**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**🙞 🕮 🙜**



**ĐỒ ÁN HỌC PHẦN**

**KHAI KHOÁNG DỮ LIỆU**

**ĐỀ TÀI**

**Phân lớp dữ liệu bình luận sản phẩm**

**trên website thegioididong.com**

**Sinh viên thực hiện:**

**Dương Trung Hiền - Mã số: B1812267 - Khóa: 44**

**Lê Duy – Mã số: B1812256 – Khoá: 44**

Cần Thơ, 12/2021

BỘ GIÁO DỤC VÀ ĐÀO TẠO

**TRƯỜNG ĐẠI HỌC CẦN THƠ**

**KHOA CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**

**🙞 🕮 🙜**



**ĐỒ ÁN HỌC PHẦN**

**KHAI KHOÁNG DỮ LIỆU**

**ĐỀ TÀI**

**Phân lớp dữ liệu bình luận sản phẩm**

**trên website thegioididong.com**

**Người hướng dẫn Sinh viên thực hiện**

**TS. Lưu Tiến Đạo Dương Trung Hiền - B1812267 - K44**

**Lê Duy - B1812256 - K44**

Cần Thơ, 12/2021

**NHẬN XÉT CỦA GIẢNG VIÊN**

------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Cần Thơ, ngày 28 tháng 12 năm 2021

(GVHD ký và ghi rõ họ tên)

**LỜI CẢM ƠN**

Để có thể hoàn thành đồ án học phần Khai Khoáng Dữ Liệu một cách hoàn chỉnh, chúng em xin được bày tỏ lòng biết ơn chân thành và sâu sắc đến giảng viên TS. Lưu Tiến Đạo – người đã trực tiếp tận tình hướng dẫn, giúp đỡ chúng em. Trong suốt quá trình thực hiện đề tài, nhờ những sự chỉ bảo và hướng dẫn quý giá đó mà đồ án đã được hoàn thành một cách tốt nhất.

Tuy có nhiều cố gắng trong quá trình thực hiện đồ án học phần, nhưng không thể tránh khỏi những sai sót. Chúng em rất mong nhận được sự đóng góp ý kiến quý báu của Thầy và các bạn để đồ án hoàn thiện tốt hơn.

Cần Thơ, ngày 26 tháng 12năm 2021

Người viết

Dương Trung Hiền Lê Duy

**MỤC LỤC**

[PHẦN 1: GIỚI THIỆU 6](#_Toc91358105)

[1. Bài toán phân lớp văn bản 6](#_Toc91358106)

[1.1. Khái niệm 6](#_Toc91358107)

[1.2. Ứng dụng 6](#_Toc91358108)

[2. Bố cục đồ án 6](#_Toc91358109)

[PHẦN 2: NỘI DUNG 7](#_Toc91358110)

[CHƯƠNG 1: MÔ TẢ BÀI TOÁN 7](#_Toc91358111)

[1. Mô tả quy trình xây dựng mô hình phân loại văn bản 7](#_Toc91358112)

[CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT 8](#_Toc91358113)

[2.1. Chuẩn bị dữ liệu 8](#_Toc91358114)

[2.2. Trực quan hoá dữ liệu 11](#_Toc91358115)

[2.2.1. Trực quan hoá tập dữ liệu 11](#_Toc91358116)

[2.2.2. Chuyển đổi nhãn 12](#_Toc91358117)

[2.3. Tiền xử lý dữ liệu 13](#_Toc91358118)

[2.3.1. Đưa về viết thường 14](#_Toc91358119)

[2.3.2. Làm sạch dữ liệu 14](#_Toc91358120)

[2.3.2.1. Xoá kí tự xuống dòng 14](#_Toc91358121)

[2.3.2.2. Xử lý dấu câu 14](#_Toc91358122)

[2.3.2.3. Xoá khoảng trắng thừa 15](#_Toc91358123)

[2.3.3. Tách từ 15](#_Toc91358124)

[2.3.4. Chuẩn hoá 15](#_Toc91358125)

[2.3.4.1. Chuẩn hóa unicode sang chuẩn unicode dựng sẵn 15](#_Toc91358126)

[2.3.4.2. Thay thế các từ viết tắt 16](#_Toc91358127)

[2.3.4.3. Xoá các từ lặp 16](#_Toc91358128)

[2.3.5. Các bước tiền xử lý đã áp dụng lên tập dữ liệu 17](#_Toc91358129)

[2.4. Xây dựng mô hình phân loại văn bản 18](#_Toc91358130)

[2.4.1. Xây dựng tập train/test 18](#_Toc91358131)

[2.4.2 Vectơ hoá văn bản 19](#_Toc91358132)

[2.4.2.1. Vectơ hoá văn bản bằng Count Vectors 19](#_Toc91358133)

[2.4.2.1. Vectơ hoá văn bản bằng TF-IDF Vectors (N-Gram level) 20](#_Toc91358134)

[2.4.3 Lưu lại model vector hoá 21](#_Toc91358135)

[2.4.4 Phân loại văn bản với Naives Bayes Multinomial 21](#_Toc91358136)

[2.4.5 Phân loại văn bản với SVM 22](#_Toc91358137)

[2.4.6 Phân loại văn bản với Logistic Regression 22](#_Toc91358138)

[2.4.7. Phân loại văn bản với DeepLearing – Phobert 23](#_Toc91358139)

[2.5. Đánh giá mô hình phân loại văn bản 29](#_Toc91358140)

[2.5.1 So sánh các vectơ hoá 29](#_Toc91358141)

[2.5.2 So sánh các mô hình 29](#_Toc91358142)

[2.6. Xây dựng Restfu API phân loại 30](#_Toc91358143)

[2.7. Xây dựng website dự đoán đánh giá 33](#_Toc91358144)

[CHƯƠNG 3: ĐÁNH GIÁ KIỂM THỬ 34](#_Toc91358145)

[PHẦN KẾT LUẬN 34](#_Toc91358146)

[TÀI LIỆU THAM KHẢO 35](#_Toc91358147)

**DANH MỤC HÌNH**

[Hình 1. Giai đoạn huấn luyện 7](#_Toc91358058)

[Hình 2. Giai đoạn kiểm thử 7](#_Toc91358059)

[Hình 3. Kiến trúc mô hình phân lớp dữ loại bình luận sản phẩm 7](#_Toc91358060)

[Hình 4. Thu nhập dữ liệu bước 1 8](#_Toc91358061)

[Hình 5. Thu nhập dữ liệu bước 2 9](#_Toc91358062)

[Hình 6. Thu nhập dữ liệu bước 3 9](#_Toc91358063)

[Hình 7. Thu nhập dữ liệu bước 4 10](#_Toc91358064)

[Hình 8. Thu nhập dữ liệu bước 5 11](#_Toc91358065)

[Hình 9. Trực quan hoá dữ liệu 11](#_Toc91358066)

[Hình 10. Sự phân bố giữa tỉ lệ các giá trị nhãn 12](#_Toc91358067)

[Hình 11. Chuyển đổi nhãn 12](#_Toc91358068)

[Hình 12. Kết quả chuyển đổi nhãn thành giá trị 0 và 1 13](#_Toc91358069)

[Hình 13. Đưa dữ liệu về viết thường 14](#_Toc91358070)

[Hình 14. Xoá kí tự xuống dòng 14](#_Toc91358071)

[Hình 15. Xử lý dấu câu 14](#_Toc91358072)

[Hình 16. Xoá khoảng trắng thừa 15](#_Toc91358073)

[Hình 17. Tách từ 15](#_Toc91358074)

[Hình 18. Chuẩn hoá unicode sang chuẩn unicode dựng sẵn 15](#_Toc91358075)

[Hình 19. Thay thế các từ viết tắt 16](#_Toc91358076)

[Hình 20. Danh sách các từ thay thế các từ viết tắt 16](#_Toc91358077)

[Hình 21. Xử lý từ lặp 17](#_Toc91358078)

[Hình 22. Cấu trúc các bước tiền xử lý dữ liệu đã áp dụng 17](#_Toc91358079)

[Hình 23. Phân chia tập dữ liệu 18](#_Toc91358080)

[Hình 24. Tập dữ liệu X\_train 18](#_Toc91358081)

[Hình 25. Tập dữ liệu y\_train 19](#_Toc91358082)

[Hình 26. Vector hóa văn bảng bằng CountVectorizer 20](#_Toc91358083)

[Hình 27. Vector hóa văn bảng bằng TF-IDF Vectors (N-Gram level) 20](#_Toc91358084)

[Hình 28. Lưu lại model vector hoá 21](#_Toc91358085)

[Hình 29. Mô hình Naïve Bayes Classifier 21](#_Toc91358086)

[Hình 30. Mô hình SVM 22](#_Toc91358087)

[Hình 31. Mô hình Logistic regression 23](#_Toc91358088)

[Hình 32. Cài đặt thư viện PhoBert 23](#_Toc91358089)

[Hình 33. Load từ điển cũng như bpe 24](#_Toc91358090)

[Hình 34. Tiền xử lý cho mô hình PhoBert 25](#_Toc91358091)

[Hình 35. Encode dữ liệu đầu vào sách các mảng subword 26](#_Toc91358092)

[Hình 36. Tạo mask để tra padding 26](#_Toc91358093)

[Hình 37. Chuyển đổi dữ liệu đầu vào thành DataLoader 27](#_Toc91358094)

[Hình 38. Tiến hành load mô hình Phobert sử dụng GPU 27](#_Toc91358095)

[Hình 39. Hàm huấn luyện mô hình PhoBert 28](#_Toc91358096)

[Hình 40. Tiến hành huấn luyện mô hình PhoBert 29](#_Toc91358097)

[Hình 41. Tiến hành dự đoán mô hình để dự đoán độ chính xác của mô hình PhoBert 29](#_Toc91358098)

[Hình 42. API của url / 30](#_Toc91358099)

[Hình 43. Đọc kết nối API của client 30](#_Toc91358100)

[Hình 44. Url /predict 30](#_Toc91358101)

[Hình 45. Load các mô hình và dự đoán 31](#_Toc91358102)

[Hình 46. Lưu kết quả dự đoán vào database 32](#_Toc91358103)

[Hình 47. Giao diện website 33](#_Toc91358104)

**TÓM TẮT**

Đồ án đề cập quy trình xây dựng mô hình phân loại văn bản cụ thể là áp dụng vào việc phân loại bình luận của người dùng trên website thegioididong.com. Sử dụng các mô hình chuyên về sử lý văn bản như: **Logistic regression**, Naives Bayes Multinomial, SVM (Support Vector Machine) và mô hình deep learning PhoBert. Sau đó, đánh giá độ chính xác của các mô hình được áp dụng. Cuối cùng tạo ra website để người dùng có thể dự đoán được bình luận thuộc phân lớp nào dựa trên mô hình đã được huấn luyện sẵn.

# PHẦN 1: GIỚI THIỆU

## Bài toán phân lớp văn bản

### Khái niệm

Bài toán phân lớp văn bản (Text Classification) là bài toán thuộc nhóm học có giám sát (Supervised learning) trong học máy. Bài toán yêu cầu dữ liệu phải có nhãn. Mô hình sẽ học từ tập dữ liệu có nhãn đó, sau đó được dùng để dữ đoán nhãn cho các tập dữ liệu chưa từng gặp qua trong tương lai [1].

### Ứng dụng

Hiện nay bài toán phân lớp văn bản đã được áp dụng vào nhiều lĩnh vực như: phát hiện tin giả về dịch bệnh Covid, phân loại thư rác, phân loại bài báo khoa học,.. Chính vì điều đó, ta có thể áp dụng bài toán phân lớp văn bản để phân lớp bình luận của người dùng trên website thegiodidong.com để biết bình luận đó là tiêu cực hay tích cực.

## Bố cục đồ án

Bố cục của đồ án khai khoáng dữ liệu gồm 3 phần chính như sau:

**Phần giới thiệu**

Giới thiệu tổng quát về đề tài.

**Phần nội dung**

**Chương 1**: Giới thiệu về bài toán

**Chương 2**: Nội dung bài toán và qui trình giải quyết bài toán

**Chương 3**: Đánh giá và kiểm thử

**Phần kết luận**

Trình bày kết quả đạt được và hướng phát triển hệ thống

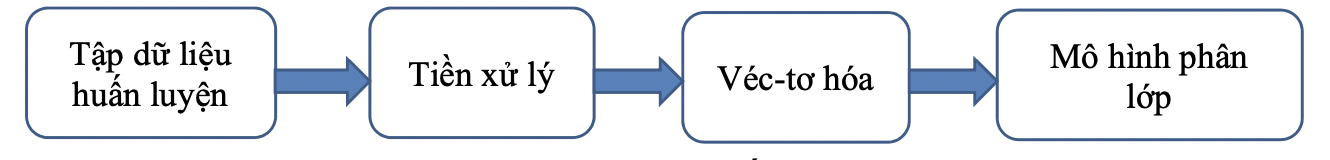
# PHẦN 2: NỘI DUNG

## CHƯƠNG 1: MÔ TẢ BÀI TOÁN

### 1. Mô tả quy trình xây dựng mô hình phân loại văn bản

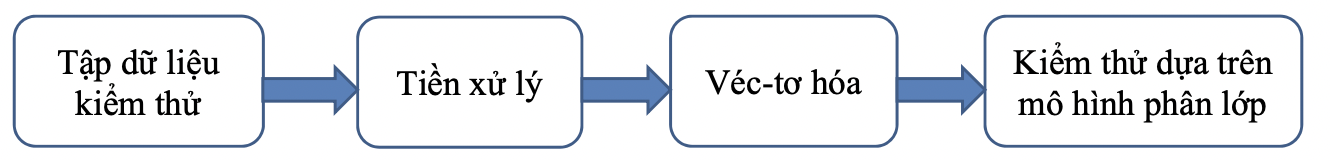
Quy trình xây dựng mô hình phân loại văn bản sử dụng các thuật toán học máy, gồm hai quá trình: huấn luyện và dự đoán.

Đầu vào của quá trình huấn luyện là các dữ liệu văn bản và các nhãn tương ứng với chủ đề cần phân loại. Quá trình này gồm 3 bước: tiền xử lý văn bản, trích xuất đặc trưng và huấn luyện sử dụng các thuật toán học máy [2].



Hình 1. Giai đoạn huấn luyện

Đầu ra của quá trình huấn luyện là mô hình đã được xây dựng và các tham số tối ưu tương ứng cho mô hình [2].



Hình 2. Giai đoạn kiểm thử

Kiến trúc tổng thể về mô hình phân lớp dữ liệu bình luận sản phẩm trên website thegioididong.com [2].

Dữ liệu bình luận

Hệ thống phân loại

Phân loại

Nộp vào

Bình luận tích cực

Bình luận tiêu cực

Hình 3. Kiến trúc mô hình phân lớp dữ loại bình luận sản phẩm

## CHƯƠNG 2: THIẾT KẾ VÀ CÀI ĐẶT

### 2.1. Chuẩn bị dữ liệu

Dữ liệu là điều tất yếu phải có trong bài toán này. Trong quá trình xây dựng bài toán phân lớp văn bản, ta cần phải quan tâm hơn cả là ở bước chuẩn bị và tiền xử lý dữ liệu vì sẽ có ảnh hưởng đến mô hình dự đoán có độ chính xác cao hay thấp ? [3]

Với bài toán “Phân lớp dữ liệu bình luận sản phẩm trên website thegioididong.com”, tập dữ liệu sẽ gồm:

* Nội dung bình luận của khách hàng
* Số lượng sao tương ứng với bình luận

Việc thu nhập dữ liệu sẽ được thực hiện bằng đoạn code Python sau:

*Lưu ý: đoạn code cào dữ liệu được viết để sử dụng trên colab của google nên có thể sẽ không chạy đối với các nền tảng khác.*

* Import và setup để tiến hành cào

Text

Description automatically generated

Hình 4. Thu nhập dữ liệu bước 1

* Ta truy cập vào đường dẫn [www.thegioididong.com/dtdd](http://www.thegioididong.com/dtdd) để tiến hành cào dữ liệu từ toàn bộ điện thoại. Tuy nhiên khi truy cập vào đường dẫn trên thì website không hiển thị toàn bộ số điện thoại hiện có mà ta cần ấn vào nút xem thêm để website có thể hiển thị toàn bộ điện thoại hiện có.

Graphical user interface, text

Description automatically generated

Hình 5. Thu nhập dữ liệu bước 2

* Ta tiến hành lấy toàn bộ đường dẫn đến từng chiếc điện thoại

Text

Description automatically generated

Hình 6. Thu nhập dữ liệu bước 3

* Hàm cào các comment dựa trên đường dẫn tìm được

Text

Description automatically generated

Hình 7. Thu nhập dữ liệu bước 4

Text

Description automatically generated

Hình 8. Thu nhập dữ liệu bước 5

### 2.2. Trực quan hoá dữ liệu

**Data visualization là gì? Trực quan hóa dữ liệu** (Data visualization) là một sự thể hiện dữ liệu hoặc là các thông tin thành biểu đồ, đồ thị hoặc về định dạng **trực quan hoá dữ liệu** khác [4].

Chúng ta cần**trực quan hóa dữ liệu** (Data visualization) vì một bản tóm tắt thông tin **trực quan hoá dữ liệu** sẽ giúp việc xác định đến những mô hình và với những xu hướng được dễ dàng hơn so với những việc xem qua hàng ngàn về bảng tính. Nó sẽ giúp bộ não của con người hoạt động. Vì với những mục đích của phân tích dữ liệu là sẽ tìm thấy về những insight ẩn sau dữ liệu, **trực quan hoá dữ liệu** sẽ có những giá trị hơn nhiều khi được **trực quan hoá dữ liệu [4].**

#### ****2.2.1. Trực quan hoá tập dữ liệu****

**Dưới đây là đoạn code viết bằng Python áp dụng lên tập dữ liệu để xem sự phân bố giữa các giá trị của nhãn.**

Text

Description automatically generated with medium confidence

Hình 9. Trực quan hoá dữ liệu

Sau khi trực quan hoá, có thể thấy: tập dữ liệu không cân bằng. Vì có sự chênh lệch lớn giữa nhãn có giá trị là 5 so với các nhãn còn lại qua hình

Chart, bar chart

Description automatically generated

Hình 10. Sự phân bố giữa tỉ lệ các giá trị nhãn

Kết luận: gộp nhãn có giá trị 5 và 4 lại là nhãn đánh giá tích cực (nhãn 1). Nhãn có giá trị 1,2 và 3 thành nhãn đánh giá tiêu cực (nhãn 0).

#### ****2.2.2. Chuyển đổi nhãn****

Để không xuất hiện tập dữ liệu mất cân bằng sẽ ảnh hưởng tới kết quả dự đoán. Vì thế ta sẽ tiến hành gộp chuyển đổi từ 5 giá trị nhãn (1, 2, 3, 4, 5) thành hai giá trị nhãn là 0 (gộp nhãn 1, 2, 3) – nhãn đánh giá tiêu cực và nhãn 1 ( gộp nhãn 4, 5) – nhãn đánh giá tích cực.

**Dưới đây là đoạn code viết bằng Python áp dụng lên tập dữ liệu để chuyển đổi giá trị nhãn tham khảo. Đoạn code được tham khảo từ [5].**

Graphical user interface, text, application

Description automatically generated

Hình 11. Chuyển đổi nhãn

Sau khi chuyển đổi nhãn, tập dữ liệu phân bố theo: (54%:46%) hay (6:4). Tập dữ không quá mất cân bằng, không ảnh hưởng đáng kể tới kết quả dự đoán.

Chart, bar chart

Description automatically generated

Hình 12. Kết quả chuyển đổi nhãn thành giá trị 0 và 1

### 2.3. Tiền xử lý dữ liệu

Bước tiền sử lý dữ liệu là quá trình **chuẩn hóa dữ liệu**và**loại bỏ các thành phần không có ý nghĩa**cho việc phân loại văn bản. Chúng ta sẽ tiến hành tiền xử lý dữ liệu trước khi đưa vào huấn luyện mô hình phân loại văn bản. Việc tiền xử lý dữ liệu là hết sức quan trọng để đảm bảo mô hình đạt được kết quả tốt [1].

Tiền xử lý dữ liệu tiếng Việt cho bài toán phân loại văn bản thường gồm các việc sau:

* Đưa về viết thường (lowercase)
* Xoá kí tự xuống dòng
* Xử lý dấu câu
* Xoá bớt khoảng trắng thừa
* Thay thế các từ viết tắt
* Xoá các kí tự lặp
* Thực hiện tách từ tiếng Việt (sử dụng thư viện tách từ như pyvi)
* Chuẩn hóa bảng mã Unicode (đưa về Unicode tổ hợp dựng sẵn)

#### 2.3.1. Đưa về viết thường

Việc đưa dữ liệu về chữ viết thường là rất cần thiết. Bởi vì đặc trưng này không có tác dụng ở bài toán phân loại văn bản. Đưa về chữ viết thường giúp giảm số lượng đặc trưng (vì máy tính hiểu hoa thường là 2 từ khác nhau) và tăng độ chính xác hơn cho mô hình.

Đoạn code viết bằng Python sau đây, sẽ mô tả việc đưa dữ liệu về viết thường:

Text

Description automatically generated

Hình 13. Đưa dữ liệu về viết thường

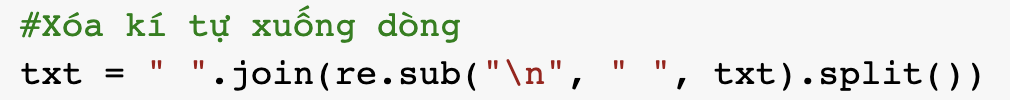
#### 2.3.2. Làm sạch dữ liệu

Tiền xử lý bao gồm việc loại bỏ các dữ liệu không có tác dụng cho việc phân loại văn bản. Việc này giúp:

* Giảm số chiều đặc trưng, tăng tốc độ học và xử lý
* Tránh làm ảnh hưởng xấu tới kết quả của mô hình

Các dấu ngắt câu và các ký tự đặc biệt khác không giúp bạn phân loại một văn bản thuộc chuyên mục nào. Do đó, chúng ta nên loại bỏ nó đi [1].

##### 2.3.2.1. Xoá kí tự xuống dòng



Hình 14. Xoá kí tự xuống dòng

##### 2.3.2.2. Xử lý dấu câu

Text

Description automatically generated

Hình 15. Xử lý dấu câu

##### 2.3.2.3. Xoá khoảng trắng thừa

Text

Description automatically generated

Hình 16. Xoá khoảng trắng thừa

#### 2.3.3. Tách từ

Một công đoạn quan trọng bậc nhất trong xử lý ngôn ngữ tự nhiên, do Tiếng Việt có độ phức tạp hơn ngôn ngữ khác (bởi có các từ ghép). Việc tách từ theo nhiều cách khác nhau có thể gây ra sự hiểu nhầm về mặt ngữ nghĩa [6].

Đối với tập dữ liệu này sẽ sử dụng thư viện nguồn mở pyvi để thực hiện tách từ:

A picture containing logo

Description automatically generated

Hình 17. Tách từ

#### 2.3.4. Chuẩn hoá

Đưa văn bản từ các dạng không đồng nhất về cùng một dạng nhằm tối ưu bộ nhớ lưu trữ và tính xác khi dự đoán.

Trong tiếng Việt có 2 loại mã Unicode là Unicode tổ hợp và Unicode dựng sẵn, làm cho mô hình máy học khó phân biệt chính xác từ vựng. Ngoài ra, cách bỏ dấu tiếng Việt và cách viết tắt không giống nhau ở mỗi khu vực cũng làm ảnh hưởng đến tỷ lệ nhận dạng từ ngữ của máy [3].

##### 2.3.4.1. Chuẩn hóa unicode sang chuẩn unicode dựng sẵn

Để chuẩn hóa dữ liệu đưa về cùng 1 chuẩn Unicode dựng sẳn như câu lệnh bên dưới:

A picture containing text

Description automatically generated

Hình 18. Chuẩn hoá unicode sang chuẩn unicode dựng sẵn

##### 2.3.4.2. Thay thế các từ viết tắt

Sử dụng bộ từ điển thay thế các kiểu gõ dấu và viết tắt.

Graphical user interface

Description automatically generated with medium confidence

Hình 19. Thay thế các từ viết tắt

Với danh sách từ viết tắt được thống kê thủ công trong tập tin acronyms\_vi được tham khảo từ [3] và đã được nhóm bổ sung các từ viết tắt thông dụng: Graphical user interface, application

Description automatically generated

Hình 20. Danh sách các từ thay thế các từ viết tắt

##### 2.3.4.3. Xoá các từ lặp

Ngoài ra, các từ có ý nghĩa nhưng bị viết trùng lắp như: hoaa, caii...cũng làm ảnh hưởng đến tỷ lệ chính xác của mô hình, vì vậy cần phải chuẩn hóa bằng đoạn code Python sau:

A picture containing chart

Description automatically generated

Hình 21. Xử lý từ lặp

#### 2.3.5. Các bước tiền xử lý đã áp dụng lên tập dữ liệu

Dưới đây là đoạn code bằng Python, áp dụng các bước tiền xử lý dữ liệu:



Hình 22. Cấu trúc các bước tiền xử lý dữ liệu đã áp dụng

Đối với bài toán này, sẽ không có bước loại bỏ stopword vì đã thử áp dụng nhưng làm giảm độ chính xác khi dự đoán.

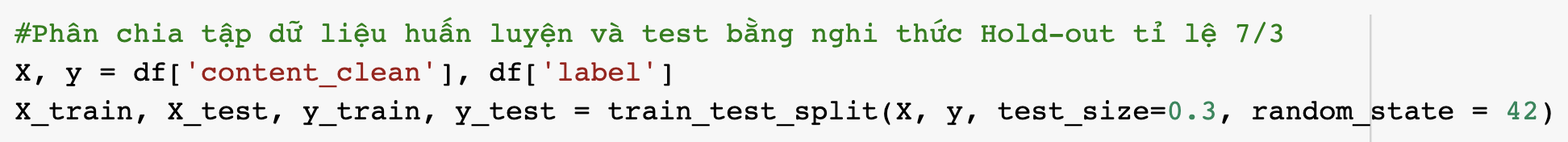
### 2.4. Xây dựng mô hình phân loại văn bản

Trước khi huấn luyện mô hình phân loại văn bản, ta cần xây dựng tập huấn luyện và tập kiểm thử. Việc này là cần thiết để đánh giá kết quả huấn luyện, lựa chọn mô hình cũng như tinh chỉnh để mô hình cho tốt hơn.

#### 2.4.1. Xây dựng tập train/test

Đối với tập dữ liệu này, ta sẽ áp dụng nghi thức hold-out: lấy ngẫu nhiên 2/3 tập dữ liệu để học và 1/3 tập dữ liệu còn lại dùng cho kiểm tra.

Dưới đây là đoạn code viết bằng Python dùng để phân chia tập dữ liệu:



Hình 23. Phân chia tập dữ liệu

Tập dữ liệu X\_train và y\_train sau khi được phân chia:

Text

Description automatically generated

Hình 24. Tập dữ liệu X\_train

Table

Description automatically generated

Hình 25. Tập dữ liệu y\_train

#### 2.4.2 Vectơ hoá văn bản

Ở bước này, chúng ta sẽ đưa dữ liệu dạng văn bản cụ thể là tập dữ liệu X\_train và X\_test đã được xử lý về dạng vector thuộc tính có dạng số học [7]. Có nhiều các khác nhau để đưa dữ liệu văn bản dạng text về dữ liệu dạng số mà chúng ta có thể thực hiện như:

* Count Vectors
* TF-IDF Vectors
  + Word level
  + N-Gram level
  + Character level
* Word Embeddings
* Text / NLP
* Topic Models

Nhưng trong bài toán này chỉ áp dụng 2 phương pháp điển hình là: Count Vectors as features và TF-IDF Vectors (N-Gram level).

##### 2.4.2.1. Vectơ hoá văn bản bằng Count Vectors

Khi sử dụng phương pháp này, chúng ta sẽ thu được một ma trận mà trong đó, mỗi hàng sẽ đại diện cho một văn bản, mỗi cột đại diện cho một từ có trong từ điển, và mỗi ô (cell) sẽ chứa tần suất xuất hiện của từ trong văn bản tương ứng. Bằng cách sử dụng thư viện sklearn, chúng ta sẽ làm như sau [7]:

Text

Description automatically generated

Hình 26. Vector hóa văn bảng bằng CountVectorizer

##### 2.4.2.1. Vectơ hoá văn bản bằng TF-IDF Vectors (N-Gram level)

Phương pháp TF-IDF (Term Frequency - Inverse Document Frequency), đây là một phương pháp cực kì phổ biến trong xử lý văn bản [7]. Nó được tính theo công thức dưới đây:

* TF(t) = (Number of times term t appears in a document) / (Total number of terms in the document)
* IDF(t) = log\_e(Total number of documents / Number of documents with term t in it)

Chúng ta có thể thực hiện TF-IDF cho các cấp độ khác nhau của văn bản như sau:

* Word Level TF-IDF : Thực hiện tính toán dựa trên mỗi thành phần là một từ riêng lẻ.
* N-gram Level TF-IDF : Kết hợp n thành phần (từ) liên tiếp nhau. Ví dụ: "thủ\_tướng đức nhận\_lời tham\_dự lễ kỷ\_niệm". Khi đó, 2-gram cho ta kết quả: {thủ\_tướng đức, đức nhận\_lời, nhận\_lời tham\_dự, tham\_dự lễ , lễ kỷ\_niệm}. Mõi phần từ là cặp 2 từ liên tiếp nhau [PhanLoaiVanBanViBLO].

Bằng cách sử dụng thư viện sklearn, chúng ta sẽ làm như sau [7]:

Text

Description automatically generated

Hình 27. Vector hóa văn bảng bằng TF-IDF Vectors (N-Gram level)

#### 2.4.3 Lưu lại model vector hoá

Để tiện sử dụng cho việc sử dụng lại dữ liệu đã được xử lý, ta sẽ lưu dữ liệu vào file .pkl:

Graphical user interface, text, application

Description automatically generated

Hình 28. Lưu lại model vector hoá

#### 2.4.4 Phân loại văn bản với Naives Bayes Multinomial

**Naïve Bayes Classifier (NBC)** là mô hình phân lớp văn bản dựa trên định lý Bayes.

Một số mô hình Naïve Bayes hay sử dụng là Multinomial Naive Bayes, Bernoulli Naive Bayes và Gaussian Naive Bayes. Mô hình này thường được sử dụng để phân lớp văn bản [8].

Đây là đoạn code áp dụng mô hình **Naïve Bayes Classifier** bằng cách sử dụng thư viện sklearn:

Graphical user interface, text, application, email

Description automatically generated

Hình 29. Mô hình Naïve Bayes Classifier

#### 2.4.5 Phân loại văn bản với SVM

**SVM (Support Vector Machine)** là 1 thuật toán học máy thuộc nhóm Supervised Learning (học có giám sát) được sử dụng trong các bài toán phân lớp dữ liệu (classification) hay hồi qui (Regression). Tuy nhiên nó được sử dụng chủ yếu cho việc phân loại [9].

Đây là đoạn code áp dụng mô hình **SVM** bằng cách sử dụng thư viện sklearn:

Graphical user interface, text, application, email

Description automatically generated

Hình 30. Mô hình SVM

#### 2.4.6 Phân loại văn bản với Logistic Regression

**Logistic regression** là thuật toán đơn giản nhưng lại rất hiệu quả trong bài toán phân loại (Classification).

**Logistic regression** được áp dụng trong bài toán phân loại nhị phân (Binary classification) tức ta sẽ có hai output, hoặc có thể gọi là hai nhãn (ví dụ như 0 và 1) [10].

Đây là đoạn code áp dụng mô hình **Logistic regression** bằng cách sử dụng thư viện sklearn:

Graphical user interface, text, application, email

Description automatically generated

Hình 31. Mô hình **Logistic regression**

#### 2.4.7. Phân loại văn bản với DeepLearing – Phobert

Phobert là mô hình deep learing được xây dựng dựa trên mô hình Bert của Google. Phobert được phát triển bởi VinAI dành cho xử lý ngôn ngữ tiếng Việt [11].

Phobert sử dụng phần encoder của mô hình Transformer để làm đầu vào huấn luyện.

Cài đặt các thư viện cần thiết để sử dụng Phobert.

*Lưu ý: đoạn code cào dữ liệu được viết để sử dụng trên colab của google nên có thể sẽ không chạy đối với các nền tảng khác.*

Graphical user interface, text

Description automatically generated

Hình 32. Cài đặt thư viện PhoBert

Tiếp theo ta tiến hành load từ điển cũng như bpe giúp encode 1 câu đầu vào thành 1 mảng các subword. Mỗi 1 subword có 1 id riêng trong từ điển của Phobert.

Text

Description automatically generated

Hình 33. Load từ điển cũng như bpe

Tiền xử lý dữ liệu đầu vào cho mô hình

Text

Description automatically generated

Hình 34. Tiền xử lý cho mô hình PhoBert

Ta bắt đầu encode dữ liệu đầu vào danh sách các mảng subword.

Text

Description automatically generated

Hình 35. Encode dữ liệu đầu vào sách các mảng subword

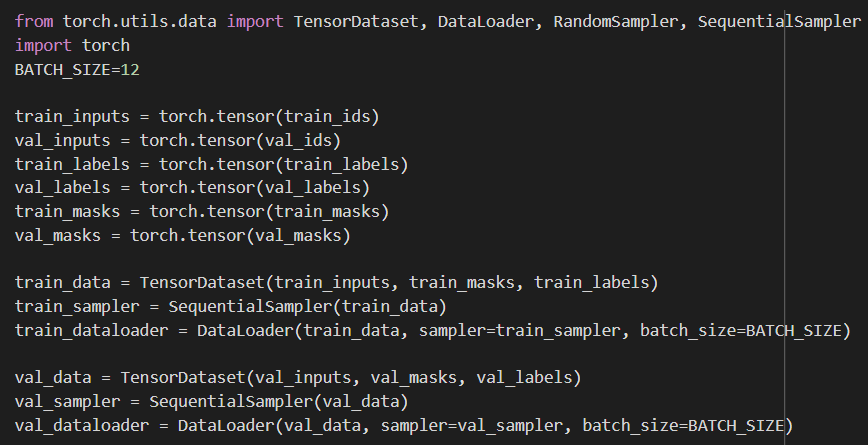
Tạo thêm các mask là mảng gồm các số 0 và 1 nhằm kiểm tra các giá trị trong chuỗi đã được padding hay chưa.

Text

Description automatically generated

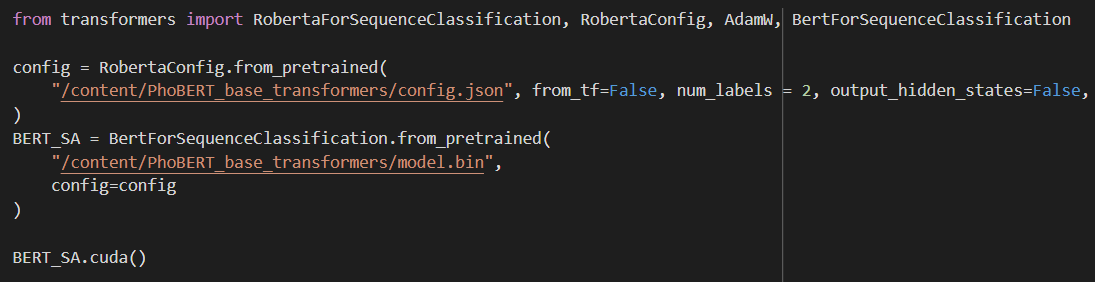
Hình 36. Tạo mask để tra padding

Chuyển đổi dữ liệu đầu vào thành DataLoader



Hình 37. Chuyển đổi dữ liệu đầu vào thành DataLoader

Tiến hành load mô hình Phobert sử dụng GPU



Hình 38. Tiến hành load mô hình Phobert sử dụng GPU

Hàm huấn luyện

Text

Description automatically generated

Hình 39. Hàm huấn luyện mô hình PhoBert

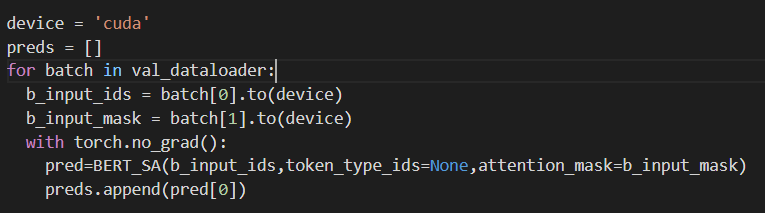
Tiến hành huấn luyện

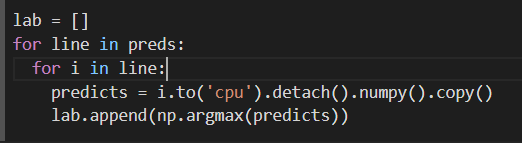
Text

Description automatically generated

Hình 40. Tiến hành huấn luyện mô hình PhoBert

Mô hình đã huấn luyện xong tiến hành dự đoán để kiểm tra độ chính xác của mô hình PhoBert.





Hình 41. Tiến hành dự đoán mô hình để dự đoán độ chính xác của mô hình PhoBert

### 2.5. Đánh giá mô hình phân loại văn bản

#### 2.5.1 So sánh các vectơ hoá

Ta có thể thấy được độ chính xác khi sử dụng TfidfVectorizer luôn cao hơn so với khi sử dụng CountVectorizer.

#### 2.5.2 So sánh các mô hình

Mô hình SVM và Logistic Regresstion sử dụng TfidfVectorizer có độ chính xác cao vượt trội hơn so với mô hình Bayes sử dụng TfidfVectorizer.

Mô hình Logistic Regresstion sử dụng CountVectorizer có độ chính xác cao hơn so với SVM.

Mô hình Logistic Regresstion luôn cao hơn so với 2 mô hình còn lại kể cả sử dụng CountVectorizer hay là sử dụng TfidfVectorizer.

### 2.6. Xây dựng Restfu API phân loại

Sử dụng ngôn ngữ python với Flask để xây dựng RESTful API để phân loại bình luận.

Ở url / ta sẽ render template index.html, thực hiện truy vấn cơ sở dữ liệu Firebase bên phía Client.

Graphical user interface, text

Description automatically generated

Hình 42. API của url /

Text

Description automatically generated

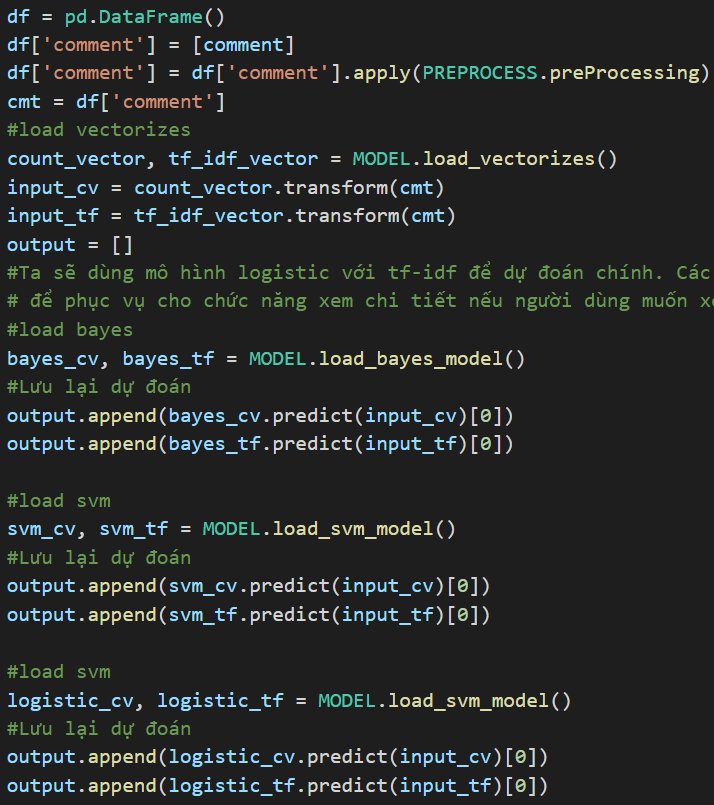
Hình 43. Đọc kết nối API của client

Ở url /predict ta sẽ lấy dữ liệu từ request gồm có nội dung bình luận và tên người dùng, sau đó tiến hành load model và thực hiện dự đoán rồi đưa dự đoán lên cơ sở dữ liệu Firebase.

Text

Description automatically generated

Hình 44. Url /predict



Hình 45. Load các mô hình và dự đoán

Text

Description automatically generated

Hình 46. Lưu kết quả dự đoán vào database

#### 2.7. Xây dựng website dự đoán đánh giá

Đường dẫn tới website phân lớp dữ liệu bình luận sản phẩm trên website thegioididong.com : <https://sentiment-analysis-tgdd.herokuapp.com/>

Timeline

Description automatically generated with medium confidence

Hình 47. Giao diện website

# CHƯƠNG 3: ĐÁNH GIÁ KIỂM THỬ

# PHẦN KẾT LUẬN

* **Kết luận**

Các bước đã thực hiện tốt trong đồ án:

* Cào dữ liệu từ website thegioididong.com. Dữ liệu cần thu thập bao gồm: nội dung bình luận và số lượng sao tương ứng với bình luận. ü
* Làm sạch dữ liệu, tách từ, chuẩn hóa từ, loại bỏ stopword. ü
* Xây dựng mô hình (sử dụng ít nhất 3 giải thuật trong đó có PhoBERT). ü
* Đánh giá mô hình. ü
* Xây dựng Restful API phân loại. ü
* Xây dựng website dự đoán đánh giá. ü
* **Hướng phát triển của đề tài**
  + Thu thập thêm dữ liệu nhằm từ bài toán phân loại 2 nhãn thành bài toán phân loại 5 nhãn ( tương ứng 5 lớp: 1 sao, 2 sao, 3 sao, 4 sao, 5sao).
  + Cải thiện giao diện website thân thiện với người dùng.
  + Đưa mô hình PhoBert lên website.

# TÀI LIỆU THAM KHẢO

1. Nguyễn Văn Hiếu, “Phân loại văn bản tiếng Việt sử dụng machine learning”. Địa chỉ: <https://nguyenvanhieu.vn/phan-loai-van-ban-tieng-viet/#chuan-bi-du-lieu> [Truy cập 01/12/2021].
2. Trần Thanh Điện, Thái Nhựt Thanh và Nguyễn Thái Nghe, “Giải pháp phân loại bài báo khoa học bằng kỹ thuật máy học” , Tạp chí *Khoa học Trường Đại Học Cần Thơ*, tập 55, số 4A, trang số 33, (2019): 29-37 .
3. Hứa Quốc Thi, *“TheGioiDiDong”,* Đồ án Khai khoáng dữ liệu, khoa Công Nghệ Thông Tin Và Truyền Thông, trường Đại Học Cần Thơ.
4. Hiền Mai, “Trực quan hoá dữ liệu là gì? Tại sao nói Data Visualization vô cùng quan trọng”, 2021-08-09 09:55:30. Địa chỉ: <https://123job.vn/bai-viet/truc-quan-hoa-du-lieu-la-gi-tai-sao-noi-data-visualization-vo-cung-quan-trong-2944.html> [Truy cập 03/12/2021].
5. VITALII MOKIN, “NLP - EDA, Bag of Words, TF IDF, GloVe, BERT”. Địa chỉ: <https://www.kaggle.com/vbmokin/nlp-eda-bag-of-words-tf-idf-glove-bert> [Truy cập 04/12/2021].
6. “Ứng dụng công nghệ trí tuệ nhân tạo trong bài toán phân loại văn bản”, 11:00 | 22/03/2019 |. Địa chỉ: <http://antoanthongtin.gov.vn/giai-phap-khac/ung-dung-cong-nghe-tri-tue-nhan-tao-trong-bai-toan-phan-loai-van-ban-105149> [Truy cập 04/12/2021].
7. Nguyễn Thành Hậu, “Phân loại văn bản tự động bằng Machine Learning như thế nào?”, 10/11/2018 1:50 PM. Địa chỉ: <https://viblo.asia/p/phan-loai-van-ban-tu-dong-bang-machine-learning-nhu-the-nao-4P856Pa1ZY3> [Truy cập 05/12/2021].
8. Tran Duc Tan, “Mô hình phân lớp Naive Bayes”, 22/06/2019 03:16 PM. Địa chỉ: <https://viblo.asia/p/mo-hinh-phan-lop-naive-bayes-vyDZO0A7lwj>

[Truy cập 12/12/2021].

1. Trương Xuân Đạt, “Giới Thiệu về Mô Hình SVM”, 04/11/2020. Địa chỉ: <https://www.stdio.vn/computer-vision/gioi-thieu-ve-mo-hinh-svm-D15jcg> [Truy cập ngày 14/12/2021].
2. Mai Trong Nghia, “Logistic regression”, 03/04/2020. Địa chỉ: <http://maitrongnghia.com/2020/04/logistic-regression/> [Truy cập ngày 16/12/2021].
3. Phạm Hữu Quang, “BERT, RoBERTa, PhoBERT, BERTweet: Ứng dụng state-of-the-art pre-trained model cho bài toán phân loại văn bản”, 19/07/2020 4:34 PM. Địa chỉ: <https://viblo.asia/p/bert-roberta-phobert-bertweet-ung-dung-state-of-the-art-pre-trained-model-cho-bai-toan-phan-loai-van-ban-4P856PEWZY3> [Truy cập ngày 20/12/2021].
4. Brandon Walsh, “Creating Web APIs with Python and Flask”, 05/12/2020. Địa chỉ: <https://programminghistorian.org/en/lessons/creating-apis-with-python-and-flask> [Truy cập ngày 25/12/2021].