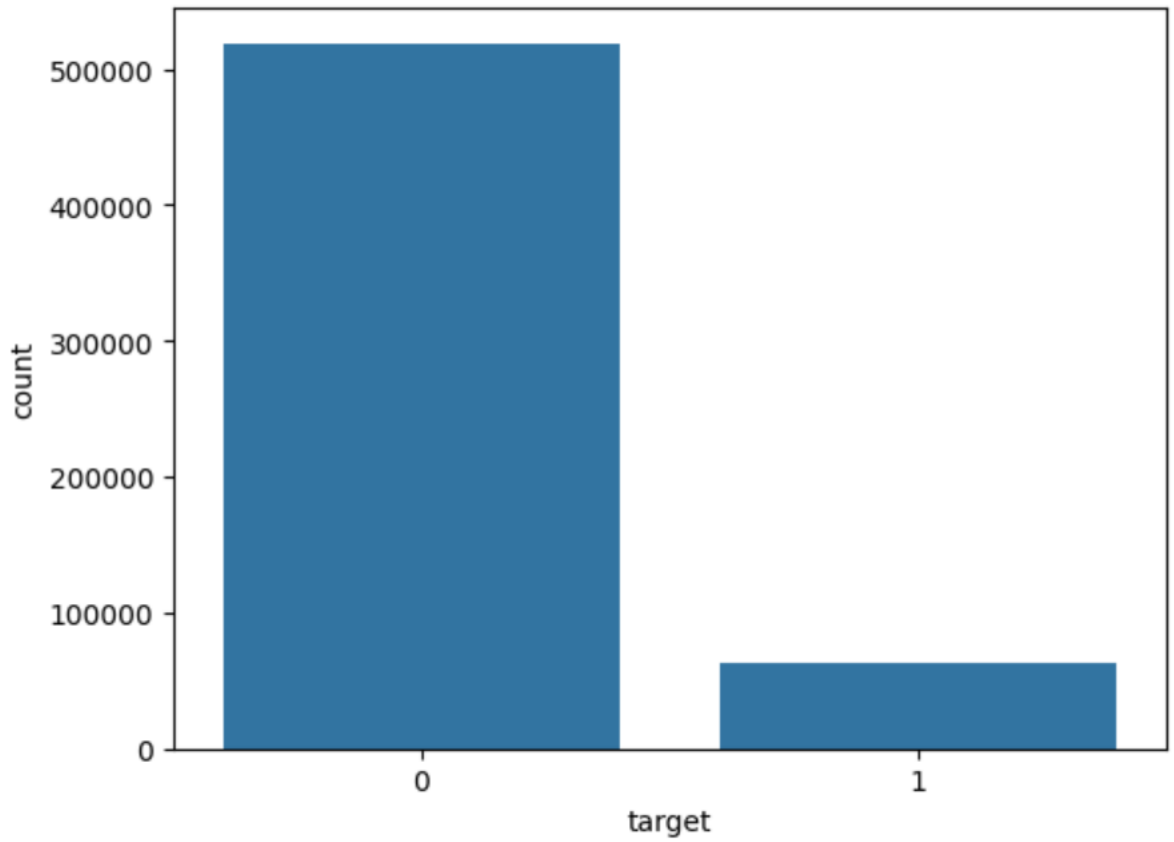
Theo tần suất mua và lần gần nhất:

Chọn các khách hàng có tần suất mua hàng 3 lần trở lên trong năm 2023 là nhóm khách hàng tiềm năng. Vì các khách hàng này có xu hướng mua hàng thường xuyên. Và các khách hàng nào mua hàng trong 3 tháng cuối năm 2023 sẽ được xem xét là nhóm khách hàng tiềm năng. Vì những khách hàng có khả năng cao sẽ tiếp tục mua hàng.

Theo giá tiền sản phẩm:

Khách hàng có xu hướng mua các sản phẩm có giá từ 50.000 đến 200.000 là nhóm khách hàng tiềm năng. Vì các sản phẩm có giá này chiếm đến 79% khách hàng mua hàng. Nên nhóm sẽ chọn các khách hàng có xu hướng mua các sản phẩm có giá từ 50.000 đến 200.000 là nhóm khách hàng tiềm năng.

Sau khi gán nhãn cho cột target ta được:



Số lượng target là 1 chỉ chiếm hơn 10% nên ta sẽ sử dụng các metrics để đánh giá mô hình là:
precision: là tỷ lệ giữa số mẫu được dự đoán chính xác là dương và tổng số mẫu được dự đoán là dương.
recall: là tỷ lệ giữa số mẫu được dự đoán chính xác là dương và tổng số mẫu dương thực sự.
f1: là một chỉ số tổng hợp của precision và recall, được tính bằng trung bình cộng trọng số của precision và recall. F1-score càng cao thì mô hình càng chính xác.

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
```

Chia tập train, test theo tỷ lệ 8:2.

2.5.2. Các mô hình dự đoán khách hàng tiềm năng

Chọn các mô hình cơ sở để phân loại các khách hàng tiềm năng, như Decision Tree, RandomForest, Gradient Boosting, Logictis Regression. Những mô hình này có tốc độ dự đoán, tốc độ huấn luyện nhanh, dễ hiểu và có khả năng xử lý nhiễu tốt. Và áp dụng GridSearchCV để tìm các siêu tham số tối ưu hóa các mô hình trên.

2.4.2.1. Decision Tree Classifier

Nguyên tác hoạt động: là một mô hình phân loại đơn giản, dễ hiểu và có thể được sử dụng để tạo ra các quy tắc kinh doanh có thể được sử dụng để xác định khách hàng tiềm năng. Mô hình này tạo ra một cây quyết định, trong đó mỗi nút đại diện cho một thuộc tính và mỗi nhánh đại diện cho một giá trị của thuộc tính đó. Điều này giúp dễ dàng hiểu cách thức hoạt động của mô hình và tại sao nó đưa ra các dự đoán cụ thể.

Thời gian chạy:

```
tree_grid_res1 = decision_tree(X_train, y_train)
116m 37.7s
```

2.4.2.2. Random Forest Classifier

Nguyên tắc hoạt động: thường có hiệu suất tốt trong các bài toán phân loại, bao gồm cả bài toán dự đoán khách hàng tiềm năng. Điều này là do mô hình Random Forest Classifier kết hợp kết quả của nhiều cây quyết định, giúp giảm thiểu khả năng overfitting. Có thể được huấn luyện nhanh chóng, ngay cả với dữ liệu lớn. Mô hình này có thể cung cấp hiệu suất tốt, dễ hiểu và giải thích, đồng thời có thể xử lý dữ liệu nhiễu tốt.

Thời gian chạy:

```python
def random_forest(X, Y):
    param_grid = {
        'n_estimators': [100, 200, 300],
        'max_depth': [int(x) for x in np.linspace(5, 20, num=4)],
        'min_samples_split': [2, 5, 10],
        'min_samples_leaf': [1, 2, 4],
        'criterion': ['gini', 'entropy']
    }
    rf = RandomForestClassifier()
    rf_grid = GridSearchCV(estimator=rf, param_grid=param_grid, scoring = ['precision', 'recall', 'f1'])
    rf_grid.fit(X, Y)
    return rf_grid
```

Python

```
rf_grid = random_forest(X_train, y_train)
1777m 18.6s
```

Python

2.4.2.3. Gradient Boosting Classifier

Nguyên tắc hoạt động: thường có hiệu suất tốt trong các bài toán phân loại, bao gồm cả bài toán dự đoán khách hàng tiềm năng. Điều này là do mô hình Gradient Boosting Classifier kết hợp kết quả của nhiều mô hình yếu, giúp giảm thiểu khả năng overfitting. Có thể được huấn luyện nhanh chóng, ngay cả với dữ liệu lớn. Mô hình này có thể cung

cấp hiệu suất tốt, dễ hiểu và giải thích, đồng thời có thể xử lý dữ liệu nhiễu tốt. Khá tương tự với mô hình Random Forest Classifier.

Thời gian chạy:

```
def gradient_boosting(X, Y):
    param_grid = {
        'n_estimators': [100, 200, 300],
        'learning_rate': [0.1, 0.05, 0.01],
        'max_depth': [int(x) for x in np.linspace(5, 20, num=4)],
        'subsample': [0.8, 0.9, 1.0],
    }
    gb = GradientBoostingClassifier()
    gb_grid = GridSearchCV(estimator=gb, param_grid=param_grid, scoring = 'f1')
    gb_grid.fit(X, Y)
    return gb_grid
```

✓ 0.0s

```
gb_grid = gradient_boosting(X_train, y_train)
```

1622m 31.5s

2.4.2.4. MultinomialNB

Nguyên tác hoạt động: được sử dụng cho dữ liệu đầu vào là các biến phân loại. Trong bài toán dự đoán khách hàng tiềm năng này thì có các biến phân loại như category, author_name, price, ... nên mô hình này là một lựa chọn phù hợp. Mô hình có hiệu suất tốt trong các bài toán phân loại dữ liệu phân loại và có thể được huấn luyện nhanh chóng, ngay cả với dữ liệu lớn.

Thời gian chạy:

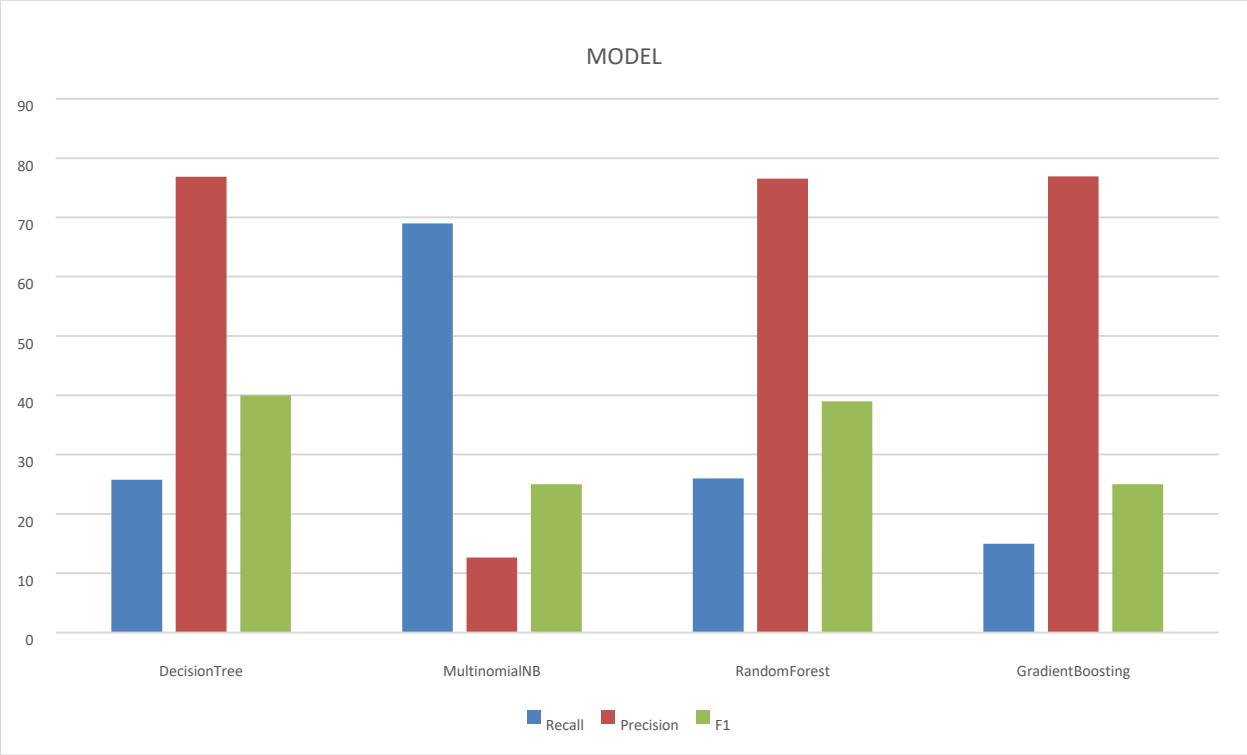
```
def multinomial_NB(X, Y):
    param_grid = {
        'alpha': [0.01, 0.1, 0.2, 0.3, 0.4 , 0.5, 1.0, 1.5, 2.0]
    }
    NB1 = MultinomialNB()
    NB1 = GridSearchCV(estimator=NB1, param_grid=param_grid, scoring = 'f1')
    NB1.fit(X, Y)
    return NB1
```

✓ 0.0s

```
NB1 = multinomial_NB(X_train, y_train)
```

✓ 13m 20.1s

2.4.2.5. Kết quả



Kết quả của 4 mô hình là tương đối tốt, với độ chính xác trung bình khoảng 70%. Điều này cho thấy rằng các mô hình này có thể dự đoán khá chính xác khả năng mua hàng của một khách hàng dựa trên dữ liệu đầu vào.

Mô hình Random Forest có độ chính xác cao nhất, với độ chính xác recall, precision và F1 lần lượt là 25%, 76,89% và 39%. Điều này cho thấy rằng mô hình này có thể dự đoán chính xác các khách hàng có khả năng mua hàng cao.

Mô hình MultinomialNB có độ chính xác thấp nhất. Điều này có thể là do mô hình này có xu hướng quá phức tạp và dễ bị overfitting.

Tuy nhiên mô hình dự đoán khách hàng tiềm năng khá phụ thuộc vào chất lượng của dữ liệu đầu vào. Để cải thiện độ chính xác có thể thu thập thêm dữ liệu đầu vào và sử dụng nhiều mô hình khác nhau và so sánh để giảm thiểu rủi ro độ chính xác thấp.

2.5 Ý nghĩa

Qua project này có thể giúp nhóm tiếp xúc được với các cách phân tích sở thích, hành vi của khách hàng. Phân tích đặc điểm các khách hàng để phân thành các nhóm để xử lý. Dựa vào insight đã tìm được ở bước trực quan hóa để tạo được Target xây dựng mô hình dự đoán khách hàng tiềm năng.

Qua đó có thể hiểu thêm được công việc thực tế của một doanh nghiệp khi phân tích về khách hàng của họ. Có thêm kinh nghiệm để có thể áp dụng vào các dự án trong tương lai từ quy trình cho đến cách thức làm việc trong một dự án dữ liệu lớn.

2.6 Đánh giá tổng quan bài toán

Bài toán khách hàng tiềm năng là một bài toán quan trọng trong kinh doanh, đặc biệt là đối với các doanh nghiệp cung cấp dịch vụ. Bài toán này nhằm xác định những khách hàng có khả năng mua hàng cao, từ đó giúp doanh nghiệp tập trung nguồn lực tiếp thị và bán hàng vào những khách hàng này, từ đó tăng hiệu quả kinh doanh.

Có các lợi ích:

- Giúp doanh nghiệp hiểu rõ hơn về nhu cầu hành vi của khách hàng → đưa ra các sản phẩm và dịch vụ phù hợp với nhu cầu của khách hàng → gia tăng sự hài lòng của khách hàng.

- Xác định phương hướng chính xác để tiếp tục giữ chân khách hàng, tập trung tiếp thị và bán hàng vào những khách hàng này.

Tóm lại, bài toán dự đoán khách hàng tiềm năng giúp doanh nghiệp tiết kiệm thời gian và chi phí trong kinh doanh.

2.7 Kết luận và hướng phát triển

Thu thập thêm dữ liệu về khách hàng để đánh giá rõ ràng hơn về các đặc điểm của khách hàng (giới tính, tuổi, thu nhập, …). Ví dụ:

- Về độ tuổi, những người trẻ tuổi có xu hướng tương tác với Facebook, Instagram, TikTok, … Thì khi quảng cáo trên các ứng dụng này về các sản phẩm phù hợp với giới trẻ sẽ phù hợp hơn, …
- Nếu thu nhập cao thì có thể tương tác với khách hàng để đăng ký VIP hoặc nâng cấp gói dịch vụ đang sử dụng, …

Xem xét thêm về reviews của khách hàng để hiểu thêm về cảm nhận của khách hàng thông qua việc xây dựng mô hình phân tích cảm xúc dựa trên reviews của khách hàng.

Ngoài ra có thể xây dựng thêm mô hình đề xuất dựa trên collaborative filtering (CF) để giảm bớt thời gian khách hàng tìm kiếm tăng sự hài lòng của khách hàng.

3. Tài liệu tham khảo

Trong quá trình thực hiện đồ án, nhóm có tham khảo cũng như tra cứu nhiều trên các nguồn online. Do quá nhiều lần tra cứu không thể liệt kê hết ở đây, nhóm chỉ trình bày các nguồn tham khảo chính ở đây.

Stack Overflow, [Online]. https://stackoverflow.com/.

Geeks for Geeks. [Online]. https://www.geeksforgeeks.org.

Meachine cơ bản. [Online]. https://machinelearningcoban.com/ .

RFM analysis for Customer Segmentation. https://clevertap.com/blog/rfm-analysis/ .