ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC BÁCH KHOA
KHOA KHOA HỌC VÀ KỸ THUẬT MÁY TÍNH

LUẬN VĂN TỐT NGHIỆP ĐẠI HỌC

PHÂN TÍCH CẢM XÚC TRONG VĂN BẢN Y KHOA

HỘI ĐỒNG: KHOA HỌC MÁY TÍNH

Giáo viên hướng dẫn:
GS. TS. Cao Hoàng Trụ

Giáo viên phản biện:
GS. TS. Phan Thị Tươi

Sinh viên thực hiện:
Nguyễn Đức Trí (51204052)
Nguyễn Diệp Phương Linh (51201899)

Thành phố Hồ Chí Minh, 12/2016

Lời cam đoan

Chúng tôi xin cam đoan rằng, đề tài luận văn tốt nghiệp "Phân tích cảm xúc trong văn bản y khoa" là công trình nghiên cứu của chúng tôi dưới sự hướng dẫn của GS. TS. Cao Hoàng Trụ, xuất phát từ nhu cầu thực tiễn của đề tài và nguyện vọng tìm hiểu, nghiên cứu của bản thân chúng tôi.

Ngoại trừ kết quả tham khảo từ các công trình khác đã ghi rõ trong luận văn, các nội dung trình bày trong luận văn này là do chính chúng tôi thực hiện và kết quả của luận văn chưa từng được công bố trước đây dưới bất kỳ hình thức nào.

Thành phố Hồ Chí Minh, ngày 16 tháng 12 năm 2016

Nhóm tác giả

Lời cảm ơn

Trước hết, chúng tôi xin gửi lời cám ơn sâu sắc nhất đến GS. TS. Cao Hoàng Trụ, giáo viên hướng dẫn luận văn và là người thầy gắn bó với chúng tôi trong nhóm nghiên cứu khoa học hơn một năm vừa qua. Chính nhờ những tri thức Thầy truyền đạt cùng với sự hướng dẫn tận tình, những góp ý khoa học của Thầy đã giúp chúng tôi hoàn thành tốt nhất đề tài luận văn này.

Chúng tôi cũng xin gửi lời cảm ơn chân thành tới quý Thầy Cô đang công tác tại Khoa Khoa học và Kỹ thuật Máy tính, Trường Đại học Bách Khoa TP.HCM, những người đã nhiệt tình truyền đạt kiến thức, kinh nghiệm trong suốt hơn bốn năm học để chúng tôi có được nền tảng vững chắc như ngày hôm nay.

Cuối cùng, chúng tôi xin gửi lời cảm ơn tới gia đình, bạn bè, những người đã động viên, giúp đỡ chúng tôi rất nhiều trong quá trình thực hiện đề tài này.

Thành phố Hồ Chí Minh, ngày 16 tháng 12 năm 2016

Nhóm tác giả

Tóm tắt luận văn

Việc nhận biết được sự phân cực cảm xúc trong báo cáo y khoa là rất quan trọng để các y bác sĩ sàng lọc, tiếp thu và tổng hợp tri thức trước khi đưa ra quyết định lâm sàng. Chúng tôi đã xem xét vấn đề này như một bài toán phân loại với dữ liệu đầu vào là một câu trong báo cáo y khoa và đầu ra là kết quả phân cực của câu: *tích cực*, *tiêu cực* hoặc *trung tính*. Để giải quyết bài toán này, chúng tôi kết hợp các kỹ thuật xử lý ngôn ngữ tự nhiên và học máy vào hệ thống xây dựng bộ phân loại cảm xúc trên câu.

Dựa trên kết quả của [1], chúng tôi tiến hành rút trích 3 đặc trưng gồm có đặc trưng N-gram, đặc trưng Chuyển đổi trạng thái và đặc trưng Phủ định để xây dựng hệ thống phân tích tính phân cực cảm xúc. Đồng thời chúng tôi đề xuất kết hợp thêm đặc trưng SO-CAL vào hệ thống. Các thí nghiệm cho thấy đặc trưng SO-CAL giúp cải thiện đáng kể hiệu quả phân loại. Kết quả hệ thống chúng tôi xây dựng đạt kết quả độ đo $F = 70.74\%$ trên tập dữ liệu 552 câu trích từ phần tóm tắt của các báo cáo y khoa. Từ kết quả đạt được, chúng tôi hy vọng đề tài sẽ cung cấp nhiều thông tin hữu ích cho các hệ thống hỗ trợ ra quyết định lâm sàng, và làm nền tảng cho các nghiên cứu sau này.

Mục lục

Danh sách hình vẽ

Danh sách bảng

Chương 1

Tổng quan

Trong Chương 1 chúng tôi sẽ trình bày giới thiệu khái quát về đề tài luận văn và những động lực đã thúc đẩy chúng tôi nghiên cứu về đề tài này, đồng thời trình bày chi tiết về mục tiêu, phạm vi đề tài và cấu trúc của luận văn.

1.1 Giới thiệu đề tài

Trong lĩnh vực nghiên cứu khoa học nói chung và y học nói riêng, quá trình học hỏi, trau dồi kiến thức nền tảng và kiểm nghiệm lại nguồn tri thức đã có là sự khởi đầu cần thiết để tìm tòi, sáng tạo nên những tri thức mới đóng góp cho cộng đồng. Ngày nay, với sự phát triển mạnh mẽ của ngành Công nghệ thông tin, nguồn dữ liệu y học đang dần được số hóa và lưu trữ ở những kho dữ liệu trên khắp thế giới, điển hình là PMC[1] và MEDLINE[2] - hai nguồn dữ liệu lớn thuộc Thư viện Y khoa Quốc gia Hoa Kỳ. Trong khi PMC hiện đã có hơn 4 triệu bài báo được cập nhật, thì kho dữ liệu MEDLINE đã tổng hợp được hơn 23 triệu tham khảo y học từ khắp các nguồn trên thế giới. Việc số hóa một lượng lớn dữ liệu y học góp phần tạo cơ hội tốt hơn, bình đẳng hơn cho mọi người tiếp cận với nguồn tri thức y học, nhưng đồng thời cũng đặt ra những thách thức mới trong việc tìm kiếm, chắt lọc và khai thác kho tàng tri thức quý báu này.

Trong hơn 10 năm trở lại đây, cuộc cách mạng mang tên "Y học thực chứng" (*Evidence-based medicine* - EBM) đã diễn ra trong giới y khoa làm thay đổi thói quen điều trị dựa trên cảm tính hay kinh nghiệm cá nhân của bác sĩ, thay vào đó là dựa vào các dữ kiện đáng tin cậy, đã qua kiểm tra một cách khoa học và có hệ thống [2]. Rào cản lớn nhất của phương pháp Y học thực chứng hiện nay là tình trạng quá tải thông tin y khoa, bởi nó đòi hỏi người điều trị phải xem xét các tài liệu liên quan trước khi đưa ra quyết định lâm sàng. Khi tra cứu và chắt lọc các báo cáo y khoa, người đọc thường quan tâm đến kết quả báo cáo và tính phân cực của cảm xúc trong kết quả này. Cực của cảm xúc (*sentiment polarity*) trong báo cáo y khoa có thể là *tích cực* (ví dụ như nghiên cứu cho thấy rằng thuốc X mang lại hiệu quả tốt cho bệnh nhân bị bệnh Y), *tiêu cực* (ví dụ như nghiên cứu cho thấy rằng thuốc X không nên áp dụng cho bệnh nhân bị bệnh Y), hoặc *trung tính* (ví dụ như nghiên cứu chỉ ra rằng thuốc X có hiệu quả nhưng đi kèm với nhiều tác dụng phụ không mong muốn, hoặc nghiên cứu không đưa ra kết luận nào).

Việc tự tổng hợp và đánh giá kết quả của một lượng lớn báo cáo y khoa tiêu tốn không ít thời gian và kém hiệu quả tại thời điểm ra quyết định lâm sàng. Các nghiên cứu gần đây trong [3] đã ghi nhận tầm quan trọng của việc phát triển hệ thống tự động phân tích tính phân cực cảm xúc bởi những lý do sau [1]:

- Đầu tiên, tính phân cực của cảm xúc trong báo cáo y khoa giúp trả lời câu hỏi về lợi ích hay tác hại của một can thiệp y tế (có thể là một phương pháp điều trị hay một loại thuốc,...).
- Thứ hai, những trường hợp nghiên cứu không đưa ra kết quả nào giúp lọc bỏ những thông tin không cần thiết khi có câu hỏi về kết quả của một can thiệp y tế.
- Thứ ba, kết quả *tiêu cực* mô tả tác dụng phụ có thể rất quan trọng cho một quyết định lâm sàng ngay cả khi không được hỏi tới.
- Cuối cùng, từ một loạt các kết quả *tích cực* hay *tiêu cực* của một can thiệp y tế vào một bệnh lý cụ thể, người điều trị có thể có cái nhìn tổng quát hơn về độ phù hợp khi sử dụng can thiệp đó cho quyết định lâm sàng.

Bên cạnh đó, hệ thống phân tích tính phân cực cảm xúc ngoài mục đích hỗ trợ cho việc ra quyết định lâm sàng còn tạo nền tảng cho các hệ thống khác rút trích, tổng hợp nhằm phát hiện ra những tri thức mới tiềm ẩn trong kho dữ liệu y học to lớn của nhân loại. Đó

[1] *PubMed Central* - https://www.ncbi.nlm.nih.gov/pmc/
[2] https://www.nlm.nih.gov/pubs/factsheets/medline.html

là lý do chúng tôi quyết định chọn đề tài "Phân tích cảm xúc trong văn bản y khoa" làm đề tài Luận văn tốt nghiệp.

1.2 Mục tiêu và phạm vi đề tài

Với đề tài "Phân tích cảm xúc trong văn bản y khoa", mục tiêu chính của chúng tôi là xây dựng bộ phân loại cảm xúc nhận dữ liệu đầu vào là một câu trong văn bản y khoa và cho kết quả đầu ra là nhãn phân cực cảm xúc (*tích cực*, *tiêu cực*, *trung tính*) của câu đó. Chúng tôi cũng đề ra mục tiêu cụ thể trong từng giai đoạn như sau:

- Giai đoạn 1: Xây dựng tập dữ liệu gồm tập các câu trích từ phần tóm tắt của báo cáo y khoa và nhãn phân cực cho mỗi câu.
- Giai đoạn 2: Nghiên cứu và hiện thực các bước rút trích đặc trưng dữ liệu.
- Giai đoạn 3: Lựa chọn và kết hợp các đặc trưng để xây dựng và tối ưu hệ thống.

Bước đầu nghiên cứu luận án, chúng tôi đã tiếp cận đề tài theo nhiều hướng thuộc nhiều tiêu chí khác nhau như: nguồn dữ liệu (trang web, báo cáo y khoa, ghi chú lâm sàng), mức tài liệu phân tích (cụm từ, câu, đoạn văn, toàn bộ tài liệu), phương pháp (dựa trên luật, dùng các giải thuật học máy),...

Về mức tài liệu phân tích, trong phạm vi luận án, chúng tôi xây dựng hệ thống phân tích cảm xúc ở mức độ câu, nghĩa là dữ liệu đầu vào là một câu trong báo cáo y khoa. Về phương pháp phân loại, chúng tôi chọn phương pháp học máy có giám sát *Support Vector Machine* bởi độ hiệu quả của phương pháp này đã được ghi nhận trong nhiều nghiên cứu như [4], [5].

Về nguồn dữ liệu, chúng tôi chọn các báo cáo y khoa làm tập dữ liệu đầu vào bởi tính phân cực cảm xúc rõ ràng thể hiện ở loại văn bản y khoa này. Vì nguồn dữ liệu y học ở Việt Nam còn hạn chế nên chúng tôi quyết định hiện thực hệ thống phân tích tính phân cực cảm xúc trên tập các báo cáo y khoa tiếng Anh. Chúng tôi đã sử dụng công cụ PubMed[1] để thu thập dữ liệu và tự xây dựng trang web để đánh nhãn dữ liệu đầu vào. Chúng tôi hy vọng kết quả của luận văn sẽ làm nền tảng để xây dựng hệ thống phân tích cảm xúc trên văn bản y khoa tiếng Việt trong tương lai.

1.3 Cấu trúc luận văn

Luận văn được chia làm 7 chương, bao gồm những khái niệm, kiến thức nền tảng và mô tả chi tiết phương pháp chúng tôi đề xuất để giải quyết bài toán "Phân tích cảm xúc trong văn bản y khoa". Trong Chương 1 (chương hiện tại), chúng tôi giới thiệu khái quát về đề tài luận văn, nêu rõ mục tiêu và phạm vi đề tài. Chương này giúp cho người đọc có cái nhìn toàn cảnh về luận án. Ở những chương sau, chúng tôi trình bày các bước xây dựng hệ thống phân tích tính phân cực cảm xúc trong văn bản y khoa, kết quả và đánh giá hệ thống. Cụ thể nội dung chính của mỗi chương như sau:

[1] https://www.ncbi.nlm.nih.gov/pubmed

Chương 2: Các công trình liên quan

Trong Chương 2, chúng tôi mô tả bối cảnh các công trình liên quan đến đề tài luận văn, giới thiệu một hướng tiếp cận đề tài và xác định hướng đi của chúng tôi để giải quyết bài toán đã nêu.

Chương 3: Kiến thức nền tảng

Trong Chương 3, chúng tôi trình bày ngắn gọn các kiến thức, công nghệ nền, cùng một số thư viện và công cụ được sử dụng trong suốt quá trình nghiên cứu và phát triển hệ thống.

Chương 4: Phương pháp đề xuất

Trong Chương 4, chúng tôi mô tả chi tiết yêu cầu bài toán "Phân tích cảm xúc trong văn bản y khoa" và đề xuất phương pháp cụ thể để giải quyết bài toán. Chúng tôi cũng mô hình hóa kiến trúc tổng quan của hệ thống và mô tả các giải thuật rút trích đặc trưng để xây dựng hệ thống.

Chương 5: Hiện thực hệ thống

Trong Chương 5, chúng tôi trình bày các chi tiết kỹ thuật của hệ thống và cách thức hiện thực từng khối chức năng của hệ thống.

Chương 6: Thí nghiệm và đánh giá

Trong Chương 6, chúng tôi trình bày cách thu thập và đánh nhãn bộ dữ liệu đầu vào, mô tả và phân tích các thí nghiệm đã thực hiện trên bộ phân loại cảm xúc có được từ hệ thống. Chúng tôi cũng giới thiệu các phương pháp đánh giá hệ thống và đưa ra kết quả đánh giá sau cùng.

Chương 7: Tổng kết

Trong Chương 7, chúng tôi tóm tắt kết quả đạt được trong quá trình làm luận án, trình bày những đóng góp và hạn chế của hệ thống phân loại, và đề xuất hướng phát triển tiếp theo.

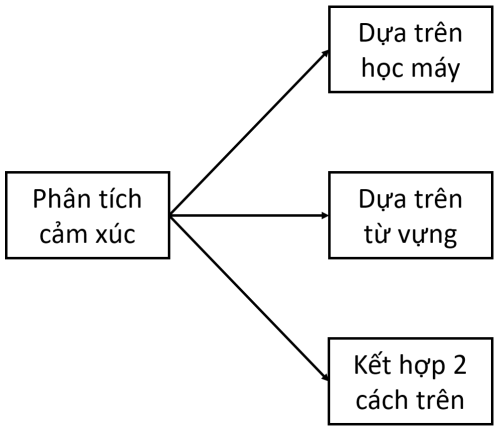
Chương 2

Các công trình liên quan

Bài toán Phân tích cảm xúc (hay Khai phá ý kiến) đã được đặt ra từ trước thế kỷ XXI và nhanh chóng trở thành chủ đề hấp dẫn trong lĩnh vực Xử lý ngôn ngữ tự nhiên bởi tính ứng dụng cao. Chủ đề Phân tích cảm xúc được phân nhỏ thành nhiều bài toán con như: phân loại tính phân cực (tích cực hay tiêu cực), phân loại ý kiến chủ quan với ý kiến khách quan, phân tích biểu cảm (vui, buồn, giận, ...), ...

Về nguồn dữ liệu, bài toán Phân tích cảm xúc đã được nghiên cứu trên nhiều lĩnh vực khác nhau như bình luận phim, bình luận trên các trang blog, các bình luận về sản phẩm, bình luận trên các diễn đàn về sức khỏe, các *Tweet* trên mạng xã hội Twitter, các văn bản y khoa, ... Trong luận án này, chúng tôi chỉ quan tâm đến nguồn dữ liệu thuộc lĩnh vực y khoa.

Những phương pháp được đề xuất để giải quyết bài toán có thể chia thành 3 nhóm[6]: phương pháp dựa trên học máy, phương pháp dựa trên từ vựng và phương pháp sử dụng kết hợp cả hai (Hình 2.1). Sau đây chúng tôi sẽ trình bày bối cảnh đề tài Phân tích cảm xúc dựa trên từng nhóm phương pháp.



Hình 2.1: Các phương pháp phân tích cảm xúc trong ngữ cảnh chung

2.1 Phương pháp dựa trên học máy

Nhiều phương pháp học máy có thể được áp dụng để giải quyết bài toán như *Naïve Bayes*, SVM (*Support Vector Machine*), *Maximun Entropy*,... trong đó nổi bật nhất là hai thuật toán học máy có giám sát *Naïve Bayes* và SVM. [7] đã thử nghiệm cả 3 giải thuật *Naïve Bayes*, SVM và *Maximun Entropy*, sử dụng các đặc trưng uni-gram, bi-gram và gán nhãn từ loại (*Part-of-speech Tagging*) để giải quyết bài toán phân loại nhị phân: phân loại các bình luận, nhận xét về phim có tính tính cực hay tiêu cực.

Nghiên cứu đạt kết quả tốt nhất là 81.6% với giải thuật SVM được sử dụng. Khác với [7], Pak và Paroubek [8] đã giải quyết bài toán phân loại đa lớp: phân tích xem các *tweet* trên Twitter thuộc loại tích cực, tiêu cực hay trung tính. Nghiên cứu đã thí nghiệm các bộ phân loại SVM, MNB (*Multinomial Naïve Bayes*), CRF (*Conditional Random Field*) sử dụng đặc trưng N-gram (gồm Uni-gram, Bi-gram), và vị trí các n-gram. Kết quả tốt nhất nhóm tác giả đạt được khi kết hợp MNB với đặc trưng N-gram và sử dụng thông tin nhãn từ loại (*pos tagging*). Ngoài ra, nhóm tác giả nhận thấy kết quả bộ phân loại tăng lên khi kích thước tập dữ liệu lớn hơn.

Trong lĩnh vực y khoa, nhiều nghiên cứu phân tích cảm xúc cũng sử dụng phương pháp dựa trên học máy. Cảm xúc trong lĩnh vực y khoa thường được hiểu khác nhau tùy vào nguồn dữ liệu phân tích. Nghiên cứu [9] giới thiệu phương pháp phân loại ý kiến bệnh nhân qua nhiều bước. Bước đầu tiên xây dựng bộ phân loại chủ đề dựa trên dữ liệu đã được gán nhãn chủ đề. Bước thứ 2 phân loại tính phân cực của các ý kiến dựa trên dữ liệu đã gán nhãn tính phân cực. Nhóm tác giả sử dụng giải thuật MNB, đạt kết quả *F-measure* là 0.67.

Nghiên cứu của nhóm tác giả Niu, Yun [1] phân tích cảm xúc trên câu sử dụng giải thuật SVM với các câu thuộc lĩnh vực y khoa (*medical text*). Cảm xúc trong nghiên cứu này được hiểu như kết quả được thể hiện trong câu. Mỗi câu được phân loại vào 1 trong 4 nhóm: tích cực, tiêu cực, trung tính hoặc không thể hiện kết quả. Kết quả tốt nhất nghiên cứu đạt được là 79.42%. Cũng như đa số các nghiên cứu trong lĩnh vực này, ngoài các đặc trưng thường dùng (uni-gram, bi-gram, gán nhãn từ loại), nhóm tác giả đề xuất sử dụng đặc trưng Chuyển đổi trạng thái (*Change phrase*) cùng với việc bổ sung kiến thức về lĩnh vực y khoa bằng cách sử dụng các khái niệm y học trong hệ thống UMLS (*Unified Medical Language System*).

2.2 Phương pháp dựa trên từ vựng

Phương pháp phân tích cảm xúc dựa trên từ vựng phụ thuộc vào các nguồn từ vựng cảm xúc. Nguồn từ vựng cảm xúc, thường được hiểu như một bộ từ điển, là tập hợp các từ ngữ thể hiện cảm xúc với mỗi từ được đánh giá tính phân cực bằng một số thực. Các từ điển này có thể được xây dựng thủ công hoặc bán thủ công. Lợi thế của phương pháp này là không cần huấn luyện, từ đó không cần dữ liệu đã được đánh nhãn. Phương pháp này thường được sử dụng cho việc phân tích cảm xúc trên các loại văn bản thông thường: các bài viết trên blog, các bình luận về phim, sản phẩm, hoặc trên các diễn đàn.

Nghiên cứu [10] sử dụng từ điển SentiWordNet để đánh giá tính phân cực của các bình luận phim. SentiWordNet là một từ điển được tạo tự động dựa trên cơ sở dữ liệu WordNet. Kết quả tốt nhất đạt được độ chính xác 69.35%. Nhóm tác giả kết luận việc sử dụng từ điển SentiWordNet đạt hiệu quả tương đương với sử dụng từ điển được xây dựng bằng tay.

Một số nghiên cứu khác tự xây dựng bộ từ điển dựa trên các nguồn khác nhau. Nghiên cứu [11] khẳng định việc xây dựng bộ từ điển giúp tạo lập một nền tảng vững chắc cho hướng tiếp cận này. Nghiên cứu tự xây dựng bộ từ điển dựa trên bộ từ điển được xây dựng thủ công General Inquirer và một số nguồn văn bản khác. Nhóm tác giả hiện thực hệ thống SO-CAL nhằm tính điểm cho tài liệu dựa trên một tập các quy tắc với bộ từ điển đã xây dựng. Kết quả nghiên cứu cho thấy bộ từ điển được xây dựng tốt hơn các từ điển được xây dựng thủ công hoặc tự động trước đây như: từ điển Google, từ điển Maryland, SentiWordNet, ... Hơn nữa, kết quả nghiên cứu cho thấy hệ thống SO-CAL đạt hiệu suất tốt trên các bình luận thuộc nhiều lĩnh vực khác nhau. Điều này có ý nghĩa quan trọng khi phương pháp phân tích cảm xúc trên văn bản thường bị phụ thuộc vào lĩnh vực, đặc biệt đối với phương pháp học máy [12].

Trong lĩnh vực y khoa, nghiên cứu [13] phân tích cảm xúc các bình luận về thuốc trên mức mệnh đề câu. Nhóm tác giả tự xây dựng 2 bộ từ điển: từ điển cho các từ vựng thông thường và từ điển các từ chuyên ngành. Trong quá trình xây dựng, nhóm tác giả đã sử dụng các từ điển có sẵn như Subjectivity Lexicon, SentiWordNet và từ điển các từ ngữ thông dụng được tập hợp từ dự án mã nguồn mở 12dict[1]. Sau đó, nhóm tác giả sử dụng công cụ xử lý ngôn ngữ NLP Stanford nhằm tạo các quan hệ và xây dựng tập các luật để tính điểm. Kết quả tốt nhất đạt độ chính xác 78%, tốt hơn thí nghiệm dùng phương pháp học máy mà nhóm tác giả đã thực hiện.

2.3 Phương pháp kết hợp học máy và từ vựng

Một trong những hạn chế của phương pháp dựa trên học máy là việc phụ thuộc vào kích thước tập huấn luyện - là tập dữ liệu đã được đánh nhãn và phải đủ lớn. Nhưng dữ liệu đã được đánh nhãn thường không phổ biến, đặc biệt trong lĩnh vực y khoa, và các nhóm nghiên cứu đa số phải tự bỏ thời gian và chi phí để đánh nhãn dữ liệu. Trong khi đó, phương pháp dựa trên từ vựng tuy không cần bộ dữ liệu huấn luyện nhưng gặp hạn chế vì không có tính chuyên sâu trên từng lĩnh vực cụ thể. Với phương pháp này, mỗi từ luôn thể hiện một tính phân cực như nhau trong mọi tình huống, điều này thường không đúng với thực tế. Một cách để vượt qua các hạn chế này là kết hợp cả 2 phương pháp trên.

Nghiên cứu [14] tiến hành so sánh các phương pháp thuộc 2 nhóm phương pháp trên, sau đó đề xuất sự kết hợp. Kết quả tốt nhất đạt *F-measure* là 0.73. Nghiên cứu [8] giải quyết bài toán phân loại nhị phân. Ban đầu, nhóm tác giả sử dụng phương pháp dựa trên từ vựng để phân loại cảm xúc ở mức thực thể (*entity level*). Sau đó, một bộ phân loại sử dụng SVM được huấn luyện bằng chính tập dữ liệu được phân loại ở bước đầu. Kết quả đạt được độ chính xác F bằng 0.749.

Trong lĩnh vực y khoa, nghiên cứu [15] thực hiện phân tích cảm xúc trên nguồn dữ liệu từ diễn đàn về sức khỏe. Cảm xúc trong các nguồn dữ liệu này thường đề cập đến ý kiến của người dùng về việc điều trị, về bác sĩ, hoặc cảm nhận của chính họ đối với sức khỏe của mình. Người dùng có thể phàn nàn rằng phương pháp điều trị gây hiệu ứng phụ, tuy nhiên kết quả điều trị vẫn tốt. Ali, Tanveer, et al [15] sử dụng các phương pháp học máy với đặc trưng *Bag-Of-Word* để phân loại cảm xúc về vấn đề "mất thính giác" với bộ dữ liệu được lấy từ ba diễn đàn y khoa[2]. Sau đó, nghiên cứu kết hợp phương pháp dựa trên từ vựng bằng cách sử dụng từ điển *Subjectivity Lexicon* giúp cải thiện kết quả hơn 4.2%.

[1] http://wordlist.sourceforge.net/

[2] http://www.medhelp.org, http://www.alldeaf.com, http://www.hearingaidforums.com

Liên quan gần nhất với phương pháp được sử dụng trong luận án của chúng tôi là nghiên cứu [16]. Nhóm tác giả sử dụng phương pháp học máy SVM, nhưng ngoài các đặc trưng thường dùng như N-gram, nhóm tác giả đề xuất sử dụng đặc trưng Hướng ngữ nghĩa (*Semantic Orientation* - SO). Bản chất của đặc trưng này là phương pháp dựa trên từ điển: điểm số *SO* cho tài liệu bằng trung bình cộng điểm số của các từ trong tài liệu đó. Điểm số các từ được tra cứu trong từ điển General Inquirer. Kết luận của nghiên cứu khẳng định việc sử dụng đặc trưng Hướng ngữ nghĩa giúp tăng độ chính xác.

Chương 3

Kiến thức nền tảng

Trong phần này chúng tôi sẽ trình bày khái niệm cơ bản về lĩnh vực Phân tích phủ định, nền tảng kiến thức của phương pháp học máy *Support Vector Machine*, phương pháp đánh giá độ đồng nhất *Fleiss's Kappa* cùng một số thư viện và công cụ chính được sử dụng trong quá trình hiện thực hệ thống.

3.1 Phương pháp phân tích phủ định

Theo [17], xấp xỉ một nửa số câu mô tả trong các văn bản lâm sàng và báo cáo y khoa chịu sự can thiệp của các yếu tố phủ định. Việc xuất hiện các cấu trúc phủ định trong câu có thể dẫn đến thay đổi hoàn toàn tính phân cực của câu. Vì vậy hiện thực tốt bước tự động phân tích phủ định góp phần quan trọng để nâng cao hiệu quả phân loại tính phân cực của câu trong văn bản y khoa.

Do đó, bài toán phân tích phủ định được đặt ra nhằm xác định cấu trúc phủ định trong câu gồm từ phủ định và tầm vực phủ định, từ đó phân tích ảnh hưởng của các yếu tố phủ định lên tính phân cực của câu. Nhiều thuật toán phân tích phủ định đã được hiện thực trên văn bản tiếng Anh ([17], [18], [19], [20]), và một số trong đó được phát triển để nhận diện phủ định cho các ngôn ngữ khác ([21], [22], [23], [24]). Ở phạm vi báo cáo luận văn này, chúng tôi xem bước phân tích phủ định như một bài toán con trong bài toán phân tích cảm xúc chung. Để giải quyết bài toán này, trước hết cần hiểu rõ cấu trúc phủ định và hình thức tồn tại yếu tố phủ định trong câu.

Cấu trúc phủ định

Cấu trúc phủ định (*negation*) là khái niệm chỉ một từ hoặc cụm từ mang ý nghĩa phủ nhận sự tồn tại của một yếu tố khác [25]. Ở ví dụ 1, từ "no" - đóng vai trò là từ phủ định - phủ nhận sự tồn tại của cụm từ "significant effect", hay nói cách khác, cụm danh từ "significant effect" chịu ảnh hưởng của yếu tố phủ định trong câu.

Ví dụ 1:

> "Early administration of oral steroid medication in patients with acute sciatica had ([no] significant effect)."

Hình thức phủ định

Trong ngữ pháp tiếng Anh [26], phủ định có thể xảy ra theo hai hình thức: phủ định hình thái (*morphological negation*) và phủ định cú pháp (*syntactic negation*). Trong đó, phủ định hình thái được tạo ra khi thay đổi từ gốc bằng những tiền tố phủ định (như "dis-", "non-", "un-") hoặc hậu tố phủ định (như "-less"), còn phủ định cú pháp là hình thức phủ định sử dụng từ ngữ phủ định hoặc mẫu cú pháp riêng biệt rõ ràng và mang ý nghĩa phủ nhận một từ hoặc cụm từ khác trong cùng câu hoặc ở câu liên quan.

Phạm vi của phủ định hình thái chỉ giới hạn ở một từ riêng lẻ nên không tác động lên yếu tố khác trong câu. Vì vậy bài toán phân tích phủ định chủ yếu tập trung phân tích dạng phủ định cú pháp, bao gồm hai thành phần chính là từ/cụm từ phủ định (gọi chung là từ phủ định) và phạm vi phủ định của từ đó. Ví dụ 1 là một trường hợp đơn giản nhất của phủ định cú pháp.

Từ phủ định

Trên thực tế, bài toán phân tích phủ định thường gặp khó khăn bởi sự đa dạng về từ phủ định cũng như vị trí tương đối giữa từ phủ định và từ bị phủ định trong câu. Ở ví dụ 2, từ phủ định không chỉ là những từ đơn giản như "no", "not" mà còn bao gồm từ phủ định khác như "without", "rule out", "exclude", ...

Ví dụ 2:

> "Mildly hyperinflated lungs ([without] focal opacity).
> (Myelomeningocele is [excluded])."

Bên cạnh đó, những động từ như "rule out", "exclude" mang ý nghĩa khác nhau khi xuất hiện trong những trường hợp đặc biệt. Ví dụ 3 minh họa câu mệnh lệnh yêu cầu khám trong ghi chú của bác sĩ, cho thấy khả năng viêm phổi (*pneumonia*) vẫn còn hiện diện. Vì thế "rule out" trong câu này không thể hiện sự phủ định.

Ví dụ 3:

> "(<Rule out> pneumonia)."

Tuy nhiên, khi được tìm thấy trong câu bị động như ở ví dụ 4, nó thể hiện sự phủ định rõ ràng khi phủ nhận khả năng bị ung thư phổi.

Ví dụ 4:

> "(The possibility of lung cancer is [ruled out])."

Mặt khác, nếu dạng bị động của những động từ này bị phủ định, sự phủ định sẽ bị loại bỏ như ở ví dụ 5.

Ví dụ 5:

> "(It is <not ruled out> that the ureterocele opens into the vagina)."

Bởi sự phức tạp trong việc nhận diện ý nghĩa phủ định của các từ phủ định nên cần thiết có một danh sách các từ phủ định được lọc và phân loại rõ ràng. Giải thuật phủ định NegEx (sẽ được đề cập rõ hơn ở Mục 4.5) đã xây dựng một danh sách thuật ngữ phủ định[1] chia làm 3 loại:

- Phủ định tiền điều kiện (*pre-condition negation*) bao gồm những từ phủ định có vị trí đứng trước những cụm từ bị nó phủ định trong câu. Ví dụ như "without", "absence of", "rule out"...
- Phủ định hậu điều kiện (*post-condition negation*) bao gồm những từ phủ định có vị trí đứng sau những cụm từ bị nó phủ định trong câu và thường ở thể bị động. Ví dụ như "be ruled out", ...
- Giả phủ định (*pseudo negation*) bao gồm những cụm từ trông có vẻ như từ phủ định nhưng không mang ý nghĩa phủ định. Ví dụ như "not certain if", "without difficulty"...

Việc phân loại các từ phủ định như trên giúp trả lời hai câu hỏi: từ nào trong câu là từ có mang ý nghĩa phủ định, và vị trí của từ bị phủ định là trước hay sau từ phủ định đó. Vấn đề còn lại là xác định tầm vực ảnh hưởng của từ phủ định trong câu.

Phạm vi phủ định

Xét về tầm vực phủ định, phủ định cú pháp được chia hai loại là phủ định liên câu (*intersentential negation*) và phủ định trong câu (*sentential negation*) [27]. Khác với phủ định liên câu - dạng phủ định mà từ phủ định có ảnh hưởng phủ định lên câu khác, phủ định trong câu có từ phủ định và từ bị phủ định cùng tồn tại trong một câu (ví dụ 6). Với đề tài luận văn này, chúng tôi chỉ xem xét đến dạng phủ định trong câu.

[1] https://code.google.com/archive/p/negex/wikis/NegExTerms.wiki

Ví dụ 6:

> Phủ định liên câu: "Is this treatment effective? [No]."
> Phủ định trong câu: "The treatment does [not] reveal the etiology of the patient's pain."

Để giải quyết vấn đề xác định phạm vi phủ định trong câu cần xây dựng một danh sách chứa các thuật ngữ kết thúc (*termination terms*)[1]. Danh sách này gồm những từ báo hiệu kết thúc sự ảnh hưởng của từ phủ định lên các thành phần không liên quan trong câu. Ở ví dụ 7, từ "but" báo hiệu kết thúc phạm vi phủ định gây ra bởi từ phủ định "denies".

Ví dụ 7:

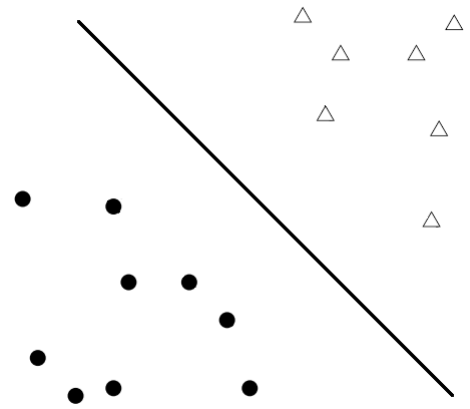
> "Patient ([denies] chest pain) but continues to experience SOB."

Nếu một từ (hoặc cụm từ) được hỏi nằm trong phạm vi phủ định thì từ đó bị phủ định. Ở ví dụ 7, từ "chest pain" bị phủ định vì nằm trong vùng phủ định của từ "denies".

3.2 Phương pháp học máy SVM

Với dữ liệu dạng văn bản, nhiều phương pháp phân loại có thể được sử dụng như SVM, *Naive Bayes*, *Expectation–maximization algorithm*, ... [5]. Trong quá trình nghiên cứu và hiện thực, nhóm chọn mô hình SVM làm giải thuật nền tảng để dán nhãn các lớp cho tập dữ liệu bởi hiệu suất cao của phương pháp này trong việc phân loại nhãn văn bản [28].

SVM được Vapnik lần đầu tiên giới thiệu vào năm 1992 và từ đó trở thành một trong những giải thuật học máy được sử dụng phổ biến nhất bởi hiệu suất phân loại tốt trên những tập dữ liệu có kích thước không quá lớn. SVM được sử dụng chủ yếu để giải quyết các bài toán phân loại và bài toán phân tích hồi quy [4],[5].

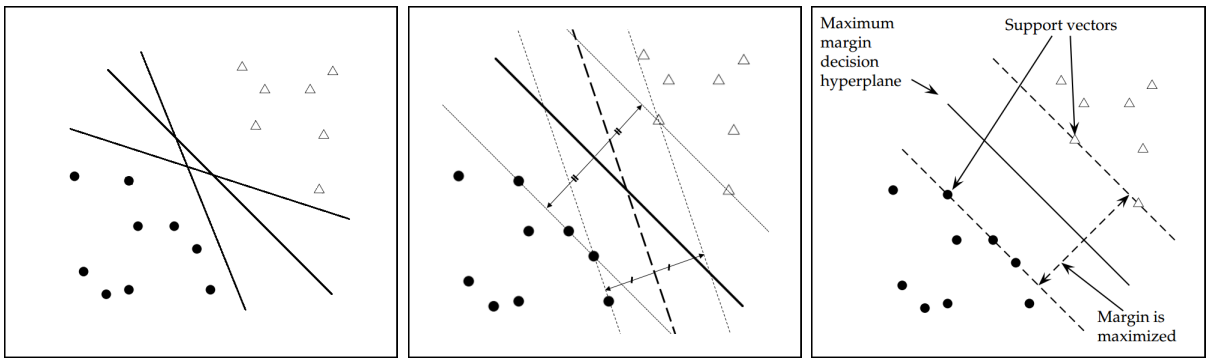


Hình 3.1: Minh họa mô hình phân loại dữ liệu có nhãn

[1]https://code.google.com/archive/p/negex/wikis/NegExTerms.wiki

Mô hình SVM chuẩn là một mô hình học máy có giám sát sử dụng thuật toán phân loại nhị phân. SVM nhận đầu vào là một tập dữ liệu biết trước nhãn của mỗi phần tử thuộc tập dữ liệu đó. Ví dụ như ở Hình 3.1, tập dữ liệu đầu vào gồm tập hợp các điểm biết trước phân lớp (hình tam giác, hình tròn). Nhiệm vụ của SVM là xây dựng một đường phân cách tuyến tính chia tập dữ liệu thành hai nhóm điểm thuộc hai lớp khác nhau, sao cho khi có một điểm mới xuất hiện chưa biết trước nhãn, từ vị trí của điểm này so với đường phân cách có thể dự đoán điểm mới thuộc nhóm phân lớp nào. Để giải quyết bài toán này, mô hình SVM sử dụng giải thuật tối ưu hóa khoảng cách giữa đường phân chia tuyến tính đến điểm dữ liệu gần nhất ở cả hai lớp.

Giải thuật tối ưu hóa khoảng cách



(a) (b) (c)

Hình 3.2: Minh họa giải thuật tối ưu hóa khoảng cách của mô hình SVM [5]

Với không gian dữ liệu tương đối đơn giản như Hình 3.2a, vấn đề đặt ra là có vô số đường tuyến tính có khả năng phân chia tập dữ liệu thành hai lớp phân biệt. Trong tập hợp các đường phân cách này, ta cần lựa chọn một đường tối ưu để tăng hiệu quả phân loại và giảm nhiễu cho tập dữ liệu bằng giải thuật tối ưu hóa khoảng cách của SVM.

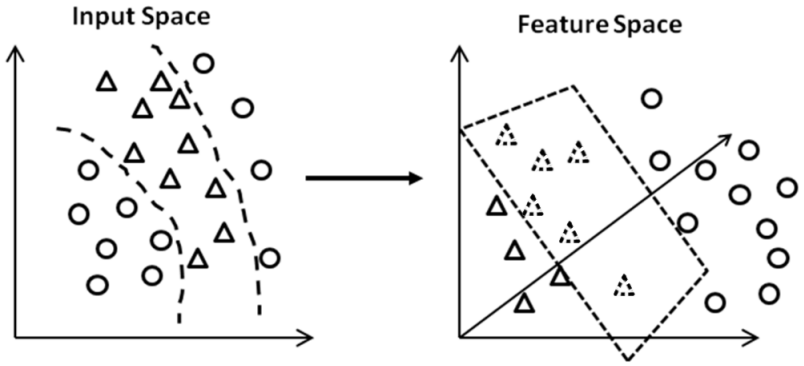
Đầu tiên, đường phân cách cần nằm cách đều hai nhóm phân loại để đảm bảo xác suất phân loại điểm mới công bằng cho cả hai lớp (Hình 3.2b). Điều này có nghĩa là khoảng cách D từ đường phân cách đến điểm gần nhất ở cả hai lớp phải bằng nhau. Tuy nhiên, số lượng đường tuyến tính đảm bảo yêu cầu trên là vô số. Lúc này ta quan tâm đến độ lớn của D: nếu D nhỏ, xác suất phân loại sai sẽ cao hơn do sai số của dữ liệu đầu vào trên thực tế. Vì vậy mô hình SVM chọn đường phân loại có khoảng cách lớn nhất đến điểm gần nhất ở cả hai lớp (Hình 3.2c). Các điểm dữ liệu thuộc hai lớp có vị trí gần nhất với đường thẳng phân loại gọi là *support vector*.

Kĩ thuật Kernel

Dựa vào đặc trưng về vị trí điểm dữ liệu, tập dữ liệu đầu vào có thể được chia làm hai loại:

- Khả phân cách tuyến tính: tồn tại ít nhất một đường thẳng thuộc không gian dữ liệu có thể phân chia tập dữ liệu xác định thành hai nhóm có nhãn khác nhau như Hình 3.1.
- Không khả phân cách tuyến tính: không tồn tại đường thẳng thuộc không gian tập dữ liệu có khả năng chia các điểm dữ liệu thành hai nhóm có nhãn khác nhau. Lúc này bài toán trở thành phân loại phi tuyến.

Trong trường hợp dữ liệu đầu vào không khả phân cách tuyến tính, cách giải quyết đầu tiên là biến đổi không gian dữ liệu trở thành khả phân cách tuyến tính. Nói cách khác, ta cần tìm một hàm ánh xạ sao cho với không gian dữ liệu sau khi ánh xạ tồn tại ít nhất một đường thẳng hoặc mặt phẳng tuyến tính có thể phân loại dữ liệu thành hai lớp. Để làm việc này ta có thể chỉnh sửa các đặc trưng của dữ liệu, hoặc suy diễn đặc trưng mới trên cơ sở những đặc trưng có sẵn. Ngoài ra, ta cũng có thể ánh xạ các điểm dữ liệu vào không gian có số chiều lớn hơn sao cho trong không gian đó có ít nhất một đường thẳng hay mặt phẳng tuyến tính có thể giúp phân loại các điểm dữ liệu (Hình 3.3). Hàm ánh xạ thường không cố định và được lựa chọn tùy theo đặc tính của dữ liệu và tính chất bài toán.



Hình 3.3: Minh họa ánh xạ tập dữ liệu không khả phân cách tuyến tính [1]

Phương pháp biên mềm (*soft-margin*)

Trên thực tế, sai số của dữ liệu đầu vào là một trong những nguyên nhân dẫn đến bài toán phân loại phi tuyến. Trong trường hợp này, phương pháp biên mềm thường được sử dụng để cải thiện giải thuật tối ưu hóa khoảng cách (Hình 3.4). Phương pháp này cho phép tồn tại một số điểm được phân chia sai lớp với một giới hạn sai số nhất định (gọi là độ lỗi).

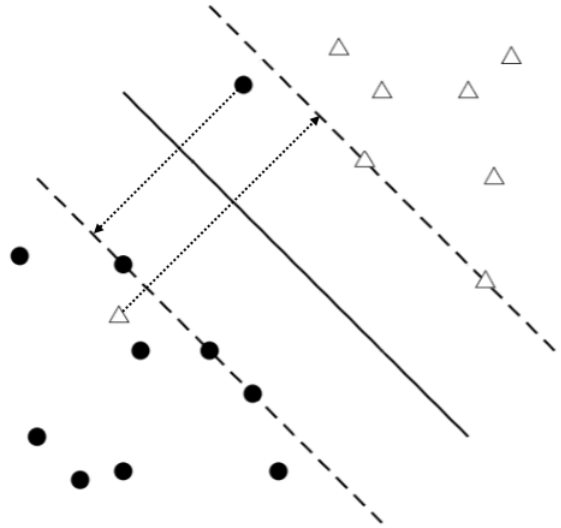
Bài toán phân loại đa lớp

Mô hình SVM dạng chuẩn như trên chỉ giúp phân loại dữ liệu thành hai lớp. Trong khi đó, bài toán thực tế đòi hỏi số lượng lớp phân loại đầu ra thường lớn hơn 2. Ví dụ như bài toán phân tích cảm xúc trên văn bản y khoa yêu cầu phân loại cảm xúc thành 3 lớp: *tích cực*, *tiêu cực* và *trung tính*. Lúc này cần áp dụng một hoặc kết hợp một số phương pháp phân loại đa lớp sử dụng mô hình SVM như OvA (One-vesus-All), OvO (One-against-one), DDAG (Decision Directed Acyclic Graph).

Bảng 3.1: Minh họa phương pháp phân loại OvA cho bài toán phân tích cảm xúc trong bệnh án điện tử

Bộ phân loại	***tích cực***	***tiêu cực***	***trung tính***
SVM_1	O	X	X
SVM_2	X	O	X
SVM_3	X	X	O

[1]Tham khảo Figure 7 của [29]



Hình 3.4: Minh họa phương pháp soft-margin

Bảng 3.1 minh họa phương pháp phân loại OvA: kết hợp nhiều mô hình SVM để phân loại dữ liệu. Trong đó, mỗi SVM giúp phân loại một lớp dữ liệu tương ứng với các lớp khác. Cụ thể với bài toán phân tích cảm xúc trong văn bản y khoa đã đề cập, ta cần ba mô hình SVM: SVM_1 phân loại dữ liệu thành lớp *tích cực* với lớp *không tích cực* (bao gồm *tiêu cực* và *trung tính*), SVM_2 phân loại dữ liệu thành lớp *tiêu cực* với lớp *không tiêu cực* (bao gồm *tích cực* và *trung tính*), và tương tự với SVM_3. Khi xuất hiện một điểm dữ liệu mới, dữ liệu đó sẽ được phân loại qua tất cả những lớp SVM đã được xây dựng.

3.3 Hệ số *Fleiss's kappa*

Hệ số *Fleiss's kappa*, gọi tắt là *kappa*, ký hiệu κ, là hệ số đánh giá mức độ đồng ý của các ý kiến đánh giá trên cùng 1 tập các đối tượng được đánh giá. Trong nghiên cứu này, chúng tôi sử dụng κ để đánh giá mức độ đồng ý của những người đánh nhãn trên tập dữ liệu. Phương pháp này được đánh giá cao bởi vì κ có xem xét đến xác suất ngẫu nhiên xảy ra sự đồng ý giữa các ý kiến đánh giá.

Tính đến nay, phương pháp tính hệ số *kappa* có 3 phiên bản:

- Phiên bản gốc do Jacob Cohen giới thiệu năm 1960, thường được gọi là *Cohen's kappa*.
- Phiên bản *kappa* có trọng số, giúp xét đến cả tỉ lệ không đồng ý.
- Phiên bản *Fleiss' kappa* có thể đo độ đồng nhất với số lượng người đánh nhãn không giới hạn, trong khi phiên bản gốc chỉ có thể đánh giá khi có đúng 2 người đánh nhãn. Lợi thế thứ 2 là mỗi người không cần đánh nhãn tất cả các câu trong tập dữ liệu đánh giá. Điều kiện cần là trong tập dữ liệu đánh giá, mỗi đối tượng đều phải có số lần đánh nhãn bằng nhau.

Trong luận án này, chúng tôi sử dụng phiên bản *Fleiss' kappa*. Cho tập gồm n đối tượng được đánh thứ tự $i = 1..n$, mỗi đối tượng có thể thuộc vào 1 trong k lớp được đánh thứ tự $j = 1..k$, mỗi đối tượng trong tập được phân loại đúng m lần. Trong bài toán của luận án,

BẢNG 3.2: Thống kê các câu được đánh nhãn

Đối tượng	**Lớp** k_1	**Lớp** k_2	**Lớp** k_3
1	x_{11}	x_{12}	x_{13}
2	x_{21}	x_{22}	x_{33}
3	x_{31}	x_{32}	x_{33}

đối tượng ở đây là câu, được phân loại vào 1 trong 3 lớp: *tích cực*, *tiêu cực* hoặc *trung tính*.

Các câu đã đánh nhãn được thống kê như Bảng 3.2, trong đó x_{ij} là số lượng câu thứ i được phân loại vào lớp thứ j. Hệ số *kappa* được tính bởi công thức:

$$\kappa = \frac{\overline{P} - \overline{P_e}}{1 - \overline{P_e}} \tag{3.1}$$

Với $\overline{P_e}$ là mức độ đồng nhất một cách ngẫu nhiên giữa các ý kiến đánh giá, $\overline{P} - \overline{P_e}$ là mức độ đồng ý thực sự. Nếu các ý kiến hoàn toàn đồng nhất, $\kappa = 1$, ngược lại nếu các ý kiến hoàn toàn không đồng nhất hoặc mức độ đồng nhất nhỏ hơn cả xác suất ngẫu nhiên thì $\kappa \leq 0$.

Bước đầu tiên, cần tính $\overline{P}$. Gọi P_i là tỉ lệ 2 ý kiến bất kỳ phân loại đối tượng thứ i thuộc cùng 1 lớp. Khi đó:

$$P_i = \frac{\text{Số lượng tổ hợp 2 ý kiến phân loại đối tượng } i \text{ thuộc cùng 1 lớp}}{\text{Số lượng tổ hợp 2 ý kiến bất kỳ}}$$

Mỗi đối tượng có chính xác m ý kiến đánh giá, suy ra số lượng tổ hợp 2 ý kiến đánh giá cho 1 đối tượng bất kỳ là:

$$\binom{m}{2} = {}^2C_m = \frac{m!}{2!(m-2)!} = \frac{m(m-1)}{2} \tag{3.2}$$

Tương tự, x_{ij} là số lượng các ý kiến đánh giá đối tượng i thuộc lớp j, suy ra số lượng tổ hợp 2 ý kiến bất kỳ đồng nhất đánh giá đối tượng i thuộc lớp j được tính bởi công thức:

$$\binom{x_{ij}}{2} = {}^2C_{x_{ij}} = \frac{x_{ij}!}{2!(x_{ij}-2)!} = \frac{x_{ij}(x_{ij}-1)}{2}$$

Từ đó, số lượng tổ hợp 2 ý kiến đánh giá đối tượng i thuộc cùng 1 lớp là:

$$\sum_{j=1}^{k} \frac{x_{ij}(x_{ij}-1)}{2} \tag{3.3}$$

Lấy (3.3) chia (3.2) ta thu được tỉ lệ các ý kiến đồng nhất đối với đối tượng i là:

$$P_i = \frac{1}{m(m-1)} \sum_{j=1}^{k} x_{ij}(x_{ij}-1)$$

$\overline{P}$ là trung bình mức độ đồng ý của các đối tượng được đánh nhãn, suy ra:

$$\overline{P} = \frac{1}{n} \sum_{i=1}^{n} P_i \tag{3.4}$$

Giá trị $\overline{P}$ đã được xác định, phần còn lại của công thức tính κ là $\overline{P_e}$. Tổng số lượng ý kiến đánh giá là $n \times m$, suy ra xác suất ngẫu nhiên 1 ý kiến bất kỳ đánh giá 1 đối tượng thuộc lớp j là:

$$p_j = \frac{1}{n \times m} \sum_{i=1}^{n} x_{ij}$$

khi đó

$$\sum_{j=1}^{k} p_j = 1$$

Từ đó, xác suất để 2 ý kiến bất kỳ ngẫu nhiên phân loại 1 đối tượng thuộc cùng 1 lớp là:

$$\overline{P_e} = \sum_{j=1}^{k} p_j^2 \tag{3.5}$$

Từ (3.1), (3.4) và (3.5), ta tính được κ.

Để dễ dàng đánh giá chất lượng tập dữ liệu đánh nhãn dựa theo κ, nghiên cứu [30] quy ước đánh giá độ đồng nhất như Bảng 3.3.

Bảng 3.3: Thang đo đánh giá độ đồng nhất dựa trên giá trị κ

Giá trị	**Mức độ đồng nhất**
$\kappa < 0$	Thấp hơn xác xuất đồng ý ngẫu nhiên
$0.1 < \kappa < 0.2$	Hơi đồng ý (*slight*)
$0.21 < \kappa < 0.40$	Mức độ khá (*fair*)
$0.41 < \kappa < 0.60$	Mức độ vừa phải (*moderate*)
$0.61 < \kappa < 0.80$	Mức độ tốt (*substantial*)
$0.81 < \kappa < 0.99$	Mức độ gần như hoàn hảo (*almost perfect*)

3.4 Các thư viện và công cụ hỗ trợ

Trong quá trình hiện thực hệ thống, chúng tôi sử dụng các thư viện và công cụ hỗ trợ như: Scikit-learn, UMLS-Metathesaurus, MetaMap, NLTK, cùng một số thư viện khác. Tất cả những thư viện, công cụ được dùng đều là mã nguồn mở và miễn phí.

Thư viện Scikit-learn

Scikit-learn [31] là bộ thư viện về lĩnh vực Học máy cho ngôn ngữ lập trình Python, được xây dựng trên nền tảng thư viện SciPy[1]. Scikit-learn cung cấp nhiều hàm chức năng cần thiết để giải quyết những vấn đề trong lĩnh vực Học máy như: các giải thuật phân loại, hồi quy, phân cụm, ...; các công cụ tiền xử lý; các giải thuật thu giảm chiều; và các công cụ giúp lựa chọn, đánh giá mô hình.

Trong nghiên cứu này, chúng tôi sử dụng thư viện Scikit-learn để hiện thực đặc trưng N-gram và giải thuật học máy SVM.

[1]https://www.scipy.org/

Bộ từ điển UMLS-Metathesaurus

UMLS (*Unified Medical Language System*) là hệ thống từ vựng y sinh do Thư viện Y khoa Quốc gia Hoa Kỳ xây dựng từ năm 1986 bao gồm tập dữ liệu lớn các thông tin y khoa đã được chuẩn hóa và những công cụ hỗ trợ tương tác với hệ thống máy tính. Tính đến năm 2004, bộ từ điển UMLS tích hợp hơn 2 triệu tên gọi cho khoảng 900 000 khái niệm y sinh và khoảng 12 triệu tên gọi cho quan hệ giữa các khái niệm này [32]. Hiện nay UMLS vẫn liên tục được cập nhật và cho phép sử dụng miễn phí phục vụ mục đích nghiên cứu khoa học.

Hệ thống UMLS chứa 3 thành phần chính:

- Kho dữ liệu Metathesaurus[1]: là bộ từ điển y sinh lớn chứa các thông tin như mã số, ngữ nghĩa của các loại từ vựng, thuật ngữ y học và nhãn liên kết giữa các từ vựng khác nhau có cùng khái niệm. Metathesaurus là thành phần lớn nhất của UMLS, được tích hợp từ mạng ngữ nghĩa và các công cụ xử lý ngôn ngữ tự nhiên trong UMLS. Kho dữ liệu Metathesaurus bao gồm nhiều thư viện y khoa được sử dụng phổ biến như MeSH[2], SNOMED CT[3],...
- Mạng ngữ nghĩa (*Semantic Network*) chứa danh mục các loại ngữ nghĩa và mối quan hệ giữa chúng.
- Công cụ từ vựng (*SPECIALIST Lexicon and Lexical Tools*) bao gồm các công cụ xử lý ngôn ngữ tự nhiên được dùng để tích hợp Metathesaurus.

Trong luận án, chúng tôi sử dụng thư viện UMLS chủ yếu để tra cứu các nhãn phân loại ngữ nghĩa cho các từ trong câu dữ liệu.

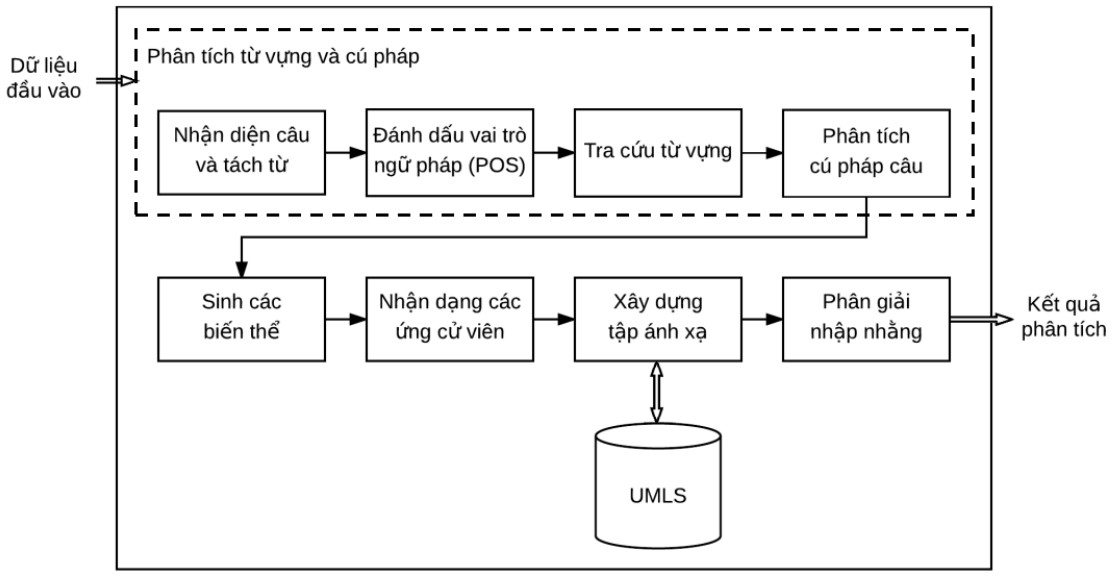
Công cụ MetaMap

MetaMap là công cụ hỗ trợ việc nhận dạng các khái niệm trong bộ từ điển UMLS-Metathesaurus từ văn bản y khoa. MetaMap nhận dữ liệu đầu vào là văn bản phi cấu trúc, thuần ngôn ngữ tự nhiên như các loại báo cáo y khoa, văn bản khám lâm sàng,... Sau quá trình xử lý, đầu ra của MetaMap là văn bản có cấu trúc - chủ yếu ở định dạng XML hoặc một số định dạng khác như MMO, HR - chứa các khái niệm nhận dạng được trong kho dữ liệu Metathesaurus từ văn bản đầu vào. Kiến trúc tổng quát của MetaMap được mô tả như Hình 3.5.

Quá trình xử lý của MetaMap có thể được tóm tắt qua 2 bước:

1. Phân tích từ vựng và cú pháp: dữ liệu đầu vào được áp dụng các tác vụ xử lý ngôn ngữ tự nhiên cơ bản như tách câu, tách từ, xác định từ loại, tra cứu từ vựng dùng công cụ từ vựng của UMLS và phân tích cú pháp câu. Sau những tác vụ này, kết quả thu được là tập hợp các cụm từ (*phrase*) được xác định từ văn bản đầu vào.
2. Phân tích chuyên sâu: ứng với mỗi cụm từ đã tìm được, MetaMap tiến hành tìm tất cả những biến thể của cụm, xác định các ứng cử viên (*candidate*) từ các khái niệm trong UMLS khớp với các biến thể được sinh ra và đánh giá độ tin cậy của từng ứng viên. Kết quả thu được là tập hợp các cụm đã có từ bước 1, và thông tin các ứng cử viên tương ứng.

[1]https://www.nlm.nih.gov/research/umls/knowledge_sources/metathesaurus/
[2]https://www.ncbi.nlm.nih.gov/mesh
[3]https://www.nlm.nih.gov/healthit/snomedct/



Hình 3.5: Kiến trúc tổng quát của MetaMap [33]

Ví dụ với đầu vào là câu: "He denied chest pain, shortness of breath or cough." Kết quả sau khi phân tích của MetaMap trả về như Hình 3.6. Các ứng cử viên được gọi tên là Meta Mapping. Câu văn được tách thành các cụm: "He", "denied", "chest pain", "shortness of breath", "or", "cough". Trong đó:

- Các cụm "He", "or" không có Meta Mapping do không tìm được ứng cử viên nào.
- Cụm "denied" có 2 Meta Mapping cùng số điểm tin cậy (1000), tương ứng với 2 khái niệm trong UMLS Metathesaurus với 2 mã định danh CUI (*Concept Unique Identifier*) là C0332319 và C2700401, kèm theo là định nghĩa, mô tả và nhóm ngữ nghĩa của khái niệm đó. Với cụm "denied" được MetaMap xác định thuộc 2 nhóm ngữ nghĩa là Khái niệm định tính (*Qualitative Concept*) và Hành động (*Activity*).
- Tương tự như cụm "denied", các cụm "chest pain", "shortness of breath", "chest pain" cũng có 2 Meta Mapping với đầy đủ mã số CUI, định nghĩa, mô tả và nhóm ngữ nghĩa của từng khái niệm.

Không chỉ xác định nhóm ngữ nghĩa của các khái niệm, MetaMap còn hỗ trợ phân tích các yếu tố phủ định có trong dữ liệu đầu vào. Khi chọn bộ lọc "-negex", kết quả phân tích sẽ thêm ký tự "N" vào trước khái niệm bị phủ định. Ví dụ như Hình 3.6, cụm "chest pain" có 1 Meta Mapping với mã khái niệm C0008031, thuộc nhóm Dấu hiệu hoặc triệu chứng (*Sign or Symptom*) bị phủ định.
Trong luận án, chúng tôi sử dụng công cụ MetaMap để gán nhãn ngữ nghĩa cho các từ chuyên ngành y khoa trong câu dữ liệu đầu vào và hỗ trợ phân tích đặc trưng Phủ định.

Bộ công cụ phân tích ngôn ngữ tự nhiên NLTK

Bộ công cụ phân tích ngôn ngữ tự nhiên NLTK (*Natural Language Toolkit*) [34] là một nền tảng hàng đầu cho việc xây dựng các chương trình Python trên máy tính để tương tác với dữ liệu dạng ngôn ngữ tự nhiên của con người[1]. NTLK tích hợp hơn 50 kho dữ liệu văn bản và từ vựng trong ngôn ngữ tự nhiên như WordNet, SentiWordNet,... cùng một bộ các thư viện hỗ trợ xử lý dữ liệu dạng văn bản. Một số chức năng NLTK cung cấp để

[1]http://www.nltk.org/

áp dụng vào các bài toán phân tích là phân loại (*classification*), biến đổi từ về dạng gốc dựa trên hình thức (*stemming*), biến đổi từ về dạng gốc dựa trên từ điển (*lemmatization*), gắn nhãn, phân tích cú pháp,... Ngoài ra NLTK còn đóng gói và cung cấp các giao diện lập trình ứng dụng (API) của một số thư viện khác (ví dụ như thư viện Stanford NLP).

Trong nghiên cứu này, chúng tôi sử dụng NLTK như công cụ chính để thực hiện một số bước tiền xử lý dữ liệu, bao gồm tách câu, biến đổi từ về dạng gốc dựa trên hình thức và dựa trên từ điển.

Các thư viện khác

Ngoài 2 thư viện kể trên, chúng tôi còn sử dụng một số thư viện hỗ trợ như:

- Thư viện NumPy: giúp thao tác trên dữ liệu dạng vector trong các chương trình Python.
- Thư viện Pandas: giúp đọc, ghi tập tin, quản lý các tập dữ liệu (gồm tập huấn luyện và tập kiểm tra).

```
Processing 00000000.tx.1: He denied chest pain, shortness of breath or cough.

Phrase: He
>>>>> Phrase
<<<<< Phrase

Phrase: denied
>>>>> Phrase
denied
<<<<< Phrase
>>>>> Mappings
Meta Mapping (1000):
  1000   C0332319:Denied (Denied (qualifier)) [Qualitative Concept]
Meta Mapping (1000):
  1000   C2700401:Denied (Deny (action)) [Activity]
<<<<< Mappings

Phrase: chest pain,
>>>>> Phrase
chest pain
<<<<< Phrase
>>>>> Mappings
Meta Mapping (1000):
  1000 N C0008031:CHEST PAIN (Chest Pain) [Sign or Symptom]
Meta Mapping (1000):
  1000   C2926613:Chest pain (Chest pain:Finding:Point in
time:^Patient:Ordinal) [Clinical Attribute]
<<<<< Mappings

Phrase: shortness of breath
>>>>> Phrase
shortness of breath
<<<<< Phrase
>>>>> Mappings
Meta Mapping (1000):
  1000 N C0013404:SHORTNESS OF BREATH (Dyspnea) [Sign or Symptom]
Meta Mapping (1000):
  1000   C2707305:Shortness of breath (Shortness of breath:-:Point in
time:^Patient:-) [Clinical Attribute]
<<<<< Mappings

Phrase: or
>>>>> Phrase
<<<<< Phrase

Phrase: cough.
>>>>> Phrase
cough
<<<<< Phrase
>>>>> Mappings
Meta Mapping (1000):
  1000 N C0010200:COUGH (Coughing) [Sign or Symptom]
Meta Mapping (1000):
  1000 N C1961131:Cough (Cough Adverse Event) [Finding]
<<<<< Mappings
```

HÌNH 3.6: Ví dụ kết quả chạy MetaMap

Chương 4

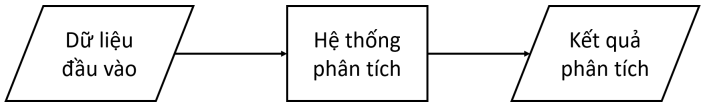
Phương pháp đề xuất

Trong phần này chúng tôi sẽ đặc tả chi tiết yêu cầu bài toán và mô hình hóa phương pháp đề xuất để xây dựng hệ thống phân tích cảm xúc trong văn bản y khoa. Đồng thời chúng tôi cũng trình bày cụ thể các thành phần trong hệ thống và cách sử dụng các thành phần này để giải quyết bài toán.

4.1 Mô tả bài toán

Với đề tài "Phân tích cảm xúc trong văn bản y khoa", chúng tôi giải quyết bài toán cụ thể sau: Cho 1 câu thuộc báo cáo nghiên cứu trong lĩnh vực y khoa, xác định cực cảm xúc của câu là tích cực, tiêu cực hay trung tính. Đây là bài toán phân loại đa lớp. Chúng tôi mô hình hóa yêu cầu bài toán như Hình 4.1, trong đó:

- Dữ liệu đầu vào là 1 câu trong bài báo nghiên cứu thuộc lĩnh vực y khoa.
- Kết quả phân tích là cực cảm xúc của câu: *tích cực*, *tiêu cực* hoặc *trung tính*.
- Hệ thống phân tích là mục tiêu mà luận án cần thực hiện.



Hình 4.1: Mô tả bài toán

Cảm xúc mà bài toán đề cập trong luận án này được hiểu như kết quả của 1 phương pháp hay 1 can thiệp, trong đó:

- Câu được phân loại *tích cực* là những câu thể hiện kết quả tốt hơn, cải thiện hơn hoặc kết quả tích cực vượt trội so với tổng thể dù vẫn có tác dụng phụ tiêu cực.

  Ví dụ 1:

  > "Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or no treatment."

- Câu được phân loại *tiêu cực* là những câu thể hiện kết quả xấu, tệ hơn hoặc thể hiện phương pháp không đem lại hiệu quả.

  Ví dụ 2:

  > "There is a suggestion that routine surgical interference may be harmful by increasing the risk of caesarean section, and this agrees with data from other trials."

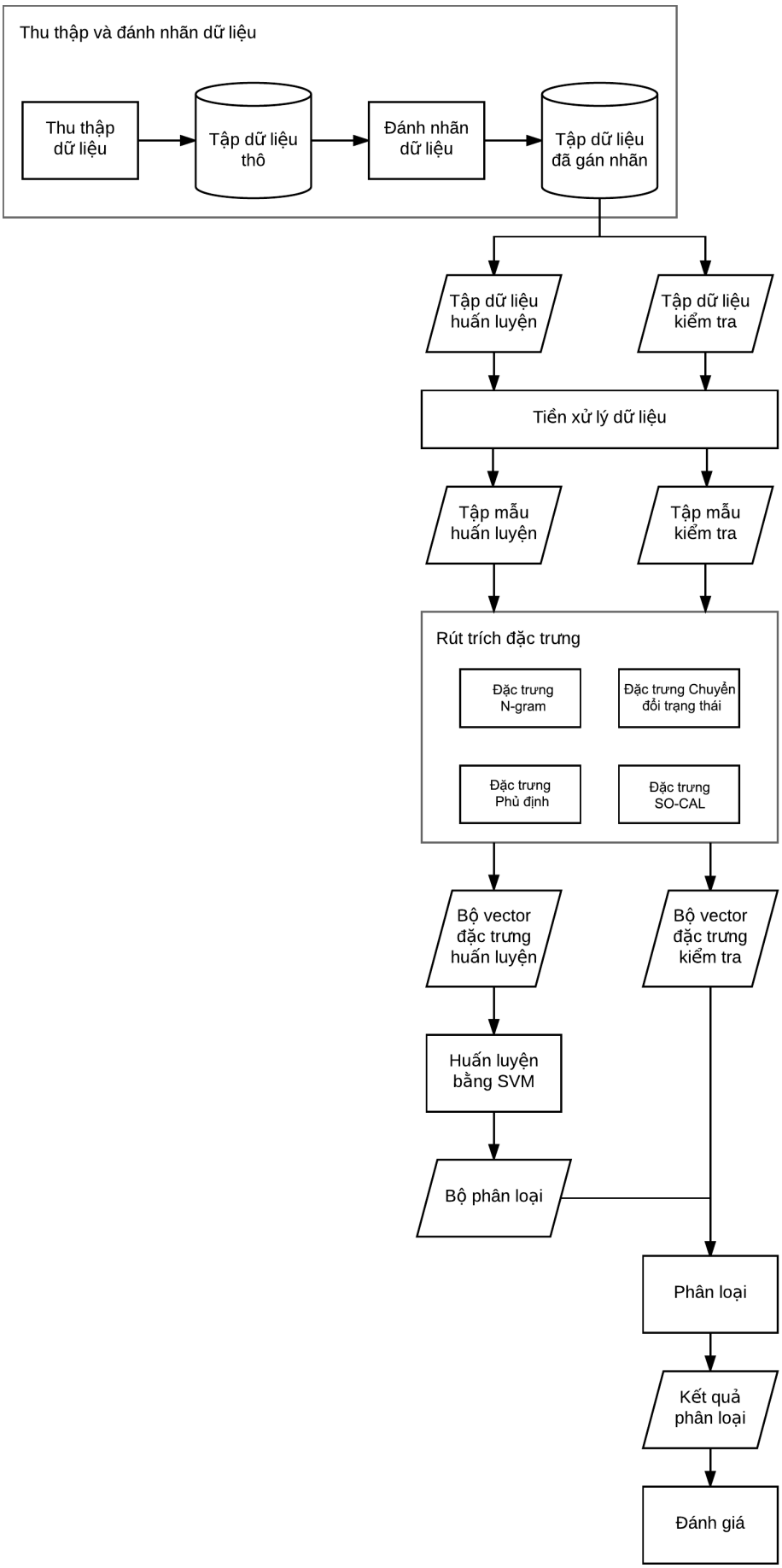
- Câu được phân loại *trung tính* là những câu không thể hiện kết quả, không có khẳng định tốt hay xấu; hoặc đồng thời nhiều ý kiến tốt xấu mà không có sự lất át rõ ràng.

  Ví dụ 3:

  > "Data extraction and analyses and quality assessment were conducted according to the Cochrane standards."

4.2 Kiến trúc tổng quan

Để giải quyết bài toán trên, chúng tôi đề xuất xây dựng hệ thống phân tích cảm xúc trong báo cáo y khoa theo kiến trúc được mô tả ở Hình 4.2. Hệ thống gồm 5 thành phần (*modules*) chính:



Hình 4.2: Kiến trúc tổng quan xây dựng hệ thống

Thu thập và đánh nhãn dữ liệu: Chúng tôi tiến hành thu thập dữ liệu từ gồm các báo cáo nghiên cứu khoa học trong lĩnh vực y khoa từ PubMed, lưu vào hệ cơ sở dữ liệu, sau đó tiến hành đánh nhãn cho mỗi câu trong tập dữ liệu. Chi tiết được trình bày tại Mục 6.2.

Tiền xử lý dữ liệu: Dữ liệu thu thập được có thể có lỗi hoặc có định dạng không đúng, cần được xử lý trước khi trích xuất đặc trưng. Để giải quyết, chúng tôi thực hiện tiền xử lý theo thứ tự các bước:

- Chuyển tất cả các ký tự trong câu thành chữ thường.
- Xóa các ký tự đặc biệt, gồm: ?, %, @, #, ∧, $, ., „, ;, :, /, ", (,), +, -, =
- Thay tất cả chữ số trong câu bằng nhãn *DIGIT*.
- Loại bỏ từ dừng (*Stop words*): Các từ dừng là những từ có tần suất xuất hiện cao trong câu nhưng lại ít có ý nghĩa phân cực. Một số từ dừng thường gặp như it, I, you, then, ...
- *Tokenization*: Sử dụng ký tự khoảng trắng để tách tách câu thành các token.
- *Lemmatization:* biến đổi một từ về dạng gốc bằng cách tra cứu từ điển. Điều này sẽ đảm bảo các từ như "goes", "went" và "go" chắc chắn có kết quả trả về như nhau, kể cả các danh từ như "mouse", "mice" cũng đều được đưa về cùng một dạng như nhau. Nhược điểm của phương pháp này là tốc độ xử lý khá chậm vì phải tra cứu trong cơ sở dữ liệu.
- *Stemming:* biến đổi một từ về dạng gốc bằng cách loại bỏ một số ký tự nằm ở cuối từ vì đó có thể là biến thể của từ. Ví dụ các từ như "walked", "walking", "walks" chỉ khác nhau ở những ký tự cuối cùng, bằng cách bỏ đi các hậu tố "-ed", "-ing", "-s" ta sẽ được từ gốc "walk". Phương pháp này tuy có tốc độ xử lý nhanh nhưng thiếu chính xác ở chỗ: với những động từ bất quy tắc như "went" hay "spoke" thì kết quả sau xử lý không thể trả ra từ gốc "go" hay "speak". Vì vậy, bước này được thực hiện sau bước *lemmatization*.

Rút trích đặc trưng: Chúng tôi đề xuất sử dụng 4 đặc trưng: N-gram, Chuyển đổi trạng thái, Phủ định và SO-CAL được trình bày trong các Mục từ 4.3 đến 4.6.

Huấn luyện: Chúng tôi thực hiện việc huấn luyện với giải thuật học máy SVM để tạo ra bộ phân loại phân cực cảm xúc.

Phân loại: Sử dụng bộ phân loại đã được xây dựng, mỗi dữ liệu đầu vào được phân loại thuộc 1 trong 3 lớp: *tích cực*,*tiêu cực*, hoặc *trung tính*.

Đánh giá: Chúng tôi đánh giá hiệu quả của bộ phân loại bằng tập dữ liệu kiểm tra.

Trong phần còn lại, chúng tôi trình bày các đặc trưng đề xuất theo cấu trúc 2 phần: Phần Mô tả trình bày ý tưởng, giới thiệu về đặc trưng, phần Rút trích trình bày cách sử dụng đặc trưng để có thể áp dụng vào hệ thống.

4.3 Đặc trưng N-gram

Mô tả

Theo kết luận của nghiên cứu [4], N-gram là đặc trưng được sử dụng phổ biến trong bài toán phân tích cảm xúc nói chung. Nhiều nghiên cứu về phân tích cảm xúc trong lĩnh vực y khoa cũng sử dụng đặc trưng này như [7], [1], [16], [35], [36], [9]

N-gram là một chuỗi gồm n phần tử liên tiếp nhau. Các phần tử này có thể là chữ cái, âm tiết hoặc đoạn văn... Trong nghiên cứu này, các phần tử là các từ đơn trong câu. Đơn vị từ được định nghĩa là chuỗi các ký tự liên tiếp nhau không chứa ký tự khoảng trắng, các từ phân biệt nhau bởi ký tự khoảng trắng. Đặc trưng N-gram mang lại hiệu quả cao trong đa số các nghiên cứu, vì vậy đặc trưng này thường được dùng như đặc trưng nền tảng (*baseline*). Báo cáo của [7] phân tích cảm xúc trên các bình luận phim, đạt độ chính xác 82.9% với chỉ một đặc trưng N-gram. Đây cũng là kết quả tốt nhất của nghiên cứu này.
Ví dụ câu: "Standard practice in pupillary monitoring yields inaccurate data"

- Với $n = 1$, N-gram được gọi là Uni-gram. Câu trên sẽ được chuyển thành các n-gram: Standard, practice, in, pupillary, monitoring, yields, inaccurate, data.
- Với $n = 2$, N-gram được gọi là Bi-gram. Câu trên sẽ được chuyển thành các n-gram: Standard practice, practice in, in pupillary, pupillary monitoring, monitoring yields, yields inaccurate, inaccurate data.
- Với $n = 3$, N-gram được gọi là Tri-gram. Câu trên sẽ được chuyển thành các n-gram: Standard practice in, practice in pupillary, in pupillary monitoring, pupillary monitoring yields, monitoring yields inaccurate, yields inaccurate data.
- Với $n > 3$, tần suất xuất hiện các n-gram thấp, dễ làm mô hình học máy bị học quá khớp (*overfitting*)

Việc phối hợp các N-gram là tùy chọn đối với mỗi nghiên cứu, và các kết quả cũng không hoàn toàn đồng nhất. Báo cáo [1] kết luận khi sử dụng Bi-gram kết hợp với Uni-gram giúp tăng độ chính xác thêm 3.01%, điều này phù hợp với báo cáo [16]. Báo cáo [16] kết luận rằng việc dùng cả Uni-gram, Bi-gram và Tri-gram giúp cải thiện kết quả rõ rệt. Trong khi đó [7] đạt kết quả cao nhất chỉ với Uni-gram. Kết luận của [7] cho thấy việc sử dụng thêm đặc trưng Bi-gram không tác động nhiều đến kết quả. Vì không có sự nhất quán trong việc sử dụng kết hợp N-gram giữa các nghiên cứu trên, chúng tôi tiến hành thử nghiệm các cách kết hợp khác nhau để tìm ra kết quả tốt nhất.

Tác giả Kerstin Denecke trong nghiên cứu [37] khẳng định rằng áp dụng kiến thức trong lĩnh vực y khoa là cần thiết để cải thiện hiệu quả phân loại cảm xúc. Trong nghiên cứu này, ý nghĩa cụ thể của các thuật ngữ y khoa không có tác dụng phân loại cảm xúc, chỉ thông tin mô tả của các thuật ngữ này có ý nghĩa.

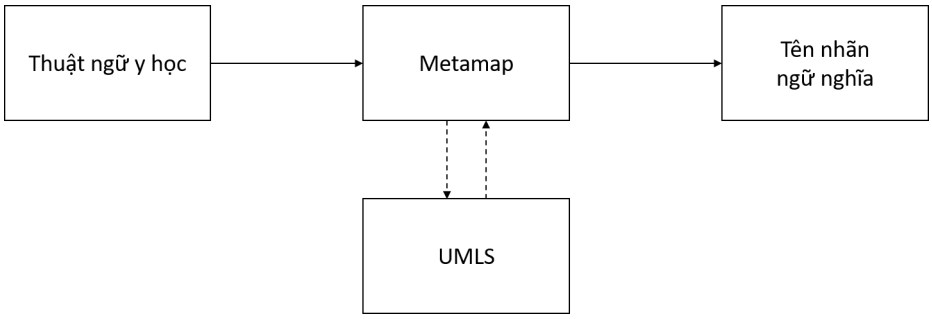
Ví dụ 1:

> "Elevated troponin level after acute stroke is common and is associated with ECG changes suggestive of myocardial ischemia and increased risk of death"

Từ "stroke" xét trong ngữ cảnh y học nghĩa là đột quỵ. Nhưng đối với bài toán phân loại cảm xúc, chúng tôi chỉ quan tâm tới ý nghĩa khái quát của từ này: "stroke" mô tả một loại bệnh. Tương tự các từ "diarrhoea, abdominal pain, nausea" chỉ cần được hiểu như

vấn đề về bụng mà không cần hiểu cụ thể như thế nào. Như vậy, các thuật ngữ y khoa thuộc cùng 1 kiểu (triệu chứng, loại bệnh, tên thuốc, ...) đều được nhóm vào một nhóm ngữ nghĩa (*semantic type*). Từ đó, giảm thiểu khả năng bộ phân loại bị nhiễu hoặc bị học quá khớp.

Một trong những hệ thống được sử dụng phổ biến trong các bài toán thuộc lĩnh vực y khoa trên dữ liệu tiếng Anh là UMLS. Đây là một hệ thống tích hợp các thuật ngữ y khoa cùng các mã chuẩn hóa nhằm tạo tiền đề cho việc xây dựng và phát triển các hệ thống thông tin y khoa cũng như các dịch vụ chăm sóc y tế khác. Để hiện thực nhiệm vụ trên, chúng tôi sử dụng công cụ MetaMap để tra cứu nhóm ngữ nghĩa [1] của các thuật ngữ y học (Hình 4.3).



Hình 4.3: MetaMap sử dụng nguồn tài nguyên UMLS, giúp tra cứu tên nhóm ngữ nghĩa của 1 thuật ngữ y học

Ví dụ 2:

> "Elevated troponin level after acute stroke is common and is associated with ECG changes suggestive of myocardial ischemia and increased risk of death"
> $\xrightarrow{MetaMap}$
> "Elevated troponin level after acute DSYN is common and is associated with ECG changes suggestive of myocardial DSYN and increased risk of death"

Từ "stroke" và "ischemia" đều thuộc kiểu loại bệnh hoặc triệu chứng, nên được thay bằng nhãn DSYN (Disease or Syndrome)

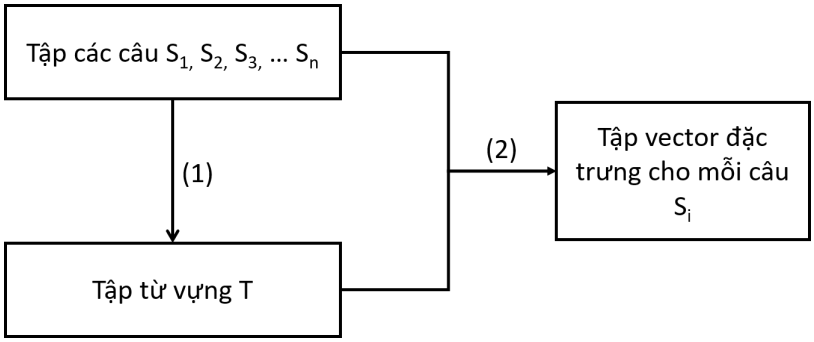
Rút trích

Giải thuật trích xuất đặc trưng N-gram trải qua 2 bước, được mô tả như Hình 4.4.

Ở bước đầu tiên (1), giải thuật nhận vào tập hợp các câu. Mỗi câu sẽ được tách ra thành các n-gram. Tất cả các n-gram từ các câu sẽ được tổng hợp lại thành tập từ vựng T. Tuy nhiên, không phải tất cả các n-gram đều được thêm vào tập từ vựng T. Giải thuật quy định 1 mức ngưỡng min_df là số câu tối thiểu cùng chứa 1 n-gram thì n-gram đó mới được thêm vào tập T. Nếu $min\_df = 1$ thì tập từ vựng T chứa tất cả các n-gram.

Nghiên cứu [16] sử dụng $min\_df = 5$, trong khi nghiên cứu [1] dùng $min\_df = 4$. Tuy nhiên cả 2 nghiên cứu trên đều không giải thích về cách chọn các giá trị trên. Trong nghiên cứu này, chúng tôi tiến hành các thí nghiệm để phân tích và chọn ra giá trị min_df tối ưu nhất.

[1] https://metamap.nlm.nih.gov/Docs/SemanticTypes_2013AA.txt



HÌNH 4.4: Giải thuật trích xuất đặc trưng N-gram

Bước còn lại (2) là *vector* hóa câu: ánh xạ 1 câu từ dạng *text* sang dạng *vector* đại diện cho câu đó. Các n-gram trong tập từ vựng T ở bước (1) được tiến hành sắp xếp. Việc sắp xếp này là tùy ý, nhưng sau khi đã sắp xếp phải giữ nguyên thứ tự. Giả sử T $= \{n\text{-}gram_1, n\text{-}gram_2, n\text{-}gram_3, \ldots, n\text{-}gram_n\}$. Khi đó, mỗi câu s_i sẽ được chuyển thành *vector* v_i có n chiều. Có 2 cách hiện thực để xác định giá trị tại chiều thứ k của *vector* v_i:

- Giá trị tại chiều thứ k của vector v_i là giá trị nhị phân, bằng 0 nếu câu đó không chứa $n\text{-}gram_k$, bằng 1 nếu câu đó chứa $n\text{-}gram_k$.
- Giá trị tại chiều thứ k của vector v_i là số nguyên, thể hiện số lần xuất hiện $n\text{-}gram_k$ trong câu đó.

Để thuận tiện khi gọi tên trong các thử nghiệm, chúng tôi quy ước đặt tên cách hiện thực thứ nhất là *vector nhị phân*, cách hiện thực thứ 2 là *vector số nguyên*.

Ví dụ 3:

> Giả sử sau khi qua bước (1), thu được tập từ vựng T gồm các n-gram: drug, risk, disturbances, associated with, disadvantage, evidence. Khi đó, nếu sử dụng cách *vector* hóa dùng *vector nhị phân*, các câu ở ví dụ 1 và 2 được chuyển thành dạng *vector* như sau:

	drug	**risk**	**disturbances**	**associated with**	**disadvantage**	**evidence**
Ví dụ 1	0	1	0	1	0	0
Ví dụ 2	0	1	0	0	1	0

Khi đó, $v_1 = (0, 1, 0, 1, 0, 0)$ và $v_2 = (0, 1, 0, 0, 1, 0)$

4.4 Đặc trưng Chuyển đổi trạng thái

Mô tả

Đặc trưng Chuyển đổi trạng thái (*Change Phrase*) được Niu, Yun et al. định nghĩa trong một nghiên cứu phân tích cảm xúc trên câu [1]. Sau đó được nhóm tác giả Sarker, Abeed, et al. sử dụng lại. Bài toán mà Sarker, Abeed, et al giải quyết cũng tương tự nhưng thay vì phân tích trên câu, nhóm tác giả phân tích trên đoạn.

Cụm từ chuyển đổi trạng thái là những cụm từ mang ý nghĩa làm thay đổi tình trạng, trạng thái: làm tốt hơn hoặc làm tệ hơn. Tính phân cực trong một câu thường biểu thị qua sự thay đổi [1], và hay xuất hiện ở những câu so sánh.

Ví dụ 1:

> "Atypical antipsychotic use is associated with an increased risk for death compared with nonuse among older adults with dementia"

Câu trên thể hiện tình trạng tệ hơn: Sử dụng "Atypical antipsychotic" làm tăng nguy cơ chết so với không sử dụng "Atypical antipsychoti". Chúng tôi sử dụng 4 nhóm để mô tả đặc trưng Chuyển đổi trạng thái:

- Nhóm thể hiện sự thay đổi trạng thái, gồm 2 nhóm:
  LESS: gồm những từ mang ý nghĩa làm giảm bớt, hạ bớt như "reduce", "decline", "fall", "less", "little",...
  MORE: gồm những từ mang ý nghĩa làm tăng thêm, hoặc duy trì như "enhance", "higher", "exceed", "increase", "improve",...
- Nhóm xác định tính phân cực, gồm 2 nhóm:
  GOOD: gồm những từ mang ý nghĩa tích cực như "benefit", "improvement", "advantage", "accuracy", "great", ...
  BAD: gồm những từ mang ý nghĩa tiêu cực như "suffer", "adverse", "hazards", "risk", "death",...

Danh sách các từ cho mỗi nhóm trên được chúng tôi tập hợp thủ công. Kết hợp 4 nhóm, ta có 4 mẫu (*pattern*) giúp mô tả những thay đổi tích cực hoặc tiêu cực như Bảng 4.1.

Ví dụ 2:

> "Atypical antipsychotic use is associated with an increased risk for death compared with nonuse among older adults with dementia"

Từ "increased" sẽ được gán nhãn MORE, "risk" được gán nhãn BAD, sau đó việc phân tích sẽ xác định được đối tượng của từ "increased" là "risk". Từ đó, câu trên được nhận dạng thuộc mẫu MORE-BAD, suy ra nó có xu hướng biểu thị tính phân cực *tiêu cực*.

Bảng 4.1: Các mẫu thay đổi của đặc trưng Chuyển đổi trạng thái

Nhóm làm thay đổi trạng thái	**Nhóm xác định đối tượng**	**Phân loại tính phân cực**
LESS	GOOD	*tiêu cực*
LESS	BAD	*tích cực*
MORE	GOOD	*tích cực*
MORE	BAD	*tiêu cực*

Rút trích

Rút trích đặc trưng Chuyển đổi trạng thái phụ thuộc vào 2 yếu tố:

- Danh sách từ trong mỗi nhóm LESS, MORE, BAD, GOOD.
- Giải thuật nhận dạng 4 mẫu LESS-GOOD, LESS-BAD, MORE-GOOD, MORE-BAD (Bảng 4.1).

Với yếu tố thứ nhất, trong nghiên cứu này, chúng tôi sử dụng danh sách từ cho mỗi nhóm tham khảo từ nghiên cứu của nhóm tác giả Sarker, Abeed, et al.[16]. Nhóm tác giả trên tự tập hợp danh sách các nhóm từ thủ công nhưng không liệt kê trong báo cáo của mình. Nhóm chúng tôi có liên hệ và nhận được mã nguồn, từ đó lấy được danh sách các nhóm

từ. Danh sách này gồm 371 từ (BAD: 223 từ, GOOD: 82 từ, MORE: 30 từ, LESS:36 từ). Sau đó, chúng tôi mở rộng danh sách bằng cách thu thập thủ công. Danh sách cuối cùng gồm 423 từ (BAD: 238, GOOD: 96, MORE: 42 từ, LESS: 47 từ).

Sau khi đã có tập hợp các từ cho mỗi nhóm, chúng tôi xem xét yếu tố thứ 2: hiện thực giải thuật nhận dạng xem 1 câu có chứa mẫu nào trong 4 mẫu LESS-GOOD, LESS-BAD, MORE-GOOD, MORE-BAD. Giải thuật nhận dạng này được thực hiện qua 2 bước.

Ở bước 1, giải thuật nhận dạng những từ mô tả sự thay đổi, bằng cách so trùng các từ trong 2 nhóm LESS và MORE với các từ trong câu. Để có thể so trùng thành công, trước tiên các từ trong 2 nhóm này được xử lý *lemmatization* và *stemming* như ở mục 4.2. Sau đó mỗi từ trong câu được so sánh với các từ trong 2 nhóm trên. Nếu từ w thuộc 1 trong 2 nhóm trên, chuyển sang bước 2, ngược lại tiếp tục với từ tiếp theo.

Bước 2 nhận diện xem câu có thuộc mẫu nào trong 4 mẫu: LESS-GOOD, LESS-BAD, MORE-GOOD, MORE-BAD hay không. Nếu trong câu có 1 từ thuộc nhóm MORE, giải thuật sẽ xác định trong phạm vi từ từ đó đến hết câu, nếu có từ nào thuộc nhóm GOOD, câu đó thuộc mẫu MORE-GOOD. Tương tự như vậy đối với 3 mẫu còn lại.

Cuối cùng, mỗi câu được ánh xạ sang 1 vector 4 chiều. Giá trị tại chiều thứ i bằng 1 nếu câu thuộc mẫu thứ i, ngược lại bằng 0. Thứ tự các mẫu được sắp xếp như sau: MORE-GOOD, MORE-BAD, LESS-GOOD, LESS-BAD

4.5 Đặc trưng Phủ định

Mô tả

Bài toán phân tích phủ định bao gồm hai nhiệm vụ chính là (1) xác định yếu tố phủ định cùng với phạm vi phủ định trong câu và (2) phân tích ảnh hưởng cũng như hiệu quả của yếu tố phủ định lên ý nghĩa phân loại tính phân cực của cả câu. Để giải quyết bài toán này chúng tôi đã tìm hiểu và hiện thực lại giải thuật phân tích phủ định NegEx[38] (chi tiết hiện thực được mô tả cụ thể ở chương 5).

Giải thuật NegEx dùng để xác định sự tồn tại của phủ định trong câu và xác định xem một cụm từ bất kỳ trong câu có chịu ảnh hưởng của yếu tố phủ định hay không. Giải thuật nhận dữ liệu đầu vào là câu văn được nghi ngờ có sự phủ định và một cụm từ thuộc câu văn đó mà cần xác định xem có bị phủ định hay không. Sau quá trình xử lý, giải thuật đưa ra câu trả lời gồm: câu văn có tồn tại sự phủ định không, xác định từ phủ định trong câu và cụm từ được hỏi có bị phủ định hay không.

Trong quá trình xử lý, NegEx dùng danh sách thuật ngữ phủ định và danh sách thuật ngữ kết thúc để giải quyết bài toán. Bên cạnh đó, giải thuật xây dựng hai biểu thức chính quy `RE` (*regular expressions*) để xác định phạm vi phủ định trong câu. Biểu thức `RE1` bao gồm tất cả các từ (từ đơn hoặc cụm từ) đứng sau thuật ngữ phủ định và sẽ kết thúc bởi một thuật ngữ kết thúc hoặc dấu kết thúc câu hoặc một thuật ngữ phủ định khác. Biểu thức `RE2` chỉ xác định khoảng 5 từ (từ đơn hoặc cụm từ), ưu tiên lĩnh vực y khoa đứng trước thuật ngữ phủ định đang xét.

Áp dụng vào bài toán, với mỗi câu trong dữ liệu đầu vào, giải thuật NegEx lặp lại theo các bước sau:

1. Xác định tất cả các từ phủ định có trong câu dựa trên danh sách thuật ngữ phủ định, ký hiệu là tập A.
2. Tìm từ phủ định đầu tiên trong câu, kí hiệu là *Neg1*.
3. Nếu *Neg1* là từ phủ định giả, bỏ qua và thực hiện bước 6.
4. Nếu *Neg1* là từ phủ định tiền điều kiện: dùng biểu thức chính quy "RE1" xác định vùng phủ định của *Neg1*.
5. Nếu *Neg1* là từ phủ định tiền điều kiện: dùng biểu thức chính quy "RE2" xác định vùng phủ định của *Neg1*.
6. Tìm từ phủ định kế tiếp (cho đến khi hết các từ trong tập A), gán cho *Neg1* và lặp lại bước 3.

Một số ví dụ khi chạy giải thuật NegEx:

Ví dụ 1:

> "Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or ([no] treatment)."
> Từ phủ định tìm được: "no" là thuật ngữ phủ định tiền điều kiện, phạm vi phủ định được xác định, từ bị phủ định là "treatment".

Ví dụ 2:

> "The patient is (tumor [free])."
> Từ phủ định tìm được: "free" là thuật ngữ phủ định hậu điều kiện, phạm vi phủ định được xác định, từ bị phủ định là "tumor".

Rút trích

Sau khi đã xác định được từ phủ định và từ chịu ảnh hưởng phủ định trong câu, chúng tôi thực hiện rút trích đặc trưng phủ định theo 3 cách sau để áp dụng vào hệ thống:

Cách thứ nhất tham khảo từ nghiên cứu [1]. Trong nghiên cứu [1], nhóm tác giả Niu, Yun, et al. chỉ xem xét 1 từ phủ định "no". Tuy nhiên, kết quả trong cho thấy đặc trưng Phủ định được thêm vào hầu như không giúp cải thiện độ chính xác. Tiếp theo, nhóm tác giả trên sử dụng công cụ Apple Pie parser để trích xuất các cụm từ, cụm từ nào có chứa từ "no" sẽ được gắn thêm hậu tố "_NO". Với cách này, yếu tố phủ định không thực sự là 1 đặc trưng mà chỉ ảnh hưởng đến hệ thống thông qua đặc trưng N-gram. Chúng tôi thử nghiệm cách này nhưng thay vì chỉ quan tâm đến từ "no", chúng tôi xem xét đến tất cả các từ được xem là từ phủ định theo giải thuật được mô tả ở phần trước. Sau đó, các từ được xem là bị phủ định được thêm hậu tố "_NEG".

Ví dụ 3:

> Câu: " Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or ([no] treatment)"
> sẽ được chuyển thành:
> " Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or no treatment_NEG"

Cách thứ 2 chúng tôi thử nghiệm có tác dụng tương tự cách trên: không thực sự là 1 đặc trưng mà chỉ ảnh hưởng đến đặc trưng N-gram. Nhưng thay vì làm thay đổi từ bị ảnh

hưởng phủ định, cách hiện thực này thay đổi từ phủ định: thay tất cả các từ phủ định bằng nhãn đại diện "NEGATION"

Ví dụ 4:

> Câu: " Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or ([no] treatment)"
> sẽ được chuyển thành:
> " Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or NEGATION treatment"

Cách còn lại cũng chỉ quan tâm đến từ phủ định, nhưng khác 2 cách trên: cách hiện thực này tạo ra 1 vector đại diện chứ không làm ảnh hưởng đến đặc trưng N-gram. Mỗi câu sẽ được ánh xạ thành 1 số nhị phân: 0 nếu câu đó không chứa yếu tố phủ định nào, 1 nếu ngược lại.

Ngoài ra, chúng tôi cũng tiến hành thử nghiệm kết hợp các cách hiện thực trên để tìm ra cách rút trích hiệu quả nhất.

4.6 Đặc trưng mở rộng SO-CAL

Mô tả

Trong nghiên cứu này, chúng tôi sử dụng đặc trưng SO-CAL dựa trên bài báo [11], tên của đặc trưng xuất phát từ tên hệ thống mà bài báo trên đã xây dựng. Theo nhóm tác giả [11], SO-CAL là chữ viết tắt của Semantic Orientation CALculator. Đây là một phương pháp phân tích cảm xúc dựa trên từ vựng.

Phân tích cảm xúc dựa trên từ vựng là một phương pháp khác, tránh được một hạn chế của phương pháp học máy là không cần qua quá trình huấn luyện. Tuy nhiên đa số các nghiên cứu này đều phân tích cảm xúc trên văn bản thông thường, không tập trung vào 1 lĩnh vực cụ thể nào [11], [10], [12].

Phương pháp này ngầm định 2 giả thiết sau đã được thỏa mãn:

- Bản thân mỗi từ có sẵn tính phân cực mà không bị phụ thuộc vào ngữ cảnh. Điều này có nghĩa là mỗi từ luôn chỉ có 1 xu hướng phân cực (tốt, xấu hoặc tích cực, tiêu cực) trong mọi câu mà từ đó xuất hiện.
- Tính phân cực của mỗi từ được đề cập ở trên có thể được biểu diễn bởi 1 số thực

Dựa trên 2 giả thiết trên, tính phân cực của một câu phụ thuộc vào số thực biểu diễn tính phân cực của các từ trong câu đó, và cũng được biểu diễn bởi 1 số thực. Sự phụ thuộc giữa tính phân cực của các từ và tính phân cực của cả câu tùy thuộc vào các nghiên cứu, có thể mô hình hóa như công thức sau:

$$Polarity_{sentence} = f(Polarity_{words-in-sentence}) \tag{4.1}$$

Rút trích

Rút trích đặc trưng SO-CAl là giải thuật tính điểm số cho mỗi câu phụ thuộc vào điểm số của mỗi từ trong câu. Sau đây, chúng tôi trình bày các vấn đề chính khi tính điểm

cho từ, bao gồm việc xây dựng từ điển, nhận biết từ loại, sử dụng những từ có tính tăng cường, xử lý phủ định.

Thứ nhất, xây dựng từ điển là một bước rất quan trọng ảnh hưởng trực tiếp đến hiệu quả của đặc trưng SO-CAL.Đây là công cụ giúp tra cứu điểm số của mỗi từ. Hiện nay, các từ điển thuộc 2 loại: xây dựng thủ công hoặc xây dựng bán tự động. Từ điển xây dựng thủ công điển hình được sử dụng nhiều trong các nghiên cứu là bộ từ điển General Inquirer. Lợi thế của loại tự điển này là được con người đánh điểm số, vì vậy giảm thiểu sai sót. Các nghiên cứu có thể dựa trên bộ từ điển này để tự mở rộng thành bộ từ điển của mình. Tuy nhiên, bất lợi của loại này là kích thước thường nhỏ. Bộ từ điển bán tự động được sử dụng phổ biến là WordNet. Từ điển thuộc loại bán tự động được xây dựng bằng cách sử dụng một tập từ hạt giống có tính phân cực cao. Các từ này có thể được tập hợp thủ công hoặc được lấy từ từ điển General Inquirer. Từ đó, điểm của các từ khác được sinh ra dựa trên tần số xuất hiện của chúng so với các từ hạt giống. Lợi thế của loại từ điển này là kích thước lớn, tạo độ bao phủ cao, từ đó nhiều từ được gán điểm số hơn. Tuy nhiên độ chính xác của loại từ điển này không cao, và kích thước lớn thường đi kèm với nhiễu.

Nghiên cứu [11] vì thế chọn phương án xây dựng một bộ từ điển thủ công dựa trên một số nguồn dữ liệu văn bản như các bình luận về phim, sách, máy tính, khách sạn,... và các từ thuộc nhóm tích cực, tiêu cực từ từ điển General Inquirer. Điểm số mỗi từ trong từ điển được xây dựng có giá trị từ -5 đến 5 với ý nghĩa: Giá trị càng nhỏ thể hiện tính phân cực về phía tiêu cực càng nhiều, và ngược lại, ví dụ:

Từ	**Giá trị**
monstrosity	-5
hate (noun and verb)	-4
disgust	-3
sham	-3
fabricate	-2
delay (noun and verb)	-1
determination	1
determination	1
inspire	2
inspiration	2
endear	3
relish (verb)	4
masterpiece	5

Thứ hai là việc nhận biết ý nghĩa của các từ loại đối với điểm số cả câu. Hầu hết các nghiên cứu ban đầu về phân tích cảm xúc dựa trên từ vựng đều chỉ tập trung vào tính từ. Điểm số của cả văn bản chỉ phụ thuộc vào điểm số của tính từ trong câu và những từ liên quan. Xét 2 ví dụ sau:

- "The young man strolled+ purposefully+ through his neighborhood+"
- "The teenaged male strutted- cockily- through his turf-"

Các dấu + ở câu thứ nhất thể hiện xu hướng bổ sung tích tích cực vào độ phân cực của cả câu, trong khi dấu - ở ví dụ thứ 2 thì ngược lại. Từ đó cho thấy, ngoài tính từ, các từ

loại khác như động từ, danh từ và phó từ cũng có tác động đến điểm số phân cực của cả câu. Các nghiên cứu sau này đã mở rộng hơn, ngoài tính từ, các từ loại động từ, danh từ và phó từ cũng được xem xét tới.

Điều này không những giúp đánh giá tốt hơn trong trường hợp ví dụ trên, ngoài ra còn giúp giải quyết một số trường hợp đặc biệt khi một từ có thể thuộc nhiều loại từ loại. Ví dụ: từ "novel" nếu xét theo từ loại danh từ thì chỉ mang ý nghĩa trung tính, nếu xét theo từ loại tính từ lại có ý nghĩa tích cực. Từ "plot" mang ý nghĩa tiêu cực nếu là động từ, nhưng mang ý nghĩa trung tính nếu xem là danh từ. Trong nghiên cứu [11], bộ từ điển được xây dựng gồm 2252 tính từ, 1142 danh từ, 903 động từ, and 745 phó từ.

Tất cả các động từ và danh từ trong từ điển đều được xử lý *lemmatization*, vì vậy các động từ ở các dạng thể hiện khác nhau đều cùng 1 điểm số. Các động từ, danh từ và tính từ đều được đánh điểm thủ công, riêng phó từ được đánh điểm tự động dựa trên tính từ. Phó từ được bỏ đuôi "-ly", sau đó so trùng với các tính từ. Ví dụ: từ "purposefully" sẽ được gán điểm số của tính từ "purposefull". Tuy nhiên, một số ngoại lệ như các phó từ "fast" được xử lý thủ công.

Thứ ba là vấn đề nhận biết và sử dụng những từ có tính tăng cường (intensification). Một từ có tính tăng cường được hiểu là các từ bản thân nó không có điểm số thể hiện tính phân cực, nhưng có khả năng tác động lên 1 từ khác, làm tăng lên hoặc hạ thấp tính phân cực của từ đó. Từ đó làm thay đổi điểm số thể hiện tính phân cực của cả cụm từ. Một số từ có tính tăng cường như: "slightly", "very", "most", "the most". Một giải thuật đơn giản có thể được sử dụng trong trường hợp này như sau: Khi gặp một từ có tác động làm tăng tính phân cực ("very"), điểm của từ bị tác động được cộng thêm 1 hằng số. Cụm từ "very good" được tính điểm bằng công thức: $Polarity(\text{"very good"}) = P(\text{"good"}) + 1$. Tương tự với trường hợp còn lại.

Tuy nhiên cách hiện thực này không thể hiện đúng bản chất của tác động tăng cường. Bởi vì cùng một từ "very" có thể có tác động mạnh yếu khác nhau tùy thuộc vào từ bị tác động. Nói cách khác, sự tác động này nên phụ thuộc vào cả 2 thành phần:

- Tính chất tăng cường của từ tác động. Tính chất này có thể được thể hiện bằng tỉ lệ phần trăm (%) như Bảng 4.2.
- Tính phân cực của từ bị tác động

Nghiên cứu [11] sử dụng công thức (4.2) để tính điểm số trong trường hợp này:

$$Polarity(\text{cụm từ}) = Polarity(\text{từ bị tác động}) * (100\% - \text{tỉ lệ tác động}) \tag{4.2}$$

Bảng 4.2: Tỉ lệ tác động của một số từ

Từ	**Tỉ lệ tác động**
slightly	-50%
somewhat	-30%
pretty	-10%
really	+15%
very	+25%
extraordinarily	+50%
(the) most	+100%

Ví dụ 1:

> "good" có điểm số là 3.0, từ đó "very good" có điểm số là: $3.0 \times (100\% + 25\%) = 3.75$

Trong trường hợp có hơn 1 từ có tính tăng cường, điểm số được tính tương tự theo cách đệ quy.

Ví dụ 2:

> "really very good": $(3 \times [100\% + 25\%]) \times (100\% + 15\%) = 4.3$

Trong trường hợp một tính từ bổ sung nghĩa cho một danh từ theo sau nó, tính từ đó được coi như là một từ có tính tăng cường. Vì vậy, bản thân tính từ đó không có điểm số, và chỉ làm thay đổi điểm số của danh từ theo sau.

Ví dụ 3:

> "This is a total failure"
> Tính từ "total" được xem là từ có tính tăng cường, nên thay vì sử dụng điểm số, "total" ảnh hưởng đến điểm số của "failure": $-3.0 \times (100\% + 50\%) = -4.5$

Thứ tư là vấn đề xử lý phủ định trong câu. Sự xuất hiện từ phủ định có thể làm đảo chiều tính phân cực cho cả câu. Tuy nhiên, một từ phân cực về phía tiêu cực mạnh (điểm số rất thấp), không có nghĩa rằng khi bị phủ định sẽ phân cực về phía tích cực mạnh (điểm số rất cao). Nghiên cứu [11] sử dụng chiến lược *shift negation*. Thay vì chỉ đơn thuần đổi dấu điểm số, *shift negation* chỉ cộng/trừ 1 lượng cố định (trong nghiên cứu này là 4) vào điểm số của từ bị phủ định.

Ví dụ 4:

> "This CD is not horrid"
> Điểm số của "horrid" là -5, khi đó, "not horrid" có điểm số là: -5 + 4 = -1

Cuối cùng, sau khi đã tính điểm cho các từ, SO-CAL tính điểm cho câu dựa trên nguyên tắc: Lấy trung bình cộng điểm số những từ có điểm số khác 0. Trường hợp tất cả các từ có điểm số bằng 0 thì điểm số của câu cũng bằng 0

Chương 5

Hiện thực hệ thống

Trong Chương 5, chúng tôi sẽ trình bày chi tiết về kỹ thuật cách rút trích các đặc trưng, hiện thực bộ phân loại cũng như cách tích hợp các yếu tố đó vào hệ thống. Ngôn ngữ lập trình được sử dụng là Python, phiên bản 2.7.

5.1 Hiện thực rút trích đặc trưng

Trong phần này, chúng tôi trình bày cách thức hiện thực để ánh xạ một câu thành một vector. Các đặc trưng nhận vào danh sách n câu, hệ thống sẽ cho ra một mảng 2 chiều $n \times m$, với m là số chiều tùy thuộc mỗi đặc trưng. Từ đó làm đầu vào cho giải thuật học máy SVM.

Đặc trưng N-gram

Scikit-learn cung cấp các công cụ để làm việc với dữ liệu dạng văn bản. Trong số này, chúng tôi sử dụng hàm chức năng sklearn.feature_extraction.text.CountVectorizer để hiện thực đặc trưng N-gram. Như mô tả ở Mục 4.3, đặc trưng N-gram được trích xuất qua 2 bước:

Bước đầu tiên: Xây dựng tập từ vựng T. Scikit-learn cung cấp lớp CountVectorizer cho tác vụ này.

```
vectorizer = CountVectorizer(input, min_df, binary, ngram_range)
```

Hàm khởi tạo CountVectorizer cung cấp 17 tham số, tuy nhiên, trong nghiên cứu này chúng tôi chỉ quan tâm đến 4 tham số, các tham số còn lại sử dụng giá trị mặc định:

input Là các câu trong tập dữ liệu huấn luyện.

min_df Là số nguyên thể hiện số câu ít nhất chứa n-gram để n-gram đó được thêm vào tập từ vựng. Tham số này đã được giải thích chi tiết tại Mục 4.3.

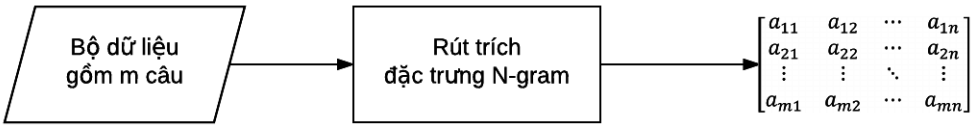
binary Là giá trị *True* hoặc *False*. Nếu $binary = True$, giải thuật sử dụng *vector nhị phân*, ngược lại sử dụng *vector số nguyên*.[1]

ngram_range Là một *tuple*, có dạng (a, b). Giải thuật sẽ sử dụng các loại n-gram từ a-gram đến b-gram.
Ví dụ $ngram\_range = (1, 3)$: Giải thuật sử dụng 3 loại N-gram: Uni-gram, Bi-gram và Tri-gram.

Bước thứ 2: xây dựng tập *vector* đại diện cho mỗi câu trong tập huấn luyện. Bước này được thực hiện bằng cách gọi hàm:

```
vectors_ngram = vectorizer.transform(raw_documents).toarray()
```

trong đó `raw_documents` là tập hợp các câu. Hàm trên trả về một mảng 2 chiều $n \times m$ với n là số lượng câu trong tập `raw_documents` và m là kích thước của tập từ vựng (Hình 5.1).



Hình 5.1: Hiện thực đặc trưng N-gram

[1] 2 khái niệm này được định nghĩa tại Mục 4.3, phần Rút trích

Đặc trưng N-gram kết hợp Metamap

MetaMap cung cấp 3 phương pháp cơ bản để nhận diện các khái niệm trong kho dữ liệu UMLS-Metathesaurus từ dữ liệu đầu vào:

- Tương tác trực tiếp thông qua giao diện web[1]: MetaMap cung cấp giao diện web (Hình 5.2) giúp người dùng, đặc biệt là người mới sử dụng, có cái nhìn trực quan nhất về cách MetaMap hoạt động bao gồm cấu trúc dữ liệu đầu vào và đầu ra, những tùy chọn và cách các tùy chọn này ảnh hưởng đến kết quả phân tích. Tuy nhiên, mỗi lần gửi dữ liệu lên máy chủ MetaMap thông qua giao diện web này, MetaMap giới hạn dữ liệu đầu vào chỉ có một tập tin dữ liệu dạng text chứa không quá 10000 ký tự vì thế không thể sử dụng cách này cho những mẩu dữ liệu quá dài. Hơn nữa việc gửi và nhận kết quả trực tuyến thông qua mạng nên không đảm bảo tốc độ xử lý và bảo mật dữ liệu. Với những hạn chế vừa nêu, sau khi đã làm quen với cách hoạt động của MetaMap, người dùng có thể chọn 2 cách sau để tương tác với MetaMap.

- Gửi tập dữ liệu lên máy chủ MetaMap[2] (*Use Batch MetaMap*): người dùng có thể gửi bộ dữ liệu lên máy chủ của MetaMap, sau quá trình xử lý MetaMap sẽ trả kết quả về địa chỉ email người dùng cung cấp. Ưu điểm của phương pháp này là tốc độ xử lý nhanh do không có tương tác trong quá trình phân tích dữ liệu. Tuy nhiên khi xảy ra lỗi trong quá trình chạy thì khó phán đoán được lỗi xảy ra ở mẩu dữ liệu nào.

- Sử dụng MetaMap cục bộ (*Use MetaMap Locally*): Nếu người dùng muốn hoàn toàn kiểm soát dữ liệu của mình thì việc cài đặt MetaMap ngay trên máy tính cá nhân là lựa chọn tốt nhất. Ưu điểm của phương pháp này là tốc độ xử lý nhanh, không phụ thuộc vào hệ thống mạng do không cần gửi dữ liệu lên máy chủ MetaMap, đảm bảo quyền kiểm soát dữ liệu và khả năng điều chỉnh cấu trúc kết quả phù hợp với nhu cầu sử dụng. Tuy nhiên nhược điểm lớn của phương pháp này là tốn khá nhiều tài nguyên hệ thống và cần máy tính có cấu hình đủ mạnh để làm máy chủ cục bộ.

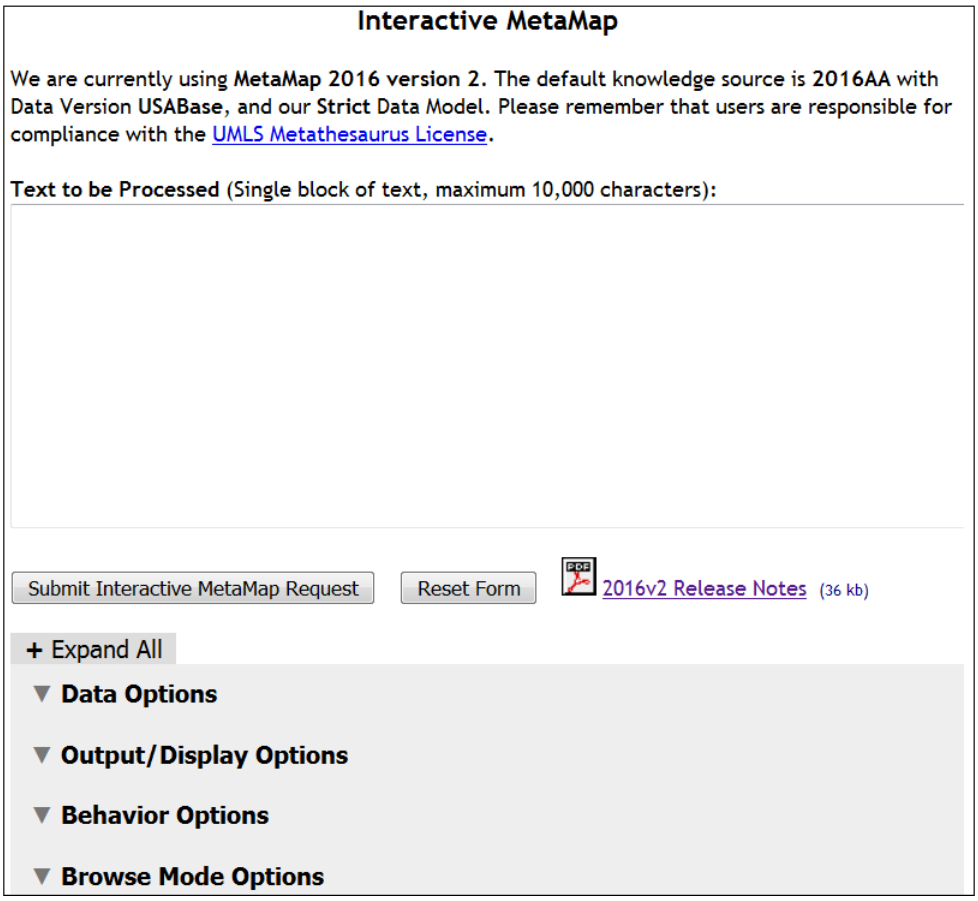
Kết quả xử lý trả về như nhau khi dùng cả 3 phương pháp. Trong hệ thống phân loại chúng tôi đã dùng MetaMap cục bộ để tối đa khả năng điều chỉnh cấu trúc dữ liệu đầu ra.

Chúng tôi sử dụng công cụ MetaMap để xác định các từ chuyên ngành tồn tại trong câu dữ liệu đầu vào. Chúng tôi đã tùy chọn bộ lọc nhãn từ vựng thuộc các nhóm ngữ nghĩa liên quan trực tiếp tới các triệu chứng, nguyên nhân, bệnh lý lâm sàng và cận lâm sàng theo [16] gồm *Pathologic Function*, *Disease or Syndrome*, *Mental or Behavioral Dysfunction*, *Cell or Molecular Dysfunction*, *Virus*, *Neoplastic Process*, *Anatomical Abnormality*, *Acquired Abnormality*, *Congenital Abnormality*, *Injury or Poisoning*. Sau đó chúng tôi thay các từ chuyên ngành bằng nhãn ngữ nghĩa của từ đó. Việc sử dụng N-gram kết hợp Metamap trải qua 2 bước (Hình 5.3):

- Tập dữ liệu sẽ được cho qua khối xử lý Metamap để tạo ra 1 tập dữ liệu mới với các từ chuyên ngành đã được thay thế bởi nhãn ngữ nghĩa.

- Tập dữ liệu mới đi qua khối rút trích N-gram để cho ra mảng 2 chiều $n \times m$ như mô tả ở trên.

[1] https://ii.nlm.nih.gov/Interactive/UTS_Required/metamap.shtml
[2] https://ii.nlm.nih.gov/Batch/UTS_Required/metamap.shtml



HÌNH 5.2: Giao diện web tương tác trực tiếp của MetaMap

Đặc trưng Chuyển đổi trạng thái

Đặc trưng Chuyển đổi trạng thái ánh xạ một câu sang 1 vector 4 chiều. Trong phần hiện thực, chúng tôi định nghĩa hàm training_change_phrase:

```
def training_change_phrase(raw_documents):
```

Hàm nhận vào danh sách các câu trong tập dữ liệu, sau đó trả về mảng 2 chiều $n \times 4$ với n là số lượng câu trong tập *raw_documents* (Hình 5.4)

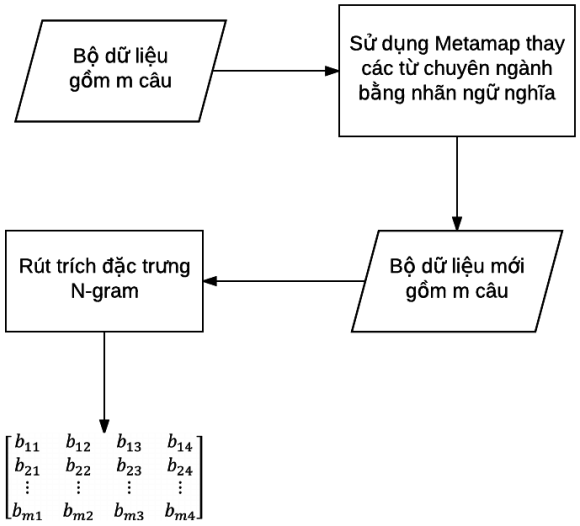
Đặc trưng Phủ định

Chúng tôi xây dựng 2 chương trình con độc lập để xác định cấu trúc phủ định trong câu, đặt tên là Meta-NegEx và Gen-NegEx. Việc sử dụng song song 2 công cụ phân tích phủ định nhằm so sánh hiệu quả giữa chúng và chọn ra phương pháp phân tích phủ định phù hợp nhất. Mô tả cụ thể 2 chương trình này như sau:

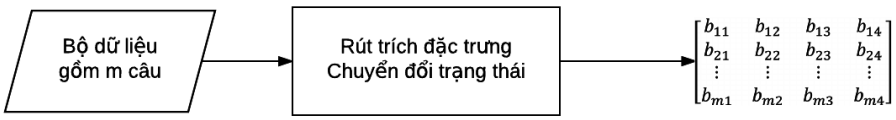
- Meta-NegEx: chúng tôi dùng công cụ MetaMap để phân tích cấu trúc phủ định của câu bằng cách bật tùy chọn `-negex`. Kết quả phân tích được lưu lại dưới định dạng JSON.
- Gen-NegEx: chúng tôi sử dụng công cụ phân tích phủ định NegEx tại mã nguồn mở của Google Code[1]. Kết quả phân tích được lưu lại dưới định dạng bảng tính XLSX.

Ví dụ như với câu dữ liệu đầu vào là "In this trial of apparently healthy persons without hyperlipidemia but with elevated high-sensitivity C-reactive protein levels, rosuvastatin

[1]https://code.google.com/archive/p/negex/wikis



HÌNH 5.3: Hiện thực đặc trưng N-gram kết hợp Metamap



HÌNH 5.4: Hiện thực đặc trưng Thay đổi trạng thái

significantly reduced the incidence of major cardiovascular events."

Khi áp dụng Meta-NegEx: kết quả lưu lại dưới dạng JSON như Hình 5.5. Trong đó, thuộc tính `id` là mã số câu, thuộc tính `negations` là mảng chứa các cấu trúc phủ định trong câu, bao gồm từ phủ định chứa trong thuộc tính `negex` và mảng các từ bị phủ định chứa trong thuộc tính `effectedWords`.

```
{
  "id": 346,
  "negations": [
    {
      "negex": "without",
      "effectedWords": [
        {
          "words": "Hyperlipidemia"
        }
      ]
    }
  ]
},
```

HÌNH 5.5: Ví dụ kết quả phân tích phủ định dùng Meta-NegEx

Khi áp dụng Gen-NegEx: kết quả lưu lại dưới dạng XLSX như Bảng 5.1. Trong đó, nội dung `negated` ở cột `Result` đánh dấu câu có cấu trúc phủ định. Chi tiết cấu trúc phủ định được ghi rõ ở cột `Negation`. Từ được đặt trong cờ `[PREN]` là từ phủ định, từ được đặt trong cờ `[NEGATED]` là từ bị phủ định, từ được đặt trong cờ `[CONJ]` là thuật ngữ kết thúc báo hiệu chấm dứt tầm vực phủ định của từ phủ định.

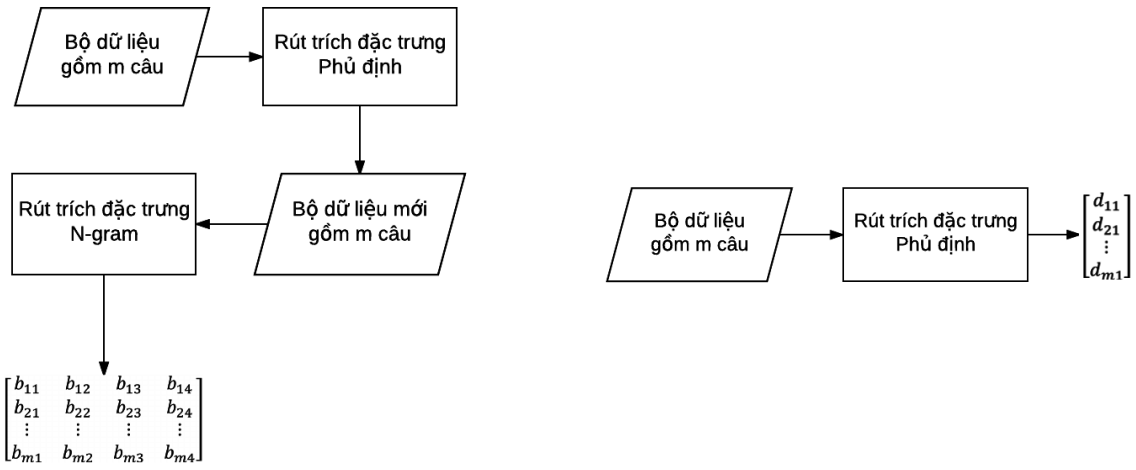
BẢNG 5.1: Ví dụ kết quả phân tích phủ định dùng Gen-NegEx

ID	Phrase	Sentence	Result	Negation
346	hyperlipidemia	In this trial of apparently healthy persons without hyperlipidemia but with elevated high-sensitivity C-reactive protein levels, rosuvastatin significantly reduced the incidence of major cardiovascular events.	negated	In this trial of apparently healthy persons [PREN]without[PREN], [NEGATED]hyperlipidemia [NEGATED] [CONJ]but[CONJ] with elevated high-sensitivity C-reactive protein levels, rosuvastatin significantly reduced the incidence of major cardiovascular events.

Kết quả của 2 công cụ được xử lý để đưa về chung 1 cấu trúc. Việc này giúp dễ dàng tích hợp yếu tố vào hệ thống các đặc trưng mà không quan tâm tới quá trình trích xuất bằng công cụ nào.

Như mô tả ở Mục 4.5, chúng tôi hiện thực việc sử dụng đặc trưng Phủ định theo 3 cách chính:

- Với 2 cách đầu, khối xử lý phủ định nhận vào tập dữ liệu và cho ra một tập dữ liệu mới với nhãn NEGATION thay thế các từ phủ định, hoặc hậu tố _NEG được thêm vào sau các từ bị phủ định. Cách thức hiện thực được mô tả như Hình 5.6a.
- Với cách hiện thực thứ 3, khối xử lý phủ định nhận vào tập dữ liệu và cho ra mảng $n \times 1$ với n là kích thước tập dữ liệu. Cách thức hiện thực được mô tả như Hình 5.6b.

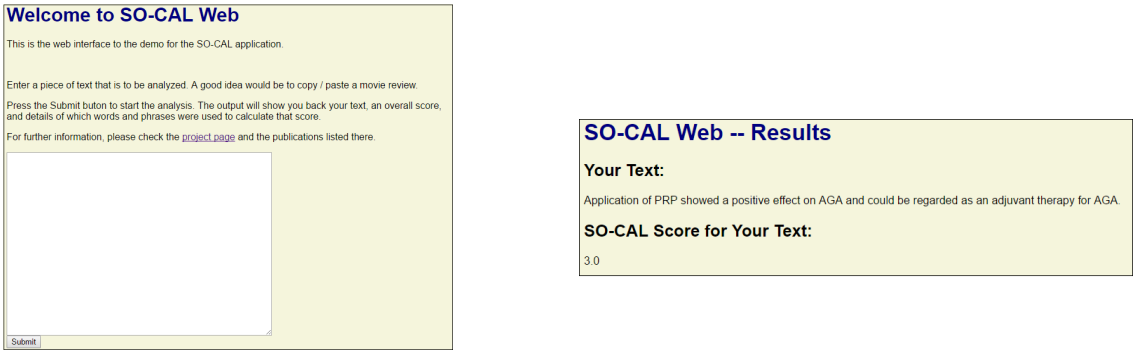


(a) Hiện thực theo cách 1 và 2 (b) Hiện thực theo cách 3

HÌNH 5.6: Hiện thực đặc trưng Phủ định theo 3 cách

Đặc trưng SO-CAL

Trong phần này, chúng tôi sử dụng phiên bản hiện thực của nghiên cứu [11]. Nhóm tác giả của nghiên cứu [11] đã hiện thực công cụ trực tuyến để tính điểm số cho 1 đoạn văn, giao diện được mô tả như Hình 5.7.

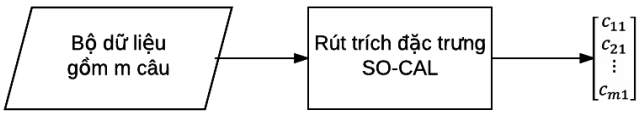


(a) Nhập 1 đoạn text sau đó nhấp submit để chuyển đến trang kết quả

(b) Giao diện kết quả

HÌNH 5.7: Công cụ online để tính điểm SO-CAL

Để tích hợp vào hệ thống, nhóm sử dụng package `re` trong Python, sử dụng phương thức `HTTP POST` để gửi yêu cầu lên trang web, sau đó *parser* phần kết quả trả về để lấy ra thông tin điểm số. Kết quả đầu ra khi rút trích đặc trưng SO-CAL là ma trận $m \times 1$ với m là kích thước tập dữ liệu (Hình 5.8).



HÌNH 5.8: Hiện thực đặc trưng SO-CAL

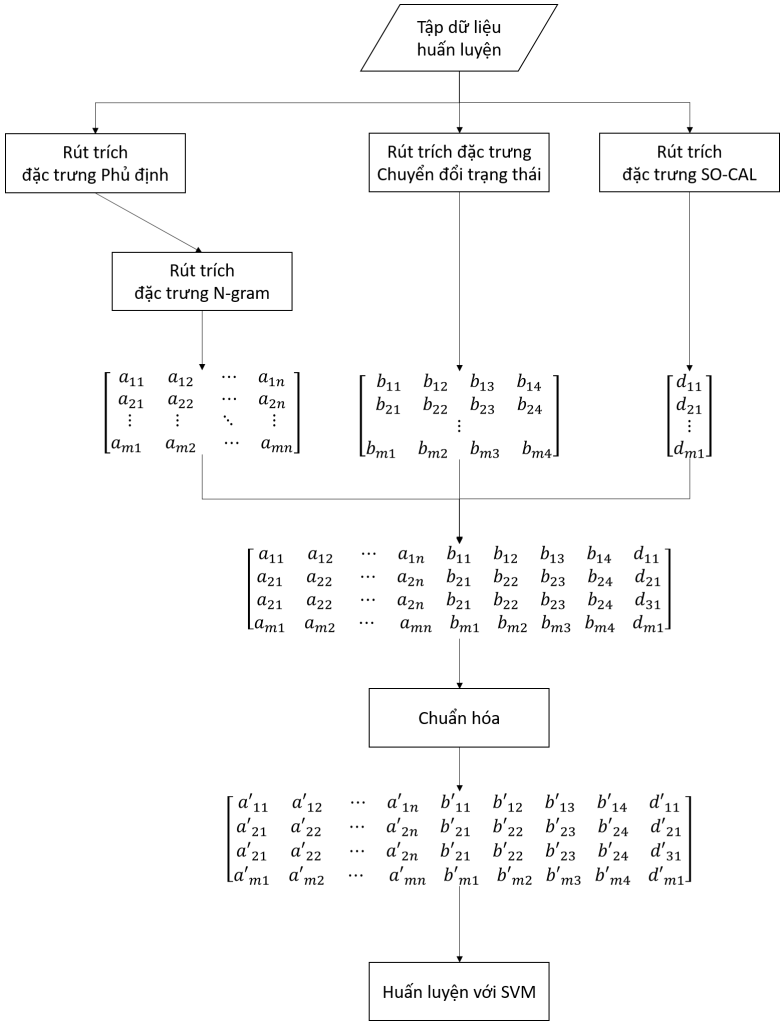
5.2 Hiện thực bộ phân loại SVM

Các đặc trưng sau khi được rút trích như đã mô tả sẽ được kết hợp lại để với mỗi câu, chỉ có 1 vector đại diện. Mô hình kết hợp các đặc trưng được miêu tả như Hình 5.9. Trong mô hình kết hợp trên, đặc trưng Phủ định được sử dụng theo cách 2 (là cách đạt hiệu quả tốt nhất, sẽ được chứng minh ở Mục 6.3). Chúng tôi sử dụng lớp SVM.SVC của thư viện Scikit-learn để hiện thực giải thuật học máy SVM.

```
clf = svm.SVC(decision_function_shape = 'ovr', C = c,                    1
              kernel = 'rbf', class_weight = 'balanced')
clf.fit(data_x, data_y)                                                  2
```

Hàm khởi tạo SVM.SVC ở dòng 1 có các tham số được sử dụng như sau:

decision_function_shape Định nghĩa cách SVC hiện thực bộ phân loại đa lớp. Nếu decision_function_shape='ovr' (one-vs-rest), SVC tạo ra 3 hàm quyết định (decision function): Một câu có thuộc lớp *tích cực* hay không, một câu có thuộc lớp *tiêu cực* hay không và một câu có thuộc lớp *trung tính* hay không. Nếu decision_function_shape='ovo' (one-vs-one), SVC tạo ra 3 hàm quyết định: Một câu



HÌNH 5.9: Kết hợp các đặc trưng trước khi đưa vào SVM huấn luyện

thuộc lớp *tích cực* hay thuộc lớp *tiêu cực*, một câu thuộc lớp *tiêu cực* hay thuộc lớp *trung tính* và một câu thuộc lớp *trung tính* hay thuộc lớp *tích cực*

C là hệ số được định nghĩa trong mô hình biên mềm của giải thuật SVM. Hệ số này đánh đổi giữa việc chấp nhận 1 vài dữ liệu bị phân loại sai, bù lại việc mặt phẳng hàm quyết định (*decision surface*) trở nên phức tạp, ít tuyến tính. Giá trị C cao thì SVC càng phân loại chính xác các dữ liệu trong tập huấn luyện, ngược lại C thấp thì mặt phẳng hàm quyết định càng đơn giản.

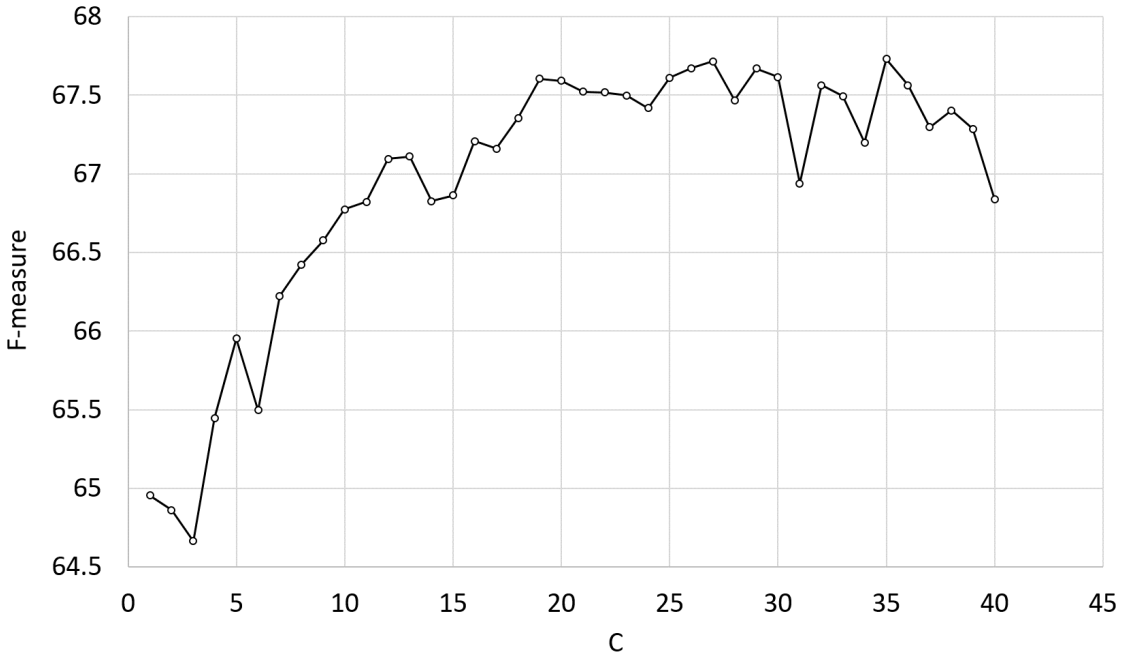
kernel Định nghĩa loại *kernel* SVC sử dụng: `linear, poly, rbf, sigmoid, precomputed`

class_weight Tùy chọn cách xử lý khi số lượng các lớp trong tập huấn luyện bị chêch lệch

Sau khi khởi tạo, hàm `clf.fit(data_x, data_y)` được gọi với `data_x` là dữ liệu tập huấn luyện sau khi đã qua các bước rút trính đặc trưng để ánh xạ từ 1 câu sang 1 *vector*, `data_y` là *vector* chứa nhãn tương ứng.

5.3 Hiện thực phương pháp kiểm tra chéo

Chúng tôi sử dụng kiểm tra chéo (*Cross validation*) để điều chỉnh tham số C trong giải thuật SVM. Chúng tôi thử nghiệm với giá trị C lần lượt 1, 10, 100, 500, 1000. Sau đó, chúng tôi co hẹp dần khoảng giá trị C và nhận thấy $C \in [25, 35]$ đạt hiệu quả tốt nhất. Từ đó, với mỗi thử nghiệm được trình bày ở Mục 6.3, chúng tôi thử các giá trị $C \in [25, 35]$ và chọn ra C với kết quả tốt nhất. Một ví dụ thể hiện sự phụ thuộc độ đo F vào C được mô tả như Hình 5.10.



Hình 5.10: Một thử nghiệm trên đặc trưng N-gram, sử dụng cross-validation để tìm tham số C

Chương 6

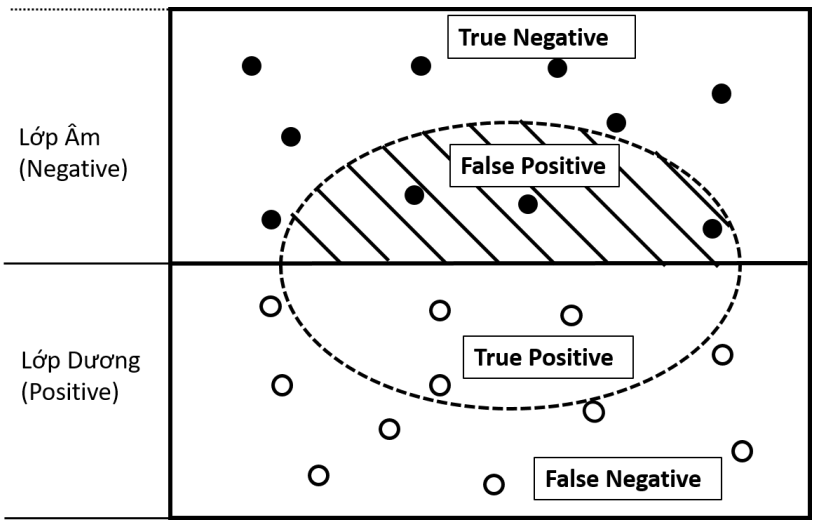
Thí nghiệm và đánh giá

Ở phần này chúng tôi sẽ trình bày chi tiết các phương pháp đánh giá, cách thức thu tập tập dữ liệu, và kết quả thí nghiệm của hệ thống, cùng với những phân tích dựa trên kết quả thí nghiệm thu được.

6.1 Phương pháp đánh giá

Sau khi huấn luyện hệ thống với tập dữ liệu huấn luyện, chúng tôi tiến hành đánh giá kết quả huấn luyện để xác định mức độ tin cậy cũng như hiệu quả của hệ thống. Chúng tôi sử dụng 3 độ đo sau để đánh giá hiệu quả của hệ thống: Độ chính xác (Precision), độ bao phủ (Recall), và độ đo F. Đây cũng là các độ đo thường được sử dụng trong các bài toán Học máy và Truy hồi thông tin. Cả 3 độ đo này chỉ được áp dụng trực tiếp vào bài toán phân loại nhị phân.

Điều kiện trước tiên để áp dụng các độ đo này là cần quy ước 1 lớp dương (*positive*) - là lớp mà hệ thống quan tâm nhiều hơn. Ví dụ trong bài toán phân loại người bị ung thư với người không bị ung thư, chúng ta có 2 lớp: bị ung thư và không bị ung thư. Vậy để áp dụng 3 độ đo trên, người thiết kế hệ thống cần quy ước lớp bị ung thư là lớp dương đối với bài toán này (người thiết kế cũng có thể quy ước ngược lại, việc này phụ thuộc vào ngữ cảnh của bài toán: hệ thống cần chọn ra những người bị ung thư trong cộng đồng, hay cần chọn ra những người không bị ung thư trong cộng đồng.



Hình chữ nhật Toàn bộ dữ liệu
Hình eclip Phần dữ liệu hệ thống cho rằng thuộc lớp dương
True Negative Phần dữ liệu thuộc lớp âm, hệ thống cũng cho rằng thuộc lớp âm
False Negative Phần dữ liệu thuộc lớp dương, hệ thống cho rằng thuộc lớp âm
True Positive Phần dữ liệu thuộc lớp âm, hệ thống cũng cho rằng thuộc lớp dương
False Positive Phần dữ liệu thuộc lớp âm, hệ thống cho rằng thuộc lớp dương

HÌNH 6.1: Các thành phần trong các phép đo Độ chính xác, Độ bao phủ và f1

Dựa trên câu trả lời là tập hợp các phần tử mà hệ thống cho rằng thuộc lớp dương, toàn bộ dữ liệu sẽ được chia thành 4 nhóm như Hình 6.1. Từ đó, định nghĩa các phép đo như sau:

Độ chính xác (Precision)

Độ chính xác P là hệ số đánh giá mức độ chính xác của câu trả lời, mức độ chính xác càng cao thì giá trị P càng lớn. Công thức tính P như sau:

$$P = \frac{\text{Tổng số câu trả lời đúng hệ thống đưa ra}}{\text{Tổng số câu trả hệ thống đưa ra}} = \frac{|True\ Positive|}{|True\ Positive| + |False\ Positive|}$$

Trong trường hợp tất cả những câu trả lời hệ thống đưa ra đều đúng, thì giá trị $P = 1$ là lớn nhất. Quy ước rằng nếu hệ thống không đưa ra câu trả lời nào, khi đó ngầm hiểu hệ thống không "sai", giá trị $P = 1$.

Độ bao phủ (Recall)

Độ bao phủ R là hệ số đánh giá mức độ bao phủ của các câu trả lời, độ bao phủ càng cao thì R càng lớn. Công thức tính R như sau:

$$R = \frac{\text{Tổng số câu trả lời đúng hệ thống đưa ra}}{\text{Tổng số câu trả lời đúng thực tế}} = \frac{|True\ Positive|}{|True\ Positive| + |False\ Negative|}$$

Trong trường hợp hệ thống đưa ra đủ tất cả các câu trả lời đúng thì giá trị $R = 1$ là lớn nhất.

F-measure

Trên thực tế, với lượng dữ liệu lớn và phức tạp, khả năng tất cả những câu trả lời hệ thống đưa ra đều đúng và hệ thống đưa ra đủ tất cả các câu trả lời đúng ($P = R = 1$) là rất thấp. Hai hệ số này thường bù trừ lẫn nhau, tức là để đạt độ bao phủ R cao, hệ thống có xu hướng đưa ra nhiều câu trả lời hơn làm cho xác suất có câu trả lời sai tăng, độ chính xác P giảm, và ngược lại. Một hệ thống nếu chỉ có P cao mà R thấp thì tuy hệ thống có câu trả lời thường đúng nhưng lại bỏ sót nhiều trường hợp đúng khác. Hệ thống chỉ có R cao mà P thấp thì hệ thống đó tuy bao quát đầy đủ tất cả các trường hợp đúng thực tế, nhưng tỉ lệ câu trả lời sai lại lớn. Một hệ thống tốt yêu cầu cả độ chính xác P và độ bao phủ đều cao R. Do đó vấn đề đặt ra là tìm một độ đo duy nhất mà đảm bảo cả P và R để thuận tiện cho việc tối ưu.

Để giải quyết vấn đề trên, ta sử dụng tiêu chí đánh giá F là trung bình điều hòa của P và R, từ đó đảm bảo rằng chỉ khi cả P và R cao thì F mới đạt giá trị cao. Công thức tính F như sau:

$$F = 2 \times \frac{P \times R}{P + R}$$

Độ chính xác (P), độ bao phủ (R), F đối với bài toán phân loại đa lớp

Để áp dụng 3 độ đo trên vào bài toán phân loại 3 lớp, chúng tôi xem như đang giải 3 bài toán con thuộc loại bài toán phân loại nhị phân. Khi đó, điểm số của bài toán lớn bằng trung bình có trọng số của điểm số từng bài toán con. Xét 3 bài toán con:

1. Phân loại một câu thuộc lớp *tích cực* hay không thuộc lớp *tích cực*. Lớp dương được chọn là lớp *tích cực*.
2. Phân loại một câu thuộc lớp *tiêu cực* hay không thuộc lớp *tiêu cực*. Lớp dương được chọn là lớp *tiêu cực*.
3. Phân loại một câu thuộc lớp *trung tính* hay không thuộc lớp *trung tính*. Lớp dương được chọn là lớp *trung tính*.

Khi đó:

$$P = \alpha_1 \times P_1 + \alpha_2 \times P_2 + \alpha_3 \times P_3$$

$$R = \alpha_1 \times R_1 + \alpha_2 \times R_2 + \alpha_3 \times R_3$$
$$F = \alpha_1 \times F_1 + \alpha_2 \times F_2 + \alpha_3 \times F_3$$

với $\alpha_1 + \alpha_2 + \alpha_3 = 1$ và $\alpha_1, \alpha_2, \alpha_3$ lần lược là tỉ lệ số lượng các câu thuộc lớp *tích cực*, *tiêu cực*, *trung tính* trong tập dữ liệu huấn luyện.

K-fold cross validation

Sau khi đã có thước đo F, chúng tôi đề xuất sử dụng phương pháp *k-fold cross validation* để đánh giá hiệu quả của hệ thống. Phương pháp này nhằm tránh trường hợp tập kiểm tra, vì được chia ngẫu nhiên, có thể rơi vào trường hợp quá dễ hoặc quá khó đối với hệ thống. Tập dữ liệu được chia ngẫu nhiên thành k phần. Phần thứ i sẽ được chọn làm tập để đánh giá, $k-1$ phần còn lại dùng cho việc học các tham số của mô hình. Tiến trình trên được thực hiện k lần với i chạy từ 1 đến k, giá trị trung bình là kết quả cuối cùng dùng để đánh giá hệ thống.

6.2 Thu thập và đánh giá dữ liệu

Thu thập dữ liệu

Trong nghiên cứu này, chúng tôi thu thập dữ liệu từ *website* PubMed [1]. Trang web cung cấp miễn phí tóm tắt (*abstract*) của các bài báo khoa học trong lĩnh vực y khoa. Phần tóm tắt của các bài báo không có cấu trúc chung, chúng tôi chỉ thu thập những tóm tắt nào có chứa phần kết luận (*Conclusion*) như Hình 6.2. Báo cáo [16] gợi ý rằng sử dụng những câu kết luận giúp tăng hiệu quả phân loại, vì vậy chúng tôi chỉ giữ lại phần kết luận, và sử dụng công cụ tìm kiếm cùng bộ lọc để tìm các bài có loại xuất bản (*publication type*) là "clinical trial".

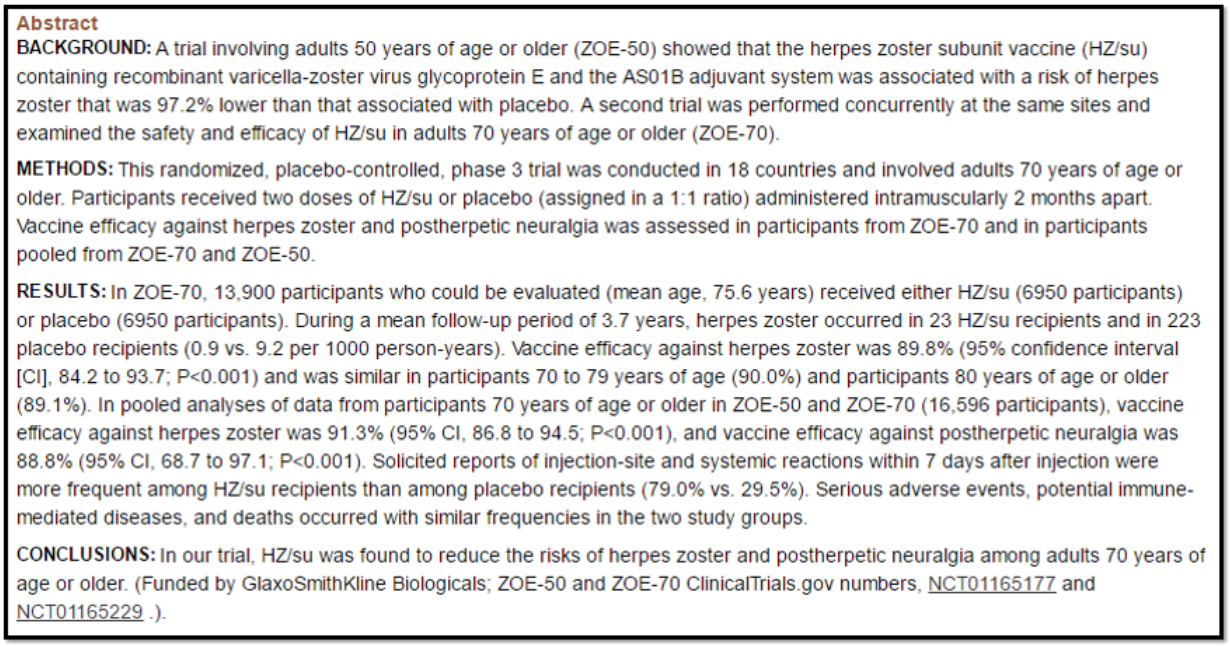
Vì *website* PubMed không chính thức hỗ trợ cung cấp dữ liệu, chúng tôi tự hiện thực công cụ phục vụ nhu cầu này. Chúng tôi sử dụng hàm chức năng `re` trong Python để gửi các yêu cầu HTTP lên *website* PubMed. Khi PubMed trả về phản hồi, thông tin được phân giải và lọc ra phần nội dung cần lấy.

Sau khi chạy công cụ trên, kết quả thu được là những đoạn văn thuộc phần kết luận. Bản thân việc phân tách câu trong đoạn cũng là 1 bài toán, thường được gọi là Định hướng ranh giới câu (*Sentence boundary disambiguation*), không nằm trong phạm vi đề tài. Trong luận án này, chúng tôi sử dụng giải thuật phân tách câu Punkt (*Punkt sentence segmenter*) [39] được hiện thực trong thư viện NLTK. Dữ liệu chúng tôi thu thập được gồm 1182 câu được gán mã số (*Id*) tuần tự và lưu trữ như Bảng 6.1.

Đánh nhãn dữ liệu

Giải thuật học máy chúng tôi sử dụng là có giám sát nên dữ liệu đầu vào cần được đánh nhãn phân loại tính phân cực cảm xúc (*tích cực*, *tiêu cực*, *trung tính*) trước khi đưa vào học và kiểm tra. Vì vậy, chúng tôi đã hiện thực một trang web phục vụ cho việc đánh nhãn dữ liệu.

[1] https://www.ncbi.nlm.nih.gov/pubmed



Abstract

BACKGROUND: A trial involving adults 50 years of age or older (ZOE-50) showed that the herpes zoster subunit vaccine (HZ/su) containing recombinant varicella-zoster virus glycoprotein E and the AS01B adjuvant system was associated with a risk of herpes zoster that was 97.2% lower than that associated with placebo. A second trial was performed concurrently at the same sites and examined the safety and efficacy of HZ/su in adults 70 years of age or older (ZOE-70).

METHODS: This randomized, placebo-controlled, phase 3 trial was conducted in 18 countries and involved adults 70 years of age or older. Participants received two doses of HZ/su or placebo (assigned in a 1:1 ratio) administered intramuscularly 2 months apart. Vaccine efficacy against herpes zoster and postherpetic neuralgia was assessed in participants from ZOE-70 and in participants pooled from ZOE-70 and ZOE-50.

RESULTS: In ZOE-70, 13,900 participants who could be evaluated (mean age, 75.6 years) received either HZ/su (6950 participants) or placebo (6950 participants). During a mean follow-up period of 3.7 years, herpes zoster occurred in 23 HZ/su recipients and in 223 placebo recipients (0.9 vs. 9.2 per 1000 person-years). Vaccine efficacy against herpes zoster was 89.8% (95% confidence interval [CI], 84.2 to 93.7; P<0.001) and was similar in participants 70 to 79 years of age (90.0%) and participants 80 years of age or older (89.1%). In pooled analyses of data from participants 70 years of age or older in ZOE-50 and ZOE-70 (16,596 participants), vaccine efficacy against herpes zoster was 91.3% (95% CI, 86.8 to 94.5; P<0.001), and vaccine efficacy against postherpetic neuralgia was 88.8% (95% CI, 68.7 to 97.1; P<0.001). Solicited reports of injection-site and systemic reactions within 7 days after injection were more frequent among HZ/su recipients than among placebo recipients (79.0% vs. 29.5%). Serious adverse events, potential immune-mediated diseases, and deaths occurred with similar frequencies in the two study groups.

CONCLUSIONS: In our trial, HZ/su was found to reduce the risks of herpes zoster and postherpetic neuralgia among adults 70 years of age or older. (Funded by GlaxoSmithKline Biologicals; ZOE-50 and ZOE-70 ClinicalTrials.gov numbers, NCT01165177 and NCT01165229 .).

Hình 6.2: Tóm tắt của 1 bài báo

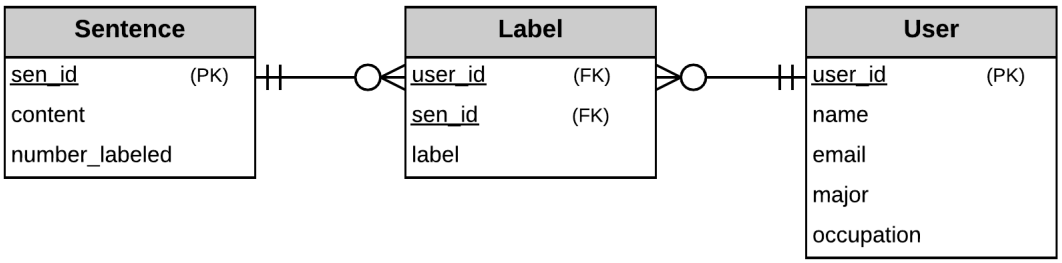
Bảng 6.1: Một số mẫu từ tập dữ liệu sau khi thu thập

Id	Sentence
10	This study was a negative study, though there was a suggestion of benefit of methylprednisolone acetate in a population of young adults with acute radicular low back pain.
17	Data extraction and analyses and quality assessment were conducted according to the Cochrane standards.
36	Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or no treatment.

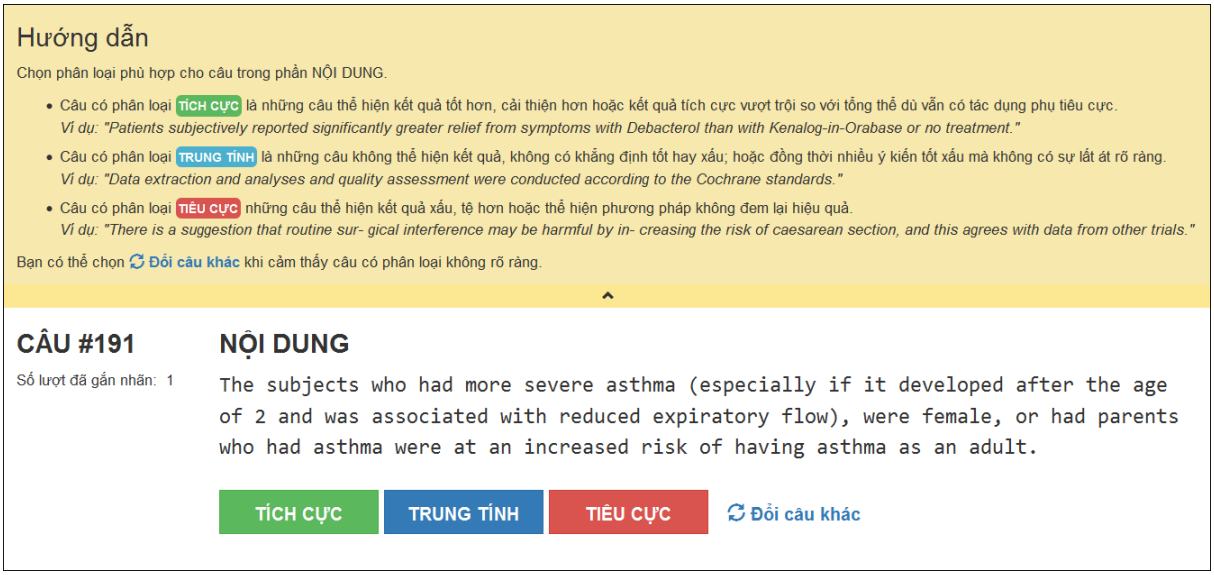
Đầu tiên, chúng tôi sử dụng hệ quản trị cơ sở dữ liệu MySQL Workbench 6.3 [1] để tạo cơ sở dữ liệu lưu trữ theo lược đồ ở Hình 6.3:

- Bảng `Sentence` chứa dữ liệu các câu gồm mã số câu (`sen_id`), nội dung câu (`content`) và số lần câu được đánh nhãn (`number_labeled`). Tập dữ liệu sau khi thu thập sẽ được thêm vào bảng này với giá trị số lần câu được đánh nhãn ở mỗi câu ban đầu mặc định bằng 0.
- Bảng `Submission` chứa nhật ký đánh nhãn gồm mã số nhãn (`id`) tăng tuần tự theo mỗi nhãn được đánh cho một câu bất kỳ, mã số câu (`sentence_id`) tương ứng với mã số câu ở Bảng `Sentence`, loại nhãn (`label`) với quy ước giá trị 0, 1, 2 lần lượt tương ứng cho các nhãn *tiêu cực*, *trung tính*, *tích cực*.

[1] http://www.mysql.com/products/workbench/



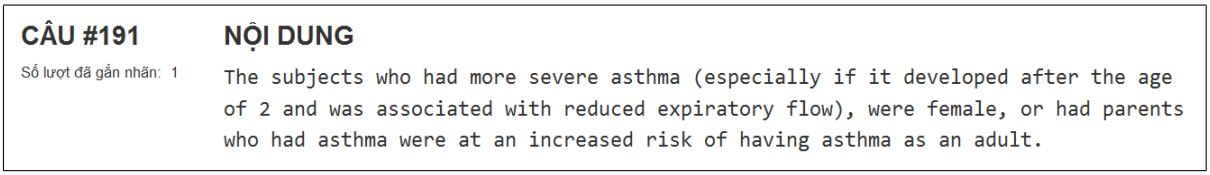
HÌNH 6.3: Mô hình thực thể liên kết tăng cường của cơ sở dữ liệu



HÌNH 6.4: Giao diện trang đánh nhãn dữ liệu

Tiếp theo, chúng tôi xây dựng một trang web hỗ trợ người dùng đánh nhãn dữ liệu[1]. Giao diện web đơn giản, trực quan (Hình 6.4), gồm hai phần chính:

- Phần hướng dẫn: mô tả cách sử dụng trang web để đánh nhãn dữ liệu, trong đó đặc tả chi tiết thế nào là phân loại *tích cực*, *tiêu cực* hay *trung tính*, lấy ví dụ cụ thể để người đọc dễ hình dung và hiểu rõ ràng về các loại nhãn phân loại.
- Phần đánh nhãn: hiển thị thông tin một câu ngẫu nhiên thuộc bảng `Sentence` trong cơ sở dữ liệu và các lựa chọn để người dùng đánh nhãn phân loại.

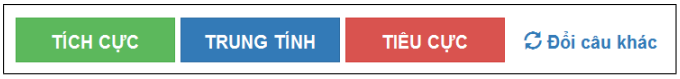


HÌNH 6.5: Thông tin một bản ghi thuộc bảng "Sentence"

Thông tin chi tiết một câu được hiển thị như Hình 6.5 bao gồm giá trị các trường thuộc bảng `Sentence` (mã số câu, nội dung câu, số lần câu được đánh nhãn). Để gắn nhãn

[1]http://anotation.mybluemix.net/

phân loại cho câu, người dùng sử dụng nhóm nút chức năng (Hình 6.6) gồm nút TÍCH CỰC (gắn nhãn *tích cực*), nút TRUNG TÍNH (gắn nhãn *trung tính*), nút TIÊU CỰC (gắn nhãn *tiêu cực*) hoặc lựa chọn "Đổi câu khác" nếu người dùng cảm thấy câu có phân loại không rõ ràng.



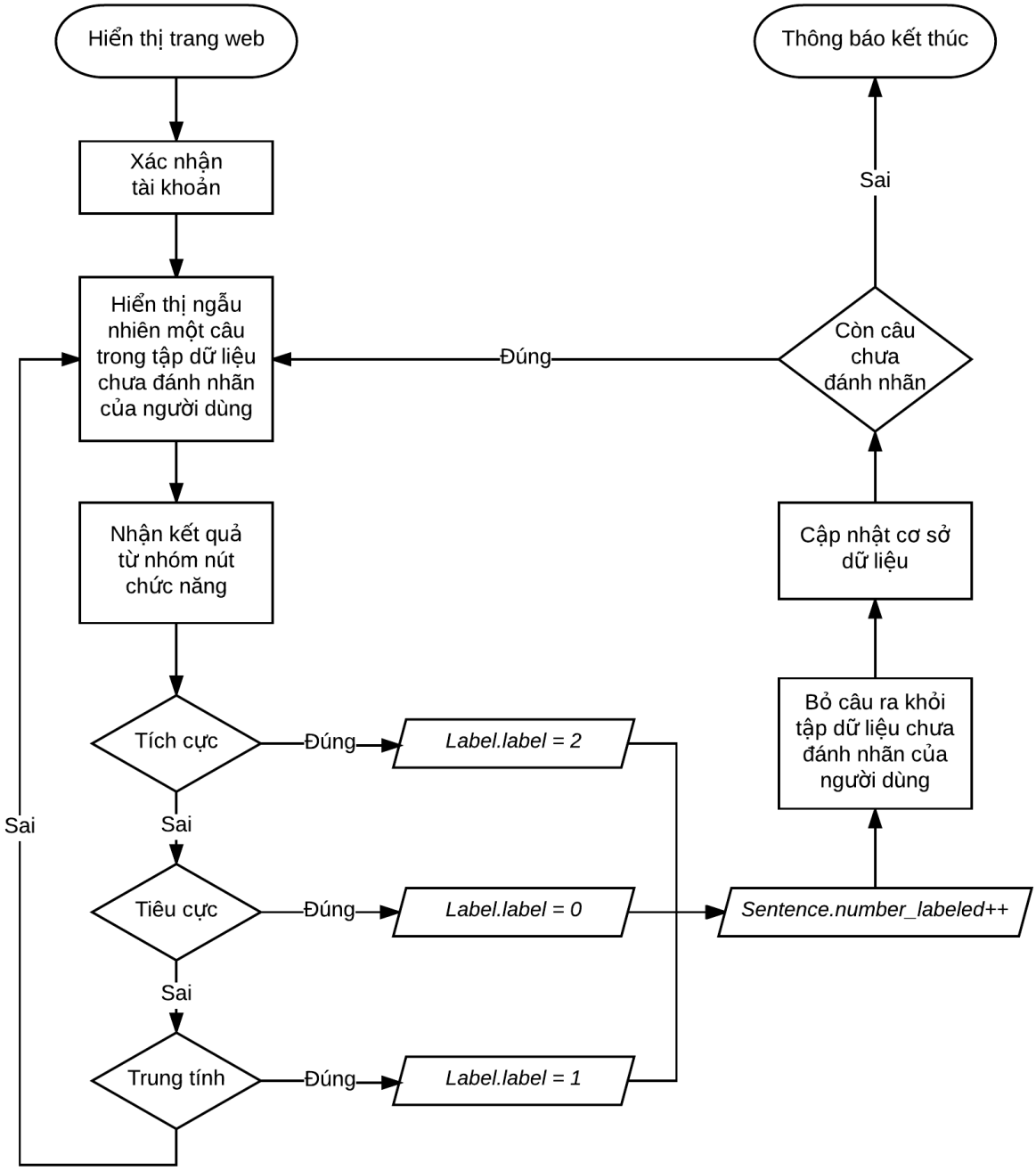
HÌNH 6.6: Nhóm nút chức năng hỗ trợ người dùng lựa chọn phân loại

Quy trình gắn nhãn dữ liệu của trang web được mô hình hóa như Hình 6.7. Mỗi lần người dùng mới truy cập hoặc tải lại trang web, hệ thống lựa chọn ngẫu nhiên một câu trong dữ liệu để hiển thị. Khi người dùng đánh nhãn *tích cực*, *tiêu cực* hoặc *trung tính* cho câu, cơ sở dữ liệu sẽ cập nhật dữ liệu trong bảng `Sentence`: tăng thêm 1 cho số lần câu được đánh nhãn với mã số câu tương ứng, đồng thời thêm một bản ghi vào bảng `Submission` với mã số câu tương ứng kèm lựa chọn phân loại của người dùng (chỉ lưu các giá trị 0, 1, 2). Nếu người dùng chọn "Đổi câu khác" hệ thống không cập nhật mà chỉ hiển thị ngẫu nhiên một câu khác trong tập dữ liệu để người dùng tiến hành gán nhãn.

Sau khi hoàn thành trang web đánh nhãn dữ liệu, chúng tôi đã tiến hành gắn nhãn cho tập dữ liệu. Kết quả tập dữ liệu sau khi gắn nhãn được lưu lại như Bảng 6.2.

BẢNG 6.2: Một số mẫu từ tập dữ liệu sau khi đánh nhãn

Id	**Sentence**	**Label**
10	This study was a negative study, though there was a suggestion of benefit of methylprednisolone acetate in a population of young adults with acute radicular low back pain.	0
17	Data extraction and analyses and quality assessment were conducted according to the Cochrane standards.	1
36	Patients subjectively reported significantly greater relief from symptoms with Debacterol than with Kenalog-in-Orabase or no treatment.	2



Hình 6.7: Quy trình xử lý gắn nhãn dữ liệu của trang web

Đánh giá dữ liệu

Sau khi xây dựng trang web hỗ trợ đánh nhãn, chúng tôi nhờ 14 bạn sinh viên đánh nhãn, với các câu được lựa chọn ngẫu nhiên. Kết quả thu được 161 câu được đánh nhãn. Để đánh giá tập dữ liệu, chúng tôi sử dụng phương pháp đánh giá độ đồng nhất *Fleiss's Kappa* trên tập gồm 46 câu, mỗi câu được đánh nhãn bởi 2 người bất kỳ, khi đó $m = 2, n = 46, k = 3$, kết quả $\kappa = 36.91\%$. Điểm số này thể hiện độ đồng nhất thuộc mức vừa (*fair argreement*).

Kết quả này còn thấp so với một số nghiên cứu tự xây dựng tập dữ liệu (nghiên cứu [16] đạt 70.6%, trong khi [15] đạt 65%). Điều này xảy ra vì sự nhập nhằng về ý nghĩa của 1 số câu và sự hiểu không đồng nhất trong quy ước về tính *tích cực*, *tiêu cực*, *trung tính*

giữa những người đánh nhãn. Để giải quyết, 2 thành viên trong nhóm chúng tôi tự đánh nhãn các câu còn lại. Tổng số câu được đánh nhãn là 552, bao gồm: 72 câu *tiêu cực*, 240 câu *trung tính*, 240 câu *tích cực*. Kết quả *kappa* trên tập này là $\kappa = 72.54\%$. Kết quả này được xem là khá tốt (*substantial agreement*).

6.3 Kết quả thí nghiệm

Với mỗi lần chạy, chúng tôi sử dụng *k-fold* với $k = 5$. Tuy nhiên, do tập dữ liệu không đủ lớn, kết quả các lần chạy có sự dao động, vì vậy, với mỗi lần thử nghiệm, chúng tôi lặp lại việc chạy *k-fold* 30 lần, sau đó lấy điểm số trung bình xem như kết quả cuối cùng. Đối với giải thuật học máy SVM, có 2 tham số cần được tùy chỉnh thích hợp: tham số *c* và *gamma*.

Tham số *gamma* được sử dụng nếu dùng *kernel rbf* quy định mức độ ảnh hưởng của 1 điểm dữ liệu đến các điểm xung quanh. Tham số này được tối ưu bởi thư viện. Để xác định tham số *c*, hệ thống chạy nhiều lần với *c* thuộc 1 khoảng cho trước (trong nghiên cứu này, *c* chạy từ 20 đến 30), từ đó chọn ra giá trị *c* tương ứng với độ đo *F* lớn nhất. Như vậy, với mỗi thử nghiệm, giá trị độ do *F* luôn là giá trị cao nhất được chọn từ những lần chạy với giá trị *c* thay đổi từ 20 đến 30.

Trong mục này chúng tôi trình bày 4 nhóm thử nghiệm: Thử nghiệm với đặc trưng N-gram, thử nghiệm với đặc trưng phủ định, thử nghiệm kết hợp các đặc trưng cơ bản và thử nghiệm kết hợp các đặc trưng cơ bản với đặc trưng mở rộng SO-CAL

Thử nghiệm với đặc trưng N-gram

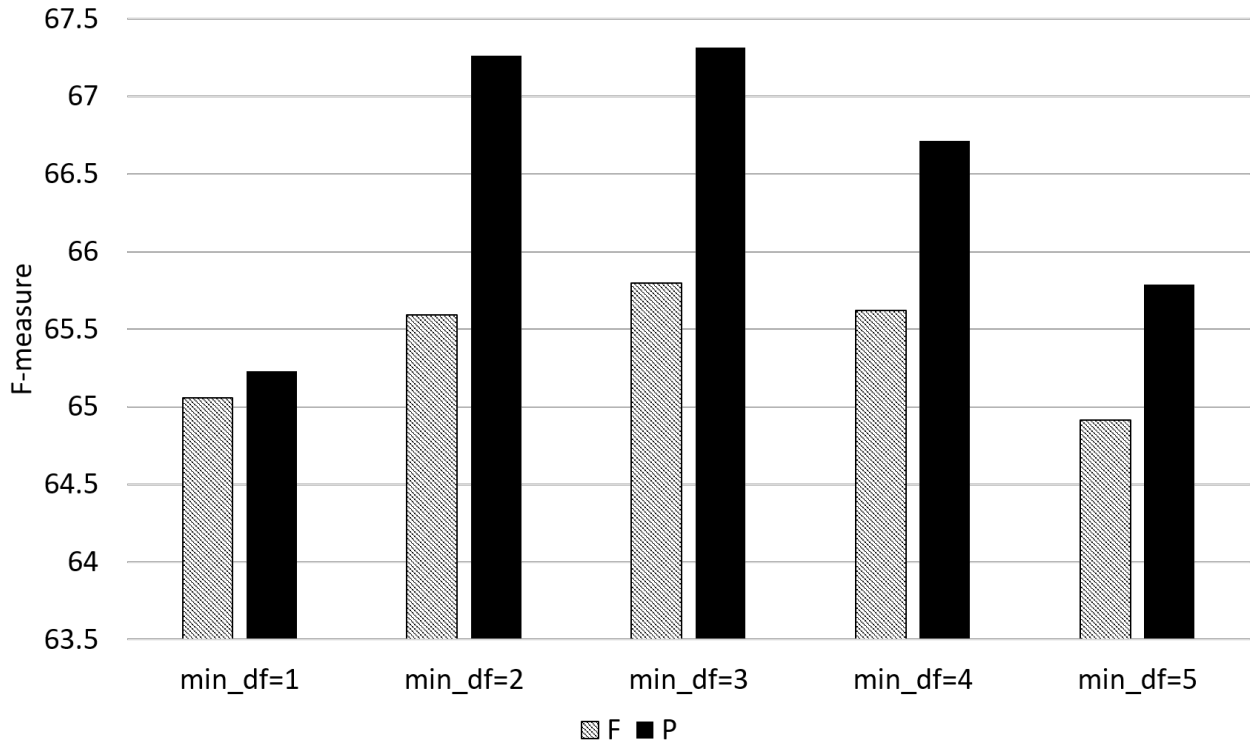
Chúng tôi tiến hành thử nghiệm với đặc trưng N-gram trước tiên vì như đã phân tích ở mục 4.3, N-gram được xem như đặc trưng nền tảng. Kết quả thử nghiệm thể hiện ở Bảng 6.3.
Các thử nghiệm từ 1 đến 10 cho thấy 2 xu hướng. Hình 6.8 thể hiện sự phụ thuộc của độ đo *F* vào tham số *min_df* và cách *vector* hóa đặc trưng N-gram. *min_df* là tham số ngưỡng, chỉ những n-gram nào có số lần xuất hiện từ *min_df* trở lên mới được thêm vào tập từ vựng S. Qua biểu đồ có thể thấy, với *min_df* quá nhỏ hoặc quá lớn đều làm giảm giá trị độ đo *F*. Khi *min_df* quá nhỏ, số lượng từ vựng quá lớn dẫn tới số lượng n-gram gây nhiễu nhiều. Ngược lại khi *min_df* quá lớn, tập từ vựng quá nhỏ dẫn tới có quá ít thông tin trong câu được giữ lại, không đủ thông tin để phân loại. Từ biểu đồ, $min\_df = 2$ hay $min\_df = 3$ không có sự khác biệt rõ rệt giá trị độ đo F. Trong nghiên cứu này, chúng tôi chọn tham số $min\_df = 3$ cho các thử nghiệm còn lại.

Hình 6.8 còn thể hiện một xu hướng khác. Kết quả của đặc trưng N-gram tốt hơn hẳn khi sử dụng phương pháp vector hóa nhị phân. Kết quả này phù hợp với báo cáo của [7]. Ngược lại, nghiên cứu [16] thực hiện phân tích cảm xúc trên đoạn văn bản, khẳng định không có sự khác biệt đáng kể giữa 2 cách vector hóa. Điều này có thể do việc lặp từ trên câu có ý nghĩa khác với lặp từ trên đoạn.

BẢNG 6.3: Các thử nghiệm nhằm tối ưu hóa đặc trưng N-gram

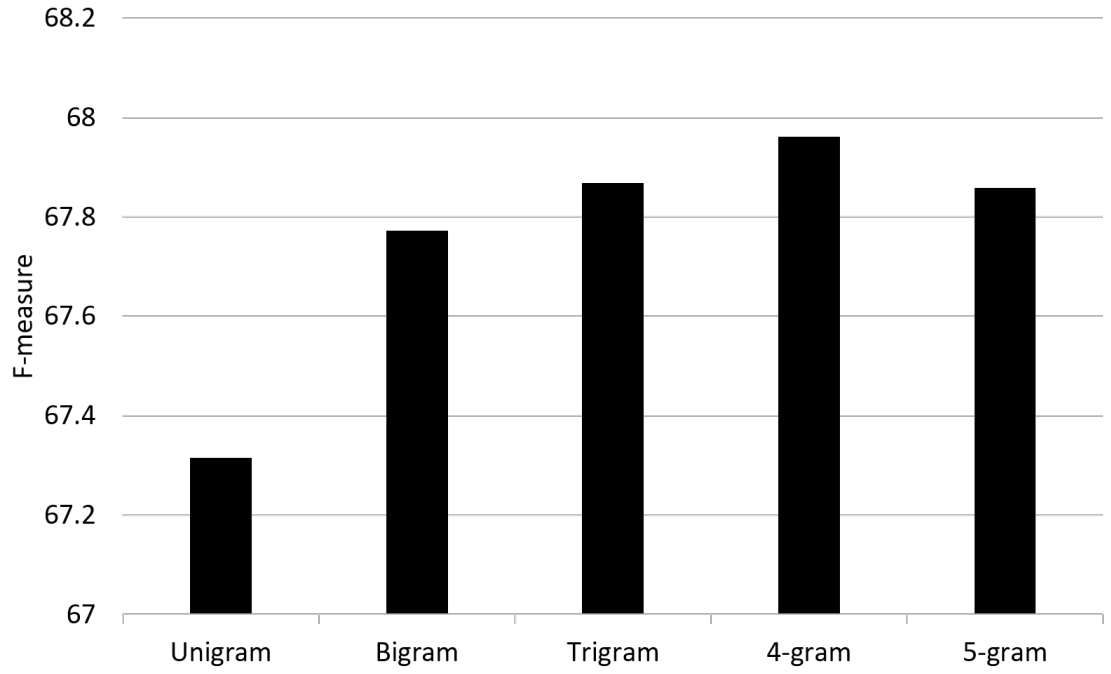
STT	**Đặc trưng**	**P (%)**	**R (%)**	**F (%)**
1	Unigram (F, min_df = 1)	66.21	65.46	65.06
2	Unigram (P, min_df = 1)	66.56	65.62	65.23
3	Unigram (F, min_df = 2)	66.50	65.55	65.59
4	Unigram (P, min_df = 2)	68.26	67.38	67.31
5	Unigram (F, min_df = 3)	66.98	65.48	65.80
6	Unigram (P, min_df = 3)	68.12	67.23	67.32
7	Unigram (F, min_df = 4)	67.02	65.13	65.62
8	Unigram (P, min_df = 4)	67.83	66.34	66.71
9	Unigram (F, min_df = 5)	66.75	64.21	64.92
10	Unigram (P, min_df = 5)	67.18	65.27	65.79
11	Unigram + Bigram (P, min_df = 3)	68.72	67.83	67.77
12	Unigram + Bigram + Trigram (P, min_df = 3)	68.68	68.00	67.87
13	**Unigram + Bigram + Trigram + 4-gram(P, min_df = 3)**	**68.76**	**68.09**	**67.96**
14	Unigram + Bigram + Trigram + 4-gram + 5-gram (P, min_df = 3)	68.81	67.98	67.86

P Quan tâm đến việc n-gram có xuất hiện trong câu hay không, nhận 2 giá trị: 1 hoặc 0
F Quan tâm đến số lần xuất hiện n-gram trong câu
min_df Số câu có n-gram đó để được thêm vào bộ từ vừng



HÌNH 6.8: Mối quan hệ giữa tham số min_df, cách vector hóa và độ đo F

Hình 6.9 thể hiện ảnh hưởng của cách kết hợp các đặc trưng N-gram. Theo đó, có sự cải thiện khi chuyển từ việc chỉ dùng Uni-gram sang dùng kết hợp Uni-gram và Bi-gram. Khi mở rộng việc kết hợp với Tri-gram và 4-gram, mặc dù chỉ số F có tăng nhưng không thực sự đáng kể. Khi kết hợp đến 5-gram, chỉ số F bắt đầu giảm. Với n càng lớn, mặc dù số lượng ngram không khác nhau nhiều (giả sử 1 câu có 20 từ, với $n = 1$ tạo ra 20 n-gram, $n = 3$ tạo ra 18 n-gram) nhưng các n-gram có tần suất xuất hiện càng thấp. Trong khi đó, hệ thống chỉ thêm n-gram vào tập từ vựng S chỉ khi n-gram đó xuất hiện từ 3 lần trở lên. Vì vậy, việc kết hợp các n-gram (với n lớn như $n = 5$) không có hiệu quả. Thay vào đó, với $n = 2, 3, 4$ giúp hệ thống nhận thêm các cụm từ như: no evidence, improve quality life, reduce risk,... Trong các thử nghiệm tiếp theo, chúng tôi dùng đặc trưng N-gram là sự kết hợp của Uni-gram, Bi-gram, Tri-gram và 4-gram.



Tên mỗi cột chỉ là đặc trưng N-gram đại diện. Ví dụ: Tri-gram đại diện cho thử nghiệm thứ 12, là kết hợp cả Uni-gram, Bi-gram và Tri-gram

Hình 6.9: Kết hợp các N-gram

Thử nghiệm với đặc trưng Phủ định

Các thử nghiệm trong phần này đều dùng kết hợp với đặc trưng N-gram. Bảng 6.4 thể hiện hiệu quả của việc rút trích yếu tố phủ định qua 7 tổ hợp, so sánh giữa 2 công cụ: Meta-NegEx và Gen-NegEx. Trong 7 thí nghiệm, cách dùng của thí nghiệm 2 cho kết quả tốt nhất, xét cho cả 2 công cụ. Trong đó, Gen-NegEx cho kết quả cao hơn. Chúng tôi sử dụng phương pháp như thí nghiệm 2 và công cụ Gen-NegEx cho các thí nghiệm ở phần sau.

Bảng 6.4: Các thử nghiệm nhằm tối ưu đặc trưng Phủ định

STT	Đặc trưng	Meta-NegEx			Gen-NegEx		
		P	**R**	**F**	**P**	**R**	**F**
1	Kiểm tra trong câu có yếu tố phủ định hay không	69.36	68.53	68.45	69.59	68.32	68.50
2	**Thay các từ phủ định trong câu bằng nhãn NEGATION**	**69.57**	**68.67**	**68.60**	**70.07**	**68.98**	**69.09**
3	Thêm nhãn "_NEG" ngay sau các từ chịu ảnh hưởng phủ định	68.93	67.91	67.88	69.11	68.12	68.03
4	Kết hợp 1 và 2	68.88	68.10	68.09	69.95	68.64	68.82
5	Kết hợp 1 và 3	69.41	68.55	68.56	69.47	67.85	68.16
6	Kết hợp 2 và 3	68.84	68.02	67.93	69.00	68.00	67.83
7	Kết hợp 1 và 2 và 3	69.35	68.40	68.44	69.62	67.96	68.22

Thử nghiệm kết hợp các đặc trưng cơ bản

Bảng 6.5: Các thử nghiệm kết hợp các đặc trưng cơ bản

STT	Đặc trưng	P	R	F
1	N-gram	68.76	68.09	67.96
2	N-gram + Chuyển đổi trạng thái	69.97	69.29	69.17
3	**N-gram + Chuyển đổi trạng thái + Phủ định**	**71.01**	**70.00**	**70.05**
4	N-gram + Chuyển đổi trạng thái + Phủ định + Metamap	70.88	70.00	69.99

Bảng 6.5 thể hiện kết quả khi kết hợp các đặc trưng cơ bản với nhau. Đặc trưng Chuyển đổi trạng thái có hiệu quả rõ rệt khi giúp tăng 1.39% so với đặc trưng nền tảng N-gram. Có thêm đặc trưng Phủ định góp phần cải thiện kết quả 2.15%. Tuy nhiên, kết quả khi sử dụng N-gram với Metamap lại không cho kết quả tốt.

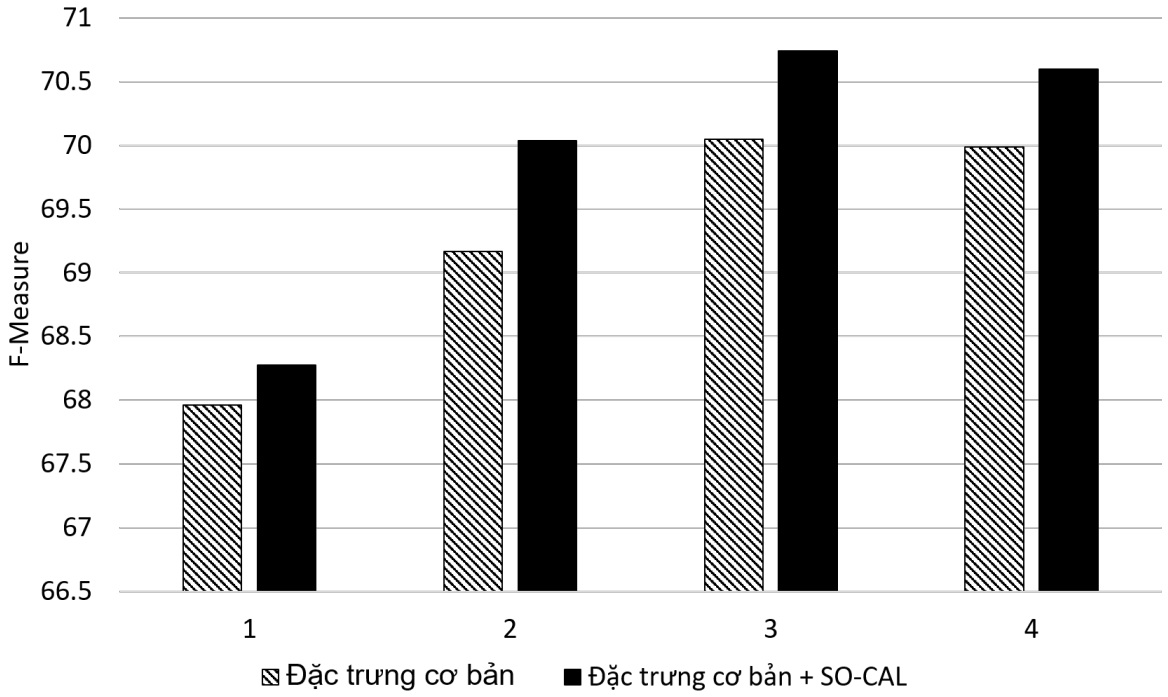
Thử nghiệm kết hợp các đặc trưng cơ bản với đặc trưng mở rộng SO-CAL

Chúng tôi tiến hành thử nghiệm kết hợp đặc trưng mở rộng SO-CAL với các đặc trưng cơ bản, kết quả thể hiện ở Bảng 6.6. Xu hướng vẫn giống kết quả của Bảng 6.5: 2 đặc trưng Chuyển đổi trạng thái và Negation khi kết hợp cùng SO-CAL đều giúp cải thiện hiệu quả (lần lượt là 1.76% và 2.47% so với N-gram kết hợp SO-CAL), tuy nhiên khi sử dụng N-gram kết hợp Metamap kết quả lại không có sự khác biệt và có phần giảm xuống.

Điểm khác biệt so với Bảng 6.5 là có sự cải thiện trong từng lần thử nghiệm giữa tập các đặc trưng có SO-CAL và không có SO-CAL (Hình 6.10). Kết quả tốt nhất trong các thử nghiệm này, cũng là kết quả tốt nhất mà hệ thống chúng tôi đạt được là **70.74%**.

Bảng 6.6: Các thử nghiệm kết hợp các đặc trưng cơ bản với đặc trưng mở rộng SO-CAL

STT	Đặc trưng	P	R	F
1	N-gram + SO-CAL	69.13	68.22	68.27
2	N-gram + Chuyển đổi trạng thái + SO-CAL	70.84	70.05	70.03
3	**N-gram + Chuyển đổi trạng thái + Phủ định + SO-CAL**	**71.64**	**70.70**	**70.74**
4	N-gram + Chuyển đổi trạng thái + Phủ định + Metamap + SO-CAL	71.49	70.54	70.60



Hình 6.10: Hiệu quả của đặc trưng SO-CAL

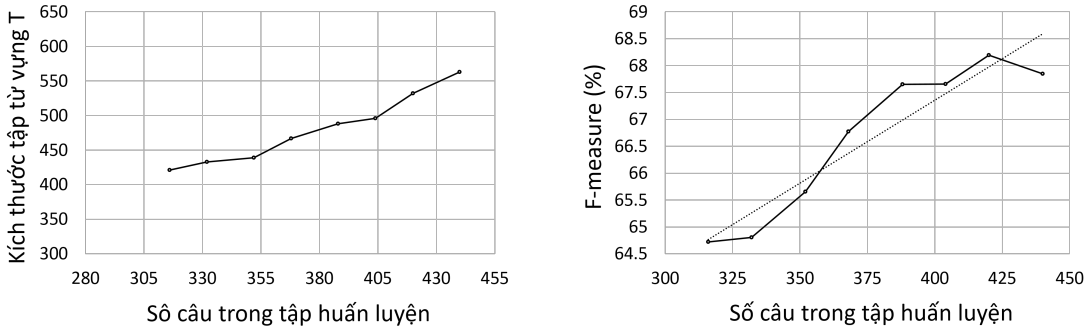
6.4 Các phân tích mở rộng

Sự phụ thuộc hiệu quả đặc trưng N-gram với kích thước tập huấn luyện

Đặc trưng N-gram được sử dụng trong luận án này như đặc trưng nền tảng, vì vậy ảnh hưởng lớn đến kết quả toàn hệ thống. Nghiên cứu [1] là nghiên cứu liên quan gần nhất đến luận án này. Nhóm tác giả sử dụng tập dữ liệu gồm 1509 câu, trong đó 1208 câu dùng để huấn luyện, đạt độ chính xác 77.87% với đặc trưng N-gram. Nghiên cứu [16] cùng lĩnh vực nhưng phân tích trên đoạn, sử dụng tập dữ liệu gồm 520 tài liệu, chứa 9221 câu cũng đạt độ chính xác cao, 74.2%.

Kích thước tập dữ liệu huấn luyện ảnh hưởng lớn đến độ chính xác thông qua kích thước tập từ vựng T. Khi kích thước tập dữ liệu huấn luyện tăng, kích thước tập từ vựng T có xu hướng tăng, từ đó có tính bao quát cao hơn, nghĩa là đặc trưng N-gram "học" được nhiều hơn (tập từ vựng T chứa nhiều từ trong tập kiểm tra hơn). Độ chính xác nhờ vậy cũng tăng lên.

Để kiểm chứng giả thuyết này, chúng tôi thực hiện thử nghiệm huấn luyện bộ phân loại chỉ với đặc trưng N-gram, sử dụng tập dữ liệu huấn luyện có kích thước khác nhau. Kết quả được trình bày ở Hình 6.11. Giả thuyết của chúng tôi cũng đồng nhất với kết luận trong nghiên cứu [16]. Kết luận của [40] cũng cho rằng hiệu suất phân loại không thể đạt kết quả tốt nếu dữ liệu không đủ.



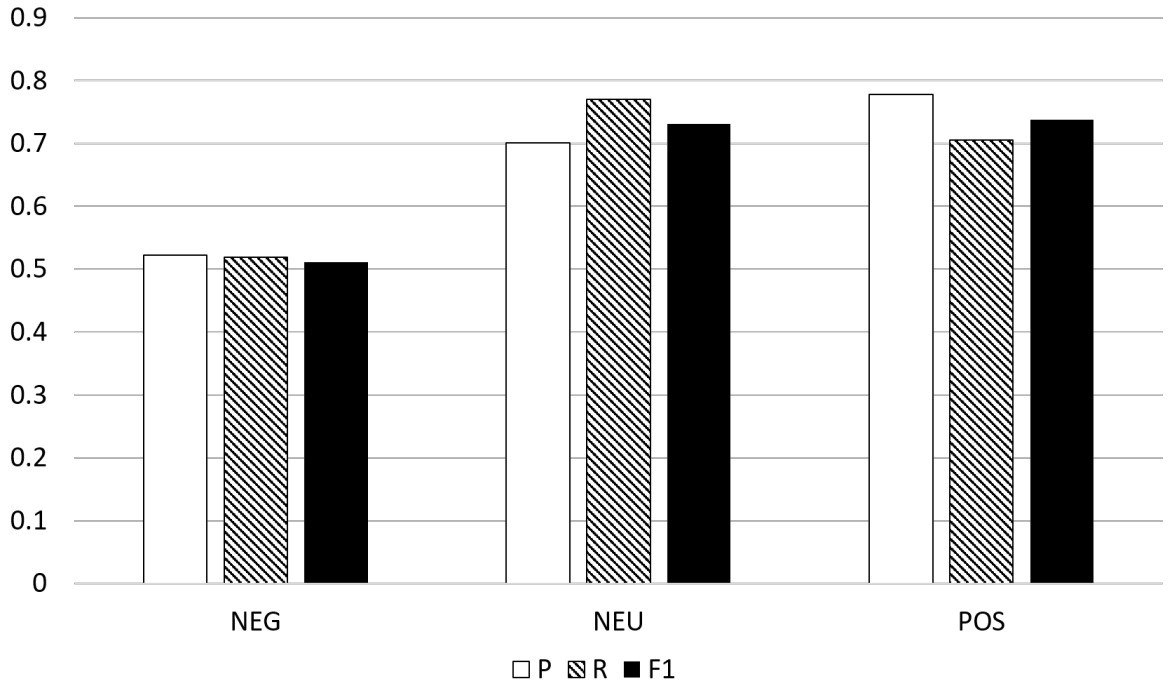
(a) Mối quan hệ giữa kích thước tập dữ liệu huấn luyện và kích thước tập từ vựng T

(b) Mối quan hệ giữa kích thước tập dữ liệu huấn luyện và độ đo F

Hình 6.11: Ảnh hưởng của kích thước tập dữ liệu huấn luyện đến đặc trưng N-gram

So sánh kết quả đối với từng lớp

Chúng tôi tiến hành phân tích kết quả cụ thể đối với từng lớp. Kết quả Hình 6.12 sử dụng lần thử nghiệm đạt kết quả tối ưu nhất, kết hợp các đặc trưng: N-gram, Chuyển đổi trạng thái, Phủ định và SO-CAL. Kết quả 2 lớp *tích cực* và *trung tính* khá đồng nhất, trong khi lớp *tiêu cực* lại cho kết quả thấp hơn rõ rệt. Vì vậy, cần thêm các phân tích cho các câu thuộc lớp *tiêu cực* để tăng độ chính xác của hệ thống.



Hình 6.12: Độ chính xác, độ bao phủ và F-measure trong từng lớp

Chương 7

Tổng kết

Trong Chương 7, chúng tôi trình bày tóm tắt những kết quả đã đạt được của luận án, những hạn chế và hướng phát triển đề tài.

7.1 Kết quả đạt được

Trong luận án này, chúng tôi đã hiện thực thành công hệ thống phân tích tính phân cực cảm xúc trong văn bản y khoa. Hệ thống đã xây dựng có khả năng phân loại câu trong những bài báo cáo nghiên cứu thuộc lĩnh vực y khoa vào 1 trong 3 lớp: *tích cực*, *tiêu cực* hoặc *trung tính*.

Tập dữ liệu chúng tôi thu thập được gồm 552 câu, được đánh nhãn thủ công bởi 16 sinh viên (2 thành viên trong nhóm và 14 bạn sinh viên khác đang theo học tại Trường Đại Học Y dược TP. Hồ Chí Minh) với hệ số *kappa* được đánh giá là khá tốt (*substantial agreement*) $\kappa = 72.54\%$.

Chúng tôi đã thực hiện các thí nghiệm trên các đặc trưng N-gram và đặc trưng Phủ định để tìm ra các thông số điều kiện để hệ thống đạt kết quả tối ưu nhất. Với đặc trưng N-gram, kết quả của chúng tôi thể hiện rằng:

- Sử dụng thông tin về sự có mặt của một n-gram trong câu cho kết quả tốt hơn so với sử dụng thông tin về số lượng một n-gram trong câu.
- Sử dụng tham số ngưỡng $min\_df = 3$ để giới hạn những n-gram nào được thêm vào tập từ vựng T, cho kết quả tốt nhất.
- Sử dụng kết hợp Uni-gram, Bi-gram, Tri-gram và 4-gram cho kết quả tốt nhất.

Với đặc trưng Phủ định, chúng tôi thí nghiệm 3 hướng hiện thực với công cụ Meta-NegEx, Gen-NegEx và kết hợp 2 công cụ này, kết quả cho thấy:

- Sử dụng công cụ Gen-NegEx cho kết quả tốt trong tất cả các cách hiện thực.
- Cách hiện thực sử dụng nhãn NEGATION thay cho các từ/cụm từ phủ định cho kết quả tốt hơn cả.

Chúng tôi sử dụng kết hợp phương pháp dựa trên học máy và phương pháp dựa trên từ vựng. Với mô hình học máy, hệ thống sử dụng giải thuật SVM với các đặc trưng cơ bản: đặc trưng N-gram, đặc trưng N-gram kết hợp MetaMap, đặc trưng Thay đổi trạng thái, đặc trưng Phủ định. Kết quả cho thấy 2 đặc trưng Thay đổi trạng thái và Phủ định giúp cải thiện kết quả phân loại, trong khi việc sử dụng N-gram kết hợp Metamap có xu hướng làm giảm độ chính xác, mặc dù sự khác biệt không lớn.

Với phương pháp dựa trên từ vựng, hệ thống sử dụng đặc trưng mở rộng SO-CAL. Qua các thử nghiệm cho thấy sự cải thiện tích cực khi kết hợp SO-CAL với các đặc trưng cơ bản. Kết quả tốt nhất chúng tôi đạt được là $F = 70.74\%$ khi kết hợp các đặc trưng N-gram, đặc trưng Thay đổi trạng thái, đặc trưng Phủ định và đặc trưng SO-CAL.

7.2 Hạn chế và hướng phát triển

Sau khi tiến hành các phân tích mở rộng (Mục 6.4), chúng tôi nhận thấy kết quả phân loại khi chỉ sử dụng đặc trưng N-gram phụ thuộc vào kích thước tập dữ liệu. Trong khi đó, đặc trưng N-gram là đặc trưng chính cơ bản. Đây có thể là yếu tố chính giúp cải thiện kết quả toàn hệ thống.

Một nhân tố khác có thể xem xét để cải thiện hệ thống là phân tích thêm lớp *tiêu cực*. So với 2 lớp *tích cực* và *trung tính*, các kết quả về độ chính xác, độ bao phủ và độ F của lớp *tiêu cực* đều thấp hơn rõ rệt. Từ đó, cần thêm các phân tích tìm hiểu để nhận dạng tốt hơn câu thuộc lớp *tiêu cực*.

Đặc trưng mở rộng SO-CAL trong luận án này được sử dụng thông qua phiên bản hiện thực của nhóm tác giả bài báo [11]. Điều này phần nào ảnh hưởng đến kết quả, vì mặc dù trong báo cáo [11] kết luận rằng SO-CAL cho kết quả tốt khi thử nghiệm với các văn bản thuộc các lĩnh vực khác nhau, nhưng các văn bản này vẫn thuộc loại văn bản thông thường (các bình luận về phim hoặc sản phẩm). Trong khi hệ thống phân tích cảm xúc trên loại văn bản báo cáo nghiên cứu y khoa, nhiều từ có ý nghĩa *tiêu cực* hoặc *trung tính* trong văn bản thông thường lại mang ý nghĩa *tích cực* trong báo cáo nghiên cứu y khoa. Vì vậy, cần thêm nhiều nghiên cứu tùy chỉnh SO-CAL sao cho phù hợp hơn với mục tiêu bài toán.

Tài liệu tham khảo

[1] Yun Niu et al. "Analysis of Polarity Information in Medical Text". In: *Proceedings of the American Medical Informatics Association Symposium* (2005), pp. 570–574.

[2] Tuấn Nguyễn. "Y học thực chứng: vài nét khái quát". In: *Hai mặt sáng tối của y học hiện đại*. 2004.

[3] Walaa Medhat, Ahmed Hassan, and Hoda Korashy. "Sentiment analysis algorithms and applications: A survey". In: *Ain Shams Engineering Journal* 5.4 (2014), pp. 1093–1113.

[4] S Chandrakala and C Sindhu. "Opinion Mining and Sentiment Classification a Survey". In: *Information and Communications Technology Academy of Tamil Nadu Journal on Soft Computing* 3.1 (2012), pp. 420–425.

[5] Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. "An Introdution to Information Retrieval". In: *Journal Information Retrieval* (2009), pp. 319–348.

[6] Nadia Felix Felipe Da Silva, Luiz Fernando Sommaggio Coletta, and Eduardo Raul Hruschka. "A Survey and Comparative Study of Tweet Sentiment Analysis via Semi-Supervised Learning". In: *ACM Computing Surveys* 49.1 (2015), pp. 1–26.

[7] Bo Pang, Lillian Lee, and Shivakumar Vaithyanathan. "Thumbs Up?: Sentiment Classification Using Machine Learning Techniques". In: *Proceedings of the 2nd Association for Computational Linguistics Conference on Empirical methods in Natural Language Processing* 10 (2002), pp. 79–86.

[8] L Zhang et al. "Combining lexicon-based and learning-based methods for Twitter sentiment analysis". In: *International Journal of Electronics, Communication and Soft Computing Science & Engineering (IJECSCSE)* 89 (2015), pp. 1–8.

[9] Lei Xia, Anna Lisa Gentile, and James Munro. "Improving Patient OpinionMining through Multi-step Classification". In: *Proceedings of the 12th International Conference on Text, Speech and Dialogue* (2009), pp. 70–76.

[10] Bruno Ohana and Brendan Tierney. "Sentiment classification of reviews using SentiWordNet". In: *9th. IT & T Conference*. 2009, p. 13.

[11] Maite Taboada et al. "Lexicon-Based Methods for Sentiment Analysis". In: *Association for Computational Linguistics* 37.2 (2011), pp. 267–307.

[12] Anastasia Giachanou and Fabio Crestani. "Like it or not: A survey of Twitter sentiment analysis methods". In: *ACM Comput Surv* 49.2 (2016), Article 28; 1–41.

[13] Jin-Cheon Na et al. "Sentiment classification of drug reviews using a rule-based linguistic approach". In: *Proceedings of the 14th International Conference on Asia-Pacific Digital Librabries* (2012), pp. 189–198.

[14] Pollyanna Gonçalves et al. "Comparing and combining sentiment analysis methods". In: *Proceedings of the 14th Association for Computing Machinery International Conference on Online social networks* (2013), pp. 27–38.

[15] Tanveer Ali et al. "Can I Hear You? Sentiment Analysis on Medical Forums.". In: *Ijcnlp* October (2013), pp. 667–673.

[16] Abeed Sarker et al. "Outcome Polarity Identification of Medical Papers". In: *Proceedings of Australasian Language Technology Association Workshop* (2011), pp. 105–144.

[17] W W Chapman et al. "Evaluation of negation phrases in narrative clinical reports.". In: *Proceedings / AMIA ... Annual Symposium. AMIA Symposium* (2001), pp. 105–9.

[18] P G Mutalik, A Deshpande, and P M Nadkarni. "Use of general-purpose negation detection to augment concept indexing of medical documents: a quantitative study using the UMLS". In: *Journal of the American Medical Informatics Association* 8.6 (2001), pp. 598–609.

[19] Peter L Elkin et al. "A controlled trial of automated classification of negation from clinical notes.". In: *BMC medical informatics and decision making* 5.1 (2005), p. 13.

[20] Qing Zeng et al. "Negation Detection using Regular Expression, Syntactic and Classification Methods". In: (2007), pp. 1–6.

[21] Farah Benamara et al. "How do Negation and Modality Impact on Opinions?". In: (2012), pp. 10–18.

[22] Wendy W Chapman et al. "Extending the NegEx lexicon for multiple languages.". In: *Studies in health technology and informatics* 192 (2013), pp. 677–81.

[23] Roberto Costumero et al. "An approach to detect negation on medical documents in Spanish". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 8609 LNAI. Springer International Publishing, 2014, pp. 366–375.

[24] Noa P Cruz Díaz and Manuel De Buenaga. "Negation and Speculation Detection in Clinical and Review Texts Detección de la Negación y la Especulación en Textos Médicos y de Opinión". In: *Procesamiento del Lenguaje Natural* (2015).

[25] Maria Skeppstedt, Carita Paradis, and Andreas Kerren. "Marker words for negation and speculation in health records and consumer reviews". In: *Proceedings of the 7th International Symposium on Semantic Mining in Biomedicine* 1650 (2016), pp. 64–69.

[26] T. Givón. "English Grammar". In: Amsterdam: John Benjamins Publishing Company, 1993.

[27] I G Councill, Ryan McDonald, and Leonid Velikovich. "What's great and what's not: learning to classify the scope of negation for improved sentiment analysis". In: *Proceedings of the ACL Workshop on Negation and Speculation in Natural Language Processing Uppsala Sweden* July (2010), pp. 51–59.

[28] Thorsten Joachims. "Text categorization with support vector machines: Learning with many relevant features". In: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* (1998), pp. 137–142.

[29] Ling Pei et al. "Using LS-SVM based motion recognition for smartphone indoor wireless positioning.". In: *Sensors (Basel, Switzerland)* 12.5 (2012), pp. 6155–75.

[30] Anthony J Viera and Joanne M Garrett. “Understanding Interobserver Agreement : The Kappa Statistic”. In: May (2005), pp. 360–363.

[31] F Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[32] Olivier Bodenreider. “The Unified Medical Language System (UMLS): integrating biomedical terminology.”. In: *Nucleic acids research* 32 (2004).

[33] Alan R Aronson and François-Michel Lang. “An overview of MetaMap: historical perspective and recent advances”. In: *Journal of the American Medical Informatics Association* (2010).

[34] Steven Bird, Ewan Klein, and Edward Loper. *Natural language processing with Python*. " O’Reilly Media, Inc.", 2009.

[35] Yun Niu, Xiaodan Zhu, and Graeme Hirst. “Using Outcome Polarity in Sentence Extraction for Medical Question-Answering”. In: *Proceedings of the American Medical Informatics Association Symposium* (2006), pp. 599–603.

[36] John Pestian et al. “Sentiment Analysis of Suicide Notes: A Shared Task”. In: *Biomedical Informatics Insights* 5 (2012), pp. 3–16.

[37] Kerstin Denecke. “Sentiment Analysis from Medical Texts”. In: *Health Web Science* (2015), pp. 75–81.

[38] Hideyuki Tanushi et al. “Negation Scope Delimitation in Clinical Text Using Three Approaches: NegEx, PyConTextNLP and SynNeg”. In: (2013).

[39] Tibor Kiss and Jan Strunk. “Unsupervised multilingual sentence boundary detection”. In: *Computational Linguistics* 32.4 (2006), pp. 485–525.

[40] Chetan Mate. “Product Aspect Ranking using Sentiment Analysis: A Survey”. In: *International Research Journal of Engineering and Technology* (2016), pp. 124–128.