**VIETNAM NATIONAL UNIVERSITY HCMC**

**UNIVERSITY OF INFORMATION TECHNOLOGY**

A blue logo with text

Description automatically generated

**PROJECT REPORT**

**TOPIC:**

**Phân loại bình luận đánh giá điện thoại**

**Instructors: Ths. Nguyễn Văn Kiệt**

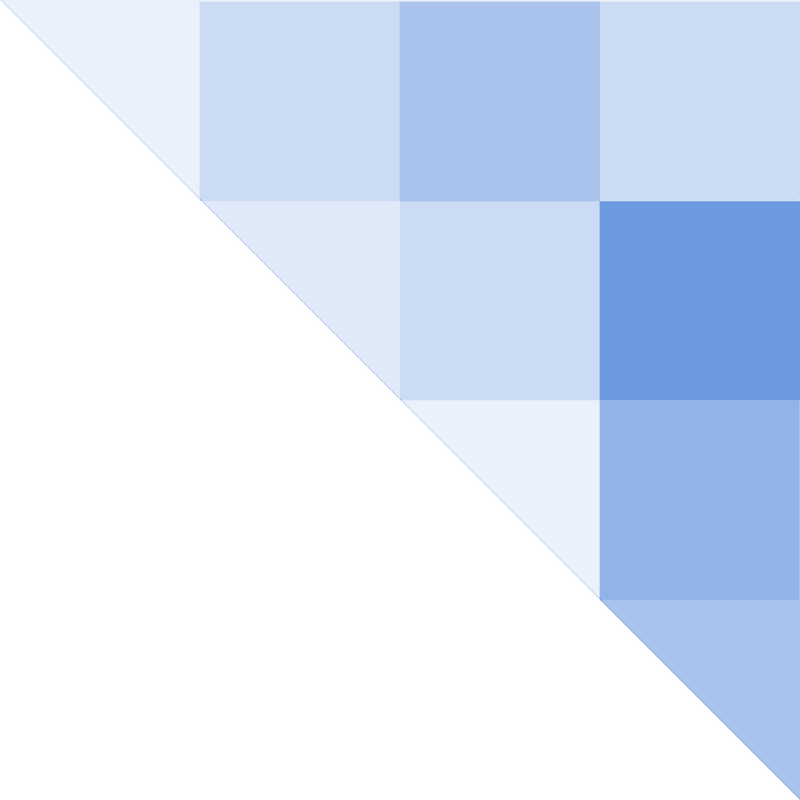
**Team members: Dương Huy Hoàng 21522087**

**Nguyễn Vũ Hoàng Long 21520058**

**Phạm Quốc Anh Khoa 21520296**

**Class: Học máy thống kê - DS102.O11.CNCL**

***Ho Chi Minh City, November 10, 2023***



[**I. Introduction 3**](#_heading=h.8m2fg1doo7bw)

[**II. Related word 3**](#_heading=h.4d34og8)

[1. Sentiment Analysis in Product Reviews: 3](#_heading=h.cji3cdgyy1e0)

[2. Multiclass Sentiment Analysis 3](#_heading=h.2tmg2bvgsoh9)

[3. Natural Language Processing for Reviews 3](#_heading=h.glt4rq3du30z)

[4. Evaluation Metrics for Multiclass Classification 4](#_heading=h.ebbtgvava8hb)

[5. Domain-Specific Sentiment Analysis 4](#_heading=h.tvvuvtpeo5te)

[6. Challenges in Multiclass Sentiment Analysis for Tech Products 4](#_heading=h.5jng43gjskat)

[**III. Dataset 4**](#_heading=h.2s8eyo1)

[1. Mô Tả Tổng Quan về Dataset 4](#_heading=h.xs3kbp4n56h7)

[2. Quy trình 4](#_heading=h.4yvgehwbr4gv)

[3. Chi Tiết Về Tập Huấn Luyện và Tập Kiểm Tra 5](#_heading=h.yyq8pyixfn0d)

[4. Tiền Xử Lý Dữ Liệu 7](#_heading=h.yf10idanl6j3)

[a. Loại bỏ cột không cần thiết 7](#_heading=h.p0iu0l2k00ij)

[b. Tạo thêm cột tương ứng với số lượng các khía cạnh 8](#_heading=h.n5fa7z6lq0sa)

[c. Label encoding 8](#_heading=h.ew65oj3wpw5)

[5. Vector hóa văn bản 9](#_heading=h.udvaqeupqxj0)

[a. Bước 1 9](#_heading=h.li2l30fttfuw)

[b. Bước 2 9](#_heading=h.f75sa3vass24)

[6. Khám phá dữ liệu (Exploratory Data Analysis - EDA) 12](#_heading=h.6ce7c0txf7qe)

[**IV. Method 16**](#_heading=h.17dp8vu)

[1. Experiment 16](#_heading=h.i0q4xtvy551q)

[a. Decision Tree 16](#_heading=h.mw71ubdp29e9)

[b. Softmax Regression 16](#_heading=h.29qgdk23jzka)

[c. SVM - Support vector machine 17](#_heading=h.2kqdk4qp0zhp)

[d. Random Forests 17](#_heading=h.oxziab3q154c)

[2. Result 18](#_heading=h.4597w2x6r2jc)

[**V. Conclusion 19**](#_heading=h.lnxbz9)

[**VI. References 20**](#_heading=h.g6e1i0a5bk3n)

# 

# Introduction

Trong bối cảnh công nghệ ngày càng phát triển nhanh chóng, điện thoại di động đã trở thành một phần không thể thiếu trong cuộc sống hàng ngày của chúng ta, là công cụ mạnh mẽ bao gồm nhiều tính năng. Khi người tiêu dùng đối mặt với sự đa dạng ngày càng tăng về các lựa chọn, nhu cầu về các phương pháp hiệu quả và chính xác để đánh giá và phân loại đánh giá điện thoại di động trở nên quan trọng. Dự án của chúng em tập trung vào việc sử dụng các phương pháp máy học để giải quyết nhiệm vụ phức tạp về phân tích cảm xúc đa lớp (multiclass sentiment analysis) trong ngữ cảnh đánh giá điện thoại di động.

# **Related word**

## **Sentiment Analysis in Product Reviews:**

Phân tích cảm xúc, hay còn gọi là khai thác ý kiến, là một phần quan trọng để hiểu thông tin chủ quan trong các đánh giá sản phẩm. Công việc liên quan này khám phá các kỹ thuật và phương pháp để xác định cảm xúc được thể hiện trong đánh giá của người dùng, tập trung vào nội dung liên quan đến sản phẩm. Nhiều phương pháp, như thuật toán học máy và mô hình học sâu, được thảo luận trong ngữ cảnh của phân tích cảm xúc cho đánh giá sản phẩm.

## **Multiclass Sentiment Analysis**

Phân tích cảm xúc đa lớp mở rộng phân tích cảm xúc nhị phân truyền thống bằng cách phân loại cảm xúc vào nhiều lớp. Công việc liên quan này sâu rộng vào các thách thức và tiến bộ trong việc phân tích cảm xúc có thể rơi vào nhiều hơn hai loại. Nó khám phá cách thuật toán có thể được điều chỉnh hoặc thiết kế để xử lý sự tinh tế của phân tích cảm xúc đa lớp, cung cấp một tổng quan toàn diện về các kỹ thuật tiên tiến trong lĩnh vực này.

## **Natural Language Processing for Reviews**

Các kỹ thuật xử lý ngôn ngữ tự nhiên (NLP) đóng vai trò then chốt trong việc trích xuất thông tin ý nghĩa từ dữ liệu văn bản, đặc biệt là trong ngữ cảnh của đánh giá sản phẩm. Công việc liên quan này điều tra các phương pháp NLP khác nhau được áp dụng vào phân tích các đánh giá, bao gồm phân đoạn, đánh dấu phần từ loại, và nhận dạng thực thể được đặt tên. Trọng tâm là cách NLP cải thiện việc hiểu cảm xúc được thể hiện trong đánh giá và đóng góp vào phân tích cảm xúc chính xác hơn.

## **Evaluation Metrics for Multiclass Classification**

Đánh giá hiệu suất của các mô hình phân tích cảm xúc đa lớp yêu cầu các thước đo chuyên biệt. Công việc liên quan này thảo luận và so sánh các thước đo đánh giá khác nhau được sử dụng để đánh giá hiệu suất của các thuật toán trong việc phân loại cảm xúc qua nhiều danh mục. Các thước đo như độ chính xác, độ chính xác, độ nhớ và điểm F1 được khám phá trong ngữ cảnh của phân tích cảm xúc đa lớp, cung cấp thông tin về ưu điểm và hạn chế của mỗi thước đo.

## **Domain-Specific Sentiment Analysis**

Phân tích cảm xúc được tùy chỉnh cho các miền cụ thể, như sản phẩm công nghệ, mang lại những thách thức độc đáo. Công việc liên quan này đào sâu vào những phức tạp của phân tích cảm xúc theo miền cụ thể và khám phá cách mô hình có thể được tùy chỉnh hoặc đào tạo để xử lý tốt hơn ngôn ngữ và cảm xúc phổ biến trong ngữ cảnh của đánh giá sản phẩm công nghệ.

## Challenges in Multiclass Sentiment Analysis for Tech Products

Phân tích cảm xúc đa lớp cho các sản phẩm công nghệ đặt ra những thách thức riêng do tính kỹ thuật của nội dung. Công việc liên quan này xác định và thảo luận về các thách thức liên quan đến việc phân tích cảm xúc trong đánh giá của các sản phẩm công nghệ, bao gồm sự hiện diện của ngôn ngữ chuyên ngành, các xu hướng công nghệ phát triển nhanh chóng, và nhu cầu phân loại cảm xúc tinh tế.

# **Dataset**

## **Mô Tả Tổng Quan về Dataset**

Đồ án thực hiện trên bộ dataset UIT-ViSFD (<https://github.com/LuongPhan/UIT-ViSFD>)

UIT-ViSFD thu thập phản hồi bằng văn bản từ khách hàng về điện thoại thông minh trên một trang thương mại điện tử lớn tại Việt Nam. Nhãn của tập dữ liệu là mười khía cạnh (*GENERAL, SCREEN, CAMERA, FEATURES, BATTERY, PERFORMANCE, STORAGE, DESIGN, PRICE, SER&ACC, OTHERS*) và ba lớp (*Positive, Negative và Neutral*)

Tập dữ liệu thành ba bộ:

* Train: 7.786 samples
* Dev: 1.112 samples
* Test: 2.224 samples

## **Quy trình**

* Phân chia dataset (đã được chia sẵn)
* Tiến hành tiền xử lý cơ bản cho dataset thô
* Loại bỏ feature không cần thiết
* Loại bỏ missing value
* Định dạng lại label và label encoding
* Standardization

Sau các bước tiền xử lý trên, nhóm tiến hành song song hai quá trình khác nhau để đánh gia hiệu suất.

***Quá trình 1:*** Chỉ áp dụng những bước tiền xử lý cơ bản như trên.

***Quá trình 2***: Tiến hành thêm các bước tiển xử lý:

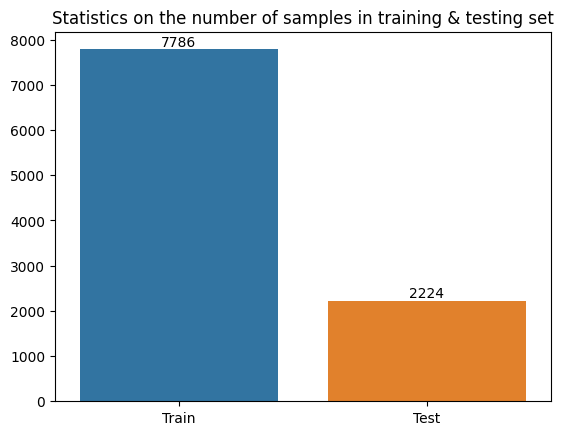
* Loại bỏ những dấu câu (punctuation removal). Các ký tự dấu câu bao gồm những ký tự như dấu chấm (.), dấu phẩy (,), dấu chấm phẩy (;), dấu chấm than (!), dấu hỏi (?), ngoặc đơn (' '), ngoặc kép (" "), và các ký tự dấu câu khác.
* Loại bỏ những ký tự do người bình luận cố tình viết dài
* Loại bỏ stopwords

Sau hai quá trình tiển xử lý khác nhau trên cùng một dữ dữ liệu thô, thực hiện tiếp tục những bước sau:

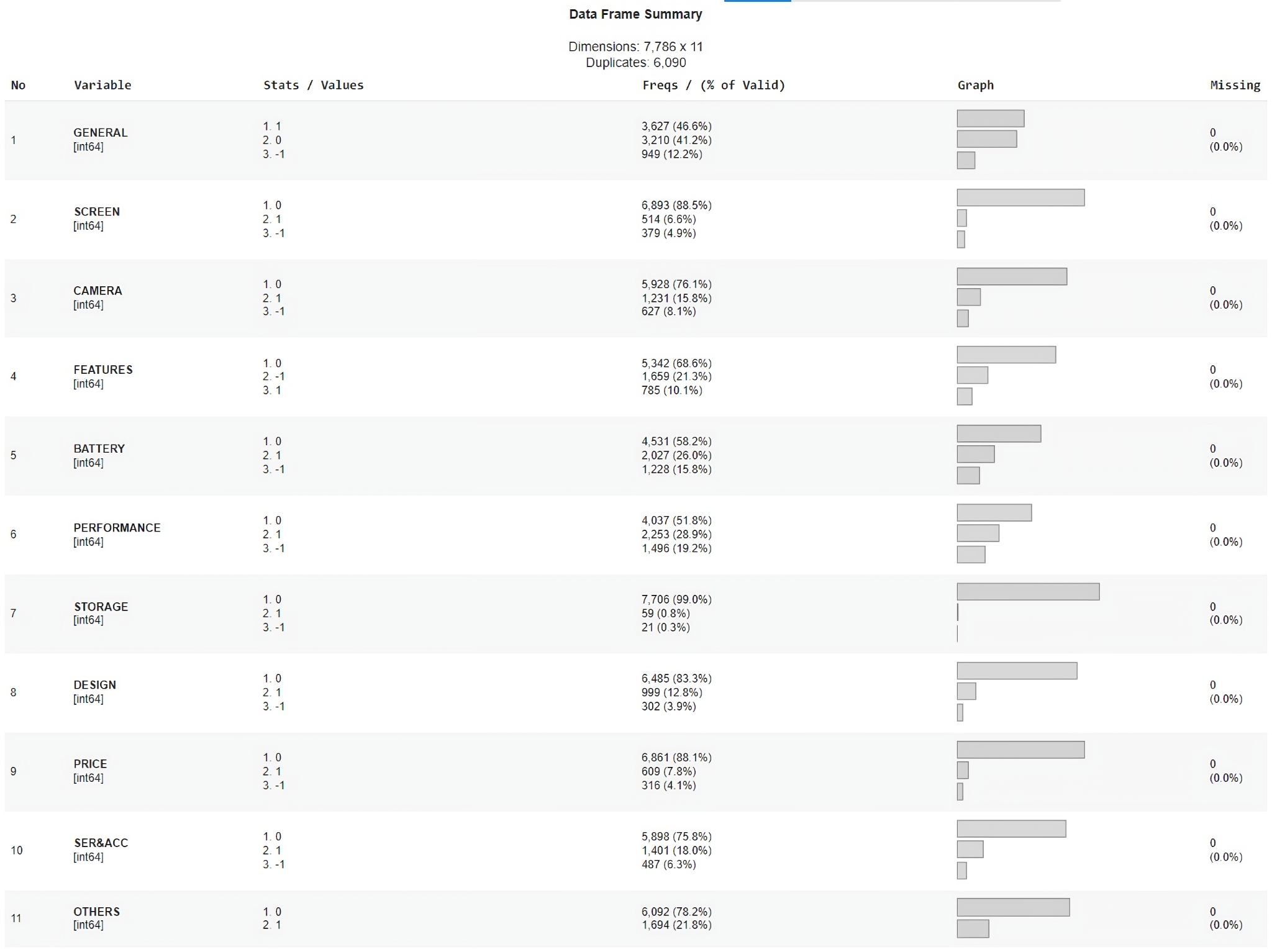
* Khám phá dữ liệu (Exploratory Data Analysis – EDA)
* Khám phá mức độ mất cân bằng của dữ liệu dữ liệu
* Khám phá tần suất xuất hiện của từ
* Thử nghiệm phương pháp giảm chiều dữ liệu (Principal Component Analysis - PCA) để khám phá sự phân tán của dữ liệu
* Tiến hành phân tích ngữ pháp câu bằng POS Tag sử dụng VNCoreNLP
* Vector hóa văn bản
* Tách từ (tokenization)
* Vector hóa văn bản (vectorization)
* Huấn luyện mô hình
* Lựa chọn các mô hình phù hợp để huấn luyện
* Đánh giá mô hình
* Lựa chọn thang đo để đánh giá mô hình
* Nhận xét, so sánh hiệu suất giữa các mô hình, giữa hai quá trình tiền xử lý khác nhau

## **Chi Tiết Về Tập Huấn Luyện và Tập Kiểm Tra**

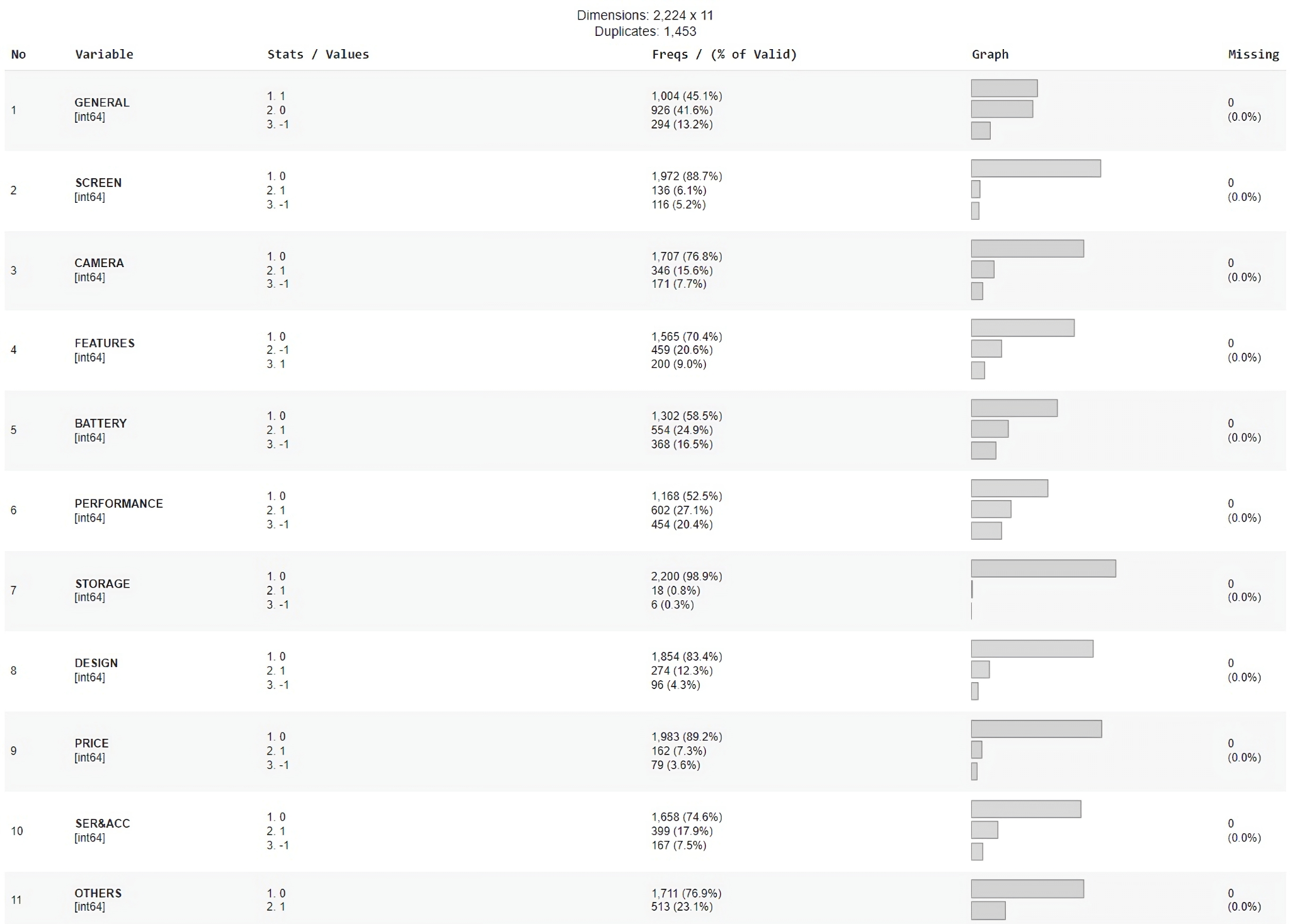
Trong đồ án này, nhóm chỉ chú trọng vào tập train và tập test. Dưới đây là biểu đồ minh họa số lượng sample ở mỗi tập.



* Về tập train có sự phân bổ các lớp như sau:



* Về tập test có sự phân bổ các lớp như sau:



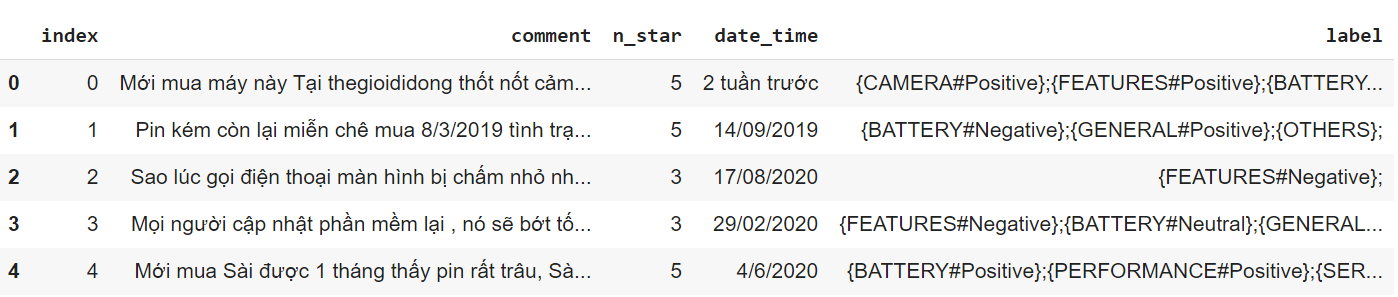
**Nhận xét**

* Một số khía cạnh bị mất cân bằng về số lượng lớp.
* Ngoài ra, dựa vào biểu đồ có thể thấy rằng, không có dòng nào chưa giá trị NaN bị loại bỏ. Số lượng sample ở tập train và tập test vẫn được giữ nguyên.

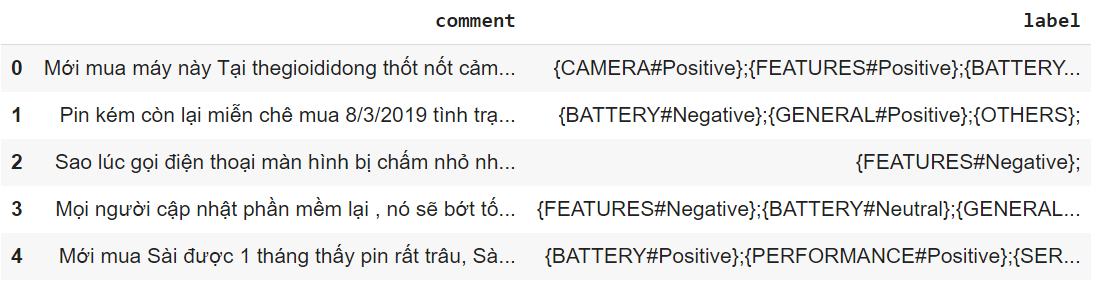
## **Tiền Xử Lý Dữ Liệu**

### Loại bỏ cột không cần thiết

Hình dưới đây là 5 dòng đầu tiên trích từ bộ dataset ban đầu. Có thể thấy một số cột là không cần thiết cũng như label cũng cần được định dạng lại để dễ dàng xử lý ở nhưng bước kế tiếp.



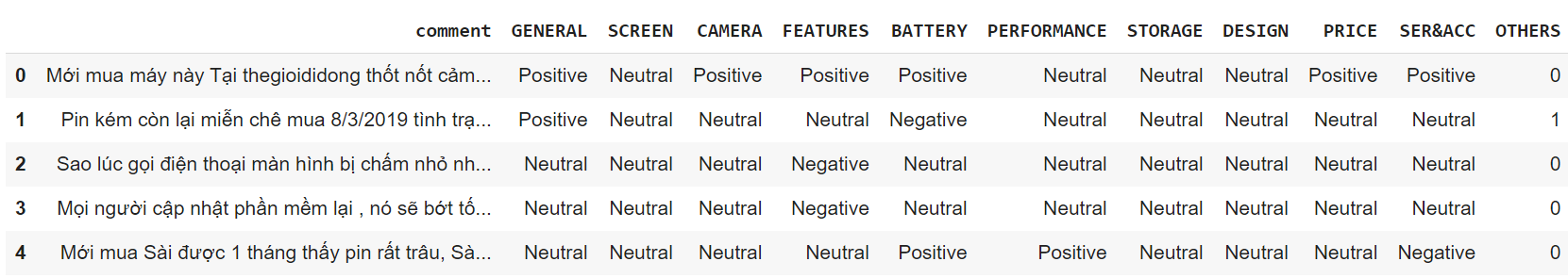
Trong đồ án này, nhóm chú trọng phân tích cảm xúc phần bình luận dưới dạng text, không bao gồm số sao đánh giá sản phẩm và thông tin ngày một bình luận được người dùng đăng lên các trang thương mại điện tử. Do đó cột **n\_star, date\_time**, **index** cần được loại bỏ. Kết quả thu được như sau:



### Tách label thành từng cột riêng

Sau khi loại bỏ những cột không cần thiết, nhóm tiến hành xử lý label ở dataset gốc. Label gốc có dạng là chuỗi các khía cạnh và lớp tương ứng nối tiếp nhau dưới dạng string rất khó nhìn và khó thao tác với trường dữ liệu này. Do đó, nhóm sẽ tách các nhãn thành các cột riêng như hình bên dưới.

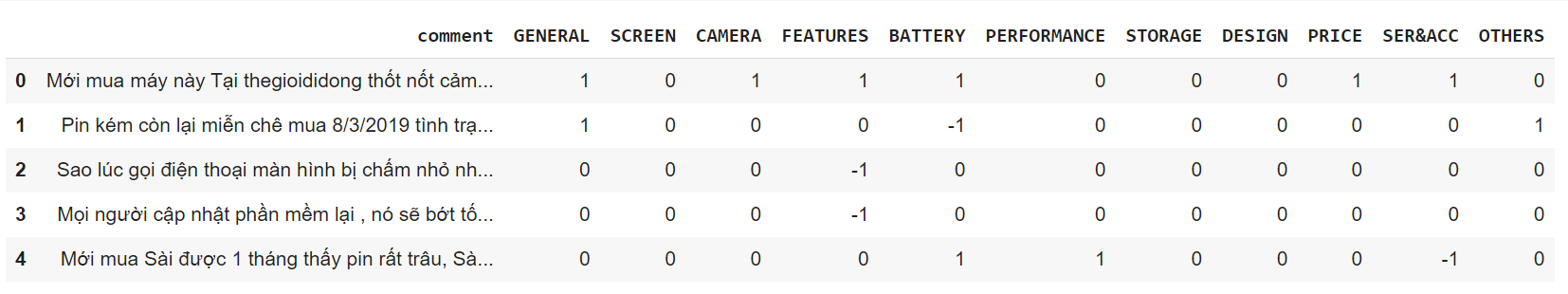
Ngoài ra, những bình luận không bao giờ để cập đến tất cả khía cạnh đang xét. Do đó, ta quy ước, nếu không có nhắc đến khía cạnh nào thì khía cạnh đó được gán lớp Neutral



Lưu ý: cột từ khóa ‘OTHERS’ không phải lúc nào cũng được nhắc đến trong bộ dataset ban đầu. Do đó, nhưng bình luận nào không nhắc đến nhưng vấn đề thuộc khía cạnh ‘OTHERS’ sẽ được gán nhãn là ‘0’

### Label encoding

Phần lớn các nhãn có 3 lớp là ‘Positive’, ‘Negative’ và ‘Neutral’ đều ở dạng chữ, vì vậy nhóm đã sử dụng kỹ thuật label encoding để chuyển ‘Positive’ thành ‘1’, ‘Neutral’ thành ‘0’ và ‘Negative’ thành ‘-1’ và thu được kết quả như sau:

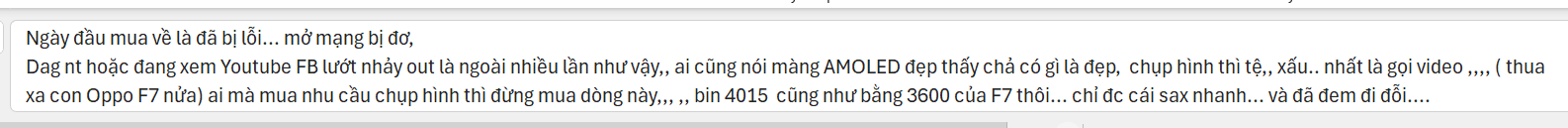


1. **Tiền xử lý riêng đối với quá trình 2**

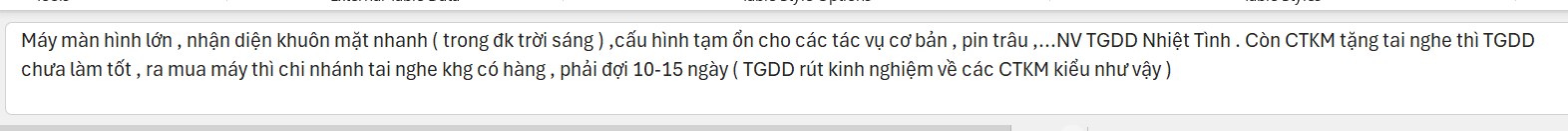
**d.1. Loại bỏ dấu câu (punctuation removal)**

*(Mục này nhóm chỉ trình bày vấn đề. Chi tiết quá trình xử lý ở phần sau)*

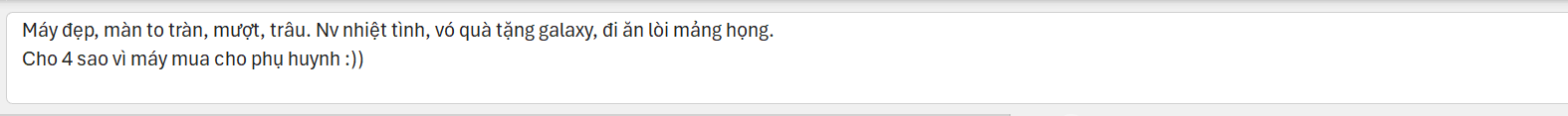
Do vô tình hoặc cố ý, một số bình luận có rất nhiều dấu câu (dấu phẩy, dấu chấm) liền kê nhau.



Ngoài chủ ý cá nhân, những dấu câu còn được sử dụng như một phần của ngữ pháp. Một số bình luận dùng dùng dấu ngoặc đơn () để chú thích và dấu 3 chấm, 4 chấm, hoặc vừa phẩy vừa 3 chấm ",...." thay cho “vân vân”.



Ngoài ra, một số bình luận sử dụng dấu câu như “:)))” hoặc “^^” để thay cho việc sử dụng emoji



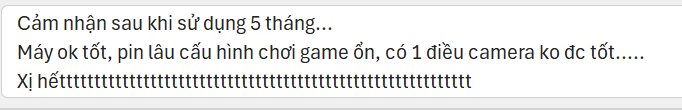
Một số đánh giá số sao ngay trong phần bình luận vừa số vừa chữ, vừa bằng dấu '\*'.

Ví dụ: **5 sao** và **5\*** là tương tự nhau.

**d.2. Loại bỏ những ký tự do người bình luận vô ý hoặc cố tình viết dài**

Một số bình luận cố tình viết dài như một cách để nhấn mạnh ý kiến cá nhân, tuy nhiên, do những ký tự viết dài có thể có cùng xuất phát từ một từ nhưng độ dài khác nhau, gây ra sự nhầm lẫn cho model.

Do đó để tránh tình trạng trên cũng như để thống nhất và chuẩn hóa văn bản, những trường hợp bên dưới cần được xử lý. *(Chi tiết quá trình xử lý ở phần sau)*

****

**d.3. Loại bỏ stopwords**

Quá trình loại bỏ stopwords là một phần quan trọng trong tiền xử lý dữ liệu văn bản tiếng Việt. Stopwords là những từ phổ biến và không mang lại nhiều ý nghĩa trong quá trình phân tích văn bản, chúng thường xuất hiện ở mức cao trong tần suất xuất hiện và không chứa nhiều thông tin đặc trưng. *(Chi tiết quá trình xử lý ở phần sau)*

Việc loại bỏ từ dừng có một số ưu điểm quan trọng:

* Giảm chiều của dữ liệu: Loại bỏ từ dừng giúp giảm số lượng từ trong văn bản, giảm chiều của dữ liệu và làm cho các vector đặc trưng trở nên nhẹ nhàng hơn, đồng thời cải thiện tốc độ xử lý.
* Tăng khả năng tập trung vào ý nghĩa: Bằng cách loại bỏ những từ không quan trọng như từ dừng, mô hình có thể tập trung vào những từ quan trọng hơn, giúp định rõ ý nghĩa và ngữ cảnh của văn bản.
* Phòng tránh hiện tượng quá trọng số: Từ dừng thường xuất hiện ở nhiều văn bản và không mang lại nhiều ý nghĩa đặc trưng, việc giữ lại chúng có thể dẫn đến hiện tượng quá trọng số khi xây dựng mô hình.
* Chuẩn hóa dữ liệu: Loại bỏ từ dừng giúp chuẩn hóa văn bản, giúp việc so sánh và xử lý dữ liệu trở nên thuận tiện hơn.

**d.4. Chi tiết quá trình xử lý cho mục d này.**

Ở đây, nhóm trình một vài dòng trong bộ dataset làm ví dụ minh họa tiền xử lý riêng đối với quá trình 2

***Đây là những văn bản thô***

Row 01: Pin kém còn lại miễn chê mua 8/3/2019 tình trạng pin còn 88% có ai giống tôi không

Row 16: Dùng ok.. Mượt k nóng..( đt có 5 6 triệu mà mở max cấu hình ròi kêu nóng) Nhân viên nhiệt tình..

Row 16: Cảm nhận sau khi sử dụng 5 tháng...

Máy ok tốt, pin lâu cấu hình chơi game ổn, có 1 điều camera ko đc tốt.....

Xị hếttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt

***Sau khi loại bỏ dấu câu***

Row 01: Pin kém còn lại miễn chê mua 832019 tình trạng pin còn 88 có ai giống tôi không

Row 16: Dùng ok Mượt k nóng đt có 5 6 triệu mà mở max cấu hình ròi kêu nóng Nhân viên nhiệt tình

Row 16: Cảm nhận sau khi sử dụng 5 tháng

Máy ok tốt pin lâu cấu hình chơi game ổn có 1 điều camera ko đc tốt

Xị hếttttttttttttttttttttttttttttttttttttttttttttttttttttttttttt

***Sau khi loại bỏ những ký tự viết dài***

Row 16: Cảm nhận sau khi sử dụng 5 tháng

Máy ok tốt pin lâu cấu hình chơi game ổn có 1 điều camera ko đc tốt

Xị hết

***Trước khi loại bỏ stopwords*** *(5 dòng dầu tiên)*

['Mới mua máy này Tại thegioididong thốt nốt cảm thấy ok bin trâu chụp ảnh đẹp loa nghe to bắt wf khỏe sóng ổn định giá thành vừa với túi tiền nhân viên tư vấn nhiệt tình',

'Pin kém còn lại miễn chê mua 832019 tình trạng pin còn 8 có ai giống tôi không',

'Sao lúc gọi điện thoại màn hình bị chấm nhỏ nháy gần camera trước vậylúc có lúc không',

'Mọi người cập nhật phần mềm lại nó sẽ bớt tốn pin mình đã thử rồi mọi thứ cũng ok nhưng vân tay ko nhạy',

'Mới mua Sài được 1 tháng thấy pin rất trâu Sài bao mượt Nhưng có 1 lỗi nhỏ là mình nghe nhạc bằng tai nghe nghe hơi lâu ko biết sao nó ko nghe được nữa mất rút tai nghe ra cắm vào lại thì nó mới nghe được']

***Sau khi loại bỏ stopwords*** *(5 dòng dầu tiên)*

['mua máy thegioididong nốt cảm ok bin trâu chụp ảnh đẹp loa to bắt wf khỏe sóng ổn định giá thành túi tiền nhân viên tư vấn nhiệt tình',

'Pin kém miễn chê mua 832019 tình trạng pin 8',

'gọi điện thoại màn hình chấm nháy camera vậylúc',

'cập nhật mềm bớt tốn pin thử ok vân ko nhạy',

'mua Sài 1 pin trâu Sài bao mượt 1 lỗi nhạc tai hơi ko ko rút tai cắm']

**Nhận xét**

* **Đối với việc loại bỏ dấu câu có một số ưu và nhược điểm sau**

**Ưu điểm**

* **Giảm nhiễu và làm sạch dữ liệu**: Loại bỏ dấu câu giúp làm sạch văn bản, giảm nhiễu và tạo ra một định dạng văn bản đồng nhất.
* **Chuẩn hóa văn bản**: Việc loại bỏ dấu câu giúp chuẩn hóa văn bản, làm cho các bình luận trở nên dễ so sánh và hiểu quả hơn.

**Nhược điểm**

* **Mất mát ý nghĩa và cảm xúc**: gây mất mát về mặt ý nghĩa và cảm xúc ở một số ngữ cảnh của bình luận.
* **Mất đi ngữ cảnh và chú thích**: gây mất mát thông tin quan trọng về ngữ cảnh, chú thích ý kiến, hoặc thể hiện cảm xúc.
* **Thách thức trong phân tích cú pháp**: gây khó khăn trong việc phân tích cú pháp và hiểu đúng cấu trúc ngữ pháp của câu.
* **Đối với việc loại bỏ ký tự viết dài**

Quá trình bỏ ký tự dài giúp chuẩn hóa độ dài của từ, tránh phát sinh thêm những từ không cần thiết vì độ dài khác nhau, từ đó làm cho dữ liệu đồng nhất hơn.

* **Đối với việc loại bỏ stopwords**
* **Giảm chiều của dữ liệu**: loại bỏ stopwords giúp giảm số lượng từ trong văn bản, giảm chiều của dữ liệu, cải thiện tốc độ xử lý.
* **Tăng khả năng tập trung vào ý nghĩa**: loại bỏ những từ không quan trọng như stopwords, model có thể tập trung vào những từ quan trọng hơn, giúp định rõ ý nghĩa và ngữ cảnh của văn bản.
* **Phòng tránh hiện tượng quá trọng số**: stopwords thường xuất hiện ở nhiều văn bản nhưng không mang lại nhiều ý nghĩa đặc trưng, việc giữ lại chúng có thể dẫn đến hiện tượng quá trọng số khi xây dựng mô hình.
* **Chuẩn hóa dữ liệu**: Loại bỏ từ dừng giúp chuẩn hóa văn bản, giúp việc so sánh và xử lý dữ liệu trở nên thuận tiện hơn.

## Vector hóa văn bản

Chia dữ liệu trước rồi tiền xử lý riêng lẻ là một phương pháp hữu ích trong quá trình xây dựng và đánh giá mô hình máy học. Lựa chọn này đặc biệt hữu ích khi chúng ta đối mặt với các tình huống mà dữ liệu không có sẵn từ trước. Do đó để mô phòng tỉnh huống thực tế, nhóm sẽ đi tiển xử lý riêng cho từng tập train và tập test.

Quy trình vector hóa văn bản gồm 2 bước chính

* Bước 1: Tách từ (tokenization)
* Bước 2: Thực hiện vector hóa (vectorization)

### Bước 1

Tách từ (tokenization) là quá trình chia một chuỗi văn bản thành các đơn vị nhỏ hơn gọi là "tokens." Tokens có thể là các từ, con số, ký tự đặc biệt, hoặc bất kỳ đơn vị nào khác mà bạn quyết định sử dụng trong quá trình xử lý ngôn ngữ tự nhiên.

*Ví dụ (trình từ tập dataset):*

*Bình luận gốc: Mọi người cập nhật phần mềm lại, nó sẽ bớt tốn pin*

*Bình luận đã được tách từ: Mọi người cập\_nhật phần\_mềm lại, nó sẽ bớt tốn pin*

Nhận xét: có thể thấy, nếu tách riêng từng từ riêng lẽ ‘cập’ - ‘nhật’ và ‘phần’ - ‘mềm’ sẽ làm mất đi ý nghĩa của chúng khi so sánh với cặp từ hoàn chỉnh mang đầy đủ ý nghĩa ‘cập nhật’ và ‘phần mềm’.

Những từ đã được tách (token) sẽ được biểu diễn từ dưới dạng vector số để có thể sử dụng trong các mô hình máy học.

### Bước 2

TF-IDF là một trong số những phương pháp được sử dụng trong đồ án này.

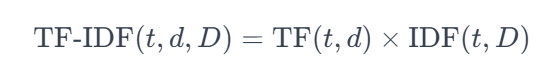
TF-IDF là viết tắt của "Term Frequency-Inverse Document Frequency".

Là một phương pháp đánh giá quan trọng của một từ trong một văn bản so với một tập hợp các văn bản

Term Frequency (TF): Đo lường tần suất xuất hiện của một từ trong một văn bản cụ thể.

Inverse Document Frequency (IDF): Đo lường độ quan trọng của từ đó trong toàn bộ tập các văn bản.

Công thức:

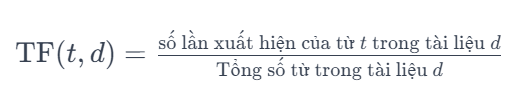


Trong đó,

*d* là từng dòng trong cột comment của bộ dataset

*D* là tập hợp các dòng có trong bộ dataset hay *D =* {*d1, d2, d3, ...*}

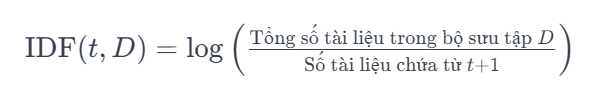
TF(*t*, *d*) là tần suất xuất hiện của từ *t* trong tài liệu *d*, thường được tính bằng cách chia số lần xuất hiện của từ đó cho tổng số từ trong tài liệu *d*.



TF(*t*, *d*) sẽ:

* Tăng khi: Số lần xuất hiện của một từ trong một tài liệu cụ thể tăng lên. Nếu một từ xuất hiện nhiều lần trong tài liệu, giá trị TF của từ đó trong tài liệu sẽ tăng cao.
* Giảm khi: Số lần xuất hiện giảm. Nếu một từ xuất hiện ít hoặc không xuất hiện trong tài liệu, giá trị TF của từ đó sẽ giảm.

IDF(*t*,*D*) là đảo ngược tần suất xuất hiện của từ *t* trong toàn bộ bộ sưu tập các tài liệu *D*. Nó được tính bằng cách lấy logarit tự nhiên của tỷ lệ giữa số tài liệu trong bộ sưu tập và số tài liệu chứa từ *t*.



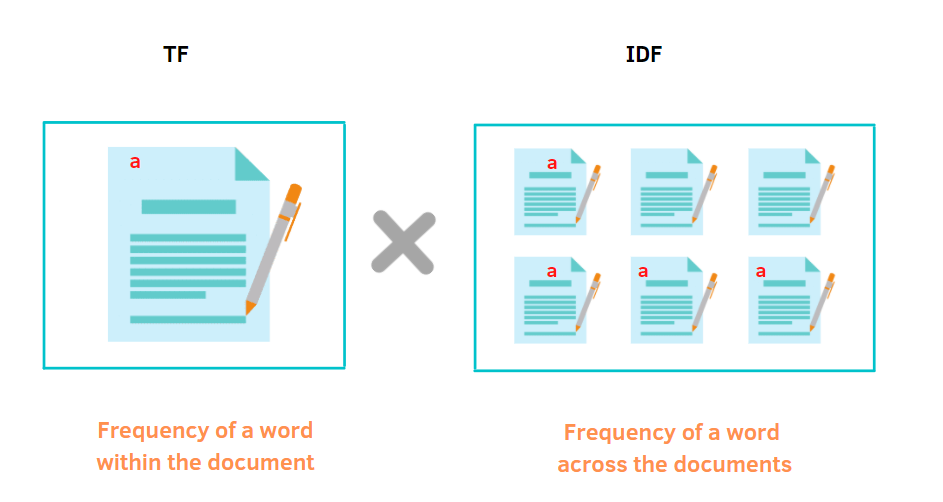
Lưu ý: mẫu số cần cộng thêm 1 để tránh trường hợp mẫu số có thể bằng 0

IDF(*t*,*D*) sẽ:

* Tăng khi: Tỷ lệ giữa tổng số tài liệu trong bộ sưu tập và số tài liệu chứa từ đó giảm. Nếu một từ xuất hiện trong ít tài liệu hơn, giá trị IDF tăng lên, vì nó được coi là độc đáo và quan trọng hơn.
* Giảm khi: Tỷ lệ giữa tổng số tài liệu và số tài liệu chứa từ đó tăng. Nếu một từ xuất hiện trong nhiều tài liệu, giá trị IDF giảm xuống, vì từ đó được coi là phổ biến và ít quan trọng hơn.

Một từ được xem là kém quan trọng hoặc quan trọng sẽ có giá trị TF và IDF khác nhau:

* Thế nào là từ kém quan trọng?
  + TF (Term Frequency): Giá trị TF của một từ kém quan trọng thường thấp, vì nó xuất hiện ít hoặc không xuất hiện nhiều trong tài liệu cụ thể. Nếu một từ xuất hiện ít, giá trị TF giảm.
  + IDF (Inverse Document Frequency): Giá trị IDF của một từ kém quan trọng thường thấp, vì nó xuất hiện trong nhiều tài liệu. Nếu một từ xuất hiện nhiều, giá trị IDF giảm.
* Thế nào là từ quan trọng?
  + TF (Term Frequency): Giá trị TF của một từ quan trọng thường cao, vì nó xuất hiện nhiều lần trong tài liệu cụ thể. Nếu một từ xuất hiện nhiều, giá trị TF tăng.
  + IDF (Inverse Document Frequency): Giá trị IDF của một từ quan trọng thường cao, vì nó xuất hiện ít trong các tài liệu khác. Nếu một từ xuất hiện ít trong bộ sưu tập, giá trị IDF tăng.



Ảnh minh họa công thức TF-IDF

Như đã nhắc đến, để mô phỏng tình huống thực tế. Cần thực hiện các bước hiện như sau:

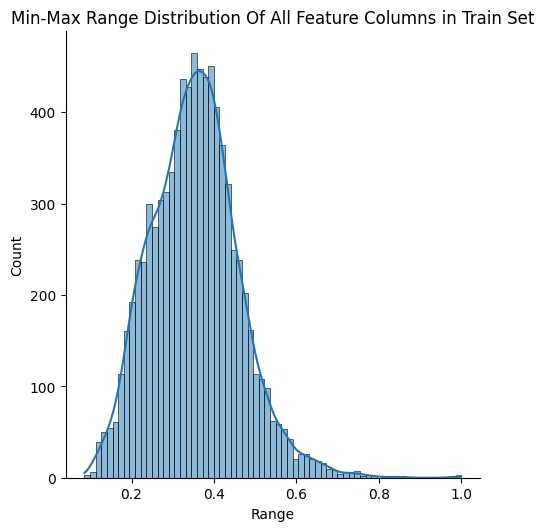
* TfidfVectorizer được sử dụng để tính toán ma trận TF-IDF cho tập huấn luyện bằng hàm fit\_transform. Ma trận này sẽ được sử dụng để huấn luyện mô hình.
* Sau đó, ma trận TF-IDF được áp dụng cho tập kiểm thử bằng cách sử dụng hàm transform. (chi tiết được trình bày ngay bên dưới)
* Vấn đề OOV (Out-of-Vocabulary) được giải quyết bằng cách kiểm tra sự khác biệt giữa từ vựng của tập kiểm thử và tập huấn luyện. Các từ OOV có thể được xử lý một cách thích hợp, ví dụ như thêm các giá trị mặc định hoặc loại bỏ chúng khỏi ma trận TF-IDF.

Đối với trường hợp này, nhóm **điền giá trị mặc định ‘0’** cho những từ trong tập test không có trong tập train

**Nhận xét và giải thích**

Lí do vì sao sử dụng transform thay vì fit\_transform là vì fit\_transform sẽ lập từ điển các từ có trong tập train rồi dựa vào đó tính trọng số TF-IDF cho từng từ. Ngược lại khi sử dụng transform, thuật toán dựa vào từ điển đã có trước đó từ fit\_transform trên tập train, từ đó mới tính trọng số TF-IDF cho từng từ trong tập test

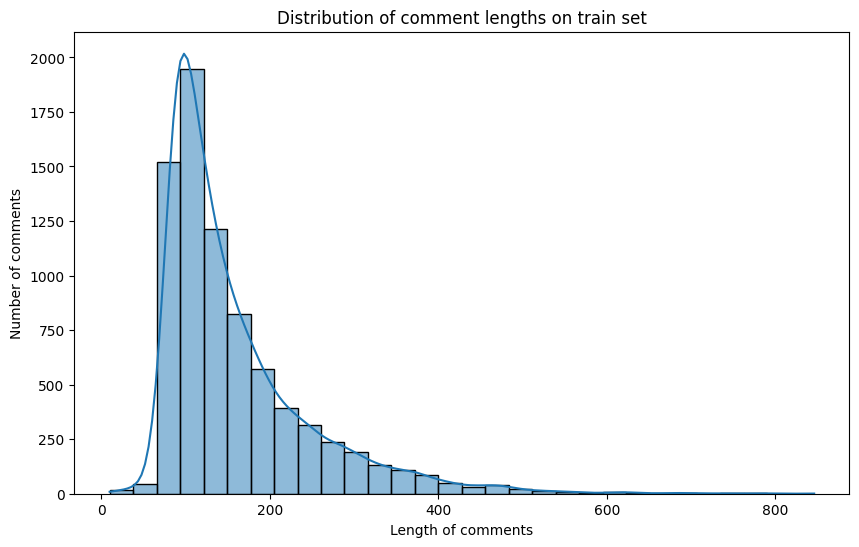
Dưới đây là biểu đồ histogram của giá trị TF-IDF min - max của từng cột tập train.

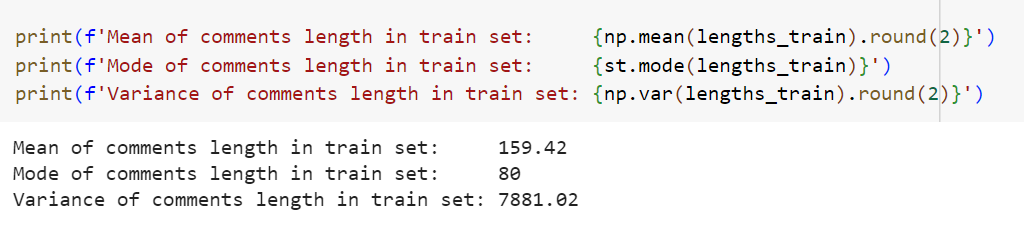


Vì khoảng chênh lệch không lớn (dưới hoặc bằng 1), do đó nhóm nhận thấy feature scaling không thật sự cần thiết.

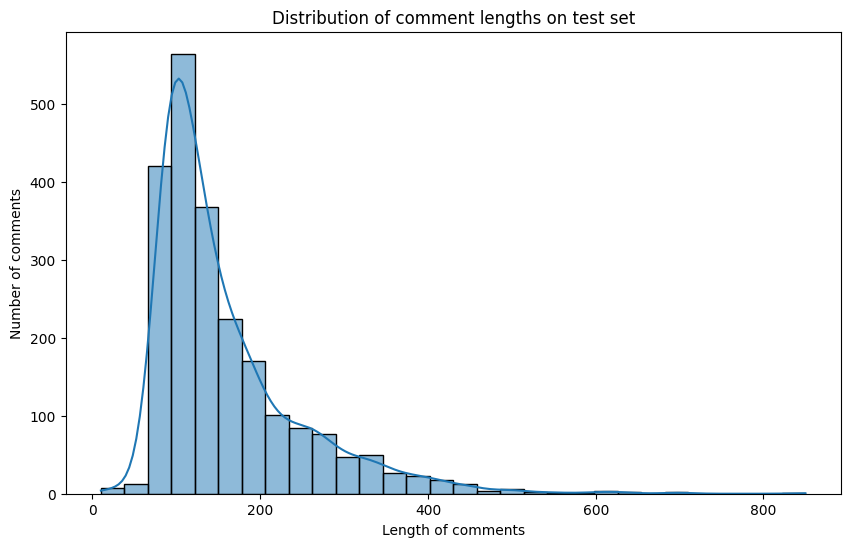
## **Khám phá dữ liệu (**Exploratory **Data Analysis - EDA)**

Độ dài câu trung bình trong tập train





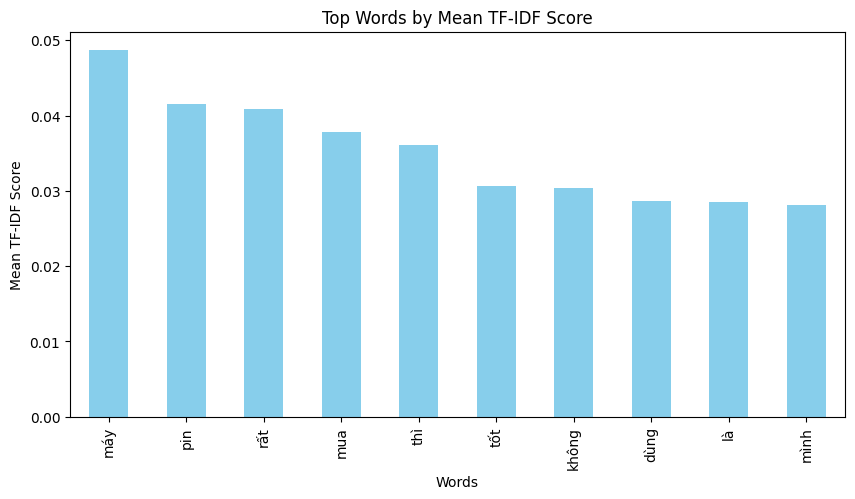
Độ dài câu trung bình trong tập test





**Đối với quá trình 1**

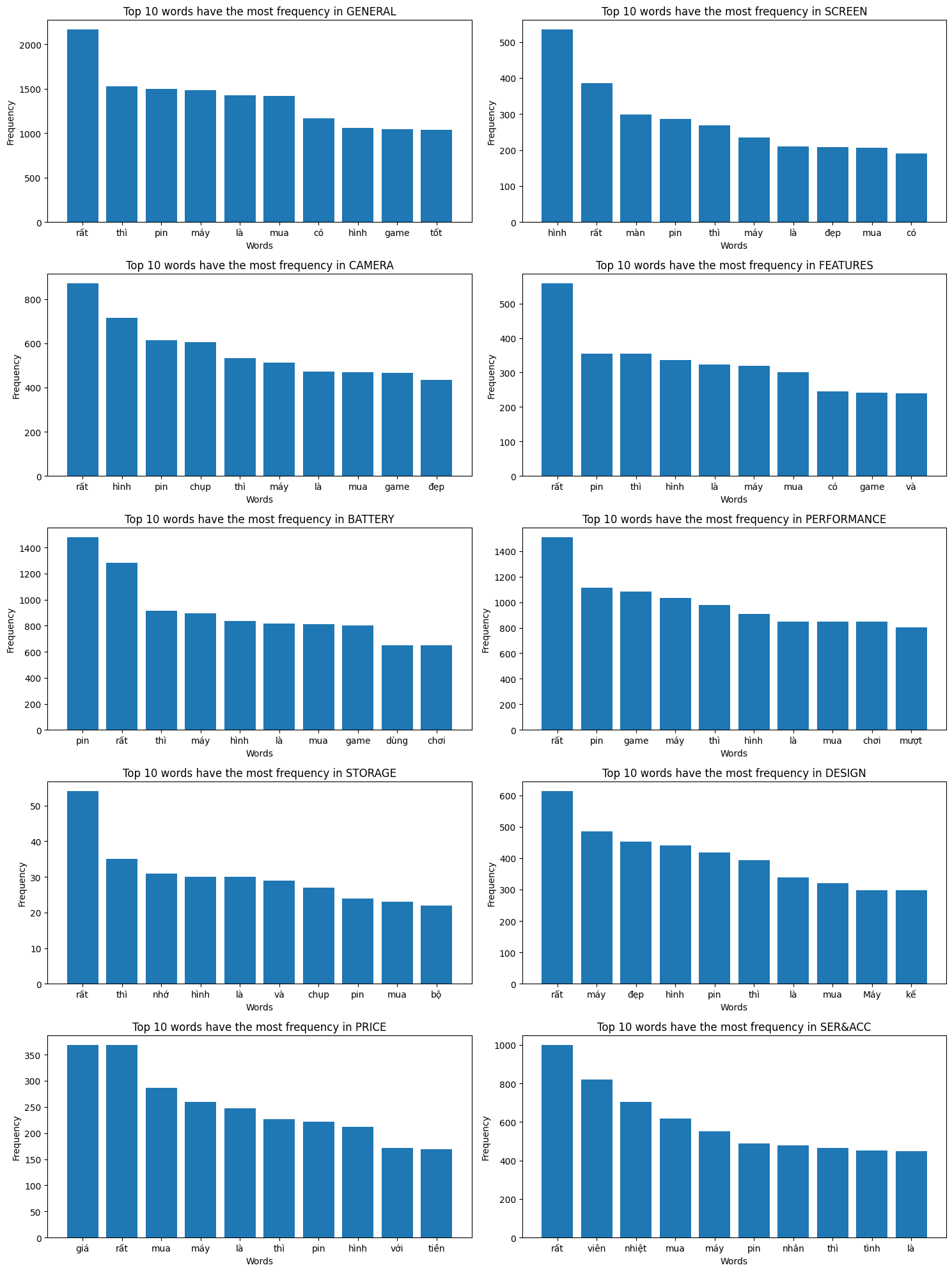
Những từ có giá trị TF-IDF trung bình cao nhất



Hình 6.1

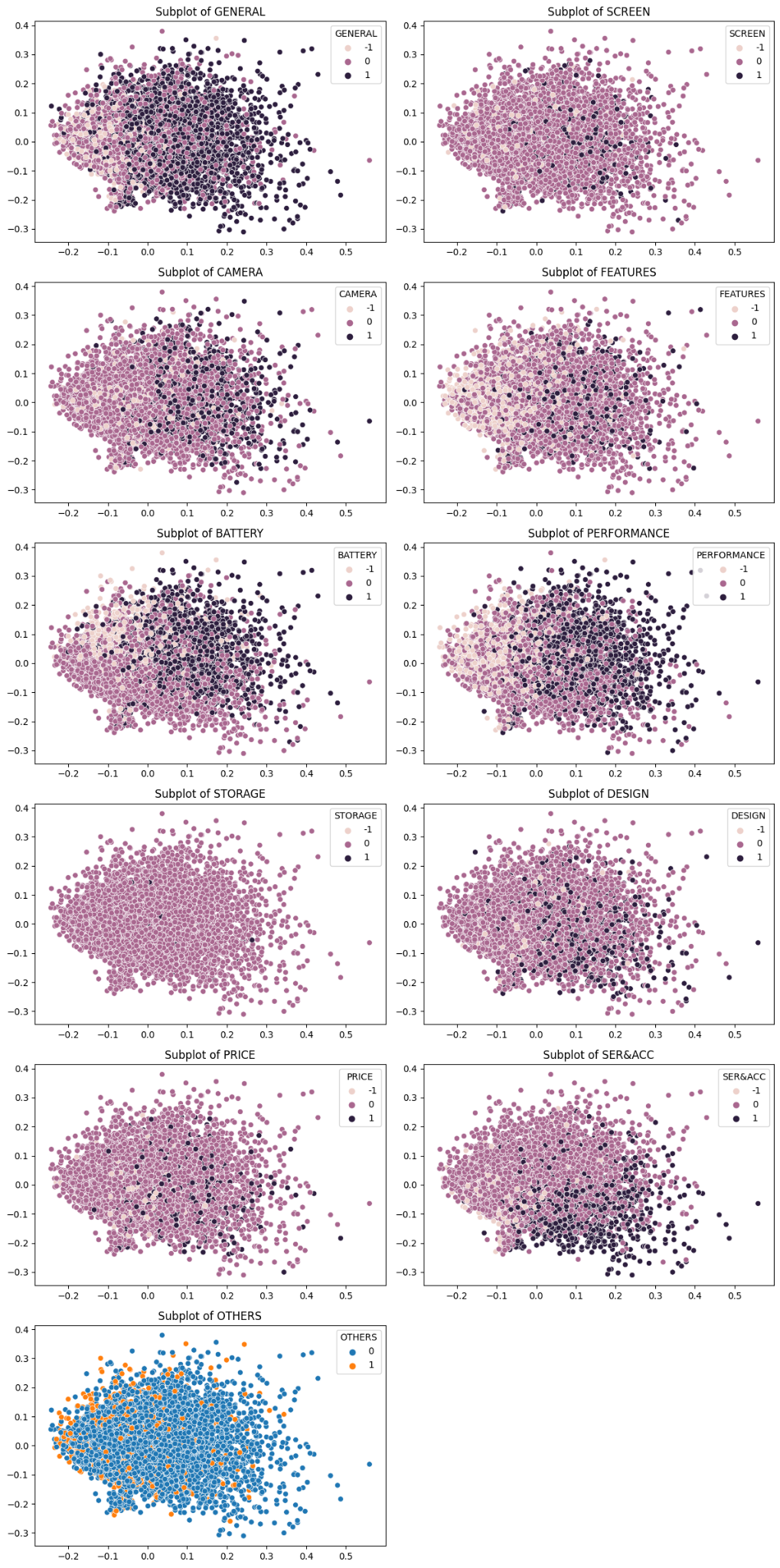
Thống kê những từ có tần suất xuất hiện nhiều nhất ở các label có class là Positive

(xem thêm biểu đồ cho các nhãn Negative và Neutral ở phần notebook)



Hình 6.2

Ngoài ra nhóm cũng tiến hành thực nghiệm giảm chiều dữ liệu.



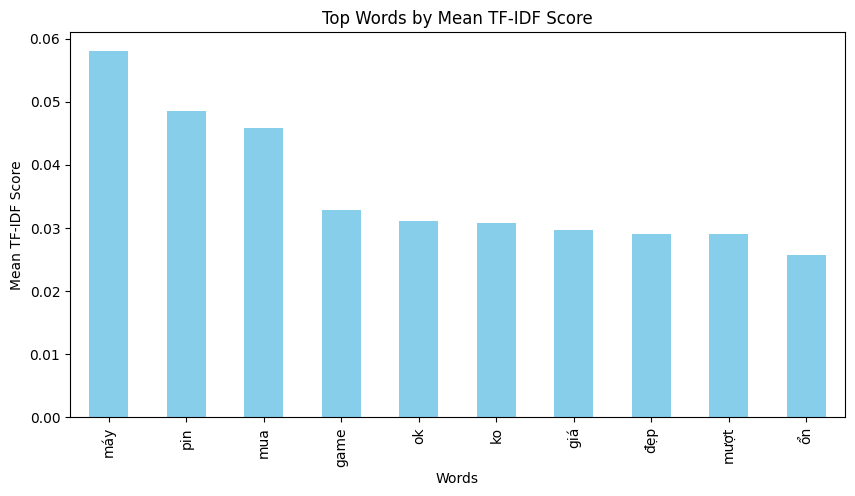
Hình 6.3

**Nhận xét**

* Đối với quá trình 1, vì không thực hiện loại bỏ stopwords nên có thể thấy những stopwords như từ “thì” , “là” lại là những từ có trọng số TF-IDF cao (hình 6.1) cũng như là nhửng từ có tần suất xuất hiện cao (hình 6.2)
* Từ thực nghiệm giảm chiều dữ liệu, ta nhận thấy các điểm dữ liệu có phân phối không tuyến tính. Dựa vào đây, ta có thể làm cơ sở để lựa chọn các thuật toán phân lớp cho mô hình

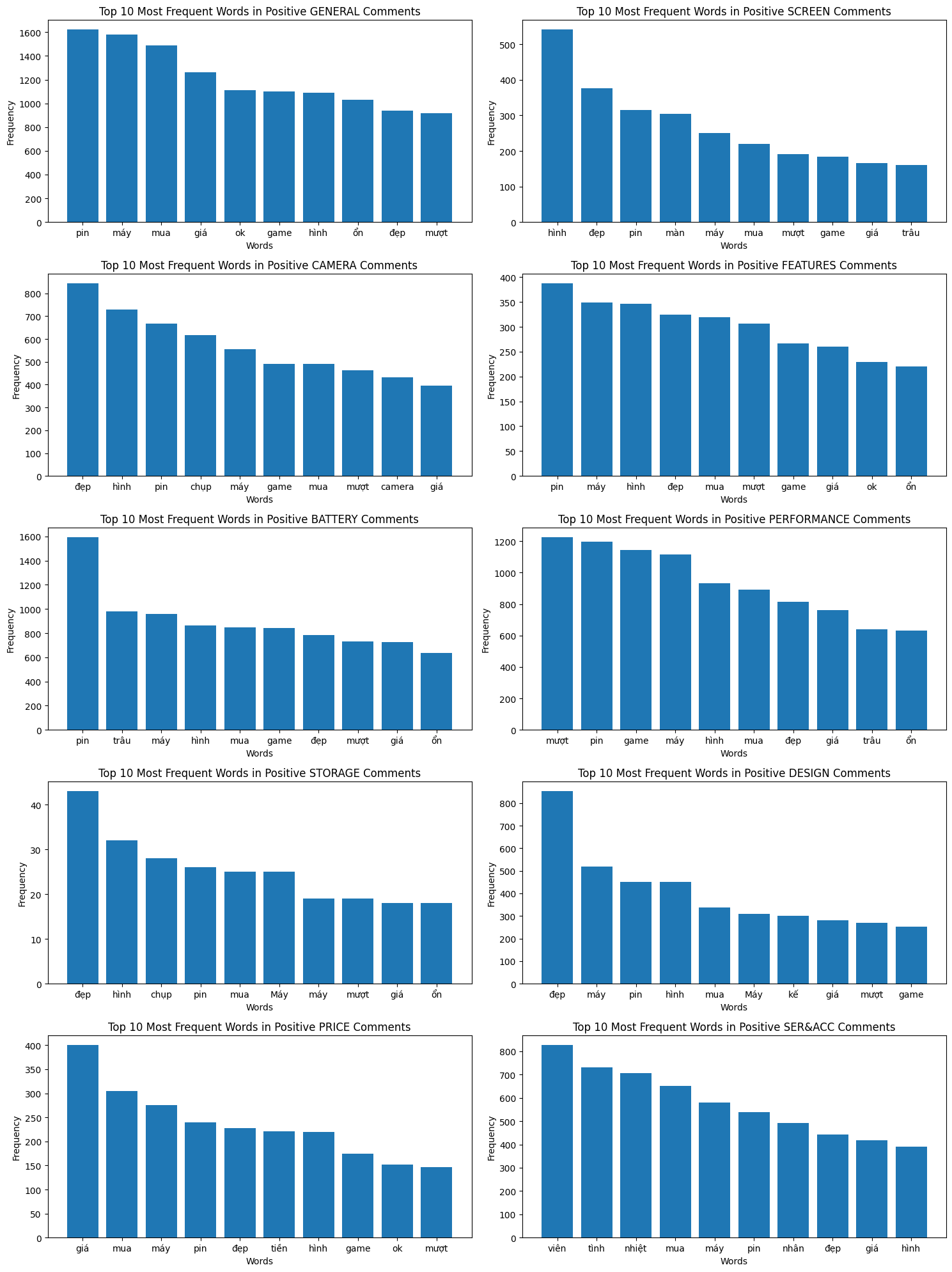
**Đối với quá trình 2**

Những từ có giá trị TF-IDF trung bình cao nhất

**** Hình 6.4

Thống kê những từ có tần suất xuất hiện nhiều nhất ở các label có class là Positive

(xem thêm biểu đồ cho các nhãn Negative và Neutral ở phần notebook)



Hình 6.5

Thực nghiệm giảm chiều dữ liệu cho quá trình 2



Hình 6.6

**Nhận xét**

* Sau khi loại bỏ stopwords trong quá trình thứ hai, ta có thể nhận thấy rằng những từ stopwords đã được loại bỏ, thay vào đó, các từ thuộc loại danh từ và tính từ, mang nhiều ý nghĩa hơn trong việc đánh giá sản phẩm, giờ đây chiếm giá trị TF-IDF cao (hình 6.3) và xuất hiện với tần suất cao hơn (hình 6.4)
* Từ thực nghiệm giảm chiều dữ liệu, ta nhận thấy các điểm dữ liệu có phân phối không tuyến tính. Dựa vào đây, ta có thể làm cơ sở để lựa chọn các thuật toán phân lớp cho mô hình

# **Method**

## Experiment

Đối với bài toán của nhóm, có thể xác định rằng đây là bài toán multioutput vì đầu ra gồm ra là dự đoán một bình luận thuộc lớp *(Positive, Negative, Neutral)* nào ở từng target (*GENERAL, SCREEN, CAMERA, FEATURES, BATTERY, PERFORMANCE, STORAGE, DESIGN, PRICE, SER&ACC, OTHERS)*

Do đó, nhóm sử dụng một phương của thư viện sklearn có tên là **MultiOutputClassifier**

Cách tiếp cận này bao gồm việc điều chỉnh mỗi bộ phân loại (classifier) cho từng target riêng biệt. Đây cách đơn giản để mở rộng các bộ phân loại vốn không hỗ trợ phân loại đa mục tiêu.

**Giải thích tham số của MultiOutputClassifier**

* **estimator**: Đây là bộ phân lớp cơ bản (base classifier) sở được sử dụng để dự đoán từng target. Bộ phân lớp cơ bản này phải hỗ trợ phân loại nhị phân. *(chi tiết bộ phân lớp cơ bản ở phần sau)*
* **n\_jobs**: Cho phép thực hiện công việc song song. Nếu giá trị là -1, thì tất cả tiến trình/luồng có sẵn sẽ được sử dụng

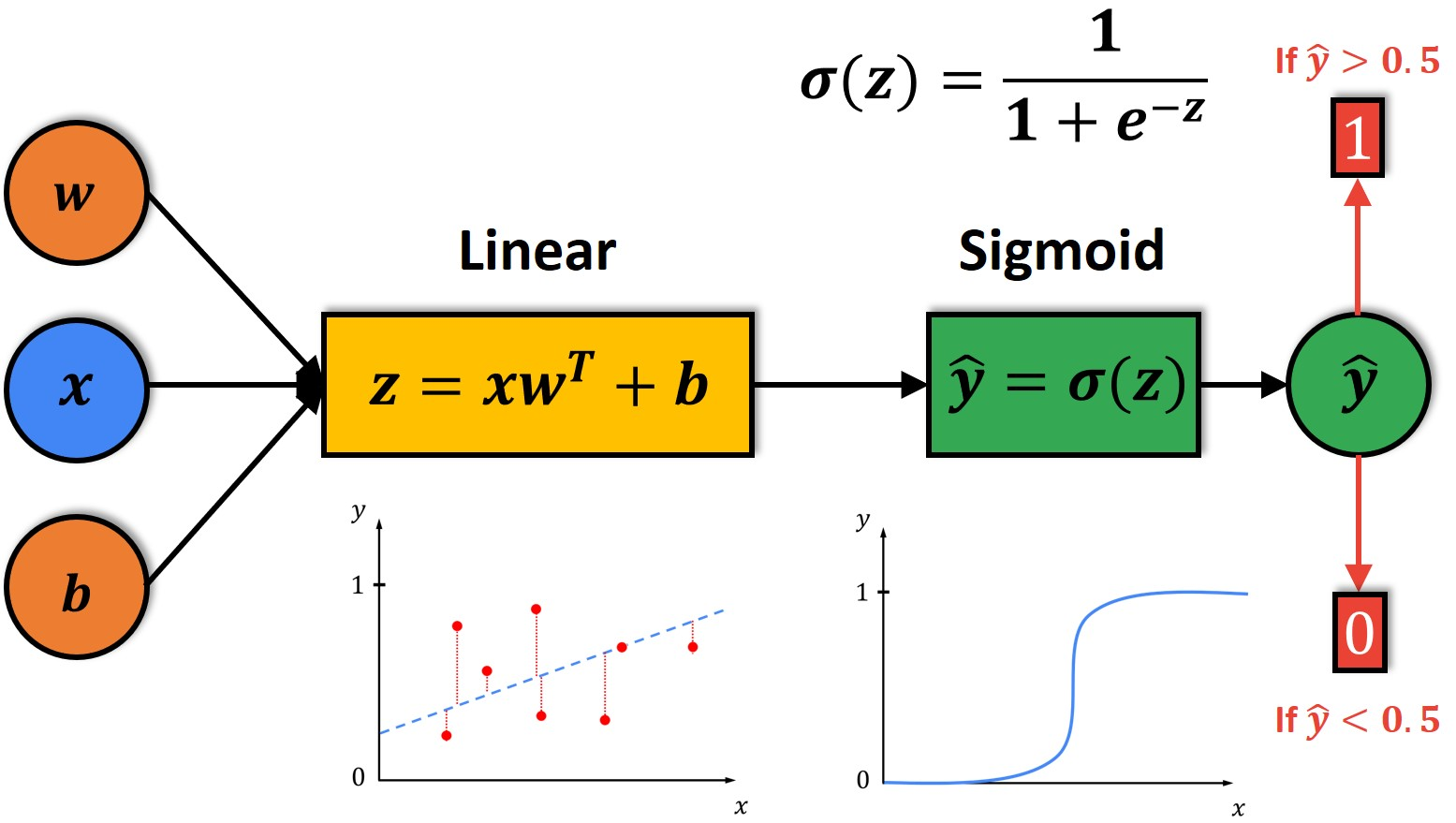
Trong đồ án này, nhóm sử dụng 3 models sau **Logistic Regression**, **Softmax** và **SVM**

Mô hình khác nhau có những tham số khác nhau cần được tùy chỉnh khác nhau. *(Chi tiết được trình bày ở phần sau)*

### Logistic Regression

Logistic Regression là một mô hình phân loại nhị phân được sử dụng khi chỉ có hai lớp cần được dự đoán. Tuy nhiên, nó cũng có thể được mở rộng để giải quyết bài toán phân loại đa lớp (multiclass) thông qua phương pháp OvR - one-vs-rest. Mô hình Logistic Regression của sklearn có tham số **multi\_class** và **solver** cần được tùy chỉnh để phù hợp với bài toán.

Đối với bài toán phân loại 3 lớp của nhóm, cài đặt multi\_class='ovr' nghĩa là mô hình sẽ được huấn luyện để phân biệt mỗi lớp với tất cả các lớp còn lại. Đầu ra của hàm logistic (hoặc hàm sigmoid) là một giá trị xác suất thể hiện xác suất của mẫu thuộc về lớp positive. Lớp với giá trị hàm quyết định cao nhất sẽ được chọn là lớp dự đoán của một sample nào đó.



Ảnh minh họa: công thức hàm sigmoid của Logistic Regression

**Giải thích tham số multi\_class = “ovr”, solver = “sag”**

* Mỗi bài toán nhị phân này sẽ được giải quyết độc lập và cho ra một giá trị xác suất cho mỗi lớp. Điều này có nghĩa là nhóm có 3 lớp, mô hình sẽ xây dựng 3 model cho 3 bài toán phân loại nhị phân và dự đoán nhãn dựa trên kết quả của những bài toán này, ngay cả khi dữ liệu chỉ có hai lớp (cột OTHERS chỉ có hai lớp 1 và 0). Lí do nhóm chọn tham số như vậy là để so sánh với mô hình softmax ở phần sau.
* Khi dự đoán, mô hình sẽ so sánh các giá trị xác suất từ các bài toán nhị phân và chọn lớp có giá trị xác suất cao nhất làm lớp dự đoán cho một sample nào đó.
* Các giá trị của solver “liblinear” và “sag” đều cũng hỗ trợ phương pháp ”‘ovr”. Tuy nhiên nhóm nhận thấy dữ liệu có nhóm khá lớn với hơn 7000 samples và mỗi sample cũng có hơn 8000 features TF-IDF nên theo hướng dẫn của sklearn là “sag” nhanh hơn đối với các tập dữ liệu lớn, trong khi ‘liblinear’ hoạt động tốt với các tập dữ liệu nhỏ

### Softmax Regression

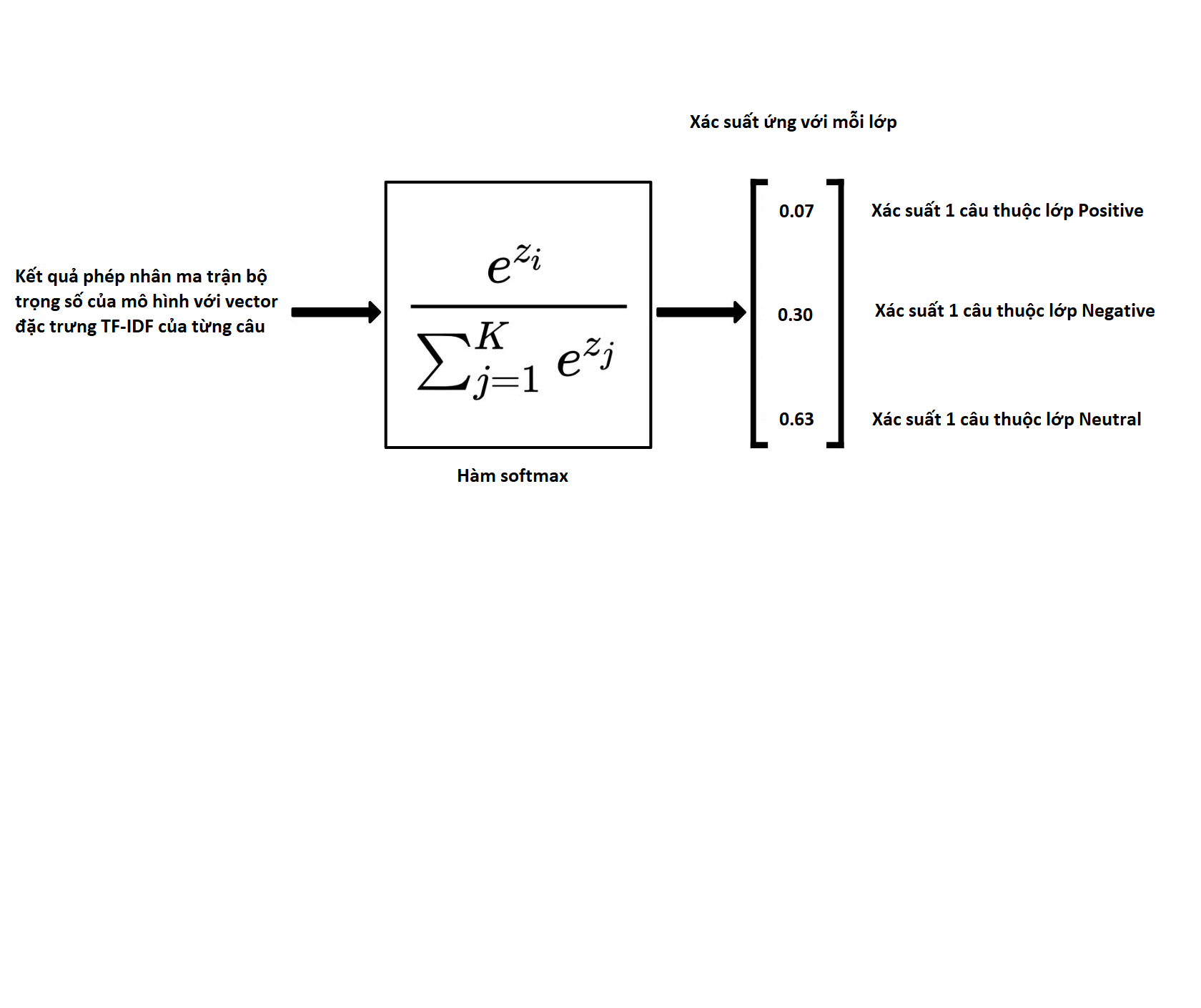
Softmax Regression là trường hợp tổng quát của Logistic Regression và được sử dụng khi có nhiều hơn hai lớp cần được dự đoán.

Mô hình Logistic Regression của sklearn có tham số **multi\_class** và **solver**

cần được tùy chỉnh để phù hợp với bài toán

Đầu ra của hàm softmax chuyển đổi thành một phân phối xác suất trên tất cả các lớp. Mỗi giá trị xác suất thể hiện xác suất của mẫu thuộc về từng lớp riêng biệt.

Lớp với xác suất cao nhất được chọn là lớp dự đoán cho mẫu đó.



Ảnh minh họa: công thức hàm softmax

**Giải thích tham số multi\_class = “multinomial”, solver = “lbfgs”**

* Với bài toán multi-classification (phần lớn mỗi label có 3 class: Positive, Nagative, Neutral), tham số **multi\_class='multinomial'** xác định cách mô hình tối thiểu hóa hàm mất mát trên toàn bộ phân phối xác suất của bài toán phân loại với nhiều hơn hai lớp hoặc ngay cả khi dữ liệu chỉ có hai lớp (cột OTHERS chỉ có hai lớp 1 và 0). Khi được thiết lập thành 'multinomial', đầu ra là một phân phối xác suất trên tất cả các lớp và luôn đảm bảo tổng xác suất trên tất cả các lớp bằng 1.
* Đối với các bài toán multiclass, **“lbfgs”** được sử dụng để xử lý hàm mất mát (multinomial loss)
* Multinomial xử lý trực tiếp vấn đề phân loại nhiều lớp, không cần tạo ra nhiều mô hình như phương pháp OvR.

### SVM - Support vector machine

Với SVM truyền thống, chỉ có thể áp dụng cho bài toán phân lớp nhị phân (binary classification). Tuy nhiên trong trường hợp này, đa phần các nhãn đều có 3 lớp vậy nên nhóm sẽ chọn thiết lập **decision\_function\_shape = “ovr”.** Ngoài ra, nhóm cũng tùy chỉnh **kernel = ‘rbf’**

"OvR" là viết tắt của "One-vs-Rest" hoặc "One-vs-All" trong ngữ cảnh của Support Vector Machines (SVM) và một số thuật toán học máy khác. Phương pháp này được sử dụng để xử lý vấn đề phân loại nhiều lớp (multi-class classification) bằng cách biến đổi nó thành nhiều bài toán phân loại nhị phân (binary classification).

* Huấn luyện SVM cho từng bài toán phân loại: SVM được huấn luyện cho mỗi lớp riêng biệt so với lớp ‘khác’. Tức là nếu huấn luyện trên một cụ thể lớp nào đó, thì hai lớp còn lại được gộp chung và xem như là một lớp ‘khác’. Mỗi SVM tạo ra một ranh giới quyết định giữa lớp riêng biệt và lớp ‘khác’.
* Dự đoán: Khi cần dự đoán lớp của một mẫu mới, tất cả các SVM được áp dụng lên mẫu đó, và lớp được chọn là lớp riêng biệt của SVM cho mẫu đó với giá trị quyết định (decision function) cao nhất.

**Giải thích tham số decision\_function\_shape = “ovr” và kernel = ‘rbf’**

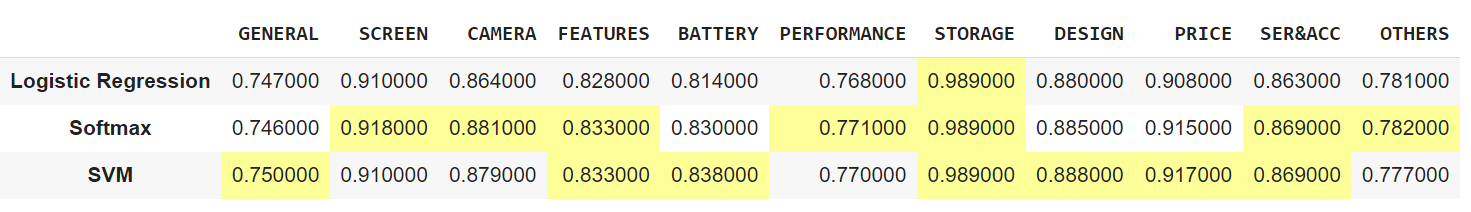
* SVM là mô hình phân lớp nhị phân (binary classification), trong khi đây là bài toán multiclass (Positive, Negative, Neutral). Do đó nhóm quyết định thiết lập  **decision\_function\_shape = “ovr”** để có thể tiếp cận với loại bài toán này
* Qua biểu đồ phân phối hình 6.3 và hình 6.4, phần lớn các trường hợp các nhãn đầu chồng lấp lên nhau, không thể hiện rõ tính tuyến tính. Do đó, nhóm chọn **kernel = “rbf”** để phù hợp với dữ liệu phi tuyến

## Result

Dưới đây nhóm sẽ tiến hành so sánh accuracy score giữa các các model, các quá trình trên từng nhãn

Quá trình 1

**Không thực hiện giảm chiều dữ liệu**

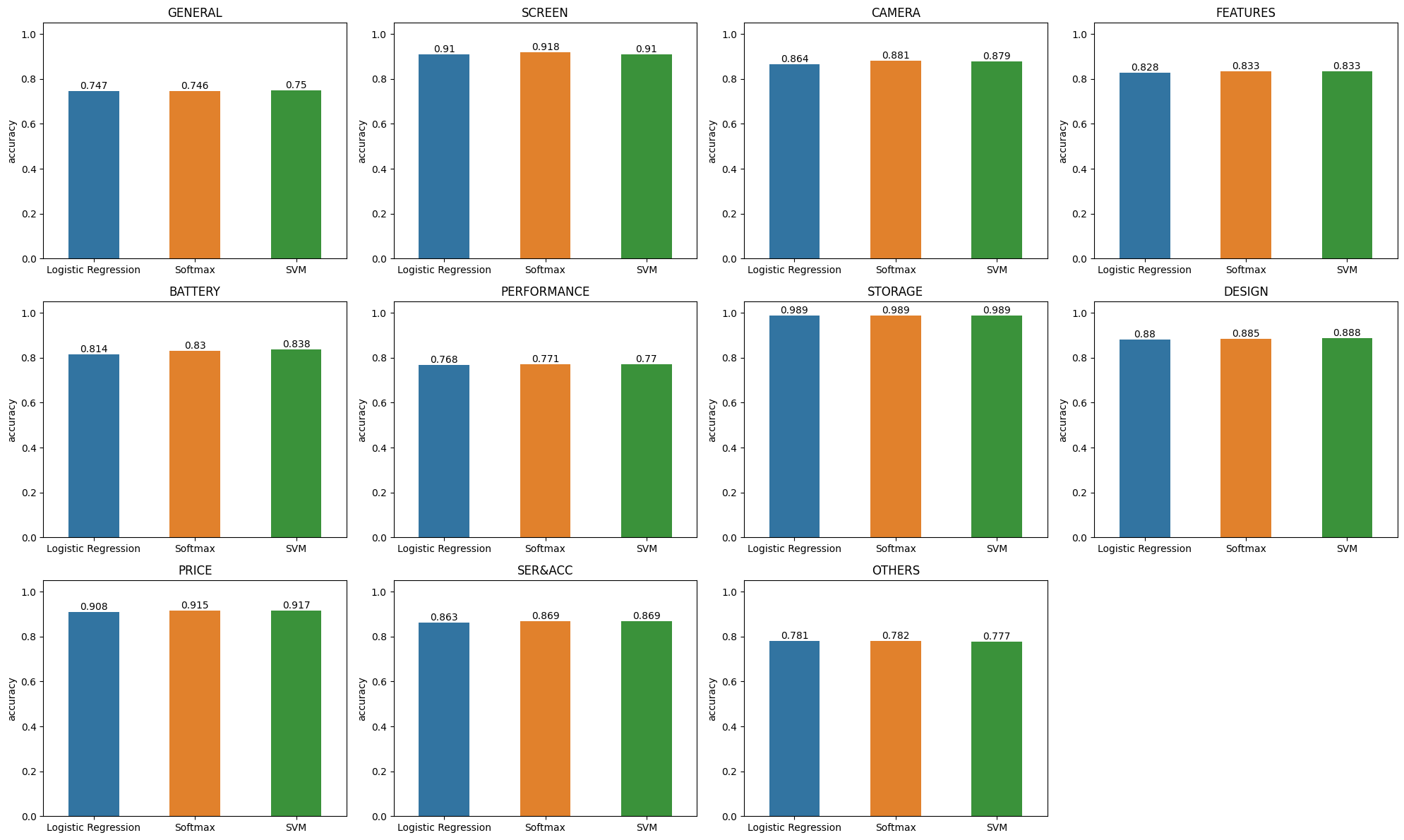


Accuracy trung bình của:

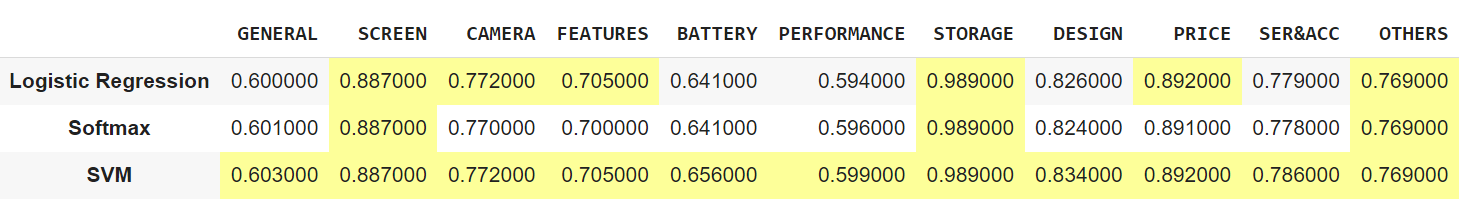
Logistic Regression: 0.8501818181818183

Softmax: 0.8562727272727273

SVM: 0.8563636363636363



**Thực hiện giảm chiều dữ liệu**



Accuracy trung bình của:

Logistic Regression: 0.7685454545454544

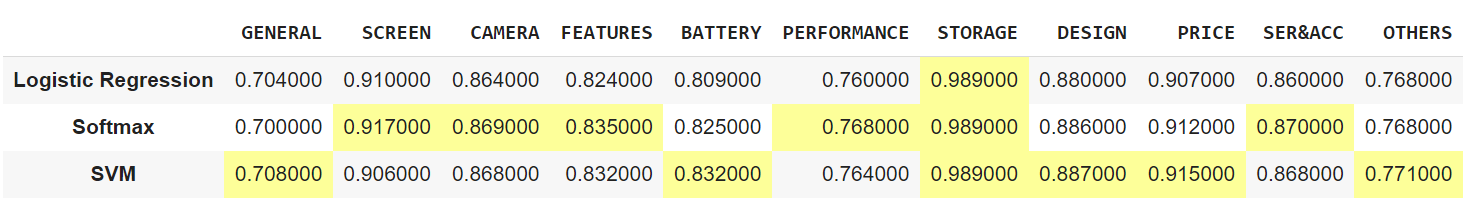
Softmax: 0.7678181818181818

SVM: 0.7720000000000001



Quá trinh 2

**Không thực hiện giảm chiều dữ liệu**

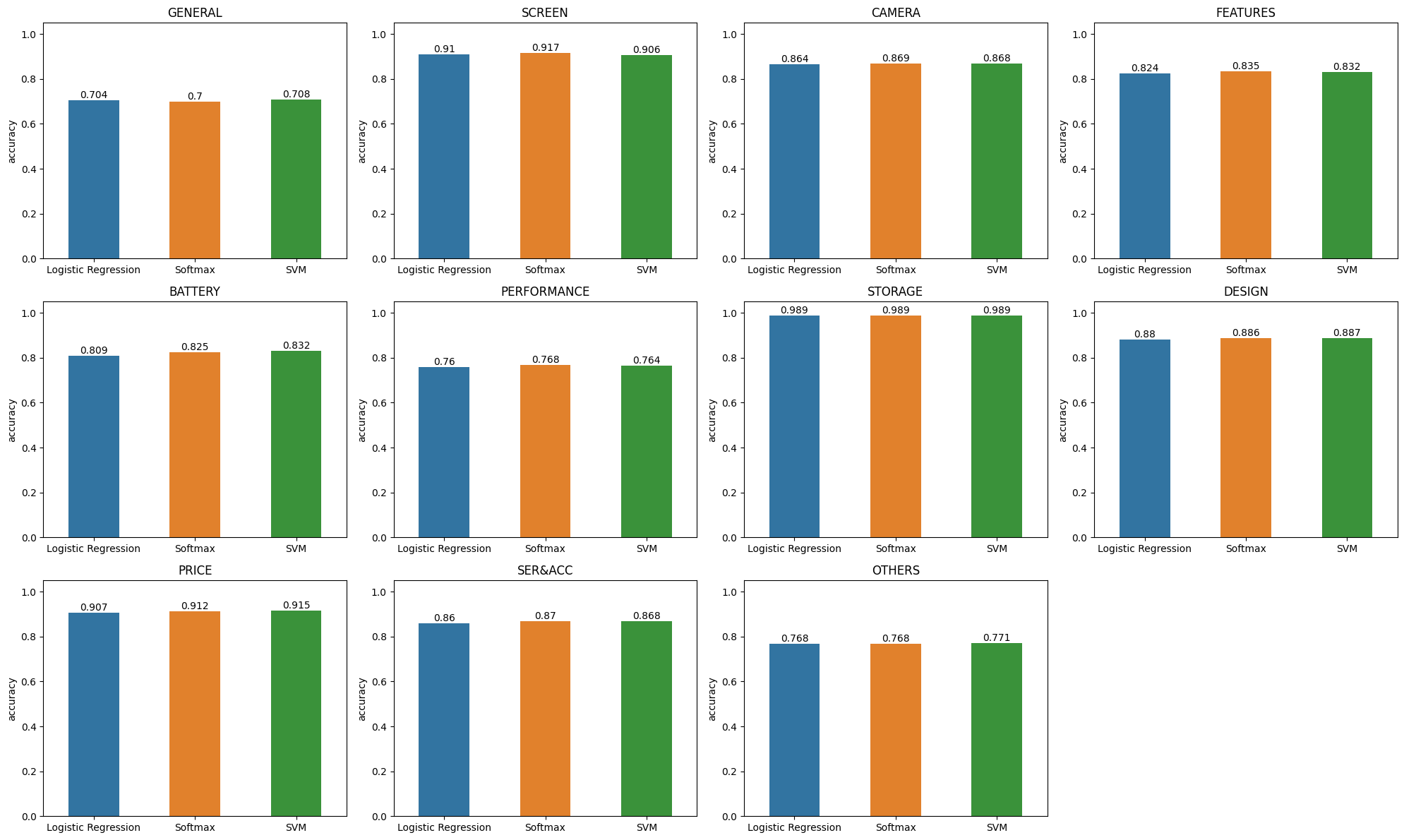


Accuracy trung bình của:

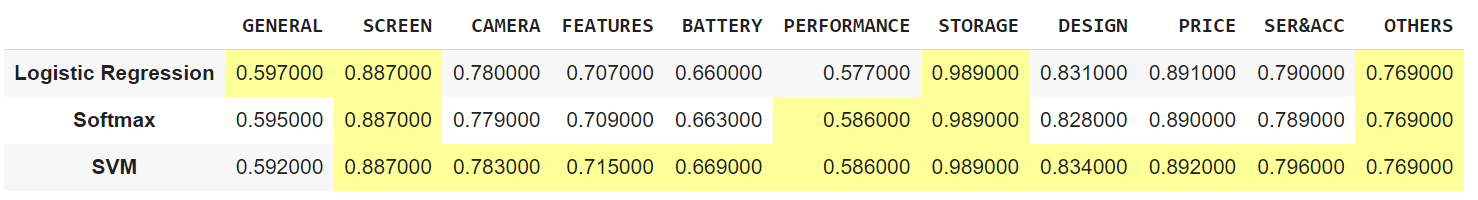
Logistic Regression: 0.8431818181818183

Softmax: 0.8490000000000001

SVM: 0.8490909090909091



**Thực hiện giảm chiều dữ liệu**

****

Accuracy trung bình của:

Logistic Regression: 0.7707272727272727

Softmax: 0.7712727272727271

SVM: 0.7738181818181818

****

# **Conclusion**

**Quá trình 1:**

**Khi không thực hiện giảm chiều dữ liệu:**

* Cả 3 mô hình đều cho kết quả gần như tương đương, đều có accuracy trung bình là 0.85. Tuy nhiên, Softmax và SVM có kết quả nhỉnh hơn Logistic Regression ở hầu hết các khía cạnh (label)
* Có thể thấy, cùng một tập train, cùng là module phân loại LogisticRegression do thư viện skleanr hỗ trợ nhưng thay đổi tham số, thay đổi hướng tiếp cận sẽ cho kết quả khác nhau.
* Với **multi\_class=”multinomial”**, Softmax tính phân phối xác suất trên toàn bộ label và luôn đảm bảo tổng của tất cả các xác suất trên toàn bộ các lớp luôn bằng một. Điều này đảm bảo rằng mỗi mẫu dữ liệu sẽ được phân loại vào một trong các lớp mà không có sự chồng lấn, giúp mô hình đưa ra dự đoán rõ ràng và mạnh mẽ cho từng lớp.
* Trong khi đó, **multi\_class=”ovr”** sử dụng một cách tiếp cận khác. Với ‘ovr’, mô hình sẽ huấn luyện một bộ phân loại riêng biệt cho mỗi lớp, so sánh lớp đó với tất cả các lớp khác. Do đó, mỗi lớp được xem xét độc lập và được so sánh với tất cả các lớp khác. Dẫn đến không có sự liên kết giữa các lớp cũng như không ràng buộc tổng các xác suất của mỗi lớp không nhất thiết phải bằng một.

**Khi thực hiện giảm chiều dữ liệu:**

Phương pháp giảm chiều dữ liệu PCA thường được sử dụng để giảm số lượng chiều của dữ liệu mà vẫn giữ được phần lớn thông tin quan trọng. Tuy nhiên, việc giảm chiều dữ liệu PCA cũng đi kèm với một số rủi ro, có thể gây mất thông tin đặc trưng. Bên cạnh đó, việc lựa chọn số lượng thành phần chính cũng cần được trải qua quá trình thực nghiệm, để đảm bảo cân đối giữa việc giảm chiều dữ liệu và việc giữ lại thông tin cũng như phản ánh đúng được mối quan hệ phức tạp giữa các feature với nhau.

Khi thực hiện giảm chiều dữ liệu, có thể dẫn đến mất đi những thông tin quan trọng. Vì vậy có thể giải thích được lí do ở cả 3 mô hình đều cho accuracy thấp hơn hẳn khi so sánh với trường hợp không thực hiện giảm chiều dữ liệu.

Nhìn chung trường hợp này, cả 3 mô hình đều cho kết quả gần như tương đương, chỉ chênh lệnh khoảng 0.01. Trong đó SVM cho kết quả nhỉnh hơn 2 mô hình còn lại. Mặc dù nhóm đã trực quan hóa 2 thành phần chính đầu tiên lên biểu đồ phân tán, từ đó làm cơ sở chọn kernel cho mô hình SVM tuy nhiên mất mát thông tin là điều không thể tránh khỏi.

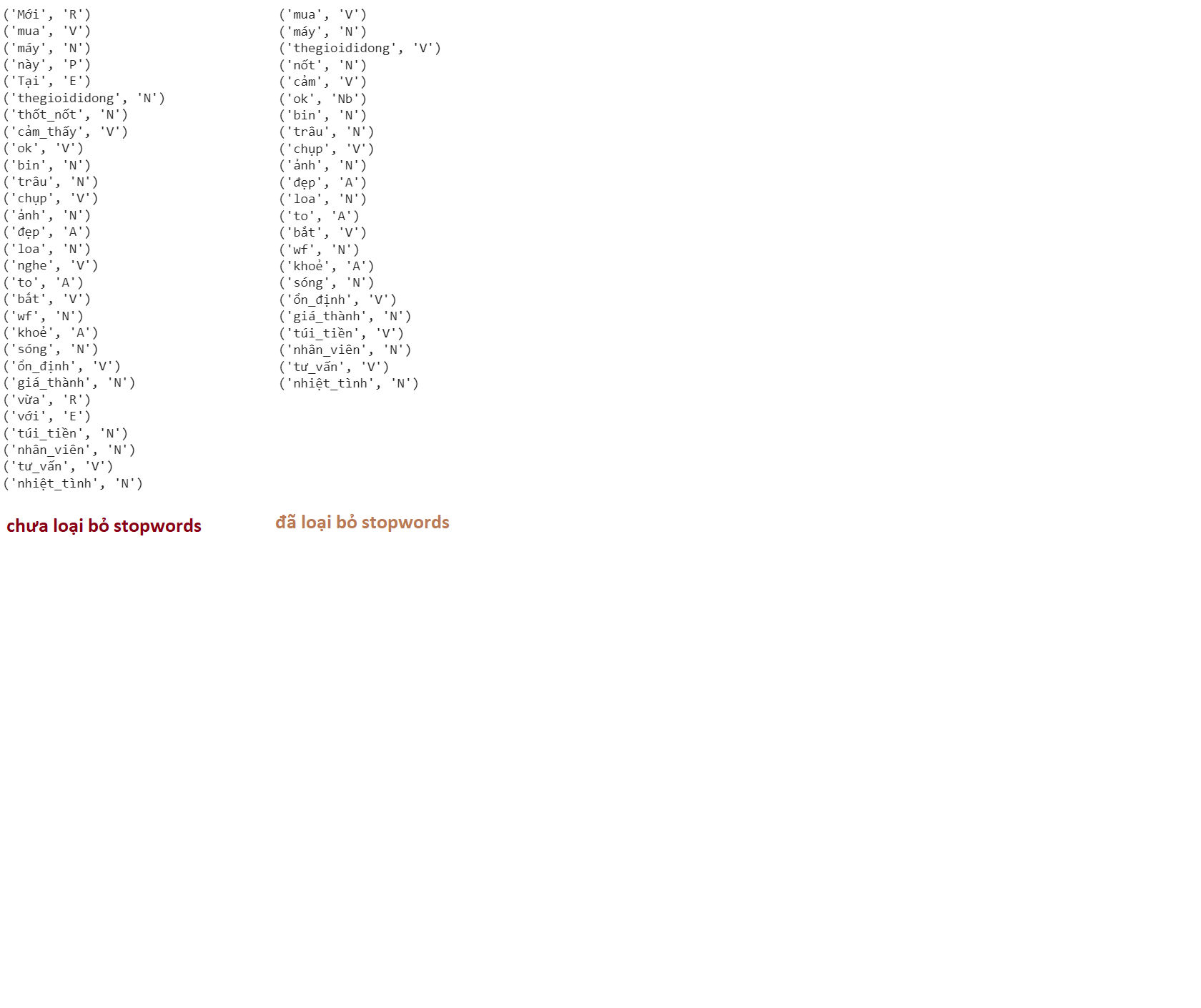
**Quá trình 2:**

**Khi không thực hiện giảm chiều dữ liệu:**

Tương tự như trên, khi giữ lại toàn bộ đặc trưng TF-IDF của câu, dương như không có sự khác biệt đáng kể giữa 3 mô hình, tuy nhiên SVM và Softmax đều nhỉnh hơn Logistic Regression. Như đã đề cập phía trước, accuracy của Softmax nhỉnh hơn Logistic Regression phụ thuộc một phần vào cách tùy nhỉnh tham số **multi\_class**.

Việc loại bỏ stopwords thường được thực hiện trong tiền xử lý dữ liệu văn bản với mục đích giảm kích thước của tập dữ liệu và tăng tốc độ đào tạo mô hình. Tuy nhiên, việc này không phải lúc nào cũng mang lại kết quả tốt. Bỏ stopword giúp loại bỏ những từ có tần suất xuất hiện nhiều nhưng lại ko mang nhiều ý nghĩa trong bài toán sentiment analysis.Dẫu vậy, thực nghiệm của nhóm cho thấy, loại bỏ stopwords lại cho accuracy thấp đi. Nhóm đưa ra những lập luận như sau để giải thích vấn đề trên:

* Trong một số trường hợp, stopwords đóng vai trò biểu diễn ngữ cảnh và ý nghĩa hoàn nhỉnh của câu.
* Thứ tự tiền xử lý cũng có thể ảnh hướng đến hiệu suất của mô hình. Có thể lấy ví dụ như trong tiếng Việt, nhiều từ được tạo ra từ 2 âm tiết, cả 2 âm tiết này đều có nghĩa riêng khi đứng độc lập, nhưng khi ghép lại sẽ mang một nghĩa khác. Nếu loại bỏ một trong hai từ vì chúng là stopwords, thì những từ tạo ra từ 2 âm tiết sẽ bị mất đi ý nghĩa, dẫn đến việc mất mát thông tin.



Hình trên so sánh kết quả pos tag của câu đầu tiên trong tập dữ liệu train trước và sau khi loại bỏ stopwords.

* Có thể thấy một số stopwords đã được loại bỏ một cách hợp lý như từ ‘mới’, từ này không mang lại bất kể thông tin gì về sản phẩm cũng như là đánh giá cá nhân.
* Ngược lại, từ “vừa” và từ “với” cũng được xem như là stopwords và nếu loại bỏ từ 2 này thì sẽ mất ý nghĩa khi mô tả khía cạnh về giá. “Vừa với túi tiền” hay “vừa túi tiền” thể hiện rõ hơn đánh giá về khía cạnh giá tiền sản phẩm hơn là chỉ nhắc đến “túi tiền”

**Khi thực hiện giảm chiều dữ liệu:**

Khi thực hiện giảm chiều dữ liệu, như đã đề cập ở phần trước có thể gây ra sụt giảm accuracy rõ rệt. Cả 3 mô hình đầu gần như cho accuracy trung bình khoảng 0.77 thấp hơn đáng kể khi so sánh với accuracy trùng bình là 0.84 của 3 mô hình khi không thực hiện giảm chiệu dữ liệu.

**Định hướng phát triển:**

* Mở rộng sang các sản phẩm khác trên các sàn thương mại điện tử
* Mở rộng thêm class và label
* Mở rộng độ dài đoạn text

# **References**

Hang, Do Thi Thuy, et al. “Hate Speech Detection on Vietnamese Social Media Text using the Bidirectional-LSTM Model.” 2019, p. 4.

Mohammad Erfan Mowlaei, et al. “Aspect-based sentiment analysis using adaptive aspect-based lexicons.” vol. 148, 2020.

Nazir, Ambreen. *Issues and Challenges of Aspect-based Sentiment Analysis: A Comprehensive Survey*. 2020.

Phan, Luong Luc. “SA2SL: From Aspect-Based Sentiment Analysis to Social Listening System for Business Intelligence.” 2021, p. 12.

Zhang, Wenxuan, et al. “arXiv.” *A Survey on Aspect-Based Sentiment Analysis: Tasks, Methods, and Challenges*, 2023, p. 21.