**PHẦN I - GIỚI THIỆU**

1. **Lý do chọn đề tài**

Những năm gần đây, chúng ta đã chứng kiến những sự tăng trưởng bùng nổ của các sàn Thương mại điện tử trong nước. Đặc biệt, trong bối cảnh dịch bệnh COVID-19, thị trường thương mại điện tử đang trở nên sôi động hơn với nhiều biến động đối với nền kinh tế. Một khảo sát mới nhất của Nielsen đưa ra tại Diễn đàn Tiếp thị trực tuyến 2020 cho thấy, số người tiêu dùng mua sắm online tăng lên 25%, trong khi ở các kênh truyền thống như siêu thị, chợ và tạp hóa chỉ tăng lần lượt 7%, 3% và 6%.

Từ đó, chúng ta có thể thấy rằng nhu cầu mua sắm online trên các trang thương mại điện tử của khách hàng đang tăng lên rất nhiều và đặc biệt trong khoảng thời gian dài giãn cách xã hội vừa qua.

Người tiêu dùng dễ dàng tiếp cận thông tin sản phẩm từ nhiều nguồn khác nhau: trang thương mại điện tử (Shopee, Tiki, Lazada…), mạng xã hội (Facebook, Instagram,…). Đặc biệt, có những trang web cung cấp dịch vụ so sánh giá sản phẩm từ các trang web khác đã hỗ trợ người tiêu dùng rất tốt trong việc mua hàng qua mạng. Tại đây cung cấp rất đa dạng các loại sản phẩm, dịch vụ đặc biệt là các mặt hàng quần áo, đồ điện tử, đồ chơi, mỹ phẩm, đồ gia dụng và cả thực phẩm.

Đặc biệt, hầu hết các sàn giao dịch thương mại điện tử đều cho phép khách hàng đánh giá sản phẩm sau khi mua. Vì thế, người tiêu dùng có thể tìm hiểu về sản phẩm một cách kỹ lưỡng trước khi quyết định mua và tiết kiệm thời gian cho người tiêu dùng, đặc biệt giúp hạn chế tiếp xúc trong thời gian dịch bệnh, giãn cách.

Và cách để người tiêu dùng mua được một sản phẩm chất lượng theo đúng lời quảng cáo của nhà cung cấp chính là đọc những phản hồi về sản phẩm của những khách hàng đã mua trước đó. Nhưng vấn đề đặt ra là chúng ta có thể đọc hết được những bình luận để đánh giá được sản phẩm đó không? Nếu dành quá nhiều thời gian để đọc hết các bình luận đó thì sẽ đi ngược với lợi ích của việc mua sắm online đề ra là tiết kiệm thời gian.

Từ ý tưởng đó, trong đồ án tốt nghiệp này em xin chọn đề tài **“Ứng dụng học máy vào bài toán chấm điểm phản hồi của khách hàng về sản phẩm”.** Với mong muốn xây dựng một hệ thống sử dụng học máy (Machine Learning) để đánh giá các sản phẩm trên các trang thương mại điện tử, cụ thể trong bài toán này là trên trang Shopee và Tiki với sản phẩm hướng đến là quần áo mùa hè.

1. **Mục tiêu**

* Thu thập dữ liệu là các bình luận trên trang TMĐT Tiki và Shopee về mặt hàng quần áo mùa hè.
* Tìm hiểu thuật toán Tokenizer để chuẩn hóa dữ liệu
* Tìm hiểu model PhoBERT và model SVM để xây dựng    mô hình cho bài toán.
* Tìm hiểu Flask Framework để xây dựng web demo

1. **Nội dung**

Báo cáo đồ án tốt nghiệp gồm có 3 chương nội dung chính, ngoài ra có các phần mở đầu, kết luận, tài liệu tham khảo... Cụ thể các chương nội dung chính như sau:

**Chương 1**: Tổng quan về học máy trong bài toán chấm điểm bình luận

**Chương 2**: Áp dụng học máy vào bài toán chấm điểm phản hồi của khách hàng về sản phẩm

**Chương 3**: Thực nghiệm và đánh giá

**CHƯƠNG 1: TỔNG QUAN VỀ HỌC MÁY TRONG BÀI TOÁN CHẤM ĐIỂM PHẢN HỒI VỀ SẢN PHẨM**

**1.1 Tổng quan về đề tài**

***1.1.1 Đặt vấn đề bài toán chấm điểm phản hồi***

Dịch bệnh Covid-19 xảy ra và ngày càng diễn biến phức tạp, nhiều người dân hạn chế ra đường để phòng, tránh lây nhiễm bệnh đã làm thay đổi xu hướng tiêu dùng. Đây chính là lý do vô cùng hợp lý để môi trường mua sắm online bùng nổ, đẩy mạnh hoạt động mua bán trên các trang thương mại điện tử và thông qua mạng xã hội.

Ưu điểm của mua sắm online là tiết kiệm và chủ động được về thời gian, dễ dàng so sánh giá cả, thanh toán an toàn và tiện lợi. Hầu hết các sàn giao dịch thương mại điện tử cho phép khách hàng đánh giá sản phẩm sau khi mua. Vì thế, người tiêu dùng có thể tìm hiểu và đọc các phản hồi về sản phẩm của những người mua trước một cách kỹ lưỡng trước khi quyết định mua.

Khác với hình thức mua sắm trực tiếp, người tiêu dùng có thể nhìn, cầm, đánh giá trực tiếp sản phẩm, thì mua sắm trực tuyến hạn chế người tiêu dùng trong việc đánh giá sản phẩm. Người tiêu dùng chỉ có thể nhìn hình ảnh sản phẩm và đọc các lời giới thiệu, quảng cáo về sản phẩm của đơn vị bán cùng cấp.

Giải pháp hữu hiệu nhất giúp khách hàng mua được sản phẩm chất lượng theo đúng lời quảng cáo của người bán đó là đọc các phản hồi về sản phẩm cũng những khách hàng đã mua. Những đánh giá chân thực này sẽ là thước đo khách quan nhất mà chúng ta có thể tham khảo trước khi đưa ra quyết định mua một sản phẩm. Việc đọc các phản hồi trước khi mua là việc rất quen thuộc của mọi người tiêu dùng hiện nay, nhưng để đọc được lần lượt các phản hồi của người mua thì sẽ rất mất thời gian, điều này là đi ngược lại với lợi ích của việc mua sắm online.

Chính vì vậy, mà chúng ta cần giải quyết được vấn đề không cần mất nhiều thời gian cho việc đọc phản hồi về sản phẩm của những người mua trước mà vẫn mua được những sản phẩm chất lượng tốt theo đúng lời quảng cáo của người bán. Và cụ thể trong bài toán này, em đã **áp dụng học máy để chấm điểm phản hồi của khách hàng về sản phẩm** trên các trang thương mại điện tử, cụ thể là trên Shopee và Tiki với sản phẩm hướng đến là quần áo mùa hè.

***1.1.2 Tính cấp thiết của đề tài***

Ngày nay, với sự phát triển vượt bậc của khoa học và công nghệ, đặc biệt là sự bùng nổ của Internet với các phương tiện truyền thông xã hội, thương mại điện tử,... đã cho phép mọi người không chia sẻ thông tin, mua sản phẩm trực tuyến mà còn thể hiện thái độ, quan điểm của mình đối với các sản phẩm, dịch vụ và các vấn đề xã hội khác. Vì vậy mà Internet đã trở lên vô cùng quan trọng và là nguồn cung cấp một lượng thông tin vô cùng lớn và quan trọng.

Chúng ta có thể nhận thấy rằng, mỗi ở các trang thương mại điện tử có rất nhiều bình luận về sản phẩm được người dùng đưa ra đánh giá về những sản phẩm đó**.** Tham khảo các thông tin phản hồi của khách hàng mua sản phẩm trước đó là một giải pháp tốt nhất để biết được chất lượng sản phẩm có tốt đúng như nhà cung cấp giới thiệu, từ đó giúp bạn đưa ra được quyết định đúng đắn là có nên hay không nên mua.

Những phản hồi, đánh giá sản phẩm này cũng là một giải pháp tuyệt vời giúp cho các khác doanh nghiệp hiểu được điểm mạnh, điểm yếu trong sản phẩm, dịch vụ của mình; từ đó nhanh chóng nắm bắt được tâm lý và nhu cầu khách hàng để mang đến cho họ sản phẩm, dịch vụ hoàn hảo nhất.

Vì có rất nhiều lượt comment, trên các trang thương mại điện tử lớn có thể lên tới hàng chục triệu lượt comment trong một ngày nên việc các khách hàngphân tích bằng tay truyền thống là điều không thể. Đây là lúc các mô hình Machine Learning thể hiện sức mạnh của mình.

Trong đồ án này, em sẽ xây dựng một model Machine Learning kết hợp giữa SVM và PhoBert để giải quyết bài toán chấm điểm phản hồi của khách hàng về sản phẩm.

**1.2  Cơ sở lý thuyết**

***1.2.1 Tổng quan về học máy***

*1.2.1.1 Khái niệm học máy*

Học máy (Machine Learning) là một tập con của Trí tuệ nhân tạo (Artificial Intelligence), nơi mà các thuật toán máy tính được sử dụng để tự học từ dữ liệu và thông tin. Trong machine learning, các máy tính không cần phải được lập trình một cách rõ ràng nhưng có thể tự thay đổi và cải thiện các thuật toán của chúng.

Trong vài năm gần đây, học máy gây nên cơn sốt công nghệ trên toàn thế giới. Trong giới học thuật mỗi năm có hàng ngàn bài báo khoa học về đề tài này. Còn đối với giới công nghiệp, các công Học máy.

Hiện nay, các thuật toán học máy cho phép máy tính có thể giao tiếp với con người, xe hơi tự lái, viết và xuất bản tường thuật các trận đấu thể thao và thậm trí tìm thấy kẻ tình nghi khủng bố. Điều này cho thấy rằng, học máy đã và đang tác động một cách sâu sắc đến mọi ngành công nghiệp và các công việc liên quan đến chúng, đây là lý do tại sao mọi nhà quản lý cần phải có ít nhất một số kiến thức về học máy và cách nó đã phát triển.

*1.2.1.2 Sự ra đời của học máy*

Điều khác biệt chính giữa con người và máy tính là con người học hỏi từ những kinh nghiệm được đúc kết từ trong quá khứ, nhưng với máy tính hoặc máy móc thì cần được thực hiện theo một quy trình có sẵn. Máy tính là những thiết bị hoạt động một cách nghiêm ngặt với ý nghĩa thông thường.

Như vậy chúng ta có thể hiểu được rằng, nếu muốn máy làm một việc gì đó chúng ta cần cung cấp cho nó những quy trình và hướng dẫn chi tiết từng bước về chính xác những việc cần làm.

Chính vì vậy, con người đã viết ra các kịch bản và lập trình để máy tính làm theo hướng dẫn và có khả năng tự học hỏi. Đây là cách mà Học máy (Machine learning) ra đời.

Khái niệm máy học chính xác là máy tính học hỏi từ các dữ liệu có trong quá khứ và rút kinh nghiệm qua thời gian để thực hiện các việc chúng ta muốn một cách hoàn chỉnh nhất.

*1.2.1.3 Cách thức làm việc của học máy*

Machine learning (Học máy) được hướng dẫn để sử dụng bộ dữ liệu đào tạo. Từ đó, tạo ra một mô hình nguyên mẫu. Kỹ thuật thuật toán này sẽ tiếp nhận dữ liệu mới, sau đó sẽ đưa ra những dự đoán phân tích dựa trên nguyên mẫu căn bản.

Những phân tích nói trên sẽ được đánh giá về độ chính xác. Nếu độ chính xác này được công nhận, thuật toán Machine learning sẽ tiến hành triển khai.  Nếu độ chính xác không được công nhận, thuật toán này sẽ được hướng dẫn lại nhiều lần với một bộ dữ liệu hướng dẫn tăng dần.

Những phân tích nói trên sẽ được đánh giá về độ chính xác. Nếu độ chính xác này được công nhận, thuật toán Machine learning sẽ tiến hành triển khai. Nếu độ chính xác không được công nhận, thuật toán này sẽ được hướng dẫn lại nhiều lần với một bộ dữ liệu hướng dẫn tăng dần.

*1.2.1.4 Phân loại các phương pháp học máy*

Xét theo phương thức học, các thuật toán Machine Learning được chia làm bốn nhóm, bao gồm “Học có giám sát” (Supervised Learning), “Học không giám sát” (Unsupervised Learning), “Học bán giám sát” (Semi-Supervised Learning) và “Học tăng cường” (Reinforcement Learning).

* **Học có giám sát (Supervised Learning)**

Mô hình học có giám sát là mô hình học trên dữ liệu có dãn nhãn, tức là mục tiêu của bài toán Machine learning cần học đã được gán nhãn sẵn trong dữ liệu huấn luyện. Dữ liệu đầu vào của quá trình học bao gồm cả vector đầu vào chứa các thuộc tính của dữ liệu lẫn giá trị đầu ra mục tiêu (gọi là nhãn của dữ liệu).

Nói cách khác học có giám sát cho phép dự đoán đầu ra *(outcome)* của một dữ liệu mới dựa trên các cặp *(input, outcome)* đã biết từ trước thu được từ bộ dữ liệu huấn luyện. Bộ dữ liệu huấn luyện bao gồm các cặp (*data, label*), tức (*dữ liệu, nhãn*). Học có giám sát là nhóm phổ biến nhất trong các thuật toán Machine Learning.

Các thuật toán học có giám sát còn được phân ra thành hai loại chính là phân lớp (Classification) và hồi quy (Regression).

* **Phân lớp:** Một bài toán được gọi là phân lớp nếu nhãn của dữ liệu đầu vào được chia thành một số hữu hạn lớp (miền giá trị là rời rạc)

Ví dụ: Trong nhận dạng chữ số viết tay, ta có ảnh của hàng nghìn trường hợp ứng với mỗi chữ số được viết bởi nhiều người khác nhau. Ta đưa các bức ảnh này vào một thuật toán học và chỉ cho nó biết “mỗi bức ảnh tương ứng với chữ số nào”. Sau khi thuật toán tạo ra một mô hình, tức là một hàm số nhận đầu vào là một bức ảnh và cho ra kết quả là một chữ số. Khi nhận được một bức ảnh mới mà mô hình “chưa từng gặp qua” và nó sẽ dự đoán xem bức ảnh đó tương ứng với chữ số nào.

Thuật toán nổi tiếng trong phân lớp: K-Nearest Neighbors, Mạng Nơron nhân tạo, SVM,…

* **Hồi quy:** Một bài toán được xem là hồi quy nếu nhãn không được chia thành các nhóm mà là một giá trị thực cụ thể (miền giá trị là liên tục). Hầu hết các bài toán dự báo (giá cổ phiếu, giá nhà, …) thường được xếp vào bài toán hồi quy

Ví dụ: Một căn nhà rộng 150 m^2, có 7 phòng và cách trung tâm thành phố 10km sẽ có giá là bao nhiêu? Lúc này kết quả dự đoán sẽ là một số thực.

Thuật toán nổi tiếng trong hồi quy: Linear Regression, Logistic Regression, …

* **Học không giám sát (Unsupervised Learning)**

Học không giám sát là một lớp mô hình học sử dụng một thuật toán để mô tả hoặc trích xuất ra các mối quan hệ tiềm ẩn trong dữ liệu. Khác với học có giám sát, học không giám sát chỉ thực thi trên dữ liệu đầu vào không cần các thuộc tính nhãn, hoặc mục tiêu của việc học. Tức là, chúng ta không biết được *outcome* hay *nhãn* mà chỉ có dữ liệu đầu vào. Các dữ liệu không được “hướng dẫn” trước như trong trường hợp học có giám sát. Các thuật toán cần học được từ dữ liệu mà không hề có bất cứ sự hướng dẫn nào.

Các bài toán Unsupervised learning tiếp tục được chia nhỏ thành hai loại là phân cụm (Clustering) và luật kết hợp (Association Rule).

* **Phân cụm**

Một bài toán phân cụm/phân nhóm toàn bộ dữ liệu X thành các nhóm/cụm nhỏ dựa trên sự liên quan giữa các dữ liệu trong mỗi nhóm. Chẳng hạn như phân nhóm khách hàng dựa vào độ tuổi, giới tính

* **Luật kết hợp**

Là bài toán khi chúng ta muốn khám phá ra một quy luật dựa trên nhiều dữ liệu cho trước. Ví dụ những khách hàng nam mua quần áo thường có xu hướng mua thêm đồng hồ hoặc thắt lưng, dựa vào đó tạo ra một hệ thống gợi ý khách hàng (Recommendation System) nhằm thúc đẩy nhu cầu mua sắm.

Một số thuật toán thuộc nhóm học không giám sát như Apriori (Association Rule), k-Means (Clustering), …

* **Học bán giám sát (Semi-Supervised Learning)**

Các bài toán khi chúng ta có một lượng lớn dữ liệu X nhưng chỉ một phần trong chúng được gán nhãn sẽ được gọi là học bán giám sát.

Học bán giám sát rất hiệu quả trong nhiều bài toán thực tế, khi tài nguyên sử dụng để gán nhãn dữ liệu rất tốn kém. Chẳng hạn trong các bài toán thị giác máy tính (dữ liệu ảnh), xử lý ngôn ngữ tự nhiên (dữ liệu văn bản), nhận dạng giọng nói (dữ liệu âm thanh).

Học bán giám sát có thể làm tăng độ chính xác cho các mô hình học có giám sát bằng cách tận dụng thông tin từ các dữ liệu không gán nhãn. Các dữ liệu này vốn rất phổ biến trên môi trường Internet và có thể dễ dàng kiếm được hơn là số lượng hạn chế các dữ liệu có gán nhãn.

* **Học tăng cường (Reinforcement Learning)**

Học tăng cường hoặc học củng cố là các bài toán giúp cho một hệ thống tự động xác định hành vi dựa trên hoàn cảnh để đạt được lợi ích cao nhất (maximizing the performance). Hiện tại, học tăng cường chủ yếu được áp dụng vào Lý Thuyết Trò Chơi (Game Theory), các thuật toán cần xác định nước đi tiếp theo để đạt được điểm số cao nhất.

Ví dụ: Trò chơi cờ vây của AlphaGo hoặc huấn luyện máy tính chơi game Mario.

*1.2.1.5 Ứng dụng của học máy*

* **Tự động phân loại**

Phân loại tin tức là một ứng dụng điểm chuẩn khác của phương pháp học máy.Vận dụng như thế nào? Như một vấn đề thực tế là bây giờ khối lượng thông tin đã tăng lên rất nhiều trên web. Tuy nhiên, mỗi người có sở thích hoặc lựa chọn cá nhân của mình. Vì vậy, để chọn hoặc thu thập một phần thông tin phù hợp trở thành một thách thức đối với người dùng từ vô số nội dung trên trang web.

Phân loại các danh mục một cách rõ ràng, dễ điều hướng giúp cho các khách hàng mục tiêu chắc chắn sẽ tăng khả năng truy cập các trang tin tức. Hơn nữa, độc giả hoặc người dùng có thể tìm kiếm tin tức cụ thể một cách hiệu quả và nhanh chóng.

Có một số phương pháp học máy trong mục đích này, tức là, máy vectơ hỗ trợ, naive Bayes, k-nearest neighbor, v.v.

* **Ứng dụng trong các mạng xã hội**

Học máy đang được sử dụng trong một loạt các ứng dụng ngày nay. Một trong những ví dụ nổi tiếng nhất là Facebook News Feed. Nguồn cấp tin tức sử dụng học máy để cá nhân hóa từng nguồn cấp dữ liệu thành viên.

Nếu một thành viên thường xuyên dừng lại để đọc hoặc thích một bài đăng của một người bạn cụ thể, News Feed sẽ bắt đầu hiển thị nhiều hơn về hoạt động của người bạn đó trước đó trong nguồn cấp dữ liệu.

Đằng sau hệ thống ấy, phần mềm sử dụng phân tích thống kê và phân tích dự đoán để xác định các mẫu trong dữ liệu người dùng và sử dụng các mẫu đó để điền vào News Feed. Nếu thành viên không còn dừng lại để đọc, thích hoặc bình luận trên các bài đăng của bạn bè, dữ liệu mới đó sẽ được bao gồm trong tập dữ liệu và News Feed sẽ điều chỉnh tương ứng.

Không chỉ riêng facebook, ta có thể bắt gặp những tính năng tương tự đó qua các mạng xã hội khác như google, instagram, ....

* **Nhận diện hình ảnh**

Nhận dạng hình ảnh là một trong những ví dụ về máy học và trí tuệ nhân tạo phổ biến nhất. Về cơ bản, nó là một cách tiếp cận để xác định và phát hiện các đặc trưng của một đối tượng trong hình ảnh kỹ thuật số. Hơn nữa, kỹ thuật này có thể được sử dụng để phân tích sâu hơn, chẳng hạn như nhận dạng mẫu, nhận diện hình khuôn, nhận dạng khuôn mặt, nhận dạng ký tự quang học và nhiều hơn nữa.

* **Trong y tế**

Machine learning là 1 xu hướng phát triển nhanh chóng trong ngành chăm sóc sức khỏe, nhờ vào sự ra đời của các thiết bị và máy cảm ứng đeo được sử dụng dữ liệu để đánh giá tình hình sức khỏe của bệnh nhân trong thời gian thực (real-time). Công nghệ Machine Learning còn giúp các chuyên gia y tế xác định những xu hướng hoặc tín hiệu để cải thiện khả năng điều trị, chẩn đoán bệnh.

***1.2.2 Các thuật toán phân lớp trong bài toán học máy***

#### *1.2.2.1* *Phương pháp K-Nearest Neighbor classifier (KNN)*

**Giới thiệu**

K-nearest neighbor (KNN) là một trong những thuật toán Supervised-Learning đơn giản nhất (mà hiệu quả trong một vài trường hợp) trong Machine Learning. Khi training, thuật toán này không học một điều gì từ dữ liệu training (đây cũng là lý do thuật toán này được xếp vào loại lazy learning), mọi tính toán được thực hiện khi nó cần dự đoán kết quả của dữ liệu mới. KNN có thể áp dụng được vào cả hai loại của bài toán Supervised learning là Classification và Regression. KNN còn được gọi là một thuật toán Instance-based hay Memory-based learning.

Với KNN, trong bài toán Phân lớp (Classification), label của một điểm dữ liệu mới (hay kết quả của câu hỏi trong bài thi) được suy ra trực tiếp từ K điểm dữ liệu gần nhất trong training set. Label của một test data có thể được quyết định bằng major voting (bầu chọn theo số phiếu) giữa các điểm gần nhất, hoặc nó có thể được suy ra bằng cách đánh trọng số khác nhau cho mỗi trong các điểm gần nhất đó rồi suy ra label.

Trong bài toán Hồi quy (Regresssion), đầu ra của một điểm dữ liệu sẽ bằng chính đầu ra của điểm dữ liệu đã biết gần nhất (trong trường hợp K=1), hoặc là trung bình có trọng số của đầu ra của những điểm gần nhất, hoặc bằng một mối quan hệ dựa trên khoảng cách tới các điểm gần nhất đó.

Một cách ngắn gọn, KNN là thuật toán đi tìm đầu ra của một điểm dữ liệu mới bằng cách chỉ dựa trên thông tin của K điểm dữ liệu trong training set gần nó nhất (K-lân cận), không quan tâm đến việc có một vài điểm dữ liệu trong những điểm gần nhất này là nhiễu.

Ví dụ trên đây là bài toán về KNN trong Classification K = 1 với 3 classes: Đỏ, Lam, Lục. Mỗi điểm dữ liệu mới (test data point) sẽ được gán label theo màu của điểm mà nó thuộc về. Trong hình này, có một vài vùng nhỏ xem lẫn vào các vùng lớn hơn khác màu. Ví dụ có một điểm màu Lục ở gần góc 11 giờ nằm giữa hai vùng lớn với nhiều dữ liệu màu Đỏ và Lam. Điểm này rất có thể là nhiễu. Dẫn đến nếu dữ liệu test rơi vào vùng này sẽ có nhiều khả năng cho kết quả không chính xác.

**Mô tả thuật toán**

* Xác định giá trị tham số K (số láng giềng gần nhất).
* Tính khoảng cách giữa điểm truy vấn phân lớp với tất cả các đối tượng trong tập dữ liệu huấn luyện.
* Sắp xếp khoảng cách theo thứ tự tăng dần và xác định K láng giềng gần nhất với điểm truy vấn phân lớp.
* Lấy tất cả các lớp của K láng giềng gần nhất đã xác định.
* Dựa vào phần lớn lớp của láng giềng gần nhất để xác định lớp cho điểm truy vấn phân lớp, lớp của điểm truy vấn phân lớp được định nghĩa là lớp chiếm đa số trong K láng giềng gần nhất.

Việc tính khoảng cách giữa các đối tượng cần phân lớp với tất cả các đối tượng trong tập dữ liệu huấn luyện được thường được sử dụng với công thức tính khoảng cách Euclidean. Cho 2 điểm P1(x1,y1) và P2(x2,y2) thì khoảng cách Euclidean distance sẽ được tính theo công thức:

*d* =

**Ưu điểm của KNN**

* Độ phức tạp tính toán của quá trình huấn luyện là bằng 0.
* Việc dự đoán kết quả của dữ liệu mới rất đơn giản (sau khi đã xác định được các điểm lân cận).
* Không cần giả sử về phân phối của các lớp.

**Nhược điểm của KNN**

* KNN rất nhạy cảm với nhiễu khi K nhỏ.
* Như đã nói, KNN là một thuật toán mà mọi tính toán đều nằm ở khâu kiểm thử, trong đó việc tính khoảng cách tới từng điểm dữ liệu trong tập huấn luyện tốn rất nhiều thời gian, đặc biệt là với các cơ sở dữ liệu có số chiều lớn và có nhiều điểm dữ liệu. Với K càng lớn thì độ phức tạp cũng sẽ tăng lên. Ngoài ra, việc lưu toàn bộ dữ liệu trong bộ nhớ cũng ảnh hưởng tới hiệu năng của KNN.

*1.2.2.2 Thuật toán Support Vector Machines (SVM)*

* **Giới thiệu**

SVM (Support Vector Machine) là một thuật toán học máy có giám sát được sử dụng rất phổ biến ngày nay trong các bài toán phân lớp (classification) hay hồi quy (Regression). Tuy nhiên nó được sử dụng chủ yếu cho bài toán phân lớp, đây là hai bài toán cơ bản và có rất nhiều ứng dụng trong tất cả các lĩnh vực như: học máy, nhận dạng, trí tuệ nhân tạo, .v.v. Và phương pháp Support Vector Machines (SVM) sẽ là một trong những phương pháp được áp hiệu quả để giải quyết bài toán trong đồ án lần này.

SVM được đề xuất bởi Vladimir N. Vapnik và các đồng nghiệp của ông vào năm 1963 tại Nga và sau đó trở nên phổ biến trong những năm 90 nhờ ứng dụng giải quyết các bài toán phi tuyến tính (nonlinear) bằng phương pháp Kernel Trick.

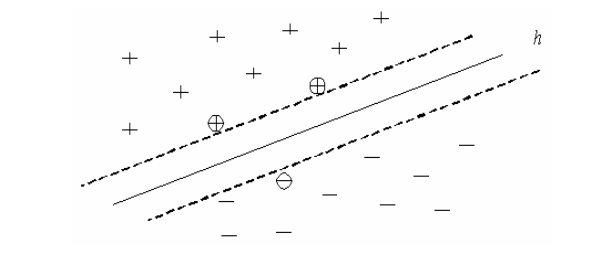
Phương pháp này thực hiện phân lớp dựa trên nguyên lý Cực tiểu hóa Rủi ro có Cấu trúc SRM (Structural Risk Minimization), được xem là một trong các phương pháp phân lớp giám sát không tham số tinh vi nhất cho đến nay. Các hàm công cụ đa dạng của SVM cho phép tạo không gian chuyển đổi để xây dựng mặt phẳng phân lớp

Gần đây, chúng cực kỳ phổ biến vì khả năng xử lý nhiều biến liên tục và phân loại.  Đưa ra một tập hợp các ví dụ đào tạo, mỗi ví dụ được đánh dấu thuộc về một hoặc hai loại khác, thuật toán đào tạo SVM xây dựng một mô hình gán các ví dụ mới cho một loại hoặc loại khác, làm cho nó trở thành một phân loại tuyến tính nhị phân không xác suất (mặc dù các phương thức chẳng hạn như chia tỷ lệ tồn tại để sử dụng SVM trong cài đặt phân loại xác suất).

* **Ý tưởng thuật toán**

Ý tưởng chính của thuật toán này là cho trước một tập huấn luyện được biểu diễn trong không gian vector trong đó mỗi tài liệu là một điểm, phương pháp này tìm ra một mặt phẳng h quyết định tốt nhất có thể chia các điểm trên không gian này thành hai lớp riêng biệt tương ứng lớp + và lớp -. Chất lượng của siêu mặt phẳng này được quyết định bởi khoảng cách (gọi là biên) của điểm dữ liệu gần nhất của mỗi lớp đến mặt phẳng này. Khoảng cách biên càng lớn thì mặt phẳng quyết định càng tốt đồng thời việc phân loại càng chính xác. Mục đích thuật toán SVM tìm ra được khoảng cách biên lớn nhất để tạo kết quả phân lớp tốt.

Hình sau minh hoạ cho thuật toán này:

**

*Hình 4. Siêu phẳng h phân chia dữ liệu huấn luyện thành 2 lớp + và - với khoảng cách biên lớn nhất. Các điểm gần h nhất là các vector hỗ trợ (Support Vector - được khoanh tròn)*

Trong chương 3 sẽ trình bày chi tiết về thuật toán SVM..

**Ưu điểm**

Là một kĩ thuật phân lớp khá phổ biến, SVM thể hiện được nhiều ưu điểm trong số đó có việc tính toán hiệu quả trên các tập dữ liệu lớn. Có thể kể thêm một số ưu điểm của phương pháp này như:

* Xử lý trên không gian số chiều cao: SVM là một công cụ tính toán hiệu quả trong không gian chiều cao, trong đó đặc biệt áp dụng cho các bài toán phân loại văn bản và phân tích quan điểm nơi chiều có thể cực kỳ lớn
* Tiết kiệm bộ nhớ: Do chỉ có một tập hợp con của các điểm được sử dụng trong quá trình huấn luyện và ra quyết định thực tế cho các điểm dữ liệu mới nên chỉ có những điểm cần thiết mới được lưu trữ trong bộ nhớ khi ra quyết định
* Tính linh hoạt - phân lớp thường là phi tuyến tính. Khả năng áp dụng Kernel mới cho phép linh động giữa các phương pháp tuyến tính và phi tuyến tính từ đó khiến cho hiệu suất phân loại lớn hơn.

**Nhược điểm**

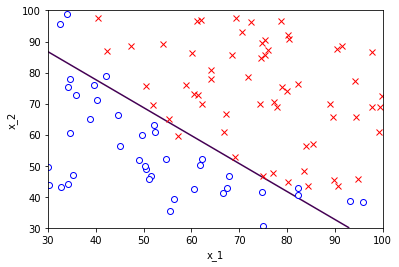
* Bài toán số chiều cao: Trong trường hợp số lượng thuộc tính (p) của tập dữ liệu lớn hơn rất nhiều so với số lượng dữ liệu (n) thì SVM cho kết quả khá tồi
* Chưa thể hiện rõ tính xác suất: Việc phân lớp của SVM chỉ là việc cố gắng tách các đối tượng vào hai lớp được phân tách bởi siêu phẳng SVM. Điều này chưa giải thích được xác suất xuất hiện của một thành viên trong một nhóm là như thế nào. Tuy nhiên hiệu quả của việc phân lớp có thể được xác định dựa vào khái niệm margin từ điểm dữ liệu mới đến siêu phẳng phân lớp mà chúng ta đã bàn luận ở trên.

#### *1.2.2.3* *Phương pháp hồi quy Logistic Regression (Logit)*

* **Giới thiệu**

Phương pháp hồi quy Logistic regression là một mô hình hồi quy nhằm dự đoán giá trị đầu ra rời rạc (discrete target variable) y ứng với một véc-tơ đầu vào x. Việc này tương đương với chuyện phân loại các đầu vào x vào các nhóm y tương ứng.

Ví dụ, xem một bức ảnh có chứa một con mèo hay không. Thì ở đây ta coi đầu ra y=1 nếu bước ảnh có một con mèo và y=0 nếu bức ảnh không có con mèo nào. Đầu vào x ở đây sẽ là các pixel một bức ảnh đầu vào.



Hình 1.5 Classification với 2 nhóm

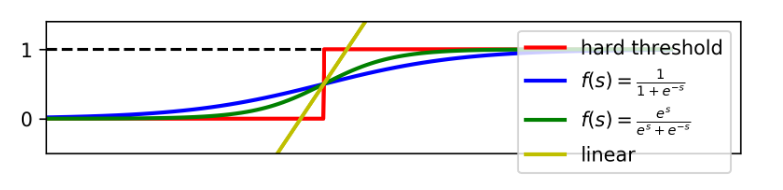
Ví dụ khác như:

* Các giao dịch thẻ tín dụng: có phải dịch gian lận hay không khi ta dựa trên thông tin lịch sử giao dịch của chúng?
* Rủi ro vay nợ: liệu một cá nhân có bị vỡ nợ với tài khoản tín dụng của mình không?
* Thị giác máy (Computer Vision): hiểu được các đối tượng xuất hiện trong ảnh.
* **Mô hình Logistic Regression**

Đầu ra dự đoán của logistic regression thường được viết chung dưới dạng:

(1-1)

Trong đó:  θ được gọi là logistic function. Một số activation cho mô hình tuyến tính được cho trong hình dưới đây:



Hình 1.6 Các activation function khác nhau

Ta nhận thấy rằng:

* Xác suất một sự kiện xảy ra nằm trong khoảng [0,1]
* Xác suất càng cao thì sự kiện đó càng chắc chắn xảy ra. Ví dụ như ôn tập càng nhiều giờ thì tỉ lệ đỗ càng lớn.

Và trong các hàm số, thì hàm sigmoid có đủ các tính chất trên và được sử dụng nhiều nhất:

f(s) = =

**Ưu điểm**

* Một điểm cộng cho Logistic Regression so với PLA là nó không cần có giả thiết dữ liệu hai class là linearly separable.
* Do tính chất đơn giản và hiệu quả, không đòi hỏi sức mạnh tính toán cao, dễ thực hiện, dễ diễn giải, được sử dụng rộng rãi bởi các nhà phân tích dữ liệu và nhà khoa học.
* Không yêu cầu mở rộng các tính năng.
* Hồi quy logistic cung cấp điểm xác suất cho các quan sát.

**Nhược điểm**

* Không thể mô hình hóa được các bài toán có hàm quyết định dạng phức tạp
* Có thể bị overﬁt dữ liệu huấn luyện. Khắc phục được bằng cách điều chỉnh trong phương pháp hợp lý cực đại (maximum likelihood)
* Không thể giải quyết vấn đề phi tuyến tính với hồi quy logistic, đó là lý do tại sao nó yêu cầu chuyển đổi các đối tượng phi tuyến tính.
* Hồi quy logistic không hoạt động tốt với các biến độc lập không tương quan với biến mục tiêu và rất giống nhau hoặc tương quan với nhau.

**CHƯƠNG 2: ÁP DỤNG HỌC MÁY VÀO BÀI TOÁN CHẤM ĐIỂM PHẢN HỒI VỀ SẢN PHẨM CỦA KHÁCH HÀNG**

## 2.1 Mô tả bài toán

## 2.2 Mô hình BERT

### 2.2.1 Sự ra đời của BERT

Thuật toán Bidirectional Encoder Representations from Transformers (BERT), một kiến trúc mới cho lớp bài toán Language Representation và bắt đầu xuất hiện trong hệ thống tìm kiếm của Google vào ngày 21 tháng 10 năm 2019, được áp dụng cho các truy vấn sử dụng ngôn ngữ là tiếng Anh và bao gồm cả các trích dẫn nổi bật (featured snippets). Thuật toán được giới thiệu là “một trong những bước nhảy vọt lớn nhất trong lịch sử tìm kiếm”.

Kể từ lúc xuất hiện BERT vẫn chưa từng nguội đi sức nóng của mình và thực tế BERT ngày càng nóng hơn bao giờ hết. Với các phiên bản cải tiến, biến thể như RoBERTa, ALBERT, DistilBERT,... BERT đã càn quét các tác vụ xử lý ngôn ngữ tự nhiên, trở lên áp đảo trong các nền tảng thi đấu như Kaggle, AIVIVN cũng như shared task của nhiều hội nghị.

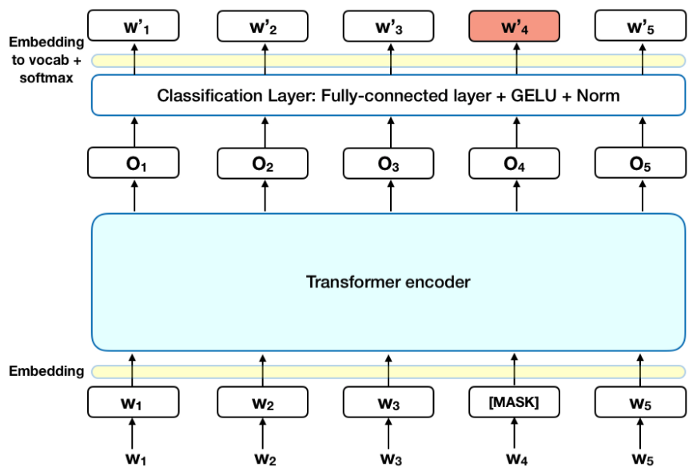
Ban đầu thuật thuật toán BERT chỉ hỗ trợ ngôn ngữ Tiếng Anh, nhưng hiện tại đã mở rộng được phạm vi áp dụng lên đến 70 ngôn ngữ, trong đó có tiếng Việt - PhoBERT. Ngoài ra, mô hình BERT này cũng sẽ được sử dụng để cải thiện chất lượng của các trích dẫn nổi bật phản hồi cho người dùng tại hơn 20 quốc gia.

### 2.2.2 Giới thiệu mô hình BERT

BERT sử dụng Transformer là một mô hình attention (attention mechanism) học mối tương quan giữa các từ (hoặc 1 phần của từ) trong một văn bản. Transformer gồm có 2 phần chính: Encoder và Decoder, encoder thực hiện đọc dữ liệu đầu vào và decoder đưa ra dự đoán. Ở đây, BERT chỉ sử dụng Encoder.

Khác với các mô hình directional (các mô hình chỉ đọc dữ liệu theo 1 chiều duy nhất - trái→phải, phải→ trái) đọc dữ liệu theo dạng tuần tự, Encoder đọc toàn bộ dữ liệu trong 1 lần, việc này làm cho BERT có khả năng huấn luyện dữ liệu theo cả hai chiều, qua đó mô hình có thể học được ngữ cảnh (context) của từ tốt hơn bằng cách sử dụng những từ xung quanh nó (phải & trái).

* BERT là một mô hình khổng lồ, với 24 khối Transformer, 1024 lớp ẩn và 340 triệu tham số.
* Mô hình này được huấn luyện trước với 40 vòng lặp trên tập huấn luyện 3.3 tỷ từ, bao gồm BooksCorpus (800 triệu từ) và Wikipedia tiếng Anh (2.5 tỷ từ).
* Mô hình chạy trên 16 TPU pods để huấn luyện.

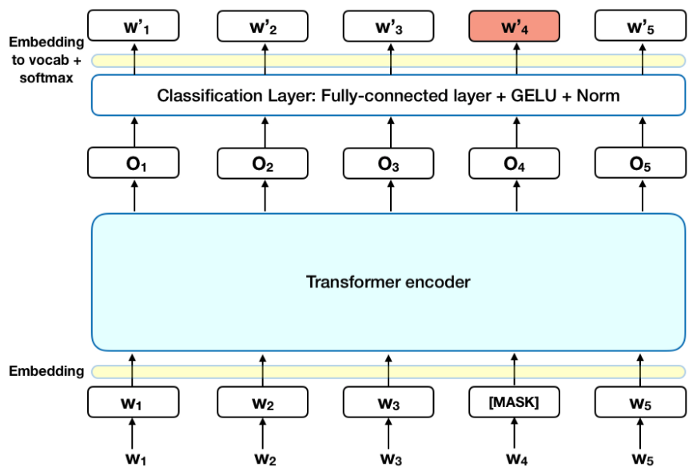


Hình trên mô tả nguyên lý hoạt động của Encoder. Theo đó, input đầu vào là một chuỗi các token w1, w2, ... được biểu diễn thành chuỗi các vector trước khi đưa vào trong mạng neural. Output của mô hình là chuỗi các vector có kích thước đúng bằng kích thước input. Trong khi huấn luyện mô hình, một thách thức gặp phải là các mô hình directional truyền thống gặp giới hạn khi học ngữ cảnh của từ. Để khắc phục nhược điểm của các mô hình cũ, BERT được train đồng thời 2 task gọi là **Masked LM** (để dự đoán từ thiếu trong câu) và **Next Sentence Prediction** (NSP – dự đoán câu tiếp theo câu hiện tại). Hai phần được train đồng thời và loss tổng sẽ là kết hợp loss của 2 task và model sẽ cố gắng minimize loss tổng này. Chi tiết 2 task này như sau:

* **Masked ML (MLM)**

Masked ML là một tác vụ cho phép chúng ta fine-tuning lại các biểu diễn từ trên các bộ dữ liệu unsupervised-text bất kỳ. Chúng ta có thể áp dụng Masked ML cho những ngôn ngữ khác nhau để tạo ra biểu diễn embedding cho chúng. Các bộ dữ liệu của tiếng anh có kích thước lên tới vài vài trăm tới vài nghìn GB được huấn luyện trên BERT đã tạo ra những kết quả khá ấn tượng.

Bên dưới là sơ đồ huấn luyện BERT theo tác vụ Masked ML



**Hình 5:** Sơ đồ kiến trúc BERT cho tác vụ Masked ML.

Theo đó:

* Trước khi đưa vào BERT, thì 15% số từ trong chuỗi được thay thế bởi token **[MASK]**, khi đó mô hình sẽ dự đoán từ được thay thế bởi **[MASK]** với context là các từ không bị thay thế bởi **[MASK]**.
* Để tính toán phân phối xác suất cho từ output, chúng ta thêm một Fully connect layer ngay sau Transformer Encoder. Hàm softmax có tác dụng tính toán phân phối xác suất. Số lượng units của fully connected layer phải bằng với kích thước của từ điển.
* Cuối cùng ta thu được véc tơ nhúng của mỗi một từ tại vị trí MASK sẽ là embedding vector giảm chiều của véc tơ Oi sau khi đi qua fully connected layer như mô tả trên hình vẽ bên phải.

Hàm lỗi (loss function) của BERT chỉ tập trung vào đánh giá các từ được đánh dấu **[MASKED]** mà bỏ qua những từ còn lại, do đó mô hình hội tụ chậm hơn so với các mô hình directional, nhưng chính điều này giúp cho mô hình hiểu ngữ cảnh tốt hơn. (Trên thực tế, con số 15% không phải là cố định mà có thể thay đổi theo mục đích của bài toán.)

* **Next Sentence Prediction (NSP)**

Trong chiến lược này, thì mô hình sử dụng một cặp câu là dữ liệu đầu vào và dự đoán câu thứ 2 là câu tiếp theo của câu thứ 1 hay không. Trong quá trình huấn luyện, 50% lượng dữ liệu đầu vào là cặp câu trong đó câu thứ 2 thực sự là câu tiếp theo của câu thứ 1  (output là 1), 50% còn lại thì câu thứ 2  (output là 0) được chọn ngẫu nhiên từ tập dữ liệu.

Cụ thể cách train như sau:

* Bước 1: Ghép 2 câu vào nhau và thêm 1 số token đặc biệt để phân tách các câu. Token [CLS] thêm vào đầu cầu Thứ nhất, token [SEP] thêm vào cuối mỗi câu.

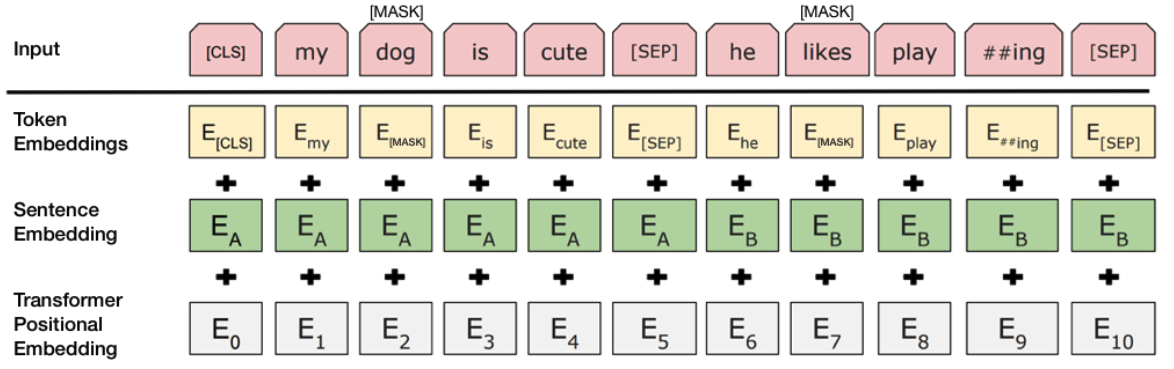
Ví dụ ghép 2 câu “Hôm nay em đi học” và “Học ở trường rất hay” thì sẽ thành [CLS] Hôm nay em đi học [SEP] Học ở trường rất vui [SEP]

* Bước 2: Mỗi token trong câu sẽ được cộng thêm một vector gọi là Sentence Embedding, thực ra là đánh dấu xem từ đó thuộc câu Thứ nhất hay câu thứ 2 thôi.

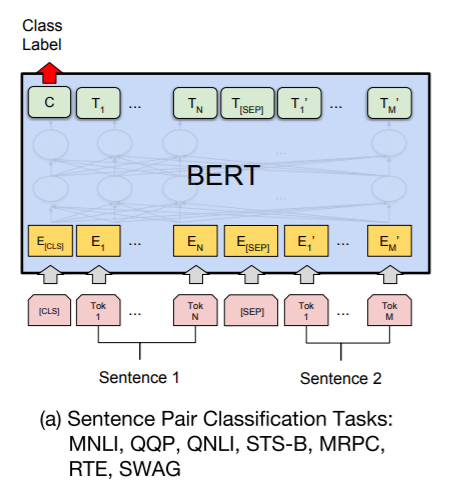
Ví dụ nếu thuộc câu Thứ nhất thì cộng thêm 1 vector toàn số “0” có kích thước bằng Word Embedding, và nếu thuộc câu thứ 2 thì cộng thêm một vector toàn số “1”.

* Bước 3: Sau đó các từ trong câu đã ghép sẽ được thêm vector Positional Encoding vào để đánh dấu vị trí từng từ trong câu đã ghép.
* Bước 4: Đưa chuỗi sau bước 3 vào mạng.
* Bước 5: Lấy encoder output tại vị trí token [CLS] được transform sang một vector có 2 phần tử [c1 c2].
* Bước 6: Tính softmax trên vector đó và output ra probality của 2 class: Đi sau và Không đi sau. Để thể hiện câu thứ hai là đi sau câu thứ nhất hay không, ta lấy chỉ cần lấy argmax.

Các bước tạo Inputs:



Cách lấy output đầu ra:



* Các kiến trúc model BERT

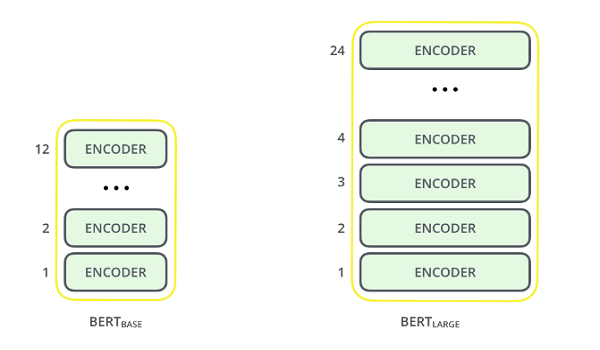
Hiện tại có nhiều phiên bản khác nhau của model BERT. Các phiên bản đều dựa trên việc thay đổi kiến trúc của Transformer tập trung ở 3 tham số là L: số lượng các block sub-layers trong transformer, H: kích thước của embedding véc tơ (hay còn gọi là hidden size), A: Số lượng head trong multi-head layer, mỗi một head sẽ thực hiện một self-attention. Tên gọi của 2 kiến trúc bao gồm:

* BERTBASE(L=12,H=768,A=12): Tổng tham số 110 triệu.
* BERTLARGE(L=24,H=1024,A=16): Tổng tham số 340 triệu.

Như vậy ở kiến trúc BERT Large chúng ta tăng gấp đôi số layer, tăng kích thước hidden size của embedding véc tơ gấp 1.33 lần và tăng số lượng head trong multi-head layer gấp 1.33 lần.

# 2.3 Giới thiệu phương pháp PhoBERT

PhoBERT là một pre-trained được huấn luyện monolingual language, tức là chỉ huấn luyện dành riêng cho tiếng Việt. Như chúng ta đã biết ở BERT thì PhoBERT cũng là một nửa của Transformer (cụ thể là nửa Encoder) với một loạt các Block Encoder chồng lên nhau (nhiều hay ít tùy model Base hay Large):



Hai phiên bản PhoBERT là "base" và "large" Hai phiên bản PhoBERT là PhoBERTbase với 12 transformers block và PhoBERTlarge với 24 transformers block, là mô hình ngôn ngữ đơn ngữ quy mô lớn đầu tiên được đào tạo trước cho tiếng Việt. Phương pháp tiếp cận đào tạo trước của PhoBERT dựa trên [RoBERTa](https://github.com/pytorch/fairseq/blob/master/examples/roberta/README.md) , tối ưu hóa quy trình đào tạo trước [BERT](https://github.com/google-research/bert) để có hiệu suất mạnh mẽ hơn.

Kết quả thử nghiệm cho thấy PhoBERT luôn vượt trội hơn so với mô hình đa ngôn ngữ được đào tạo trước tốt nhất gần đây XLM-R (Conneau và cộng sự, 2020) và cải thiện tính năng hiện đại trong nhiều tác vụ NLP dành riêng cho Việt Nam bao gồm gắn thẻ Phần của giọng nói , Phân tích cú pháp phụ thuộc, Nhận dạng thực thể được đặt tên và Suy luận ngôn ngữ tự nhiên.

PhoBERT được train trên khoảng 20GB dữ liệu bao gồm khoảng 1GB Vietnamese Wikipedia corpus và 19GB còn lại lấy từ Vietnamese news corpus. Đây là một lượng dữ liệu khả ổn để train một mô hình như BERT.

PhoBERT sử dụng RDRSegmenter của VnCoreNLP để tách từ cho dữ liệu đầu vào trước khi qua BPE encoder.

PhoBERT chỉ sử dụng task Masked Language Model để train, bỏ đi task Next Sentence Prediction.

# 2.3 Mô hình SVM

SVM sử dụng thuật toán học nhằm xây dựng một siêu phẳng làm cực tiểu hoá độ phân lớp sai của một đối tượng dữ liệu mới. Độ phân lớp sai của một siêu phẳng được đặc trưng bởi khoảng cách bé nhất tới siêu phẳng đấy. SVM có khả năng rất lớn cho các ứng dụng được thành công trong bài toán phân lớp văn bản.

Ở chương 2, em đã giới thiệu sơ bộ về mô hình SVM. Ở chương này, em sẽ đi chi tiết về nội dung phương pháp SVM.

#### *2.3.1 Cơ sở lý thuyết*

SVM thực chất là một bài toán tối ưu, mục tiêu của thuật toán này là tìm được một không gian F và siêu phẳng quyết định f trên F sao cho sai số phân loại là thấp nhất.

Cho tập mẫu (x1, y1), (x2, y2), … (xf, y f ) } với ∈ Rn , thuộc vào hai lớp nhãn: yi ∈{-1,1} là nhãn lớp tương ứng của các xi (-1 biểu thị lớp I, 1 biểu thị lớp II).

Ta có, phương trình siêu phẳng chứa vectơ trong không gian:

. + b = 0

Đặt f() = sign(+b) =

Như vậy, f() biểu diễn sự phân lớp của )vào hai lớp như đã nêu. Ta nói yi=+1 nếu € lớp I và yi = -1 nếu € lớp II . Khi đó, để có siêu phẳng f ta sẽ phải giải bài toán sau:

Tìm min với W thỏa mãn điều kiện sau:

yi (sin(+b))1 với i

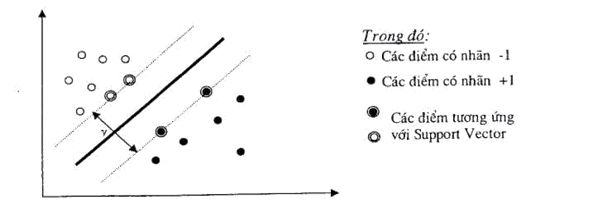
Bài toán SVM có thể giải bằng kỹ thuật sử dụng toán tử Lagrange để biến đổi về thành dạng đẳng thức. Một đặc điểm thú vị của SVM là mặt phẳng quyết định chỉ phụ thuộc các Support Vector và nó có khoảng cách đến mặt phẳng quyết định là 1/ . Cho dù các điểm khác bị xóa đi thì thuật toán vẫn cho kết quả giống như ban đầu. Đây chính là điểm nổi bật của phương pháp SVM so với các phương pháp khác vì tất cả các dữ liệu trong tập huấn luyện đều được đùng để tối ưu hóa kết quả.

**TÓM LẠI:**Trong trường hợp nhịphân phân tách tuyến tính, việc phân lớpđược thực hiện qua hàm quyết định *f(x) = sign(<w.x> + b),* hàm này thu được bằng việc thay đổi vectơ chuẩn *w*, đây là vectơ để cực đại hóa viền chức năng.

#### *2.4.2* *Bài toán phân 2 lớp với SVM*

Bài toán đặt ra là: Xác định hàm phân lớp để phân lớp các mẫu trong tương lai, nghĩa là với một mẫu dữ liệu mới xi thì cần phải xác định xi được phân vào lớp +1 hay lớp -1

Để xác định hàm phân lớp dựa trên phương pháp SVM, ta sẽ tiến hành tìm hai siêu phẳng song song sao cho khoảng cách *y* giữa chúng là lớn nhất có thể để phân tách hai lớp này ra làm hai phía. Hàm phân tách tương ứng với phương trình siêu phẳng nằm giữa hai siêu phẳng tìm được



*Hình 6: Minh họa bài toán 2 phân lớp bằng phương pháp SVM*

Các điểm mà nằm trên hai siêu phẳng phân tách được gọi là các Support Vector. Các điểm này sẽquyết định đến hàm phân tách dữliệu.

#### *2.4.3* *Bài toán nhiều phân lớp với SVM*

Để phân nhiều lớp thì kỹ thuật SVM nguyên thủy sẽ chia không gian dữ liệu thành 2 phần và quá trình này lặp lại nhiều lần. Khi đó hàm quyết định phân dữ liệu vào lớp thứ i của tập n , 2-Iớp sẽ là:

*fi(x)* = wiix + bi

Những phần tử *x* là support vector sẽ thỏa điều kiện

+1 nếu thuộc lớp i



*fi (x)* =

-1 nếu thuộc phần còn lại

Nhƣ vậy, bài toán phân nhiều lớp sử dụng phương pháp SVM hoàn toàn có thể thực hiện giống như bài toán hai lớp. Bằng cách sử dụng chiến lược *"một-* *đối-một”*(*one - against - one*).

Giả sử bài toán cần phân loại có k lớp (k > 2), chiến lược *"một-đối-một”*sẽ tiến hành k(k-l)/2 lần phân lớp nhị phân sử dụng phương pháp SVM. Mỗi lớp sẽ tiến hành phân tách với k-1 lớp còn lại để xác định k-1 hàm phân tách dựa vào bài toán phân hai lớp bằng phương pháp SVM.

#### 2.3.4 Các bước chính của phương pháp SVM

Phương pháp SVM yêu cầu dữ liệu được diễn tả như các vector của các số thực. Như vậy nếu đầu vào chưa phải là số thì ta cần phải tìm cách chuyển chúng về dạng số của SVM

* Tiền xử lý dữ liệu: Thực hiện biến đổi dữ liệu phù hợp cho quá trình tính toán, tránh các số quá lớn mô tả các thuộc tính. Thường nên co giãn (scaling) dữ liệu để chuyển về đoạn [-1, 1] hoặc [0, 1].
* Chọn hàm hạt nhân: Lựa chọn hàm hạt nhân phù hợp tƣơng ứng cho từng bài toán cụ thể để đạt được độ chính xác cao trong quá trình phân lớp.
* Thực hiện việc kiểm tra chéo để xác định các thám số cho ứng đụng. Điều này cũng quyết định đến tính chính xác của quá trình phân lớp.
* Sử dụng các tham số cho việc huấn luyện với tập mẫu. Trong quá trình huấn luyện sẽ sử dụng thuật toán tối ưu hóa khoảng cách giữa các siêu phẳng trong quá trình phân lớp, xác định hàm phân lớp trong không gian đặc trưng nhờ việc ánh xạ dữ liệu vào không gian đặc trưng bằng cách mô tả hạt nhân, giải quyết cho cả hai trường hợp dữ liệu là phân tách và không phân tách tuyến tính trong không gian đặc trưng.
* Kiểm thử tập dữ liệu Test

**CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ**

**3.1 Ngôn ngữ, công cụ và thư viện sử dụng**

**3.1.1 Ngôn ngữ**

Trong đồ án này, ngôn ngữ em sử dụng để giải quyết bài toán toán Python.

Python là một ngôn ngữ lập trình bậc cao, thông dịch, hướng đối tượng, đa mục đích và cũng là một ngôn ngữ lập trình động.

Cú pháp của Python là khá dễ dàng để học và ngôn ngữ này cũng mạnh mẽ và linh hoạt không kém các ngôn ngữ khác trong việc phát triển các ứng dụng. Python hỗ trợ mẫu đa lập trình, bao gồm lập trình hướng đối tượng, lập trình hàm và mệnh lệnh hoặc là các phong cách lập trình theo thủ tục.

Python không chỉ làm việc trên lĩnh vực đặc biệt như lập trình web, và đó là tại sao ngôn ngữ này là đa mục đích bởi vì nó có thể được sử dụng với web, enterprise, 3D CAD, …

Ta không cần sử dụng các kiểu dữ liệu để khai báo biến bởi vì kiểu của nó là động, vì thế ta có thể viết a=15 để khai báo một giá trị nguyên trong một biến.

Với Python, việc phát triển ứng dụng và debug trở nên nhanh hơn bởi vì không cần đến bước biên dịch và chu trình edit-test-debug của Python là rất nhanh.

* **Lịch sử phát triển**

Python được phát triển bởi Guido Van Rossum vào cuối những năm 80 và đầu những năm 90 tại Viện toán-tin ở Hà Lan. Python kế thừa từ nhiều ngôn ngữ như ABC, Module-3, C, C++, Unix Shell, …

Ngôn ngữ Python được cập nhật khá thường xuyên để thêm các tính năng và hỗ trợ mới. Phiên bản mới nhất hiện nay của Python là **Python 3.3** được công bố vào 29/9/2012 với nguyên tắc chủ đạo là "bỏ cách làm việc cũ nhằm hạn chế trùng lặp về mặc chức năng của Python".

* **Đặc điểm**

Dưới đây là một số đặc điểm chính của Python:

Dễ dàng để sử dụng: Python là một ngôn ngữ bậc cao rất dễ dàng để sử dụng. Python có một số lượng từ khóa ít hơn, cấu trúc của Python đơn giản hơn và cú pháp của Python được định nghĩa khá rõ ràng, … Tất cả các điều này là Python thực sự trở thành một ngôn ngữ thân thiện với lập trình viên.

Ta có thể đọc code của Python khá dễ dàng. Phần code của Python được định nghĩa khá rõ ràng và rành mạch.

* Python có một thư viện chuẩn khá rộng lớn. Thư viện này dễ dàng tương thích và tích hợp với UNIX, Windows, và Macintosh.
* Python là một ngôn ngữ thông dịch. Trình thông dịch thực thi code theo từng dòng (và ta không cần phải biên dịch ra file chạy), điều này giúp cho quá trình debug trở nên dễ dàng hơn và đây cũng là yếu tố khá quan trọng giúp Python thu hút được nhiều người học và trở nên khá phổ biến.
* Python cũng là một ngôn ngữ lập trình hướng đối tượng. Ngoài ra, Python còn hỗ trợ các phương thức lập trình theo hàm và theo cấu trúc.

Ngoài các đặc điểm trên, Python còn khá nhiều đặc điểm khác như hỗ trợ lập trình GUI, mã nguồn mở, có thể tích hợp với các ngôn ngữ lập trình khác, …

**3.1.2 Công cụ (chưa làm)**

* + 1. ***Thư viện sử dụng***

#### *3.1.3.1 Thư viện Numpy*

Numpy (Numeric Python): là một thư viện toán học rât phổ biến và mạnh mẽ của Python. NumPy được trang bị các hàm số đã được tối ưu, cho phép làm việc hiệu quả với ma trận và mảng, đặc biệt là dữ liệu ma trận và mảng lớn với tốc độ xử lý nhanh hơn nhiều lần khi chỉ sử dụng Python đơn thuần.

Nếu bạn muốn trở thành một lập trình viên khoa học dữ liệu chuyên sâu, bạn cần phải nắm rõ numpy. Đây là một trong những thư viện hữu ích nhất của python, đặc biệt đối với các lập trình viên khoa học dữ liệu chuyên sâu phải làm việc thường xuyên liên quan đến các con số. Vì phần lớn Khoa học Dữ liệu và Máy học xoay quanh Thống kê, nên việc thực hành trở nên quan trọng hơn nhiều.

Sử dụng NumPy, lập trình viên có thể thực hiện các thao tác sau:

* Các phép toán toán học và logic trên mảng.
* Các biến đổi Fourier và các quy trình để thao tác shape.
* Các phép toán liên quan đến Đại số tuyến tính. NumPy tích hợp sẵn các hàm cho đại số tuyến tính và tạo số ngẫu nhiên.
  + - 1. *Thư viện Pandas*

Pandas là một thư viện mã nguồn mở được phát triển bởi Wes McKinney vào năm 2008. Pandas được sử dụng chủ yếu để thao tác phân tích dữ liệu và dọn dẹp dữ liệu, được thiết kế để cho phép bạn làm việc với dữ liệu được gắn nhãn hoặc quan hệ theo cách trực quan hơn

* Có thể xử lý tập dữ liệu khác nhau về định dạng: chuỗi thời gian, bảng không đồng nhất, ma trận dữ liệu
* Khả năng import dữ liệu từ nhiều nguồn khác nhau như CSV, DB/SQL
* Có thể xử lý vô số phép toán cho tập dữ liệu: subsetting, slicing, filtering, merging, groupBy, re-ordering, and re-shaping,..
* Xử lý dữ liệu mất mát theo ý người dùng mong muốn: bỏ qua hoặc chuyển sang 0
* Xử lý, phân tích dữ liệu tốt như mô hình hoá và thống kê
* Tích hợp tốt với các thư viện khác của python
* Cung cấp hiệu suất tốt
  + - 1. *Selenium*

Selenium là một trong những công cụ kiểm thử phần mềm tự động mã nguồn mở (open source test automation tool) mạnh nhất hiện nay cho việc kiểm thử ứng dụng Web. Selenium script có thể chạy được trên hầu hết các trình duyệt như IE, Mozilla FireFox, Chrome, Safari, Opera; và hầu hết các hệ điều hành như Windows, Mac, Linux.

Selenium WebDriver (Selenium 2) là thư viện cho phép người dùng lập trình (scripting) test script trên các ngôn ngữ lập trình khác nhau như Python, Java, C#, Ruby…

* + - 1. *Thư viện torch*

Đây là thư viện máy học mã nguồn mở , khung máy tính khoa học và ngôn ngữ tập lệnh dựa trên ngôn ngữ lập trình Lua của Facebook . Nó cung cấp một loạt các thuật toán cho học sâu và đã được Facebook, IBM, Yandex và những người khác điều chỉnh để giải quyết các vấn đề phần cứng cho luồng dữ liệu.

* + - 1. *Thư viện Underthesea*

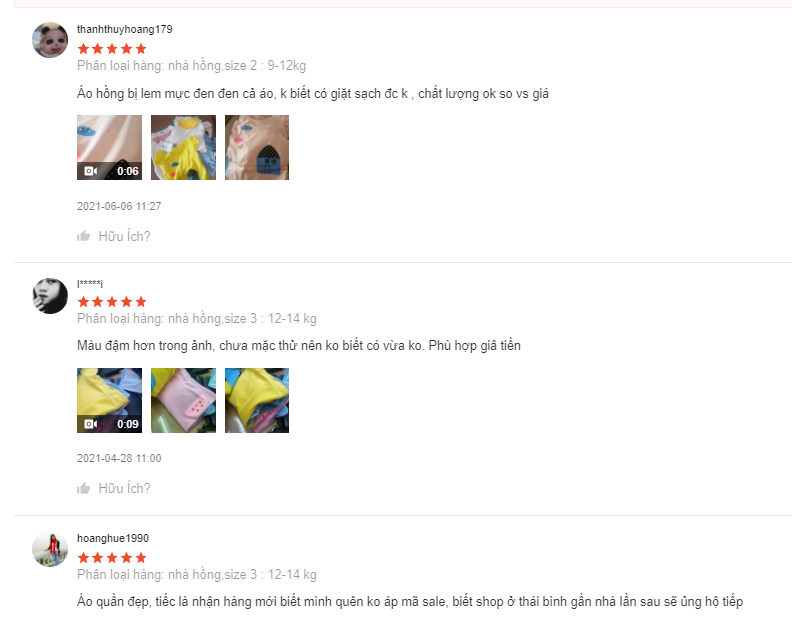
Underthesea là tập hợp các dự án, nguồn dữ liệu mở, tài liệu hướng dẫn với mục đích hỗ trợ việc nghiên cứu và phát triển trong lĩnh vực xử lý ngôn ngữ tự nhiên tiếng Việt.

* + - 1. *Thư viện Sklearn*

**3.2 Các bước thực hiện**

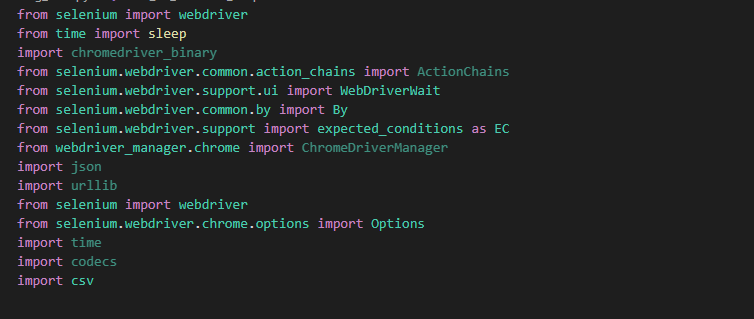
Bước 1: Thu thập dữ liệu

Dữ liệu để xây dựng bài toán là các bình luận tiếng Việt đi kèm với chấm điểm tương ứng. Dữ liệu này được lấy trên các trang thương mại điện tử như Shoppe, Tiki và được dùng để xây dựng mô hình chấm điểm.

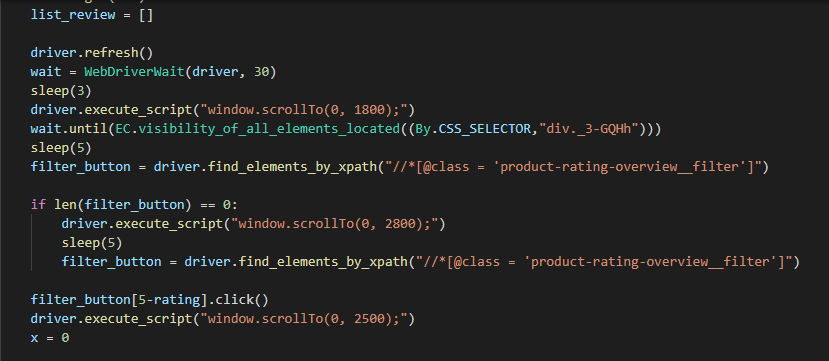


Để lấy được dữ liệu, cần sử dụng thư viện Selenium webdriver để tương tác với giao diện.

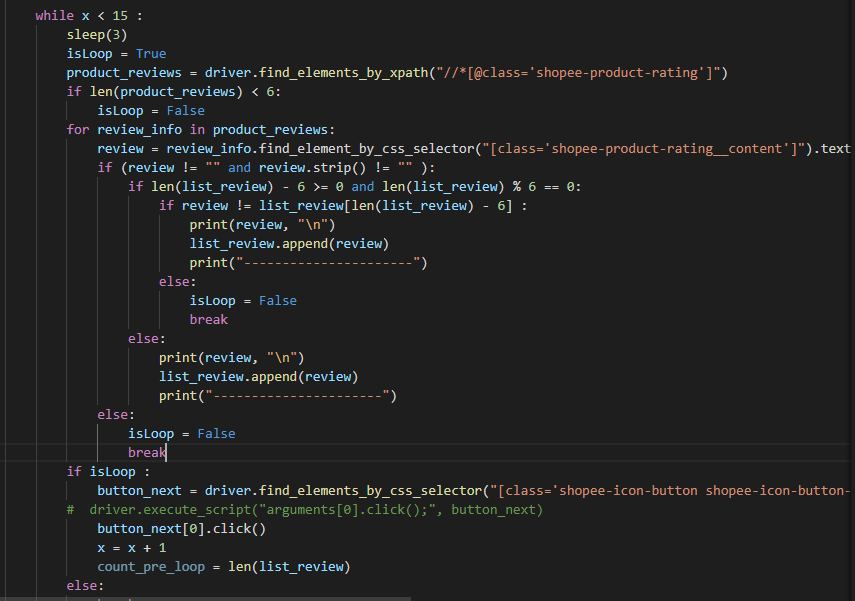
Import các thư viện cần sử dụng



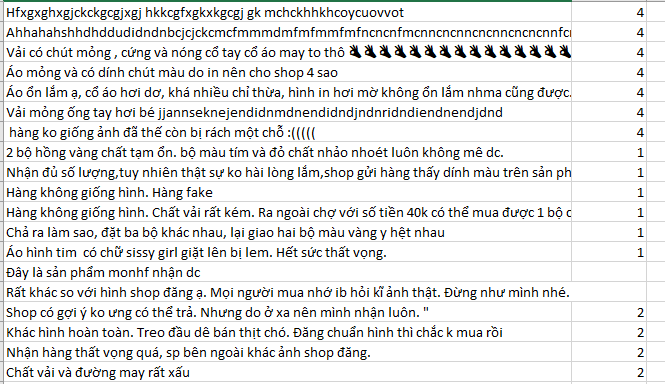
Lần lượt thao tác mở trình duyệt của mỗi sản phẩm và cuộn tới vùng hiển thị các bình luận. Mỗi sản phẩn sẽ có các danh sách bình luận tương ứng 1/2/3/4/5 sao. Lần lượt thu thâp dữ liệu tương ứng với số sao:



Lấy tối đa 15 trang bình luận cho mỗi loại điểm bình luận.



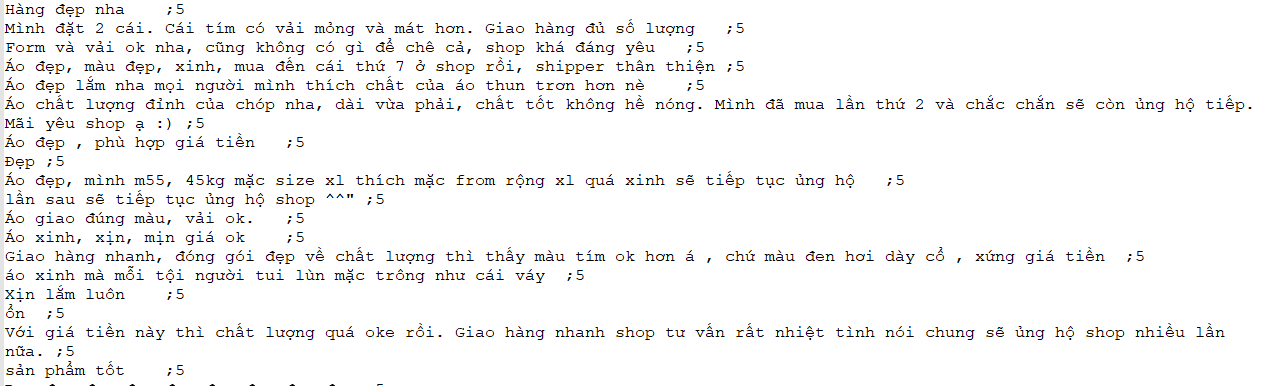
Kết quả được lưu lại trong file csv như sau:



Bước 2: Tiền xử lý dữ liệu

Với dữ liệu được thu thập về, có thể thấy, trong mỗi bình luận còn có chữ viết tắt như: “ko”/k tương ứng “không”, “sp” tương ứng “sản phẩm”,...một số từ tiếng anh được người dùng việt hóa như: ok/okie, shop, fake,size…Ngoài ra, có thể thấy, người dùng thường sử dụng các icon 👌👌👌👌👌👌/ 😖, kí hiệu :(((((, từ viết sai chính tả viết tắt tùy tiện như: “Thấy chất vải lần này k dc đẹp = lần trc”/” Tuyệt lắm nên mua nhaaaaa”/…, và các dấu câu !/@/,/... Hoặc các bình luận vô nghĩa, chỉ mang tính chất nhận xu từ các sàn thương mại điện tử…

Vì vậy, việc làm sạch dữ liệu là vô cùng cần thiết. Trước hết cần loại bỏ bằng tay các câu vô nghĩa, việt hóa hoặc thay các từ viết tắt thành các từ có nghĩa để bình luận vẫn còn ý nghĩa, xóa đi các icon xuất hiện, một số từ được sử dụng thông thường như ok, shop, size vẫn được giữ lại.



Sau đó, có thể xóa đi các kí tự đặc biệt, dấu câu có ở trong câu



Lúc này, dữ liệu đã được làm sạch hơn nhiều và tất cả kí tự đều đưa về dạng viết thường.

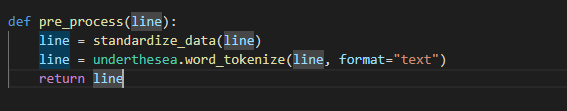
Bước 3: Trích xuất đặc trưng văn bản

Trước khi đưa vào trích xuất, mỗi bình luận được vector hóa lại. Nghĩa là, một bình luận là vector trong đó mỗi một từ, cụm từ là phần tử.

Ví dụ: “Chiếc áo này rất đẹp” đưa về vector như sau [“chiếc”, “áo”, “này”, “rất”, “đẹp”]

Hoặc: “Kiểu này hợp với học sinh” đưa về vector như sau: [“Kiểu”, “này”, “hợp”, “với”, “học\_sinh”]

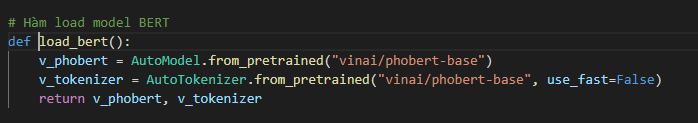
Sử dụng thư viện Underthesea để vector hóa, đồng thời, phương pháp này làm giảm độ dài vector.



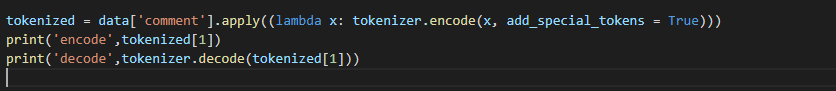
***Sử dụng mô hình phoBert để trích xuất đặc trưng***

Sau đó, đặc trưng được đưa vào mô hình phoBert để trích xuất đặc trưng. Tuy nhiên, cần tokenize bằng bộ tokenizer của PhoBert. Chú ý rằng khi tokenize ta sẽ thêm 2 token đặc biệt là [CLS] và [SEP] vào đầu và cuối câu.

Load mô hình PhoBert:



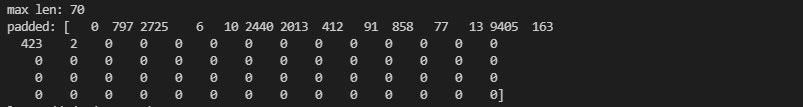
Vector hóa bằng bộ tokenizer của phoBert



Kết quả như sau:

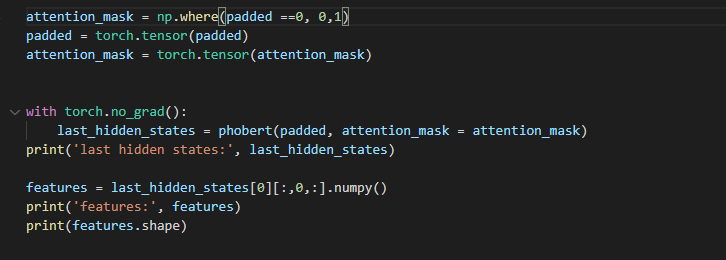


Với bộ data, tìm độ dài lớn nhất và thêm 0 vào sau các vector để các bình luận có cùng độ dài max\_length: max\_length = 70 và mỗi vector sẽ có dạng:

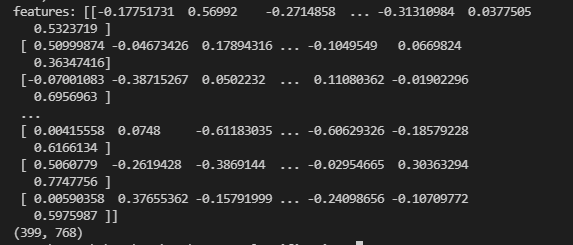


Nếu data có 399 bản ghi và max\_length = 70, sau khi vector hóa, ta được một tập dữ liệu 2 chiều có kích thước (399,70).

Cuối cùng, đưa dữ liệu này vào mô hình phoBert để lấy đặc trưng :



Đặc trưng thu được là feature có định dạng như hình và kích thước (399\*768):



**Trích xuất đặc trưng bằng kĩ thuật TF-IDF (Term Frequency–Inverse Document Frequency).**

Bước 4: Xây dựng mô hình phân lớp

**3.3 Mô tả dữ liệu**

**3.4 Kết quả mô hình**

**3.5 Kết quả đánh giá thực nghiệm**

*3.5.1 Kết quả đề tài đạt được*

*3.5.2 Hạn chế của đề tài*

*3.5.3 Hướng phát triển và tối ưu*

**TÀI LIỆU THAM KHẢO**