BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC LẠC HỒNG

****

TRẦN NGỌC PHÚC

PHÂN LOẠI NỘI DUNG TÀI LIỆU WEB

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

Đồng Nai, 2012

BỘ GIÁO DỤC VÀ ĐÀO TẠO

TRƯỜNG ĐẠI HỌC LẠC HỒNG

****

TRẦN NGỌC PHÚC

PHÂN LOẠI NỘI DUNG TÀI LIỆU WEB

Chuyên ngành: CÔNG NGHỆ THÔNG TIN

Mã số: 60.48.02.01

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC

TS PHẠM TRẦN VŨ

Đồng Nai, 2012

LỜI CAM ĐOAN

Tôi xin cam đoan đây là công trình nghiên cứu của bản thân. Các số liệu, kết quả trình bày trong luận văn này là trung thực. Những tư liệu được sử dụng trong luận văn có nguồn gốc và trích dẫn rõ ràng, đầy đủ.

Học viên

Trần Ngọc Phúc

LỜI CẢM ƠN

Tôi xin bày tỏ lòng biết ơn sâu sắc đến TS Phạm Trần Vũ đã hướng dẫn nhiệt tình, tận tâm trong suốt quá trình tôi thực hiện luận văn này.

Tôi xin chân thành cảm ơn Quý thầy cô trong Khoa Công nghệ thông tin trường Đại học Lạc Hồng đã tạo điều kiện thuận lợi cho tôi trong suốt thời gian học tập và nghiên cứu tại trường.

Tôi cũng xin chân thành cảm ơn Quý thầy cô ngoài trường đã tận tâm dạy bảo tôi trong suốt quá trình học tập và giúp đỡ tôi trong quá trình nghiên cứu.

Xin chân thành cảm ơn những người thân trong gia đình, cùng các anh chị em, bạn bè, đồng nghiệp đã giúp đỡ, động viên tôi trong quá trình thực hiện và hoàn thành luận văn này.

Đồng Nai, ngày 10 tháng 6 năm 2012

Học viên

Trần Ngọc Phúc

MỤC LỤC

DANH MỤC HÌNH

DANH MỤC BẢNG

DANH MỤC CÁC TỪ VIẾT TẮT

Từ viết tắt	Ý nghĩa
CRFs	Conditional Random Fields
IDF	Inverse Document Frequency
KNN	K-Nearest Neighbor
LDA	Latent Drichlet Allocation
LLSF	Linear Least Square Fit
Maxent	Maximum Entropy
MM	Maximum Matching
NB	Naïve Bayes
pLSA	Probabilistic Latent Semantic Analysis
SVM	Support Vector Machine
TBL	Transformation - based Learning
TF	Term Frequency
WFST	Weighted Finit State Transducer

MỞ ĐẦU

Trong những năm gần đây, sự phát triển vượt bậc của Công nghệ thông tin đã làm tăng số lượng giao dịch thông tin trên mạng Internet một cách đáng kể đặc biệt là thư viện điện tử, tin tức điện tử, … Do đó mà số lượng văn bản xuất hiện trên mạng Internet cũng tăng với một tốc độ chóng mặt, và tốc độ thay đổi thông tin là cực kỳ nhanh chóng. Với số lượng thông tin đồ sộ như vậy, một yêu cầu lớn đặt ra là làm sao tổ chức và tìm kiếm thông tin, dữ liệu có hiệu quả nhất. Bài toán phân lớp là một trong những giải pháp hợp lý cho yêu cầu trên. Nhưng một thực tế là khối lượng thông tin quá lớn, việc phân lớp dữ liệu thủ công là điều không thể. Hướng giải quyết là một chương trình máy tính tự động phân lớp các thông tin dữ liệu trên.

Trong các loại dữ liệu thì văn bản là loại dữ liệu phổ biến mà con người thường gặp phải nhất. Mô hình biểu diễn văn bản phổ biến hiện nay là mô hình không gian vector, trong đó mỗi văn bản được biểu diễn bằng một vector của các từ khóa. Tuy nhiên bài toán khai phá dữ liệu văn bản thường gặp phải một số khó khăn như tính nhiều chiều của văn bản, tính nhập nhằng của ngôn ngữ… Đồng thời, khi xử lý các bài toán phân lớp tự động thì cũng gặp phải một số khó khăn là để xây dựng được bộ phân lớp có độ tin cậy cao đòi hỏi phải có một lượng các mẫu dữ liệu huấn luyện tức là các văn bản đã được gán nhãn chủ đề lớp tương ứng. Các dữ liệu huấn luyện này thường rất hiếm và đắt vì đòi hỏi thời gian và công sức của con người. Do vậy, cần phải có hệ thống xử lý văn bản hiệu quả và một phương pháp học không cần nhiều dữ liệu được phân loại và có khả năng tận dụng được các nguồn dữ liệu chưa phân loại rất phong phú như hiện nay. Nhận thấy đây là lĩnh vực mang tính khoa học cao, ứng dụng rất nhiều trong các bài toán thực tế ví dụ như: ứng dụng lọc nội dung văn bản, bài toán phân lớp sau tìm kiếm, … Tác giả quyết định chọn đề tài "**Phân loại nội dung tài liệu web**" là một việc làm không chỉ có ý nghĩa khoa học, mà còn mang tính thực tiễn.

Trong luận văn sẽ trình bày một số thuật toán phân lớp tiêu biểu và đưa ra hướng thực nghiệm cho hệ thống phân lớp.

Luận văn áp dụng phân tích chủ đề ẩn cụ thể là thuật toán Latent Dirichlet Allocation để xác định chủ đề phục vụ cho việc tiến hành phân lớp. Thực nghiệm cho thấy độ chính xác cao, phù hợp để áp dụng vào hệ thống phân lớp tự động.

Mục tiêu của luận văn:

- Nghiên cứu các kỹ thuật xử lý ngôn ngữ tiếng Việt
- Phân loại nội dung tài liệu trên văn bản tiếng Việt.

Đối tượng nghiên cứu

Các tài liệu văn bản tin tức dạng text chuẩn tiếng Việt, không chứa hình ảnh, âm thanh, …

Phạm vi nghiên cứu

Phân loại văn bản vào các thể loại phổ biến giống như trên các trang báo điện tử hiện nay, như trang http://vnexpress.net, http://vietnamnet.vn, các thể loại được nghiên cứu xử lý trong luận văn: đời sống, kinh doanh, khoa học, ô tô – xe máy, pháp luật, thế giới, thể thao, văn hóa, vi tính, xã hội.

Những vấn đề cần giải quyết trong phạm vi luận văn:

- Tìm hiểu tổng quan về vấn đề nghiên cứu.
- Tìm hiểu cách thức tương tác với tài liệu, văn bản tiếng Việt.
- Tìm hiểu các phương pháp học máy.
- Xây dựng bộ dữ liệu chủ quan dựa trên văn bản đã được phân loại sẵn.
- Nghiên cứu các thuật toán xử lý và so khớp văn bản.
- Xây dựng quy trình phân loại văn bản.
- Hiện thực quy trình phân loại văn bản.

Bố cục trình bày của luận văn

Chương 1: Giới thiệu tổng quan về bài toán phân lớp văn bản và đưa ra quy trình phân lớp văn bản.

Chương 2: Trình bày cụ thể hơn về quy trình phân lớp văn bản và đề cập đến các vấn đề liên quan trong quá trình thực hiện bài toán.

Chương 3: Trình bày các bước thực hiện quy trình và đưa ra kết quả chương trình thực nghiệm.

Kết luận những điểm chính, chỉ ra những điểm cần khắc phục đồng thời đặt ra hướng phát triển.

CHƯƠNG 1: TỔNG QUAN VỀ PHÂN LOẠI TÀI LIỆU

1.1 Tổng quan về phân loại tài liệu

1.1.1 Giới thiệu về bài toán phân loại

Phân lớp văn bản là một trong nhiều lĩnh vực được chú ý nhất và đã được nghiên cứu trong những năm gần đây.

Phân lớp văn bản [1] (hay Text Categorization hoặc Document Classificant) là quá trình gán các văn bản vào một hay nhiều lớp văn bản đã được xác định từ trước. Người ta có thể phân lớp các văn bản một cách thủ công, tức là đọc nội dung từng văn bản và gán nó vào một lớp nào đó. Hệ thống quản lý tập gồm nhiều văn bản cho nên các này sẽ tốn nhiều thời gian, công sức và do đó là không khả thi. Do vậy mà phải có các phương pháp phân lớp tự động. Để phân lớp tự động, người ta sử dụng các phương pháp học máy trong trí tuệ nhân tạo như Cây quyết định, Naïve Bayes, K láng giềng gần nhất, …

Một trong những ứng dụng quan trọng nhất của phân lớp văn bản tự động là ứng dụng trong các hệ thống tìm kiếm văn bản. Từ một tập con văn bản đã phân lớp sẵn, tất cả các văn bản trong miền tìm kiếm sẽ được gán chỉ số lớp tương ứng. Trong câu hỏi của mình, người dùng có thể xác định chủ đề hoặc lớp văn bản mà mình mong muốn tìm kiếm để hệ thống cung cấp đúng yêu cầu của mình.

Một ứng dụng khác của phân lớp văn bản là trong lĩnh vực hiểu văn bản. Phân lớp văn bản có thể được sử dụng để lọc các văn bản hoặc một phần văn bản chứa dữ liệu cần tìm mà không làm mất đi tính phức tạp của ngôn ngữ tự nhiên.

Trong phân lớp văn bản, sự tương ứng giữa một văn bản với một lớp hoặc thông qua việc gán giá trị đúng sai (True – văn bản thuộc lớp, hay False – văn bản không thuộc lớp) hoặc thông qua một độ phụ thuộc (đo độ phụ thuộc của văn bản vào lớp). Trong trường hợp có nhiều lớp thì phân loại đúng sai sẽ là việc xem một văn bản có thuộc vào một lớp duy nhất nào đó hay không.

1.1.2 Tổng quan

Xử lý ngôn ngữ, phân loại nội dung tài liệu văn bản trong những năm gần đây là lĩnh vực đang được quan tâm của cộng đồng khoa học trong và ngoài nước. Các công trình liên quan đến vấn đề xử lý ngôn ngữ tự nhiên và phân loại dữ liệu đã được công bố như:

Ngoài nước:

- Đề tài "*Active Learning for Text Classification"* [19] tạm dịch "Hoạt động huấn luyện để phân loại văn bản" của tác giả Rong Hu, đang làm việc tại School of Computing, Dublin Institute of Technology.

Đề tài thực hiện đưa các thông tin vào học máy dùng các thuật toán gom cụm để tạo ra bộ dữ liệu mẫu. Đề tài tập trung vào việc tối ưu cho việc học máy tích cực.

- Bài báo *"Text Categorization with Support Vector Machines: Learning with Many Relevant Features"* [21] tạm dịch "Phân loại văn bản dùng Support Vector Manchines: Huấn luyện với nhiều tính năng liên quan" của tác giả Thorsten Joachims, trường Đại học Dortmund, Đức.

Bài báo trình bày về việc sử dụng và cải tiến kỹ thuật Support Vector Machines (SVM) cho việc học máy có hiệu quả trong việc phân loại văn bản..

- Bài báo *"Text Categorization"* [17] của tác giả Fabrizio Sebastiani, trường Đại học Padova, Ý.

Bài báo trình bày 3 giai đoạn trong 1 hệ thống phân loại văn bản: lập chỉ mục tài liệu văn bản dùng LSI, học tập phân loại văn bản dùng SVM và Boosting, và đánh giá phân loại văn bản.

- Bài báo *"Text Categorization Based on Regulazired Linear Classification Methods"* [22] tạm dịch "Phân loại văn bản dựa trên phương pháp phân loại tuyến tính chính quy" của nhóm tác giả Tong Zhang và Franks J.

Oles, Mathematical Sciences Department, IBM T.J. Watson Research Center, NewYork.

Bài báo trình bày phương pháp phân loại văn bản tuyến tính dựa vào các kỹ thuật Linear Least Squares Fit, Logistic Regression, SVM.

❖ Hầu hết các đề tài trên đều tập trung xử lý cho phần học máy là chính. Mặt khác, các đề tài dành cho xử lý ngôn ngữ tiếng nước ngoài, cụ thể là tiếng Anh. Để áp dụng cho các tài liệu văn bản bằng tiếng Việt thì không có được độ chính xác như mong muốn.

Trong nước có những công trình như:

- Bài báo *"Social-aware Document Similarity Compution for Recommender System"* [23] của tác giả Tran Vu Pham, Le Nguyen Thach, Faculty of Computer Science and Engineering, Ho Chi Minh City University of Technology, Ho Chi Minh City, Vietnam.

Bài báo nói về tính toán sự tương đồng trong văn bản dựa vào kỹ thuật tính toán sự tương đồng qua 3 khía cạnh của văn bản: Content, Tag, User. Tác giả nghiên cứu áp dụng kỹ thuật này để tính toán sự tương đồng của văn bản so với dữ liệu mẫu đã được học máy trước đó.

- Bài báo *"Dynamic Profile Representation and Matching in Distributed Science Networks"* [24] tạm dịch Biểu diễn và so sánh động hồ sơ cá nhân trong các mạng khoa học của tác giả Phạm Trần Vũ, Trường Đại học Bách Khoa – Đại học Quốc gia TP.HCM, đăng trên Journal of Science and Technology Development, Vol. 14, No. K2, 2011.

Bài báo có đề cập tới phương pháp so trùng các hồ sơ dựa trên các phân tích về mặt ngữ nghĩa (LSA). Các phương pháp này không cần sử dụng ontology, nhưng vẫn có khả năng thực hiện các so sánh liên quan đến ngữ nghĩa, dựa vào các phương pháp thống kê.

- Đề tài *"Phân lớp tài liệu Web độc lập ngôn ngữ"* [6] của Nguyễn Thị Thùy Linh, ngành Công nghệ thông tin, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội năm 2006.

Đề tài này nghiên cứu, đề xuất một phương pháp phân loại nội dung Web độc lập ngôn ngữ. Phương pháp cho phép tích hợp thêm các ngôn ngữ mới vào bộ phân lớp vài giải quyết vấn đề bùng nổ đặc trưng thông qua hướng tiếp cận kỹ thuật học máy Entropy cực đại để xây dựng mô hình phân lớp và sử dụng chiến lược tối ưu hóa hàm nhiều biến. Đề tài này tập trung vào việc học máy.

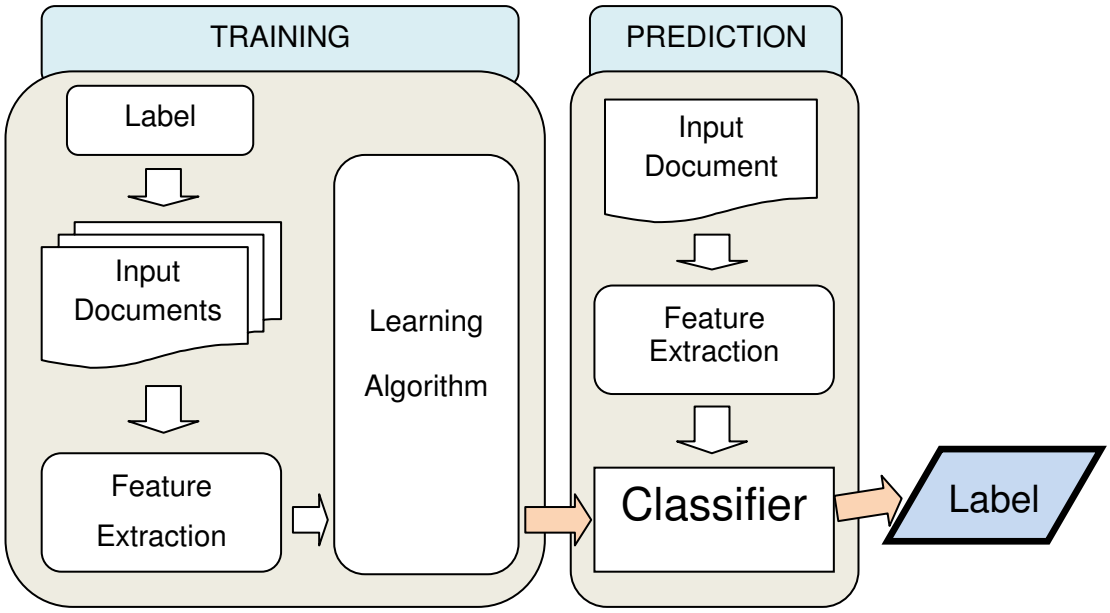
- Đề tài "*Phát triển thuật toán gom cụm văn bản HTML và ứng dụng"* [10] của tác giả Nguyễn Thế Quang.

Mục tiêu đề tài là nghiên cứu các khái niệm toán học nền tảng về mô hình không gian vectơ, mô hình Latent Semantic Indexing (LSI) được sử dụng để lập chỉ mục, quản lý và truy xuất trên các tập văn bản lớn và thuật toán gom cụm.

❖ Các đề tài trên đều có những ưu điểm nhất định của nó, tuy nhiên phạm vi xử lý văn bản của nó quá rộng, hầu như không xác định cụ thể cho một loại văn bản nào. Do đó, kết quả cho ra độ chính xác không được đồng nhất và khó để đánh giá.

1.2 Quy trình phân loại văn bản

Qua tìm hiểu nghiên cứu, tác giả rút ra quy trình phân loại văn bản chung cho hầu hết mọi phương pháp phân loại.



Hình 1.1 Quy trình phân loại văn bản

Để tiến hành phân loại văn bản nói chung, chúng ta sẽ thực hiện các bước như sau:

Bước 1: Xây dựng bộ dữ liệu chủ quan dựa vào tài liệu văn bản đã được phân loại sẵn. Tiến hành học cho bộ dữ liệu, xử lý và thu thập được dữ liệu của quá trình học là các đặc trưng riêng biệt cho từng chủ đề.

Bước 2: Dữ liệu cần phân loại được xử lý, rút ra đặc trưng kết hợp với đặc trưng được học trước đó để phân loại và rút ra kết quả.

Các phần xử lý của từng quá trình sẽ được trình bày chi tiết trong các chương tiếp theo.

CHƯƠNG 2: MỘT SỐ KỸ THUẬT TRONG PHÂN LOẠI VĂN BẢN

2.1 Xử lý văn bản

2.1.1 Đặc điểm của từ trong tiếng việt

Tiếng Việt là ngôn ngữ đơn lập [3][11]. Đặc điểm này bao quát tiếng Việt cả về mặt ngữ âm, ngữ nghĩa, ngữ pháp. Khác với các ngôn ngữ châu Âu, mỗi từ là một nhóm các ký tự có nghĩa được cách nhau bởi một khoảng trắng. Còn tiếng Việt, và các ngôn ngữ đơn lập khác, thì khoảng trắng không phải là căn cứ để nhận diện từ.

Tiếng:

➢ Trong tiếng Việt trước hết cần chú ý đến đơn vị xưa nay vẫn quan gọi là tiếng. Về mặt ngữ nghĩa, ngữ âm, ngữ pháp, đều có giá trị quan trọng.

➢ Sử dụng tiếng để tạo từ có hai trường hợp:

✓ Trường hợp một tiếng: đây là trường hợp một tiếng được dùng làm một từ, gọi là từ đơn. Tuy nhiên không phải tiếng nào cũng tạo thành một từ.

✓ Trường hợp hai tiếng trở lên: đây là trường hợp hai hay nhiều tiếng kết hợp với nhau, cả khối kết hợp với nhau gắn bó tương đối chặt chẽ, mới có tư cách ngữ pháp là một từ. Đây là trường hợp từ ghép hay từ phức.

Từ:

Có rất nhiều quan niệm về từ trong tiếng Việt, từ nhiều quan niệm về từ tiếng Việt khác nhau đó chúng ta có thể thấy đặc trưng cơ bản của "từ" là sự hoàn chỉnh về mặt nội dung, từ là đơn vị nhỏ nhất để đặt câu.

Người ta dùng "từ" kết hợp thành câu chứ không phải dùng "tiếng", do đó quá trình tách câu thành các "từ" cho kết quả tốt hơn là tách câu bằng “tiếng”.

2.1.2 Tách từ

Có nhiều phương pháp tách từ [3][11] trong tiếng Việt. Luận văn sẽ trình bày các phương pháp tách từ phổ biến.

2.1.2.1 Phương pháp Maximum Matching: Forward / Backward

Phương pháp so khớp tối đa (MM-Maximum Matching) hay còn gọi là LRMM - Left Right Maximum Matching. Ở phương pháp này, chúng ta sẽ duyệt một ngữ hoặc câu từ trái sang phải và chọn từ có nhiều âm tiết nhất có mặt trong từ điển và cứ thực hiện lặp lại như vậy cho đến hết câu.

Dạng đơn giản của phương pháp dùng để giải quyết nhập nhằng từ đơn. Giả sử chúng ta có một chuỗi ký tự C_1, C_2, …, C_n. Chúng ta sẽ áp dụng phương pháp từ đầu chuỗi. Đầu tiên kiểm tra xem C_1 có phải là từ hay không, sau đó kiểm tra xem C_1C_2 có phải là từ hay không. Tiếp tục thực hiện như thế cho đến khi tìm được từ dài nhất .

Dạng phức tạp: Quy tắc của dạng này là phân đoạn từ. Thông thường người ta chọn phân đoạn ba từ có chiều dài tối đa. Thuật toán bắt đầu từ dạng đơn giản, cụ thể là nếu phát hiện ra những cách tách từ gây nhập nhằng, như ở ví dụ trên, giả sử C_1 là từ và C_1C_2 cũng là một từ, khi đó chúng ta kiểm tra ký tự kế tiếp trong chuỗi C_1, C_2, ..., C_n để tìm tất cả các đoạn ba từ có bắt đầu với C_1 hoặc C_1C_2.

Ví dụ : Giả sử chúng ta có được các đoạn sau:

- C_1 C_2 C_3 C_4
- C_1C_2 C_3C_4 C_5
- C_1C_2 C_3C_4 C_5C_6

Khi đó chuỗi dài nhất sẽ là chuỗi thứ ba. Do đó từ đầu tiên của chuỗi thứ ba (C_1C_2) sẽ được chọn. Thực hiện các bước cho đến khi được chuỗi từ hoàn chỉnh.

Nhận xét :

Phương pháp này thực hiện tách từ đơn giản, nhanh và chỉ cần dựa vào từ điển để thực hiện. Tuy nhiên, khuyết điểm của phương pháp này cũng

chính là từ điển, nghĩa là độ chính xác khi thực hiện tách từ phụ thuộc hoàn toàn vào tính đủ, tính chính xác của từ điển.

2.1.2.2 Phương pháp Transformation – based Learning (TBL)

Phương pháp này tiếp cận dựa trên tập ngữ liệu đã đánh dấu. Theo cách tiếp cận này để cho máy tính có thể nhận biết ranh giới giữa các từ để có thể tách từ chính xác, chúng ta sẽ cho máy học các câu mẫu trong tập ngữ liệu đã được đánh dấu ranh giới giữa các từ đúng. Chúng ta thấy phương pháp rất đơn giản, vì chỉ cần cho máy học các tập câu mẫu và sau đó máy sẽ tự rút ra qui luật của ngôn ngữ và để từ đó sẽ áp dụng chính xác khi có những câu đúng theo luật mà máy đã rút ra. Và để tách từ được hoàn toàn chính xác trong mọi trường hợp thì đòi hỏi phải có một tập ngữ liệu tiếng Việt thật đầy đủ và phải được huấn luyện lâu để có thể rút ra các luật đầy đủ.

2.1.2.3 Mô hình tách từ bằng WFST và mạng Neural

Mô hình mạng chuyển dịch trạng thái hữu hạn có trọng số Weighted Finit State Transducer (WFST) đã được áp dụng trong tách từ từ năm 1996. Ý tưởng cơ bản là áp dụng WFST với trọng số là xác suất xuất hiện của mỗi từ trong kho ngữ liệu. Dùng WFST để duyệt qua các câu cần xét, khi đó từ có trọng số lớn nhất là từ được chọn để tách. Phương pháp này cũng đã được sử dụng trong công trình đã được công bố của tác giả Đình Điền năm 2001, tác giả đã sử dụng WFST kèm với mạng Neural để khử nhập nhằng khi tách từ, trong công trình tác giả đã xây dựng hệ thống tách từ gồm tầng WFST để tách từ và xử lý các vấn đề liên quan đến một số đặc thù riêng của ngôn ngữ tiếng Việt như từ láy, tên riêng, ... và tầng mạng Neural dùng để khử nhập nhằng về ngữ nghĩa sau khi đã tách từ (nếu có).

Chi tiết về 2 tầng này như sau:

a Tầng WFST gồm có 3 bước

Bước 1: Xây dựng từ điển trọng số: theo mô hình WFST, thao tác phân đoạn từ được xem như là một sự chuyển dịch trạng thái có xác suất.

Chúng ta miêu tả từ điển D là một đồ thị biến đổi trạng thái hữu hạn có trọng số.

Giả sử:

- H là tập các từ chính tả tiếng Việt (còn gọi là "tiếng")

  - P là từ loại của từ .

- Mỗi cung của D có thể là:

  - Từ một phần tử của H tới một hần tử của H

  - Các nhãn trong D biểu diễn một chi phí được ước lượng theo công thức: Cost = -log(f/N)

  Trong đó: f là tần số của từ, N là kích thước tập mẫu.

Bước 2: Xây dựng các khả năng phân đoạn từ: Để giảm sự bùng nổ tổ hợp khi sinh ra dãy các từ có thể từ một dãy các tiếng trong câu, tác giả đã đề xuất phương pháp kết hợp dùng thêm từ điển để hạn chế sinh ra các bùng nổ tổ hợp, cụ thể là nếu phát hiện thấy một cách phân đoạn từ nào đó không phù hợp (không có trong từ điển, không có phải là tứ láy, không phải là danh từ riêng,…) thì tác giả loại bỏ các nhánh xuất phát từ cách phân đoạn đoạn đó.

Bước 3: Lựa chọn khả năng phân đoạn từ tối ưu: Sau khi có được danh sách các cách phân đoạn từ có thể có của câu, tác giả đã chọn trường hợp phân đoạn từ có trọng số bé nhất.

b Tầng mạng Neural

Mô hình được sử dụng để khử nhập nhằng khi tách từ bằng cách kết hợp so sánh với từ điển.

Nhận xét: Mô hình này đạt được độ chính xác trên 97% theo như công bố trong công trình của tác giả, bằng việc sử dụng thêm mạng Neural kết hợp với từ điển để khử các nhập nhằng có thể có khi tách ra các được nhiều từ từ một câu và khi đó tầng mạng Neural sẽ loại bỏ đi các từ không phù hợp bằng cách kết

hợp với từ điển. Bên cạnh đó, cũng tương tự như phương pháp TBL điểm quan trọng của mô hình này cần tập ngữ liệu học đầy đủ.

2.1.2.4 Phương pháp tách tách từ tiếng Việt dựa trên thống kê từ Internet và thuật giải di truyền

Phương pháp tách tách từ tiếng Việt dựa trên thống kê từ Internet và thuật giải di truyền – IGATEC (Internet and Genetics Algorithm based Text Categorization for Documents in Vietnamese) do H. Nguyễn đề xuất năm 2005 như một hướng tiếp cận mới trong tách từ với mục đích phân loại văn bản mà không cần dùng đến một từ điển hay tập ngữ liệu học nào. Trong hướng tiếp cận này, tác giả kết hợp giữa thuật toán di truyền với dữ liệu thống kê được lấy từ Internet .

2.1.2.5 Loại bỏ từ dừng

Từ dừng (stop-words) dùng để chỉ các từ mà xuất hiện quá nhiều trong các câu văn bản của toàn tập kết quả, thường thì không giúp ích gì trong việc phân biệt nội dung của các tài liệu văn bản. Ví dụ, những từ "và", "hoặc", "cũng", "là", "mỗi", "bởi", …

2.1.3 Đặc trưng văn bản

Các phương pháp rút trích thông tin [6][11][16] cổ điển thì coi mỗi một văn bản như là tập các từ khóa và gọi tập các từ khóa này là tập các term. Một phần tử trong tập term thì đơn giản là một từ, mà ngữ nghĩa của từ này giúp tạo thành nên nội dung của văn bản. Vì vậy, tập term được sử dụng để tạo các chỉ mục và tóm lược nội dung của văn bản.

Giả sử cho một tập term của một văn bản nào đó, chúng ta có thể nhận thấy rằng không phải tất cả các từ trong tập term này đều có mức độ quan trọng như nhau trong việc mô tả nội dung văn bản. Ví dụ, bây giờ chúng ta xét một tập gồm một trăm ngàn văn bản, giả sử có một từ A nào đó xuất hiện trong một trăm ngàn văn bản này thì chúng ta có thể khẳng định rằng từ A này không quan trọng và chúng ta sẽ không quan tâm đến nó, bởi vì chắc chắn là nó sẽ không cho

chúng ta biết được về nội dung của các văn bản này. Vì vậy từ A sẽ bị loại ra khỏi tập các term, khi chúng ta xây dựng tập term cho văn bản để miêu tả nội dung ngữ nghĩa của các văn bản này. Kết quả này có được thông qua thao tác xác định trọng số cho mỗi một từ trong tập term của một văn bản .

Đặt k_i là từ thứ i trong tập term, d_j là văn bản j, và $w_{ij} >= 0$ là trọng số của từ k_i trong văn bản d_j. Giá trị của trọng số này thì rất là quan trọng trong việc miêu tả nội dung của văn bản.

Đặt t là số lượng các từ trong tập term của hệ thống. $K = \{k_1, k_2, k_3, \ldots, k_t\}$ là tập tất cả các từ trong tập term, trong đó k_i là từ thứ i trong tập term. Trọng số $w_{ij} > 0$ là trọng số của từ k_i trong văn bản d_j. Với mỗi một từ, nếu nó không xuất hiện trong văn bản thì $w_{ij} = 0$. Do đó, văn bản d_j thì được biểu diễn bằng vector d_j, trong đó vector $d_j = \{w_{j1}, w_{j2}, w_{j3}, \ldots, w_{jt}\}$.

Các đặc trưng của văn bản khi biểu diễn dưới dạng vector:

- Số nhiều không gian đặc trưng thường lớn .
- Các đặc trưng độc lập nhau.
- Các đặc trưng rời rạc: vector đặc trưng d_i có thể có nhiều thành phần mang giá trị 0 do có nhiều đặc trưng không xuất hiện trong văn bản d_i (nếu chúng ta tiếp cận theo cách sử dụng giá trị nhị phân 1, 0 để biểu diễn cho việc có xuất hiện hay không một đặc trưng nào đó trong văn bản đang được biểu diễn thành vector), tuy nhiên nếu đơn thuần cách tiếp cận sử dụng giá trị nhị phân 0, 1 này thì kết quả phân loại phần nào hạn chế là do có thể đặc trưng đó không có trong văn bản đang xét nhưng trong văn bản đang xét lại có từ khóa khác với từ đặc trưng nhưng có ngữ nghĩa giống với từ đặc trưng này, do đó một cách tiếp cận khác là không sử dụng số nhị phân 0, 1 mà sử dụng giá trị số thực để phần nào giảm bớt sự rời rạc trong vector văn bản.

2.2 Biểu diễn văn bản

Có nhiều cách biểu diễn văn bản [1], luận văn trình bày các phương pháp biểu diễn văn bản phổ biến.

2.2.1 Mô hình logic

Theo mô hình này, các từ có nghĩa trong văn bản sẽ được đánh chỉ số và nội dung văn bản được quản lý theo các chỉ số Index đó. Mỗi văn bản được đánh chỉ số theo quy tắc liệt kê các từ có nghĩa trong các văn bản với vị trí xuất hiện của nó trong văn bản. Từ có nghĩa là từ mang thông tin chính về các văn bản lưu trữ, khi nhìn vào nó, người ta có thể biết chủ đề của văn bản cần biểu diễn.

Tiến hành Index các văn bản đưa vào theo danh sách các từ khóa nói trên. Với mỗi từ khóa người ta sẽ đánh số thứ tự vị trí xuất hiện của nó và lưu lại chỉ số đó cùng với mã văn bản chứa nó. Cách biểu diễn này cũng được các máy tìm kiếm ưa dùng.

Ví dụ, có 2 văn bản với mã tương ứng là VB1, VB2:

- VB1 là: “Ngân hàng cổ phần thương mại”.
- VB2 là: “Công ty thương mại hàng hóa”

Khi đó, ta có cách biểu diễn như sau:

Bảng 2.1: Biểu diễn văn bản trong mô hình Logic

Từ mục	Mã VB_Vị trí XH
Ngân	VB1(1)
Hàng	VB1(2),VB2(5)
Cổ	VB1(3)
Phần	VB1(4)
Thương	VB1(5),VB2(3)

Mại	VB1(6)VB2(4)
Công	VB2(1)

Một số ưu điểm, nhược điểm

- **Ưu điểm**

Việc tìm kiếm trở nên nhanh và đơn giản. Thật vậy, giả sử cần tìm kiếm từ "computer". Hệ thống sẽ duyệt trên bảng Index để trỏ đến chỉ số Index tương ứng nếu từ "computer" tồn tại trên hệ thống. Việc tìm kiếm này khá nhanh và đơn giản khi trước đó ta đã sắp xếp bảng Index theo vẫn chữ cái. Phép tìm kiếm trên có độ phức tạp cấp $\theta(n\log_2 n)$, với n là số từ trong bảng Index. Tương ứng với chỉ số index trên sẽ cho ta biết các tài liệu chứa từ khóa tìm kiếm. Như vậy, việc tìm kiếm liên quan đến k từ thì các phép toán cần thực hiện là $k*n*\log_2 n$ (n là số từ trong bảng Index).

- **Nhược điểm**

Đòi hỏi người sử dụng phải có kinh nghiệm và chuyên môn trong lĩnh vực tìm kiếm vì câu hỏi đưa vào dưới dạng Logic nên kết quả cũng có giá trị Logic (Boolean). Một số tài liệu sẽ được trả lại khi thỏa mãn mọi điều kiện đưa vào. Như vậy muốn tìm được tài liệu theo nội dung thì phải biết đích xác về tài liệu.

Việc Index các tài liệu rất phức tạp và làm tốn nhiều thời gian, đồng thời cũng tốn không gian để lưu trữ các bảng Index.

Các tài liệu tìm được không được sắp xếp theo độ chính xác của chúng. Các bảng Index không linh hoạt vì khi các từ vựng thay đổi (thêm, sửa, xóa, …) dẫn tới chỉ số Index cũng phải thay đổi theo.

2.2.2 Mô hình phân tích cú pháp

Trong mô hình này, mỗi văn bản đều phải được phân tích cú pháp và trả lại thông tin chi tiết về chủ đề của văn bản đó. Sau đó, người ta tiến hành Index các chủ đề của từng văn bản. Cách Index trên chủ đề cũng giống như Index trên văn bản nhưng chỉ Index trên các từ xuất hiện trong chủ đề.

Các văn bản được quản lý thông qua các chủ đề này để có thể tìm kiếm được khi có yêu cầu, câu hỏi tìm kiếm sẽ dựa trên các chủ đề trên.

Một số ưu điểm, nhược điểm của phương pháp này

- **Ưu điểm**

Tìm kiếm theo phương pháp này khá hiệu quả và đơn giản, do tìm kiếm nhanh và chính xác.

Đối với những ngôn ngữ đơn giản về mặt ngữ pháp thì việc phân tích trên có thể đạt được mức độ chính xác cao và chấp nhận được.

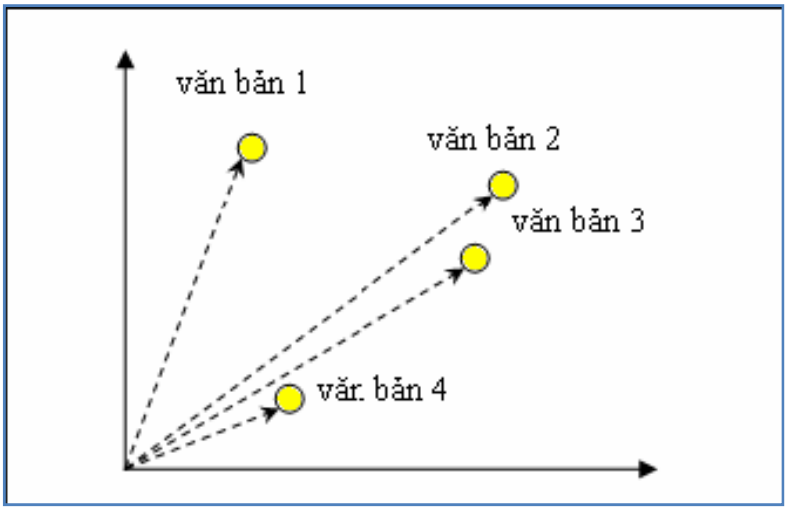
- **Nhược điểm**

Chất lượng của hệ thống theo phương pháp này hoàn toàn phụ thược vào chất lượng của hệ thống phân tích cú pháp và đóan nhận nội dung tài liệu. Trên thực tế, việc xây dựng hệ thống này rất phức tạp, phụ thuộc vào đặc điểm của từng ngôn ngữ và đa số chưa đạt đến độ chính xác cao.

2.2.3 Mô hình không gian vector

Cách biểu diễn văn bản thông dụng nhất là thông qua vector biểu diễn theo mô hình không gian vector (Vector Space Model). Đây là một cách biểu diễn tương đối đơn giản và hiệu quả.

Theo mô hình này, mỗi văn bản được biểu diễn thành một vector. Mỗi thành phần của vector là một từ khóa riêng biệt trong tập văn bản gốc và được gán một giá trị là hàm f chỉ mật độ xuất hiện của từ khóa trong văn bản.



Hình 2.1: Biểu diễn vector văn bản trong không gian 2 chiều

Giả sử ta có một văn bản và nó được biểu diễn bởi vector $V(v_1,v_2, \ldots, v_n)$. Trong đó, v_i là số lần xuất hiện của từ khóa thứ i trong văn bản. Ta xét 2 văn bản sau:

VB1: Life is not only life

VB2: To life is to fight

Sau khi qua bước tiền xử lý văn bản, ta biểu diễn chúng như sau:

Bảng 2.2: Biểu diễn văn bản mô hình Vector

Từ	Vector_VB1	Vector_VB2
Life	2	1
Fight	0	1
Only	1	0

Trong các cơ sở dữ liệu văn bản, mô hình vector là mô hình biểu diễn văn bản được sử dụng phổ biến nhất hiện nay. Mối quan hệ giữa các trang văn bản được thực hiện thông qua việc tính toán trên các vector biểu diễn vì vậy được thi hành khá hiệu quả. Đặc biệt, nhiều công trình nghiên cứu về mối quan hệ "tương

tự nhau" giữa các trang web (một trong những quan hệ điển hình nhất giữa các trang web) dựa trên mô hình biểu diễn vector .

2.2.3.1 Mô hình boolean

Một mô hình biểu diễn vector với hàm f cho ra giá trị rời rạc với duy nhất hai giá trị đúng và sai (true và false, hoặc 0 và 1) gọi là mô hình Boolean. Hàm f tương ứng với từ khóa t_i sẽ cho ra giá trị đúng nếu và chỉ nếu từ khóa t_i xuất hiện trong văn bản đó.

Mô hình Boolean được xác định như sau:

Giả sử có một cơ sở dữ liệu gồm ***m*** văn bản, $\boldsymbol{D = \{d_1, d_2, \ldots d_m\}}$. Mỗi văn bản được biểu diễn dưới dạng một vector gồm ***n*** từ khóa $\boldsymbol{T = \{t_1, t_2, \ldots t_n\}}$. Gọi $\boldsymbol{W = \{w_{ij}\}}$ là ma trận trọng số, trong đó $\boldsymbol{w_{ij}}$ là giá trị trọng số của từ khóa $\boldsymbol{t_i}$ trong văn bản $\boldsymbol{d_j}$.

$$W_{ij} = \begin{cases} 1 \text{ nếu } t_i \text{ có mặt trong } d_j \\ 0 \text{ nếu ngược lại} \end{cases}$$

Trở lại với 2 văn bản trên, áp dụng mô hình Boolean ta có biểu diễn sau:

Bảng 2.3: Biểu diễn văn bản mô hình Boolean

Từ	Vector_VB1	Vector_VB2
Life	1	1
Fight	0	1
Only	1	0

2.2.3.2 Mô hình tần suất

Trong mô hình tần suất, ma trận $W = \{w_{ij}\}$ được xác định dựa trên tần số xuất hiện của từ khóa t_i trong văn bản d_j hoặc tần số xuất hiện của từ khóa t_i trong toàn bộ cơ sở dữ liệu. Sau đây là một số phương pháp phổ biến:

a. Phương pháp dựa trên tần số từ khóa (TF – Term Frequency)

Các giá trị w_{ij} được tính dựa trên tần số (hay số lần) xuất hiện của từ khóa trong văn bản. Gọi f_{ij} là số lần xuất hiện của từ khóa t_i trong văn bản d_j, khi đó w_{ij} được tính bởi một trong ba công thức:

$$w_{ij} = f_{ij}$$

$$w_{ij} = 1 + \log(f_{ij})$$

$$w_{ij} = \sqrt{f_{ij}}$$

Trong phương pháp này, trọng số w_{ij} tỷ lệ thuận với số lần xuất hiện của từ khóa t_i trong văn bản d_j. Khi số lần xuất hiện từ khóa t_i trong văn bản d_j càng lớn thì điều đó có nghĩa là văn bản d_j càng phụ thuộc vào từ khóa t_i, hay nói cách khác từ khóa t_i mang nhiều thông tin trong văn bản d_j.

Ví dụ, khi văn bản xuất hiện nhiều từ khóa máy tính, điều đó có nghĩa là văn bản đang xét chủ yếu liên quan đến lĩnh vực tin học.

Nhưng suy luận trên không phải lúc nào cũng đúng. Một ví dụ điển hình là từ "và" xuất hiện nhiều trong hầu hết các văn bản, nhưng trên thực tế từ này lại không mang nhiều ý nghĩa như tần suất xuất hiện của nó. Hoặc có những từ không xuất hiện trong văn bản này nhưng lại xuất hiện trong văn bản khác, khi đó ta sẽ không tính được giá trị của $\log(f_{ij})$. Một phương pháp khác ra đời khắc phục được nhược điểm của phương pháp TF, đó là phương pháp IDF.

b. Phương pháp dựa trên nghịch đảo tần số văn bản (IDF – Inverse Document Frequency)

Trong phương pháp này, giá trị w_{ij} được tính theo công thức sau:

$$W_{ij} = \begin{cases} \log\frac{m}{hi} = \log(m) - \log(h_i) \text{ nếu } t_i \text{ xuất hiện trong } d_j \\ 0 \text{ nếu ngược lại} \end{cases}$$

Trong đó m là số lượng văn bản và h_i là số lượng văn bản mà từ khóa t_i xuất hiện.

Trọng số w_{ij} trong công thức này được tính dựa trên độ quan trọng của từ khóa ti trong văn bản d_j. Nếu t_i xuất hiện trong càng ít văn bản, điều đó có nghĩa là khi nó xuất hiện trong d_j thì trọng số của nó đối với văn bản d_j càng lớn hay nó là điểm quan trọng để phân biệt văn bản d_j với các văn bản khác và hàm lượng thông tin trong nó càng lớn.

c. Phương pháp TF × IDF

Phương pháp này là tổng hợp của hai phương pháp TF và IDF, giá trị của ma trận trọng số được tính như sau:

$$W_{ij} = \begin{cases} [1+\log(f_{ij})] \log(\frac{m}{hi}) \text{ nếu } f_{ij} \geq 1 \\ 0 \text{ nếu ngược lại} \end{cases}$$

Đây là phương pháp kết hợp được ưu điểm của cả hai phương pháp trên. Trọng số w_{ij} được tính bằng tần số xuất hiện của từ khóa t_i trong văn bản d_j và độ hiếm của từ khóa t_i trong toàn bộ cơ sở dữ liệu.

Một số ưu, nhược điểm của phương pháp biểu diễn này

- **Ưu điểm**

Các tài liệu có thể được sắp xếp theo mức độ liên quan đến nội dung yêu cầu.

Tiến hành lưu trữ và tìm kiếm đơn giản hơn phương pháp Logic.

- **Nhược điểm**

Việc xử lý sẽ chậm khi hệ thống các từ vựng là lớn do phải tính toán trên toàn bộ các vector của tài liệu.

Khi biểu diễn các vector với các hệ số là số tự nhiên sẽ làm tăng mức độ chính xác của việc tìm kiếm nhưng làm tốc độ tính toán giảm đi rất nhiều do các phép nhân vector phải tiến hành trên các số tự nhiên hoặc số thực, hơn nữa việc lưu trữ các vector sẽ tốn kém và phức tạp.

Hệ thống không linh hoạt khi lưu trữ các từ khóa. Chỉ cần một thay đổi rất nhỏ trong bảng từ vựng sẽ kéo theo hoặc là vector hóa lại toàn bộ các tài liệu lưu trữ, hoặc là sẽ bỏ qua các từ có nghĩa bổ sung trong các tài liệu được mã hóa trước đó.

Một nhược điểm nữa, chiều của mỗi Vector theo cách biểu diễn này là rất lớn, bởi vì chiều của nó được xác định bằng số lượng các từ khác nhau trong tập hợp văn bản. Ví dụ số lượng các từ có thể có từ 103 đến 105 trong tập hợp các văn bản nhỏ, còn trong tập hợp các văn bản lớn thì số lượng sẽ nhiều hơn, đặc biệt trong môi trường Web.

2.3 Độ tương đồng

2.3.1 Khái niệm độ tương đồng

Trong toán học, một độ đo là một hàm số cho tương ứng với một "chiều dài", một "thể tích" hoặc một "xác suất" với một phần nào đó của một tập hợp cho sẵn. Nó là một khái niệm quan trọng trong giải tích và trong lý thuyết xác suất.

Ví dụ, độ đo đếm được định nghĩa bởi $\mu(S)$ = số phần tử của S

Rất khó để đo sự giống nhau, sự tương đồng. Sự tương đồng là một đại lượng (con số) phản ánh cường độ của mối quan hệ giữa hai đối tượng hoặc hai đặc trưng. Đại lượng này thường ở trong phạm vi từ -1 đến 1 hoặc 0 đến 1. Như vậy, một độ đo tương đồng có thể coi là một loại Scoring Function (hàm tính điểm).

Ví dụ, trong mô hình không gian vector, ta sử dụng độ đo Cosine để tính độ tương đồng giữa hai văn bản, mỗi văn bản được biểu diễn bởi một vector.

2.3.2 Độ tương đồng

Phát biểu bài toán độ tính tương đồng như sau: Xét 2 văn bản d_i và d_j. Mục tiêu của bài toán là tìm ra một giá trị của hàm $S(d_i,d_j)$ với $S \in (0,1)$. Hàm $S(d_i,d_j)$ được gọi là độ đo sự tương đồng giữa 2 văn bản d_i và d_j. Giá trị càng cao thì sự giống nhau về nghĩa của hai văn bản càng nhiều.

Ví dụ: Xét hai câu sau: "Tôi là nam" và "Tôi là nữ", bằng trực giác có thể thấy rằng hai câu trên có sự tương đồng khá cao.

Độ tương đồng ngữ nghĩa là một giá trị tin cậy phản ánh mối quan hệ ngữ nghĩa giữa hai câu. Trên thực tế, khó có thể lấy một giá trị có chính xác cao bởi vì ngữ nghĩa chỉ được hiểu đầy đủ trong một ngữ cảnh cụ thể.

2.3.3 Các phương pháp tính độ tương đồng

Bài toán độ tương đồng ngữ nghĩa được sử dụng phổ biến trong lĩnh vực xử lý ngôn ngữ tự nhiên và có nhiều kết quả khả quan. Một số phương pháp tính độ tương đồng [3][4][11] được sử dụng để tính độ đo này như sau :

- Phương pháp sử dụng thống kê: độ đo Cosine, độ đo khoảng cách Euclide, Manhattan, …

- Phương pháp sử dụng các tập dữ liệu chuẩn về ngôn ngữ để tìm ra mối quan hệ giữa các từ: Wordnet, Brown Corpus, Penn TreeBank…

Các phương pháp tính độ tương đồng sử dụng kho ngữ liệu Wordnet được đánh giá cho ra kết quả cao. Tuy nhiên, kho ngữ liệu Wordnet chỉ hỗ trợ ngôn ngữ tiếng Anh, việc xây dựng kho ngữ liệu này cho các ngôn ngữ khác đòi hỏi sự tốn kém về mặt chi phí, nhân lực và thời gian. Nhiều phương pháp được đề xuất để thay thế Wordnet cho các ngôn ngữ khác, trong đó việc sử dụng phân tích chủ đề ẩn [13] hay sử dụng mạng ngữ nghĩa Wikipedia để thay thế Wordnet được xem như là các phương án khả thi và hiệu quả. Các phương pháp này tập

trung vào việc bổ sung các thành phần ngữ nghĩa hỗ trợ cho độ đo tương đồng Cosine.

2.3.3.1 Phương pháp tính độ tương đồng sử dụng độ đo Cosine

Trong phương pháp tính độ này, các câu sẽ được biểu diễn theo một mô hình không gian vector. Mỗi thành phần trong vector chỉ đến một từ tương ứng trong danh sách mục từ chính. Danh sách mục từ chính thu được từ quá trình tiền xử lý văn bản đầu vào, các bước tiền xử lý gồm: tách câu, tách từ, gán nhãn từ loại, loại bỏ những câu không hợp lệ (không phải là câu thực sự) và biểu diễn câu trên không gian vectơ.

Không gian vector có kích thước bằng số mục từ trong danh sách mục từ chính. Mỗi phần tử là độ quan trọng của mục từ tương ứng trong câu. Độ quan trọng của từ j được tính bằng TF như sau:

$$W_{i,j} = \frac{tf_{i,j}}{\sqrt{\sum_j tf_{i,j}^2}}$$

Trong đó, $tf_{i,j}$ là tần số xuất hiện của mục từ i trong câu j.

Với không gian biểu diễn tài liệu được chọn là không gian vector và trọng số TF, độ đo tương đồng được chọn là Cosine của góc giữa hai vector tương ứng của hai văn bản là S_i và S_k. Vector biểu diễn hai câu lần lượt có dạng:

$S_i = <w_1^i, \ldots, w_t^i>$, với w_t^i là trọng số của từ thứ t trong không gian i.

$S_i = <w_1^k, \ldots, w_t^k>$, với w_t^k là trọng số của từ thứ t trong không gian k.

Độ tương tự giữa chúng được tính theo công thức:

$$Sim(S_i,S_j) = \frac{\sum_{j=1}^{t} w_j^i w_j^k}{\sqrt{\sum_{j=1}^{t} (w_j^i)^2 * \sum_{j=1}^{t} (w_j^k)^2}}$$

Trên các vector biểu diễn cho các câu lúc này chưa xét đến các quan hệ ngữ nghĩa giữa các mục từ, do đó các từ đồng nghĩa sẽ không được phát hiện, dẫn đến kết quả xét độ tương tự có xét đến ngữ nghĩa chưa tốt.

2.3.3.2 Phương pháp tính độ tương đồng dựa vào độ đo khoảng cách Euclide

Khoảng cách Euclide là một phương pháp phổ biến để xác định mức độ tương đồng giữa các vector đặc trưng của hai văn bản.

Cho hai vector v_a và v_b, khoảng cách Euclide được định nghĩa như sau:

$$\text{euc\_dist}\left(\vec{V}_a, \vec{V}_b\right) = \sqrt{\sum_{i=1}^{n}\left(w_{ai} - w_{bi}\right)^2}$$

Vì $\frac{\text{euc\_dist}\left(\vec{V}_a, \vec{V}_b\right)}{n}$ nằm trong khoảng 0 và 1, do đó mức độ tương đồng giữa hai vector này được xác định bằng công thức như sau:

$$euc\_sim\left(\vec{v}_a, \vec{v}_b\right) = 1 - \frac{\text{euc\_dist}\left(\vec{V}_a, \vec{V}_b\right)}{n} = 1 - \frac{1}{n}\sqrt{\sum_{i=1}^{n}\left(w_{ai} - w_{bi}\right)^2}$$

2.3.3.3 Phương pháp tính độ tương đồng dựa vào độ đo khoảng cách Manhattan

Khoảng cách Manhattan là một phương pháp thứ ba dùng để xác định mức độ tương đồng giữa các vector đặc trưng của hai văn bản.

Cho hai vector v_a và v_b , khoảng cách Manhattan được định nghĩa như sau:

$$man\_\text{dist}\left(\vec{V}_a, \vec{V}_b\right) = \sum_{i=1}^{n}\left|w_{ai} - w_{bi}\right|$$

Vì $\frac{man\_\text{dist}\left(\vec{V}_a, \vec{V}_b\right)}{n}$ nằm trong khoảng 0 và 1, do đó mức độ tương đồng giữa hai vector này được xác định bằng công thức như sau:

$$man\_sim\left(\vec{v}_a, \vec{v}_b\right) = 1 - \frac{man\_\text{dist}\left(\vec{V}_a, \vec{V}_b\right)}{n} = 1 - \frac{1}{n}\sum_{i=1}^{n}\left|w_{ai} - w_{bi}\right|$$

Kết luận: Các phương pháp nêu trên cho kết quả tốt như nhau trong việc xác định mức độ tương đồng giữa các vector, nên tùy vào mục tiêu mà chọn phương pháp nào là phù hợp.

2.4 Các phương pháp phân loại văn bản

Hiện nay đã có rất nhiều công trình nghiên cứu về phân loại văn bản [11][15][16][18][19] và đã có được những kết quả đáng khích lệ, như là: Naïve Bayes, Support Vector Machine, K–Nearest Neighbor, Linear Least Squares Fit, Neural Network, Naïve Bayes, Centroid–Based… Điểm chung của các phương pháp này đều dựa vào xác suất thống kê hoặc dựa vào trọng số của các từ, cụm từ trong văn bản. Trong mỗi phương pháp đều có cách tính toán khác nhau, tuy nhiên các phương pháp này đều phải thực hiện một số bước chung, như: đầu tiên mỗi phương pháp sẽ dựa vào thông tin về sự xuất hiện của các từ trong văn bản (tần số xuất hiện trong tập văn bản, …) để biểu diễn thành dạng vector, sau đó tùy từng bài toán cụ thể mà chúng ta sẽ quyết định chọn áp dụng phương pháp nào, công thức tính toán nào cho phù hợp để phân loại tập văn bản dựa trên tập các vector đã xây dựng được ở bước trên, nhằm mục đích đạt được kết quả phân loại tốt nhất.

2.4.1 Phương pháp pháp Naïve Bayes (NB)

Naïve Bayes là phương pháp phân loại dựa vào xác suất được sử dụng rộng rãi trong lĩnh vực máy học và nhiều lĩnh vực khác như trong các công cụ tìm kiếm, các bộ lọc mail, …

Ý tưởng cơ bản của cách tiếp cận này là sử dụng xác suất có điều kiện giữa từ hoặc cụm từ và chủ đề để dự đoán xác suất chủ đề của một văn bản cần phân loại. Điểm quan trọng của phương pháp này chính là ở chỗ giả định rằng sự xuất hiện của tất cả các từ trong văn bản đều độc lập với nhau. Như thế NB không tận dụng được sự phụ thuộc của nhiều từ vào một chủ đề cụ thể. Chính giả định đó làm cho việc tính toán NB hiệu qủa và nhanh chóng hơn các phương pháp khác

với độ phức tạp theo số mũ vì nó không sử dụng cách kết hợp các từ để đưa ra phán đoán chủ đề.

Mục đích chính là làm sao tính được xác suất $Pr(C_j, d')$, xác suất để văn bản d'nằm trong lớp C_j. Theo luật Bayes, văn bản d' sẽ được gán vào lớp C_j nào có xác suất $Pr(C_j, d')$ cao nhất.

Công thức để tính $Pr(C_j, d')$ như sau:

$$H_{BAYES}(d') = \operatorname{argmax}\left(\frac{Pr(C_j) \cdot \prod_{i=1}^{|d'|} Pr(w_i \mid C_j)}{\sum_{c' \in c} Pr(c') \cdot \prod_{i=1}^{|d'|} Pr(w_i \mid C')} \right)_{c_j \in C}$$

Với :

- $TF(w_i, d')$ là số lần xuất hiện của từ w_i trong văn bản d'
- $|d'|$ là số lượng các từ trong văn bản d'
- w_i là một từ trong không gian đặc trưng F với số chiều là $|F|$
- $Pr(C_j)$ được tính dựa trên tỷ lệ phần trăm của số văn bản mỗi lớp tương ứng.

$$Pr(C_j) = \frac{\|C_j\|}{\|C\|} = \frac{\|C_j\|}{\sum_{C' \in C} \|C'\|}$$

trong tập dữ liệu huấn luyện

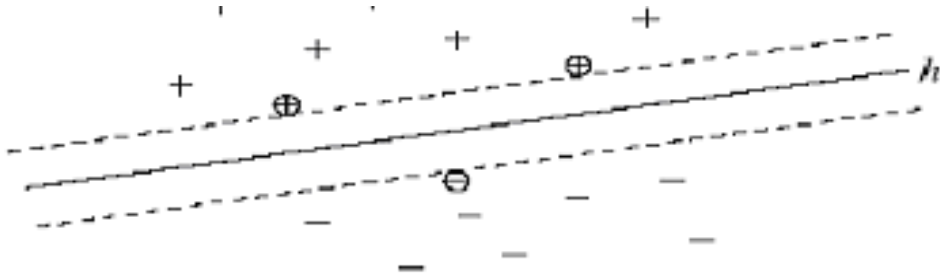
- $$Pr(w_i \mid C_j) = \frac{1 + TF(w_i, c_j)}{|F| + \sum_{w' \in |F|} TF(w', c_j)}$$

Ngoài ra còn có các phương pháp NB khác có thể kể ra như ML Naïve Bayes, MAP Naïve Bayes, Expected Naïve Bayes. Nói chung Naïve Bayes là một công cụ rất hiệu quả trong một số trường hợp. Kết quả có thể rất xấu nếu dữ

liệu huấn luyện nghèo nàn và các tham số dự đoán (như không gian đặc trưng) có chất lượng kém. Nhìn chung, đây là một thuật toán phân loại tuyến tính thích hợp trong phân loại văn bản nhiều chủ đề. NB có ưu điểm là cài đặt đơn giản, tốc độ thực hiện thuật toán nhanh, dễ dàng cập nhật dữ liệu huấn luyện mới và có tính độc lập cao với tập huấn luyện.

2.4.2 Phương pháp Support Vector Machine (SVM)

SVM là phương pháp phân loại rất hiệu quả được Vapnik giới thiệu năm 1995. Ý tưởng của phương pháp là cho trước một tập huấn luyện được biểu diễn trong không gian vector, trong đó mỗi một văn bản được xem như một điểm trong không gian này. Phương pháp này tìm ra một siêu mặt phẳng h quyết định tốt nhất có thể chia các điểm trên không gian này thành hai lớp riêng biệt tương ứng, tạm gọi là lớp + (cộng) và lớp – (trừ). Chất lượng của siêu mặt phẳng này được quyết định bởi một khoảng cách (được gọi là biên) của điểm dữ liệu gần nhất của mỗi lớp đến mặt phẳng này. Khoảng cách biên càng lớn thì càng có sự phân chia tốt các điểm ra thành hai lớp, nghĩa là sẽ đạt được kết quả phân loại tốt. Mục tiêu của thuật toán SVM là tìm được khoảng cách biên lớn nhất để tạo kết qủa phân loại tốt.



Hình 2.2: Mô hình SVM

Có thể nói SVM thực chất là một bài toán tối ưu , mục tiêu của thuật toán là tìm được một không gian H và siêu mặt phẳng quyết định h trên H sao cho sai số khi phân loại là thấp nhất, nghĩa là kết qủa phân loại sẽ cho kết quả tốt nhất.

Phương trình siêu mặt phẳng chứa vector d_i trong không gian như sau:

$$\vec{d_i} \cdot \vec{w} + b = 0$$

$$h\left(\vec{d_i}\right) = \text{sign}\left(\vec{d_i}.\vec{w}\right) = \begin{cases} +, \vec{d_i}.\vec{w} + b > 0 \\ -, \vec{d_i}.\vec{w} + b < 0 \end{cases}$$

Như thế vector $h(d_i)$ biểu diễn sự phân lớp của vector d_i vào hai lớp. Gọi Y_i mang giá trị +1 hoặc -1, khi đó Y_i = +1 văn bản tương ứng với vector d_i thuộc lớp (+) và ngược lại nó sẽ thuộc vào lớp (-). Khi này để có siêu mặt phẳng h, ta sẽ giải bài toán sau:

Tìm Min $\left\|\vec{w}\right\|$ với $\vec{w}$ và b thỏa điều kiện: $\forall i \in 1,n : y_i(\text{sign}(d_i - w + b)) \geq 1$

Chúng ta thấy rằng SVM là mặt phẳng quyết định chỉ phụ thuộc vào các vector hỗ trợ có khoảng cách đến mặt phẳng quyết định là $1/w_i$. Khi các điểm khác bị xóa đi thì thuật toán vẫn cho kết quả giống như ban đầu. Chính đặc điểm này làm cho SVM khác với các thuật toán khác như KNN, LLSF, Nnet, NB vì tất cả dữ liệu trong tập huấn luyện đều được dùng để tối ưu hóa kết quả.

2.4.3 Phương pháp K-Nearest Neighbor (KNN)

K-Nearest Neighbor là phương pháp truyền thống khá nổi tiếng theo hướng tiếp cận thống kê đã được nghiên cứu trong nhiều năm qua. KNN được đánh giá là một trong những phương pháp tốt nhất được sử dụng từ những thời kỳ đầu trong nghiên cứu về phân loại văn bản.

Ý tưởng của phương pháp này đó là khi cần phân loại một văn bản mới, thuật toán sẽ xác định khoảng cách (có thể áp dụng các công thức về khoảng cách như Euclide, Cosine, Manhattan, …) của tất cả các văn bản trong tập huấn luyện đến văn bản này để tìm ra K văn bản gần nhất, gọi là K Nearest Neighbor – K láng giềng gần nhất, sau đó dùng các khoảng cách này đánh trọng số cho tất cả các chủ đề. Khi đó, trọng số của một chủ đề chính là tổng tất cả các khoảng cách ở trên của các văn bản trong K láng giềng có cùng chủ đề, chủ đề nào không xuất hiện trong K láng giềng sẽ có trọng số bằng 0. Sau đó các chủ đề sẽ

được sắp xếp theo giá trị trọng số giảm dần và các chủ đề có trọng số cao sẽ được chọn làm chủ đề của văn bản cần phân loại.

Trọng số của chủ đề c_j đối với văn bản x được tính như sau :

$$W\left(\vec{x}, c_j\right) = \sum_{\vec{d_i} \in \{kNN\}} sim\left(\vec{x}, \vec{d_i}\right).y\left(\vec{d_i}, c_j\right) - b_j$$

Trong đó :

y (d_i, c) thuộc {0,1} , với :

- y = 0: văn bản d_i không thuộc về chủ đề c_j

- y = 1: văn bản d_i thuộc về chủ đề c_j

sim(x , d): độ giống nhau giữa văn bản cần phân loại x và văn bản d. Chúng ta có thể sử dụng độ đo Cosine để tính khoảng cách :

$$sim\left(\vec{x}, \vec{d_i}\right) = \cos\left(\vec{x}, \vec{d_i}\right) = \frac{\vec{x} \cdot \vec{d_i}}{\left\|\vec{x}\right\| \left\|\vec{d_i}\right\|}$$

- b_j là ngưỡng phân loại của chủ đề c_j được tự động học sử dụng một tập văn bản hợp lệ được chọn ra từ tập huấn luyện.

Để chọn được tham số k tốt nhất cho thao tác phân loại, thuật toán cần được chạy thử nghiệm trên nhiều giá trị K khác nhau, giá trị K càng lớn thì thuật toán càng ổn định và sai sót càng thấp.

2.4.4 Phương pháp Linear Least Square Fit (LLSF)

LLSF là một cách tiếp cận ánh xạ được phát triển bởi Yang và Chute vào năm 1992. Ban đầu LLSF được thử nghiệm trong lĩnh vực xác định từ đồng nghĩa sau đó sử dụng trong phân loại vào năm 1994. Các thử nghiệm cho thấy hiệu suất phân loại của LLSF có thể ngang bằng với phương pháp KNN kinh điển.

Ý tưởng của LLSF là sử dụng phương pháp hồi quy để học từ tập huấn luyện và các chủ đề có sẵn.

Tập huấn luyện được biểu diễn dưới dạng một cặp vector đầu vào và đầu ra như sau:

- Vector đầu vào là một văn bản bao gồm các từ và trọng số.

- Vector đầu ra gồm các chủ đề cùng với trọng số nhị phân của văn bản ứng với vector đầu vào .

Giải phương trình các cặp vector đầu vào, đầu ra chúng ta sẽ thu được ma trận đồng hiện của hệ số hồi quy của từ và chủ đề .

Phương pháp này sử dụng công thức :

$$F_{LS} = \arg_F \min \|FA - B\|^2$$

Trong đó :

- A, B là ma trận đại diện tập dữ liệu huấn luyện (các cột trong ma trận tương ứng là các vector đầu vào và đầu ra).

- F_{LS} là ma trận kết quả chỉ ra một ánh xạ từ một văn bản bất kỳ vào vector của chủ đề đã gán trọng số.

Nhờ vào việc sắp xếp trọng số của các chủ đề, chúng ta được một danh sách chủ đề có thể gán cho văn bản cần phân loại. Nhờ đặt ngưỡng lên trọng số của các chủ đề mà ta tìm được chủ đề thích hợp cho văn bản đầu vào. Hệ thống tự động học các ngưỡng tối ưu cho từng chủ đề, giống với KNN. Mặc dù LLSF và KNN khác nhau về mặt thống kê, nhưng chúng ta vẫn tìm thấy điểm chung trong cách làm của hai phương pháp này là quá trình học ngưỡng tối ưu.

2.4.5 Phương pháp Centroid – based vector

Là một phương pháp phân loại đơn giản, dễ cài đặt và tốc độ nhanh do có độ phức tạp tuyến tính O(n).

Ý tưởng của cách tiếp cận này là mỗi lớp trong dữ liệu huấn luyện sẽ được biểu diễn bằng một vector trọng tâm. Việc xác định lớp của một văn bản bất kỳ sẽ thông qua việc tìm vector trọng tâm nào gần với vector biểu diễn văn bản thứ nhất. Lớp của văn bản chính là lớp mà vector trọng tâm đại diện và khoảng cách được xác định theo độ đo Cosine.

Chúng ta có công thức tính vector trọng tâm của lớp i :

$$\vec{C_i} = \frac{1}{\|\{i\}\|} \sum_{d_j \in \{i\}} \vec{d_j}$$

Độ đo khoảng cácg giữa vector x và vector C_i :

$$\cos\left(\vec{x}, \vec{C_i}\right) = \frac{\vec{x} \cdot \vec{C_i}}{\left\|\vec{x}\right\| \cdot \left\|\vec{C_i}\right\|}$$

Trong đó :

- x là vector văn bản cần phân loại

- {i} là tập hợp các văn bản thuộc chủ đề C_i

Chủ đề của vector x là C_x thỏa mãn $\cos(x, C_x) = \arg\max(\cos(x, C_i))$.

2.4.6 Kết luận

Các thuật toán phân loại trên từ thuật toán phân loại hai lớp (SVM) đến các thuật toán phân loại đa lớp (KNN) đều có điểm chung là yêu cầu văn bản phải được biểu diễn dưới dạng vector đặc trưng. Ngoài ra các thuật toán như KNN, NB, LLSF đều phải sử dụng các ước lượng tham số và ngưỡng tối ưu khi phân loại văn bản, trong khi thuật toán SVM có thể tự xác định các tham số tối ưu này trong quá trình thực hiện thuật toán. Xét về mặt thời gian, các phương pháp có thời gian huấn luyện khác nhau, các phương pháp KNN, NB, LLSF có thời gian

huấn luyện và phân loại văn bản nhanh hơn so với các thuật toán còn lại, đồng thời dễ dàng cài đặt hơn.

Có 3 yếu tố quan trọng tác động đến kết quả phân loại văn bản:

1) Cần một tập dữ liệu huấn luyện chuẩn và đủ lớn để cho thuật toán học phân loại. Nếu chúng ta có được một tập dữ liệu chuẩn và đủ lớn thì quá trình huấn luyện sẽ tốt và khi đó chúng ta sẽ có kết qủa phân loại tốt sau khi đã được học.

2) Các phương pháp trên hầu hết đều sử dụng mô hình vector để biểu diễn văn bản, do đó phương pháp tách từ trong văn bản đóng vai trò quan trọng quá trình biểu diễn văn bản bằng vector. Yếu tố này rất quan trọng, vì có thể đối với một số ngôn ngữ như tiếng Anh thì thao tác tách từ trong văn bản đơn giản chỉ là dựa vào các khoảng trắng, tuy nhiên trong các ngôn ngữ đa âm tiết như tiếng Việt và một số ngôn ngữ khác thì sử dụng khoảng trắng khi tách từ là không chính xác, do đó phương pháp tách từ là một yếu tố quan trọng.

3) Thuật toán sử dụng để phân loại phải có thời gian xử lý hợp lý, thời gian này bao gồm: thời gian học, thời gian phân loại văn bản, ngoài ra thuật toán này phải có tính tăng cường (incremental function) nghĩa là không phân loại lại toàn bộ tập văn bản khi thêm một số văn bản mới vào tập dữ liệu mà chỉ phân loại các văn bản mới mà thôi, khi đó thuật toán phải có khả năng giảm độ nhiễu (noise) khi phân loại văn bản.

CHƯƠNG 3: CHƯƠNG TRÌNH THỬ NGHIỆM

3.1 Quy trình thực hiện

3.1.1 Xử lý dữ liệu

Dữ liệu đầu vào cho quá trình học máy hay dữ liệu đầu vào để phân loại đều là dạng văn bản đã qua công đoạn tiền xử lý. Công đoạn tiền xử lý này rất quan trọng và cần thiết, nó làm tối ưu hóa dữ liệu trong việc lưu trữ và xử lý. Các công đoạn trong quá trình tiền xử lý văn bản trong luận văn gồm: tách từ tiếng Việt, loại bỏ các từ dừng, từ tầm thường. Sau đó, rút trích đặc trưng và biểu diễn văn bản.

3.1.1.1 Tách từ tiếng Việt

Do phân loại văn bản dựa vào đặc trưng của văn bản, đặc trưng của văn bản có tốt cho quá trình phân loại không chủ yếu là dựa vào phần tách từ có chính xác không, nên độ chính xác việc tách văn bản thành các từ có nghĩa rất quan trọng.

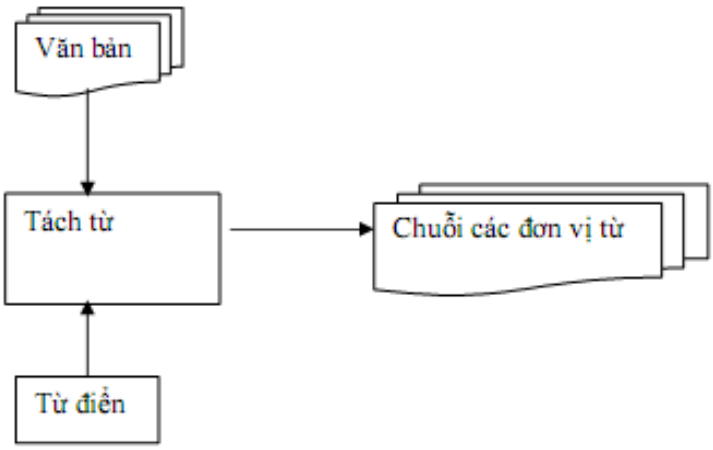
Như đã trình bày ở Chương 2, có nhiều cách tách từ thông dụng, trong luận văn, tác giả đề xuất sử dụng kỹ thuật tách từ Maximum Matching với công cụ tách từ vnTokenizer [26]. Công cụ này thuộc nhánh đề tài "Xử lý văn bản tiếng Việt", (chủ trì nhánh này là GS. Hồ Tú Bảo), nằm trong Đề tài thuộc Chương trình Khoa học Công nghệ cấp Nhà nước KC01/06-10 "Nghiên cứu phát triển một số sản phẩm thiết yếu về xử lý tiếng nói và văn bản tiếng Việt" (VLSP) chủ nhiệm đề tài là PGS. TS. Lương Chi Mai [27]. Công cụ sử dụng kết hợp từ điển và ngram, trong đó mô hình ngram được huấn luyện sử dụng VietTreebank (70.000 câu đã được tách từ) cho độ chính xác trên 97%.

Giới thiệu công cụ vnTokenizer [12]

VnTokenizer là công cụ tách từ tiếng Việt được nhóm tác giả Nguyễn Thị Minh Huyền, Vũ Xuân Lương và Lê Hồng Phương phát triển dựa trên phương pháp so khớp tối đa (Maximum Matching) với tập dữ liệu sử dụng là bảng âm tiết tiếng Việt và từ điển từ vựng tiếng Việt.

Công cụ được xây dựng bằng ngôn ngữ Java, mã nguồn mở. Có thể dễ dàng sửa đổi nâng cấp và tích hợp vào các hệ thống phân tích văn bản tiếng Việt khác.

Quy trình thực hiện tách từ theo phương pháp khớp tối đa:



Hình 3.1: Quy trình tách từ.

- Đầu vào của công cụ tách từ vnTokenizer là một câu hoặc một văn bản được lưu dưới dạng tệp.

- Đầu ra là một chuỗi các đơn vị từ được tách.

- Các đơn vị từ bao gồm các từ trong từ điển cũng như các chuỗi số, chuỗi kí từ nước ngoài, các hình vị ràng buộc (gồm các phụ tố), các dấu câu và các chuỗi kí tự hỗn tạp khác trong văn bản (ISO, 2008). Các đơn vị từ không chỉ bao gồm các từ có trong từ điển, mà cả các từ mới hoặc các từ được sinh tự do theo một quy tắc nào đó (như phương thức thêm phụ tố hay phương thức láy) hoặc các chuỗi kí hiệu không được liệt kê trong từ điển.

Công cụ sử dụng tập dữ liệu đi kèm là tập từ điển từ vựng tiếng Việt, danh sách các đơn vị từ mới bổ sung, được biểu diễn bằng ôtômat tối tiểu hữu hạn trạng thái, tệp chứa các biểu thức chính quy cho phép lọc các đơn vị từ đặc biệt (xâu dạng số, ngày tháng, …), và các tệp chứa các thống kê unigram và bigram trên kho văn bản tách từ mẫu.

Với các đơn vị từ đã có trong từ điển, khi thực hiện tách từ cũng được xử lý hiện tượng nhập nhằng bằng cách kết hợp với các thống kê unigram và bigram. Chẳng hạn trong tiếng Việt thường gặp các trường hợp nhập nhằng như:

- Xâu AB vừa có thể hiểu là 1 đơn vị từ, vừa có thể hiểu là chuỗi 2 đơn vị từ A-B.

- Xâu ABC có thể tách thành 2 đơn vị AB-C hoặc A-BC.

3.1.1.2 Loại bỏ từ dừng, từ tầm thường

Để phục vụ cho việc xử lý và lưu trữ hiệu quả hơn, với các văn bản đã qua công đoạn tách từ, ta loại bỏ các từ dừng, và từ tầm thường.

Trong quá trình nghiên cứu, tác giả nhận thấy, sau khi loại bỏ từ dừng, giữ lại các từ có từ loại là danh từ thì vẫn giữ lại ý nghĩa đầy đủ của văn bản. Do đó, quá trình này tác giả đề xuất chỉ giữ lại những từ nào là danh từ, vừa giữ nguyên ý nghĩa của văn bản, vừa giảm chi phí cho việc lưu trữ và vừa giảm chi phí cho việc tính toán, thay vì phải mất một khoản chi phí tính toán thêm các từ không có ý nghĩa.

Để thực hiện quá trình này, từ văn bản đã được tách từ, sử dụng kết hợp từ điển từ dừng và công cụ gán nhãn từ [8] loại để thu về các danh từ. Với công cụ gán nhãn từ loại, sử dụng công cụ JvnTagger [26], công cụ thuộc đề tài VLSP đã nêu ở phần trên.

Giới thiệu về công cụ gán nhãn từ loại JvnTagger [5].

a Giới thiệu

JVnTagger là công cụ gán nhãn từ loại tiếng Việt dựa trên Conditional Random Fields (Lafferty et al., 2001) và Maximum Entropy (Nigam et al., 1999). JVnTagger được xây dựng trong khuôn khổ đề tài cấp nhà nước VLSP với dữ liệu huấn luyện khoảng 10.000 câu và 20,000 câu của Viet Treebank. Thử nghiệm với phương pháp 5-fold cross

validation trên VTB-10,000 cho thấy kết quả gán nhãn với CRFs có thể đạt giá trị F1 lớn nhất là 93.45% và 10-fold cross validation với Maxent trên VTB-20,000 đạt giá trị F1 lớn nhất là 93.32%.

b Cơ sở lý thuyết

b.1 Giới thiệu Maximum Entropy

Tư tưởng chính của Maximum Entropy là *"ngoài việc thỏa mãn một số ràng buộc nào đó thì mô hình càng đồng đều càng tốt"*. Để rõ hơn về vấn đề này, ta hãy cùng xem xét bài toán phân lớp gồm có 4 lớp. Ràng buộc duy nhất mà chúng ta chỉ biết là trung bình 40% các tài liệu chứa từ "professor" thì nằm trong lớp ***faculty***. Trực quan cho thấy nếu có một tài liệu chứa từ "professor" chúng ta có thể nói có 40% khả năng tài liệu này thuộc lớp ***faculty***, và 20% khả năng cho các khả năng còn lại (thuộc một trong 3 lớp còn lại).

Mặc dù Maximum Entropy có thể được dùng để ượng lượng bất kì một phân phối xác suất nào, chúng ta xem xét khả năng maximum entropy cho việc gán nhãn dữ liệu chuỗi. Nói cách khác, ta tập trung vào việc học ra phân phối điều kiện của chuỗi nhãn tương ứng với chuỗi (xâu) đầu vào cho trước.

Các ràng buộc và đặc trưng

Trong Maximum Entropy, người ta dùng dữ liệu huấn luyện để xác định các ràng buộc trên phân phối điều kiện. Mỗi ràng buộc thể hiện một đặc trưng nào đó của dữ liệu huấn luyện. Mọi hàm thực trên quan sát đầu vào và nhãn đầu ra có thể được xem như là đặc trưng $f_i(o,s)$. Maximum Entropy cho phép chúng ta giới hạn các phân phối mô hình lý thuyết gần giống nhất các giá trị kì vọng cho các đặc trưng này trong dữ liệu huấn luyện D. Vì thế người ta đã mô hình hóa xác suất $P(o \mid s)$ như sau (ở đây, o là quan sát đầu vào và s là quan sát đầu ra)

$$P(o \mid s) = \frac{1}{Z(o)} \exp\left(\sum_i \lambda_i f_i(o,s) \right) \qquad (2.1)$$

Ở đây $f_i(o,s)$ là một đặc trưng, λ_i là một tham số cần phải ướng lượng và $Z(o)$ là thừa số chuẩn hóa đơn giản nhằm đảm bảo tính đúng đắn của định nghĩa xác suất (tổng xác suất trên toàn bộ không gian bằng 1) $Z(o) = \sum_c \exp \sum_c \lambda_i f_i(o,s)$.

Lưu ý, mỗi hàm đặc trưng $f_i(o,s)$ là một ánh xạ từ <ngữ cảnh, nhãn> →[0,1]. Một ví dụ về một hàm đặc trưng là f(từ hiện tại là "học_sinh", nhãn danh từ N) = 1.

Một số phương pháp huấn luyện mô hình từ dữ liệu học bao gồm: IIS (Improved Iterative Scaling), GIS, L-BFGS, …

b.2 Giới thiệu Conditional Random Fields

CRFs là mô hình trạng thái tuyến tính vô hướng (máy trạng thái hữu hạn được huấn luyện có điều kiên) và tuân theo tính chất Markov thứ nhất. CRFs đã được chứng minh rất thành công cho các bài toán gán nhãn cho chuỗi như tách từ, gán nhãn cụm từ, xác định thực thể, gán nhãn cụm danh từ, etc.

Gọi $\mathbf{o} = (\mathbf{o_1}, \mathbf{o_2}, \ldots, \mathbf{o_T})$ là một chuỗi dữ liệu quan sát cần được gán nhãn. Gọi S là tập trạng thái, mỗi trạng thái liên kết với một nhãn $l \in L$. Đặt $\mathbf{s} = (\mathbf{s_1}, \mathbf{s_2}, \ldots, \mathbf{s_T})$ là một chuỗi trạng thái nào đó, CRFs xác định xác suất điều kiện của một chuỗi trạng thái khi biết chuỗi quan sát như sau:

$$p_\theta(\mathbf{s} \mid \mathbf{o}) = \frac{1}{Z(\mathbf{o})} \exp\left[\sum_{t=1}^{T} \sum_k \lambda_k f_k(s_{t-1}, s_t, \mathbf{o}, t) \right]. \qquad (\mathbf{1})$$

Gọi $Z(\mathbf{o}) = \sum_{s'} \exp\left(\sum_{t=1}^{T} \sum_k \lambda_k f_k(s'_{t-1}, s'_t, \mathbf{o}, t) \right)$ là thừa số chuẩn hóa trên toàn bộ các chuỗi nhãn có thể. f_k xác định một hàm đặc trưng và λ_k là

trọng số liên kết với mỗi đặc trưng f_k. Mục đích của việc học máy với CRFs là ước lượng các trọng số này. Ở đây, ta có hai loại đặc trưng f_k : đặc trưng trạng thái (per-state) và đặc trưng chuyển (transition).

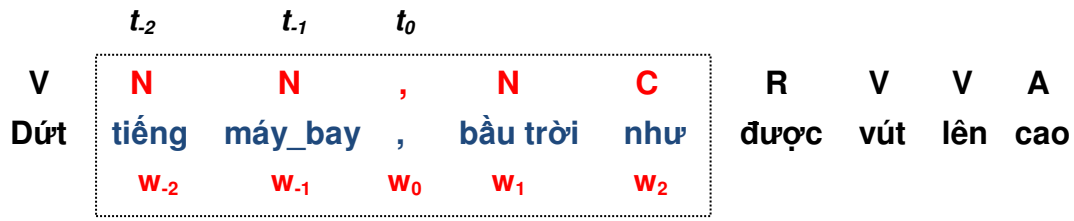
$$f_k^{(per-state)}(s_t, \mathbf{o}, t) = \delta(s_t, l) x_k(\mathbf{o}, t). \quad (2)$$

$$f_k^{(transition)}(s_{t-1}, s_t, t) = \delta(s_{t-1}, l)\delta(s_t, l). \quad (3)$$

Ở đây δ là Kronecker-δ. Mỗi đặc trưng trạng thái (2) kết hợp nhãn l của trạng thái hiện tại s_t và một vị từ ngữ cảnh - một hàm nhị phân $x_k(\mathbf{o},t)$ xác định các ngữ cảnh quan trọng của quan sát **o** tại vị trí t. Một đặc trưng chuyển (3) biểu diễn sự phụ thuộc chuỗi bằng cách kết hợp nhãn l' của trạng thái trước s_{t-1} và nhãn l của trạng thái hiện tại s_t.

Người ta thường huấn luyện CRFs bằng cách làm cực đại hóa hàm likelihood theo dữ liệu huấn luyện sử dụng các kĩ thuật tối ưu như L-BFGS. Việc lập luận (dựa trên mô hình đã học) là tìm ra chuỗi nhãn tương ứng của một chuỗi quan sát đầu vào. Đối với CRFs, người ta thường sử dụng thuật toán qui hoạch động điển hình là Viterbi để thực hiện lập luận với dữ liệu mới.

b.3 Lựa chọn đặc trưng



Hình 3.2: Cửa sổ trượt với kích cỡ size = 5 chuyển động dọc theo dữ liệu

Các mẫu ngữ cảnh cho việc lựa chọn đặc trưng với Maximum Entropy và Conditional Random Fields được cho trong bảng sau:

Bảng 3.1: Ngữ cảnh trong việc chọn đặc trưng với Maxent và CRFs

Loại	**Ngữ cảnh**	**Giải thích**
Mẫu ngữ cảnh cho cả Maxent và CRFs		
Mẫu ngữ cảnh cơ bản (loại 1)	w:-2; w:-1; w:0; w:1; w:2	w:i cho biết từ tại vị trí thứ i trong chuỗi đầu vào (nằm trong cửa sổ trượt với kích cỡ 5)
	w:0:1; w:1:2; w:-1:1	w:i:j kết hợp từ thứ i và từ thứ j trong chuỗi đầu vào
	is_all_capitalized(i) (i=0;1); is_initial_capitalized(i) (i=0;1); is_number(i) (i=-1;0;1); contain_numbers(i) (i, contain_hyphen, contain_comma, is_marks	Kiểm tra một số thuộc tính của từ thứ i trong cửa sổ hiện tại như: từ có phải là toàn chữ viết hoa hay có kí tự đầu viết hoa hay không, có chứa số, v.v...
Mẫu ngữ cảnh từ điển (loại 2)	tags_in_dictionary(i) (i=0,1)	Các từ loại có thể gán cho từ thứ i trong cửa sổ hiện tại (V, N, A, ...)
Mẫu ngữ cảnh đặc trưng tiếng Việt (loại 3)	is_full_repretative(0), is_partial_repretative(0)	Kiểm tra xem một từ có phải từ láy toàn bộ hay một phần không
Mẫu ngữ cảnh dựa vào suffix (loại 4)	prf(0), sff(0)	Âm tiết đầu tiên (ví dụ "sự" trong "sự hướng dẫn"), cuối cùng trong từ hiện tại ("hóa" trong "công nghiệp hóa")
Mẫu cho đặc trưng cạnh của CRFs		
$t_{-1}\, t_0$	Nhãn của từ trước đó và nhãn của từ hiện tại. Đặc trưng này được trích chọn trực tiếp từ dữ liệu bởi FlexCrfs	

c Kết quả gán nhãn từ loại với CRFs và Maximum Entropy

Dữ liệu VietTreebank gồm 10,000 câu được chia thành 5 folds. Đánh giá gán nhãn từ loại với CRFs và Maximum Entropy với phương pháp 5-fold-cross-validation, lấy lần lượt 4 fold để huấn luyện và thử nghiệm trên fold còn lại sau đó lấy trung bình độ đo F1 trên 5 thử nghiệm, thu được kết quả như bảng sau:

Bảng 3.2: Kết quả gán nhãn từ loại của JvnTagger

Fold	**F1-measure (CRFs)**	**F1-measure (Maxent)**
Fold1	93.01%	93.18%
Fold2	93.33%	**93.26%**
Fold3	**93.46%**	93.25%
Fold4	93.15%	93.09%
Fold5	92.90%	93.11%
Trung bình	**93.17%**	**93.18%**

3.1.2 Xây dựng bộ dữ liệu tập đặc trưng phục vụ cho phân loại

Để thực hiện phân loại, đòi hỏi phải có bộ dữ liệu chuẩn và chính xác đáp ứng được yêu cầu. Quá trình này, tác giả sử dụng mô hình phân tích chủ đề ẩn để thực hiện. Từ các tập văn bản được phân loại chủ quan theo từng thể loại riêng biệt, qua các bước tiền xử lý, thực hiện phân tích chủ đề ẩn, rút ra được các tập đặc trưng của từng thể loại.

3.1.2.1 Giới thiệu mô hình phân tích chủ đề ẩn

Vấn đề biểu diễn dữ liệu một cách hiệu quả để khai thác mối quan hệ giữa các dữ liệu ngày càng trở nên tinh vi và phức tạp hơn. Đã có rất nhiều nghiên cứu nhằm giải quyết về vấn đề này. Các mô hình chủ đề ẩn [2][7][9][13][14] là một bước tiến quan trọng trong việc mô hình quá dữ liệu văn bản. Chúng được dựa trên ý tưởng rằng mỗi tài liệu có một xác suất phân phối vào các chủ đề, và mỗi chủ đề là sự phân phối kết hợp giữa các từ. Biểu diễn các từ và tài liệu dưới dạng phân phối xác suất có lợi ích rất lớn so với mô hình không gian véc tơ thông thường.

Một ý tưởng của các mô hình chủ đề ẩn là xây dựng những tài liệu mới dựa theo phân phối xác suất. Trước hết, để tạo ra một tài liệu mới, ta cần chọn ra một phân phối những chủ đề cho tài liệu đó, điều này có nghĩa tài liệu được tạo nên từ những chủ đề khác nhau, với những phân phối khác nhau. Tiếp

đó, để sinh các từ cho tài liệu ta có thể lựa chọn ngẫu nhiên các từ dựa vào phân phối xác suất của các từ trên các chủ đề.

Một cách hoàn toàn ngược lại, cho một tập các tài liệu, ta có thể xác định một tập các chủ đề ẩn cho mỗi tài liệu và phân phối xác suất của các từ trên từng chủ đề.

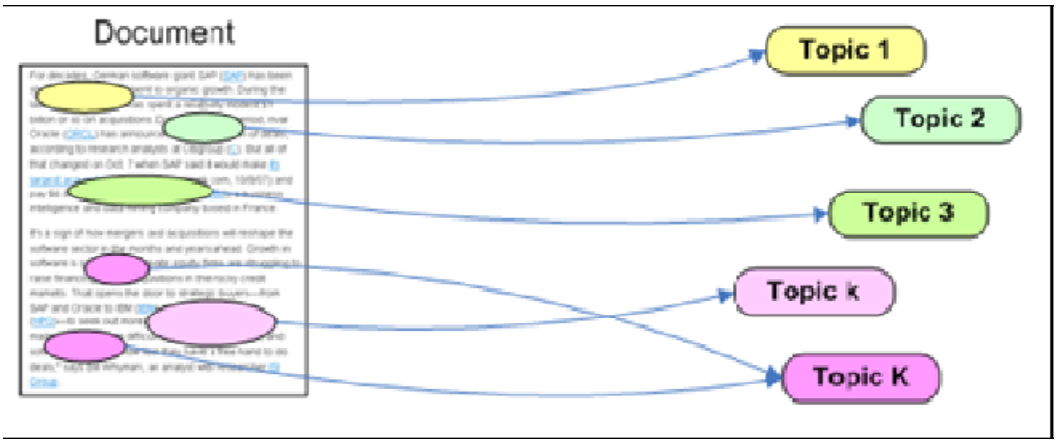
Hai ví dụ về phân tích chủ đề sử dụng mô hình ẩn là Probabilistic Latent Semantic Analysis (pLSA) and Latent Dirichlet Allocation (LDA).

PLSA là một kĩ thuật thống kê nhằm phân tích những dữ liệu xuất hiện đồng thời [20]. Nó được phát triển dựa trên Latent Semantic Analysis kết hợp với một mô hình xác suất. Tuy nhiên, theo phân tích của Blei và các cộng sự (2003), mặc dù pLSA là một bước quan trọng trong việc mô hình hóa dữ liệu văn bản, tuy nhiên nó vẫn còn chưa hoàn thiện ở chỗ chưa xây dựng được một mô hình xác suất tốt ở mức độ tài liệu. Điều đó dẫn đến vấn đề gặp phải khi phân phối xác suất cho một tài liệu nằm ngoài tập dữ liệu học, ngoài ra số lượng các tham số có thể tăng lên một cách tuyến tính khi kích thước của tập dữ liệu tăng.

LDA, là một mô hình hoàn thiện hơn so với pLSA và có thể khắc phục được những nhược điểm ở trên. Mô hình chủ đề ẩn LDA này sẽ được sử dụng trong việc xây dựng dữ liệu cho hệ thống.

3.1.2.2 Mô hình Latent Dirichlet Allocation

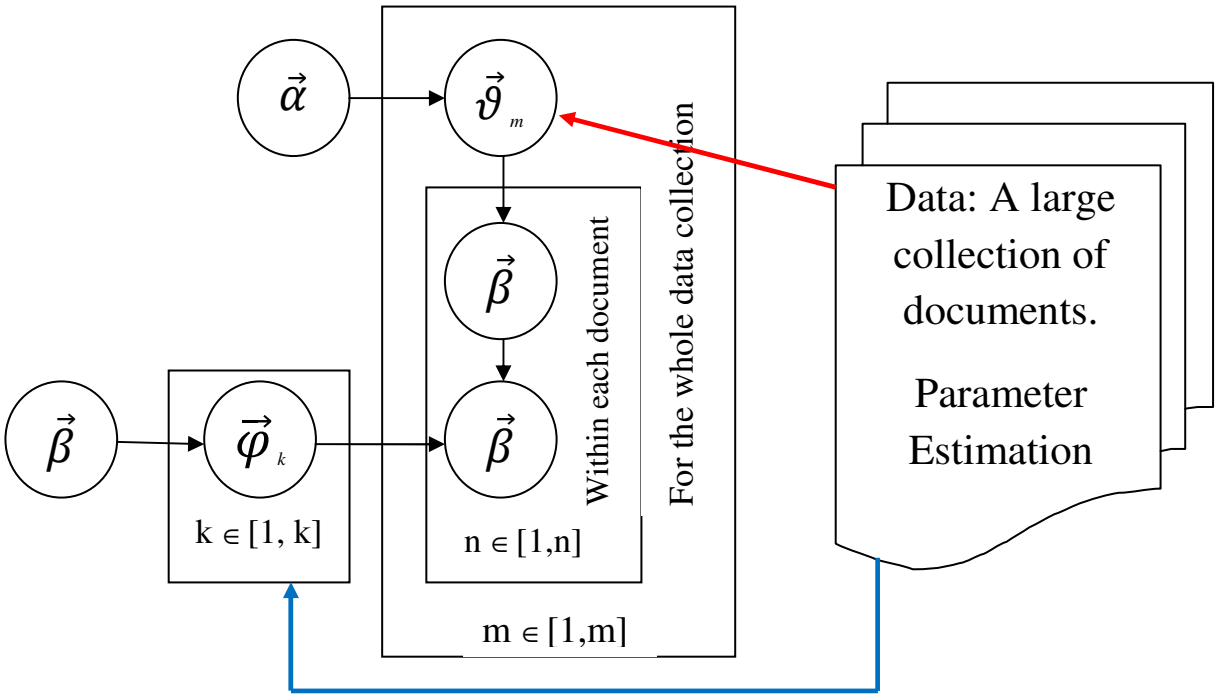
LDA [2][9][7][13][14] là một mô hình sinh xác suất cho tập dữ liệu rời rạc như text corpora. LDA dựa trên ý tưởng: mỗi tài liệu là sự trộn lẫn của nhiều chủ đề (topic). Về bản chất, LDA là một mô hình Bayesian 3 cấp (three-level hierarchical Bayes model: corpus level, document level, word level) trong đó mỗi phần của mô hình được coi như một mô hình trộn hữu hạn trên cơ sở tập các xác suất chủ đề.



Hình 3.3: Tài liệu với K chủ đề ẩn.

Ước lượng tham số cho mô hình LDA:

Cho một corpus của M tài liệu biểu diễn bởi $D=\{d_1,d_2, \ldots, d_M\}$, trong đó, mỗi tài liệu m trong corpus bao gồm Nm từ wi rút từ một tập từ vựng của các mục từ $\{t_1, \ldots, t_v\}$, V là số lượng các mục từ t trong tập từ vựng. LDA cung cấp một mô hình sinh đầy đủ chỉ ra kết quả tốt hơn các phương pháp trước. Quá trình sinh ra văn bản như sau:



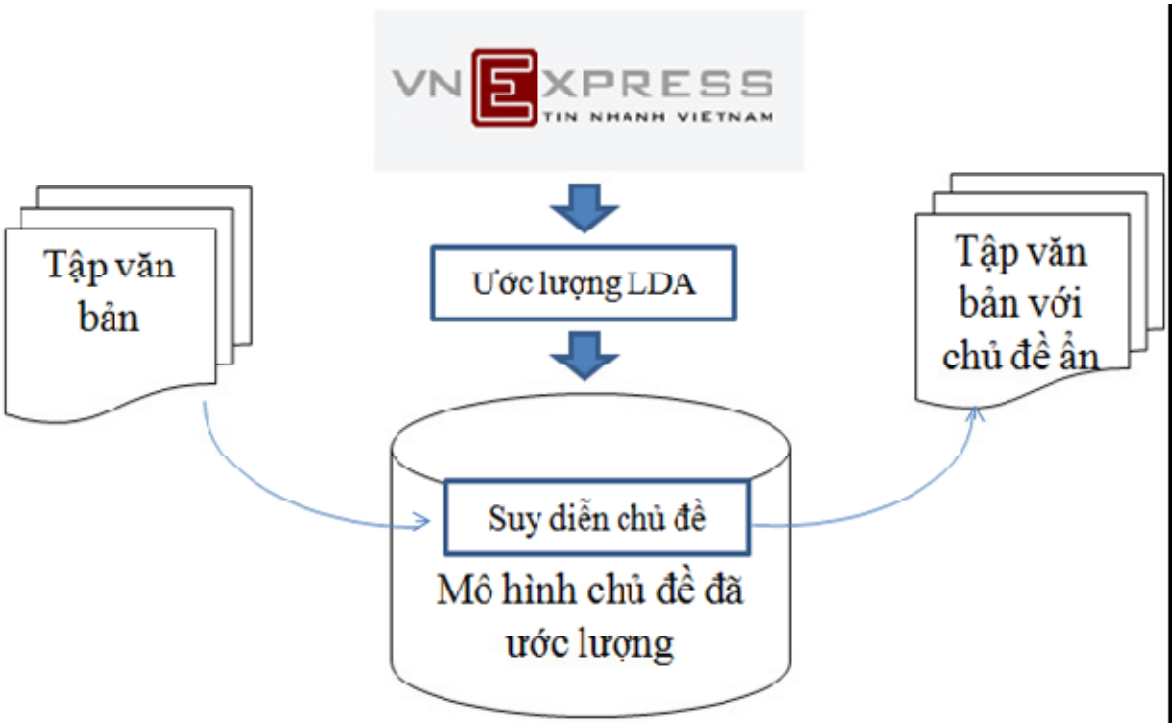
Hình 3.4: Ước lượng tham số cho tập dữ liệu.

Ước lượng tham số cho mô hình LDA bằng phương pháp cực đại hóa hàm likelihood trực tiếp và một cách chính xác có độ phức tạp thời gian rất cao và không khả thi trong thực tế. Người ta thường sử dụng các phương pháp xấp xỉ như Variational Methods và Gibbs Sampling. Gibbs Sampling được xem là một thuật toán nhanh, đơn giản, và hiệu quả để huấn luyện LDA.

Suy luận chủ đề :

Theo Nguyễn Cẩm Tú [13], với một mô hình chủ đề đã được huấn luyện tốt dựa trên tập dữ liệu toàn thể (Universial Dataset) bao phủ miền ứng dụng, ta có thể thực hiện một tiến trình quá trình suy diễn chủ đề cho các tài liệu mới tương tự như quá trình ước lượng tham số (tức là xác định được phân phối trên các chủ đề của tài liệu qua tham số theta). Tác giả cũng chỉ ra rằng sử dụng dữ liệu từ trang VnExpress.net huấn luyện được các mô hình có ưu thế hơn trong các phân tích chủ đề trên dữ liệu tin tức, trong khi các mô hình được huấn luyện bởi dữ liệu từ Wiki tốt hơn trong phân tích chủ đề các tài liệu mang tính học thuật.

Dựa trên những nghiên cứu đó, tác giả chọn mô hình được chủ đề được huấn luyện bởi tập dữ liệu toàn thể thu thập từ trang Vnexpress.net cho phân tích chủ đề. Một tiến trình phân tích chủ đề tổng quát được minh họa như sau:



Hình 3.5: Suy luận chủ đề cho các tin tức thu thập từ vnexpress.net

Công cụ JGibbsLDA của Nguyễn Cẩm Tú đã hiện thực quá trình ước lượng và suy luận chủ đề ẩn cho kết quả rất tốt, tác giả sử dụng công cụ này để xây dựng tập đặc trưng cho từng thể loại và thu được kết quả khả quan.

3.1.3 Phân loại văn bản sử dụng tần suất chủ đề

Dữ liệu cần phân loại cũng phải được qua các bước tiền xử lý như dữ liệu học (tách từ, loại bỏ từ dừng, từ phổ biến) để thu được các từ đặc trưng cho văn bản cô đọng nhất mà vẫn thể hiện được đầy đủ ý nghĩa của văn bản. Lần lượt so sánh tần suất xuất hiện của từng chủ đề trên đặc trưng của văn bản vừa thu được. Tần suất của thể loại nào xuất hiện nhiều hơn thì thuộc thể loại đó.

3.1.4 Phân loại văn bản sử dụng hệ số Cosine

Trong các phương pháp tính độ tương đồng, luận văn sử dụng phương pháp tính độ đo Cosine do phương pháp này đơn giản, dễ cài đặt mà vẫn đạt quả như các phương pháp kia.

Thể hiện các từ đặc trưng đó trong mô hình không gian vector. Với văn bản d sau khi xử lý chỉ còn n từ, ta biểu diễn d trong không gian vector: $\vec{d}$ ={(w_1:p_1),

(w_2:p_2), …, (w_n:p_n)}, trong đó, w_i là từ thứ i, p_i là tần suất của từ w_i trong văn bản đó. P_i được tính bằng công thức:

$$p_i = \frac{N(w_i)}{\sum_{i=1}^{n} w_i}$$

$N(w_i)$ là số lần xuất hiện của từ w_i trong văn bản.

$\sum_{i=1}^{n} w_i$ là tổng số từ trong văn bản.

Các tập đặc trưng sau khi suy luận LDA cũng cho ra tập đặc trưng dạng vector T dạng: $\overrightarrow{T_j}$ = {(w_{j1}:p_{j1}),(w_{j2},p_{j2}), …, (w_{jm},p_{jm})}, trong đó w_{ji} là từ thứ i của tập đặc trưng thứ j, p_{ji} là tần suất của từ thứ w_{ji}, tần suất này tự sinh ra trong quá trình suy luận của LDA.

Với vector $\vec{d}$ và từng vector đặc trưng $\overrightarrow{T_j}$ có được, dùng công thức Cosine tính độ tương đồng 2 vector và ra được kết quả, với kết này dễ dàng so sánh độ tương đồng của thể loại nào lớn hơn thì thuộc thể loại đó.

3.2 Kết quả thực nghiệm

3.2.1 Môi trường thực nghiệm

3.2.1.1 Môi trường

Bảng 3.3: Môi trường thực nghiệm

Thành phần	Chỉ số
CPU	Core 2 Duo T7300 2.0 Ghz
RAM	2Gb
HDD	160Gb
OS	Window 7 Ultimate

3.2.1.2 Công cụ

Công cụ mã nguồn mở được sử dụng trong chương trình:

Bảng 3.4: Công cụ mã nguồn mở sử dụng

Tên công cụ	Công dụng và nguồn
vnTokenizer	Công cụ tách từ tiếng Việt [26] http://vlsp.vietlp.org:8080/demo/?page=resources
JvnTagger	Công cụ gán nhãn từ loại tiếng Việt [26] http://vlsp.vietlp.org:8080/demo/?page=resources
JGibbsLDA	Công cụ phân tích chủ đề ẩn [25] http://jgibblda.sourceforge.net/

Chương trình xây dựng trên nền JDK 1.7.0 với công cụ hỗ trợ lập trình NetBean 7.1.1.

Ngoài các công cụ trên, tác giả xây dựng các module xử lý sau:

Module thu thập dữ liệu tin tức từ trang Vnexpress.net. Thu thập dữ liệu từ trang báo điện tử Vnexpress.net qua các URL, hoặc tự động thu thập tin tức theo từng thể loại. Từ các URL phân tích loại bỏ các phần không cần thiết, thu lại nội dung và coi như một tài liệu phục vụ cho quá trình học máy và kiểm thử.

Module học máy. Lưu trữ các văn bản theo từng thể loại đã được phân loại trước để phục quá trình rút trích đặc trưng.

Module phân loại văn bản. Biểu diễn các vector văn bản và tính toán độ tương đồng, so sánh và phân loại văn bản.

3.2.1.3 Dữ liệu

Dữ liệu bao gồm 11185 văn bản đã được rút trích đặc trưng phục vụ cho việc phân tích chủ đề ẩn ước lượng và suy luận.

Hơn 2000 tin tức văn bản thu được từ trang báo điện tử vnexpress.net và vietnamnet.vn theo phân loại như trong chính trong báo phục vụ cho việc học máy.

10 bộ dữ liệu (mỗi bộ khoảng 100 tin tức) thu thập từ báo điện tử vnexpress.net mới nhất thuộc 10 thể loại, phục vụ cho việc kiểm thử chương trình.

3.2.2 Kết quả thực nghiệm

Số lượng thể loại cần phân loại là 10 thể loại thông dụng về tin tức hiện nay gồm: đời sống, khoa học, kinh doanh, ô tô – xe máy, pháp luật, thể thao, thế giới, văn hóa, vi tính, xã hội.

Số chủ đề suy diễn LDA là 100 chủ đề được suy diễn, mỗi chủ đề gồm 100 từ và trọng số của nó.

3.2.2.1 Tiền xử lý văn bản

Một văn bản (văn bản chuẩn tiếng Việt, không có hình ảnh, ký hiệu toán học, …) đưa vào trước khi phân loại được trải qua bước tiền xử lý.

Ví dụ, một văn bản có nội dung như sau:

"15h ngày 11/4, sau tiếng động mạnh, người dân ở làng Quốc tế Thăng Long (Cầu Giấy, Hà Nội) hốt hoảng thấy cụ bà nằm bên vũng máu, cạnh đó là một chiếc chăn.

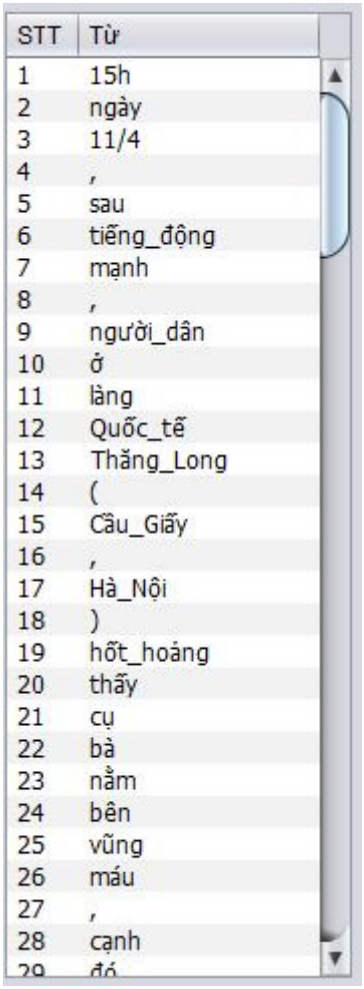
"Khi nghe thấy tiếng 'bịch', tôi vội chạy ra nhìn xung quanh thì phát hiện một người đang phát hiện bà cụ đã chết", một bảo vệ toà nhà chia sẻ.

Hiện trường vụ tai nạn.

Nạn nhân được xác định là Đỗ Thị Thư (64 tuổi), đến chơi nhà con gái ở tầng 16 (chung cư 16 tầng) được vài ngày. Theo hàng xóm, 10h sáng cùng ngày, bà Thư mang chăn lên tầng thượng của chung cư để phơi. Có thể, lúc bà cụ lên rút chăn xuống thì bị ngã, rơi xuống đất.

Khi vụ việc xảy ra, vợ chồng người con gái của nạn nhân đều đi làm."

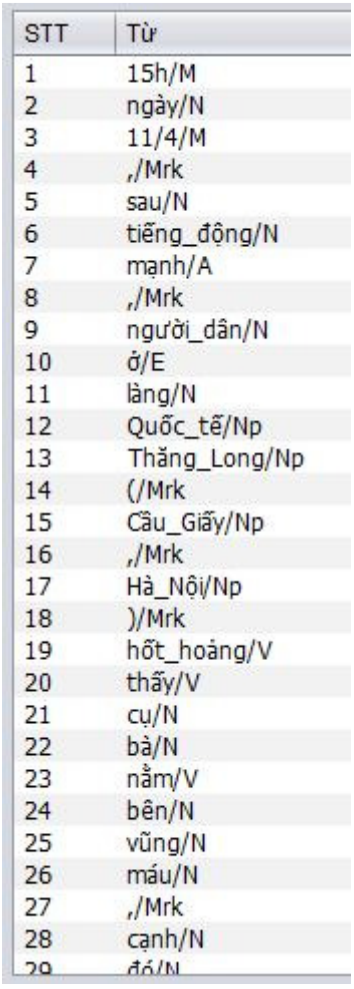
Qua công đoạn tách từ :



STT	Từ
1	15h
2	ngày
3	11/4
4	,
5	sau
6	tiếng_động
7	mạnh
8	,
9	người_dân
10	ở
11	làng
12	Quốc_tế
13	Thăng_Long
14	(
15	Cầu_Giấy
16	,
17	Hà_Nội
18	)
19	hốt_hoảng
20	thấy
21	cụ
22	bà
23	nằm
24	bên
25	vũng
26	máu
27	,
28	cạnh
29	đó

Hình 3.6: Văn bản tách ra thành các từ.

Công đoạn gán nhãn từ loại :



STT	Từ
1	15h/M
2	ngày/N
3	11/4/M
4	,/Mrk
5	sau/N
6	tiếng_động/N
7	mạnh/A
8	,/Mrk
9	người_dân/N
10	ở/E
11	làng/N
12	Quốc_tế/Np
13	Thăng_Long/Np
14	(/Mrk
15	Cầu_Giấy/Np
16	,/Mrk
17	Hà_Nội/Np
18	)/Mrk
19	hốt_hoảng/V
20	thấy/V
21	cụ/N
22	bà/N
23	nằm/V
24	bên/N
25	vũng/N
26	máu/N
27	,/Mrk
28	cạnh/N
29	đó/N

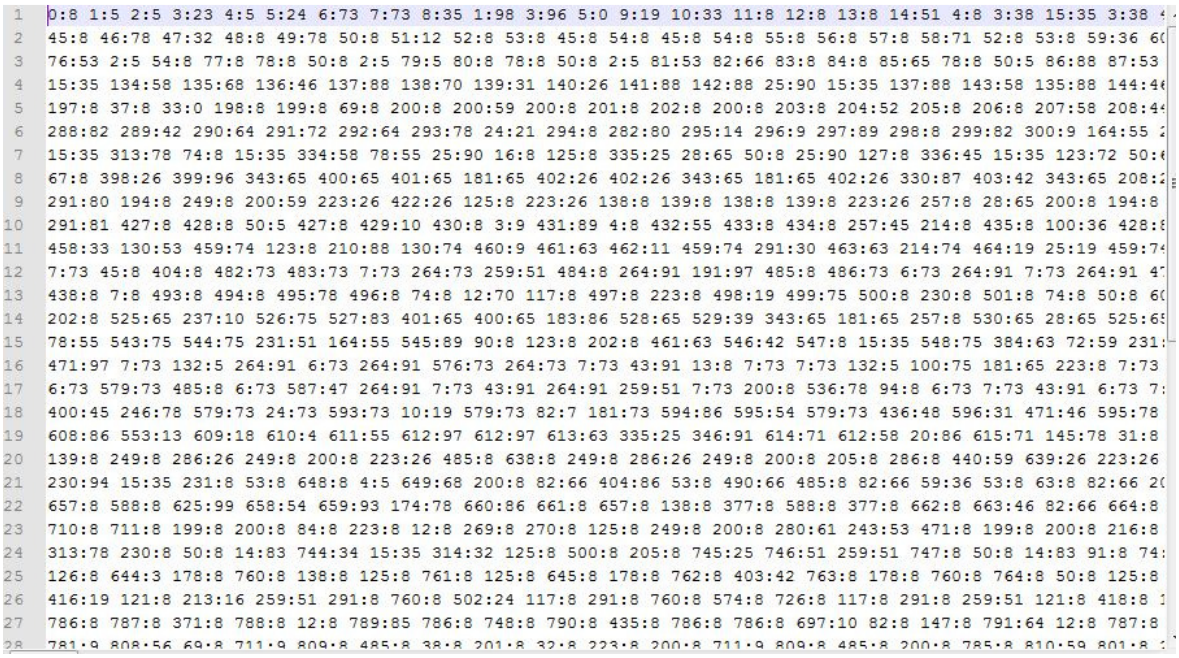
Hình 3.7: Gán nhãn từ loại cho các từ.

Loại bỏ các từ không là danh từ kết hợp với loại bỏ từ dừng thu được đặc trưng:"tiếng_động người_dân làng quốc_tế thăng_long cầu_giấy hà_nội vũng máu cạnh chiếc khi tiếng bịch xung_quanh người bà_cụ nhà hiện_trường vụ tai_nạn nạn_nhân đỗ_thị_thư tuổi nhà con_gái tầng chung_cư tầng hàng_xóm thư tầng thượng chung_cư bà_cụ vụ_việc vợ_chồng người con_gái nạn_nhân".

3.2.2.2 Tìm đặc trưng cho từng thể loại

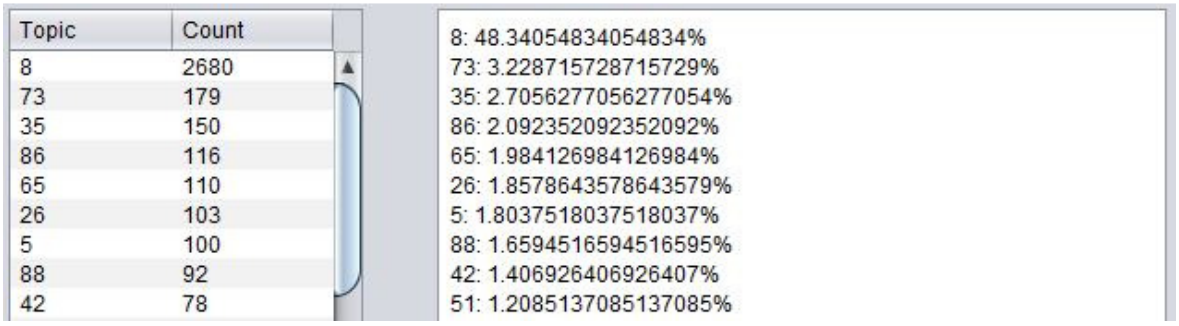
Hơn 2000 tin thức thu thập được sử dụng cho việc học máy nhằm tìm ra đặc trưng tối ưu nhất cho từng thể loại. Tiến hành suy luận chủ đề các thể loại. Kết quả thu được của một số chủ đề:

- Chủ đề kinh doanh: trong file .tassign, hiển thị dạng (word : topic):



Hình 3.8: Suy luận với thể loại kinh doanh

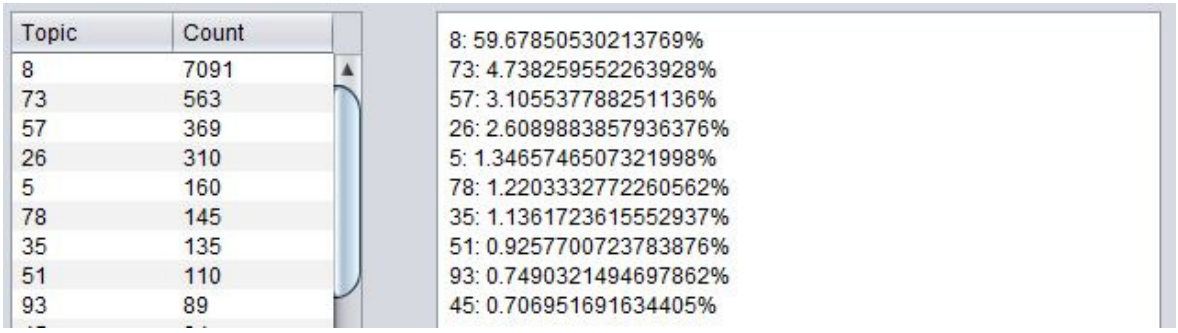
Nhận thấy xuất hiện topic 8 chiếm tỉ lệ cao > 48% trong toàn bộ thể loại kinh doanh, trong khi topic cao thứ hai là topic 73 chiếm khoảng 3%.



Topic	Count
8	2680
73	179
35	150
86	116
65	110
26	103
5	100
88	92
42	78

Hình 3.9: Topic có tỉ lệ cao thuộc thể loại kinh doanh

Tiếp tục đưa vào học máy với số lượng tin nhiều hơn (1000 tin). Số lượng từ ở chủ đề 8 tăng nhiều, ở các chủ đề còn lại tăng số lượng không đáng kể.

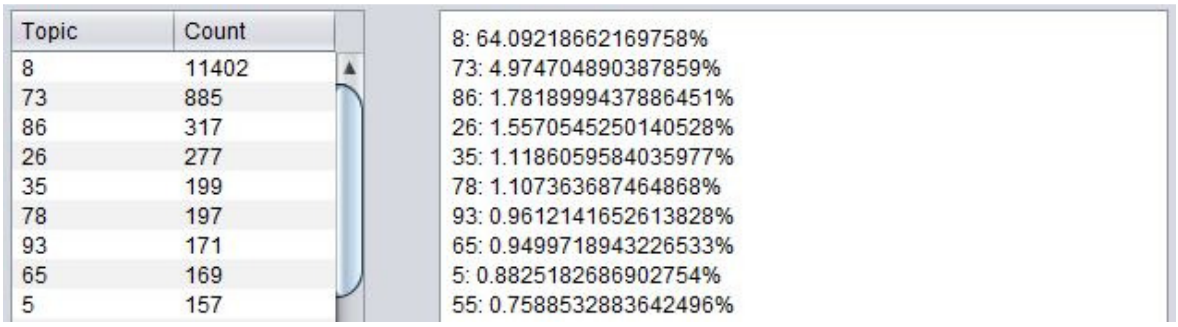


Topic	Count
8	7091
73	563
57	369
26	310
5	160
78	145
35	135
51	110
93	89

8: 59.67850530213769%
73: 4.738259552263928%
57: 3.105537788251136%
26: 2.6089883857936376%
5: 1.3465746507321998%
78: 1.2203332772260562%
35: 1.1361723615552937%
51: 0.9257700723783876%
93: 0.7490321494697862%
45: 0.706951691634405%

Hình 3.10: Topic có tỉ lệ cao thuộc thể loại kinh doanh với 1000 tin.

Tiếp tục đưa vào học máy với số lượng 1500 tin, số lượng từ ở chủ đề 8 tăng đáng kể, các chủ đề khác số lượng từ tăng không đáng kể với lượng từ ở chủ đề 8.

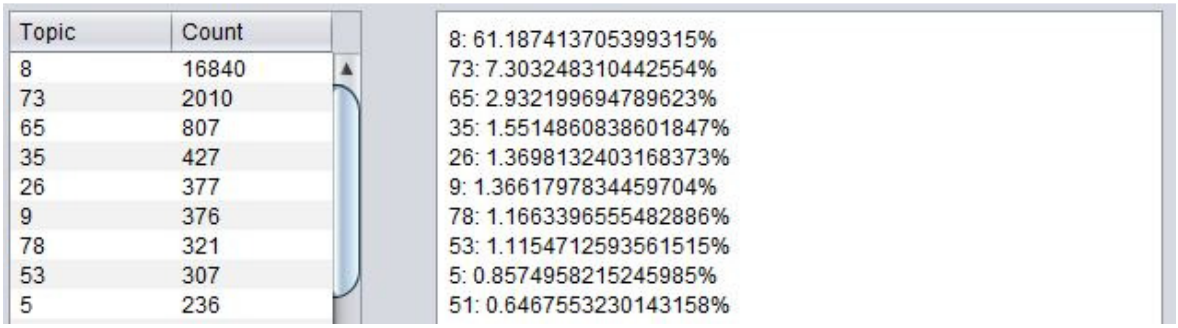


Topic	Count
8	11402
73	885
86	317
26	277
35	199
78	197
93	171
65	169
5	157

8: 64.09218662169758%
73: 4.974704890387859%
86: 1.7818999437886451%
26: 1.5570545250140528%
35: 1.1186059584035977%
78: 1.107363687464868%
93: 0.9612141652613828%
65: 0.9499718943226533%
5: 0.8825182686902754%
55: 0.7588532883642496%

Hình 3.11: Topic có tỉ lệ cao thuộc thể loại kinh doanh với 1500 tin

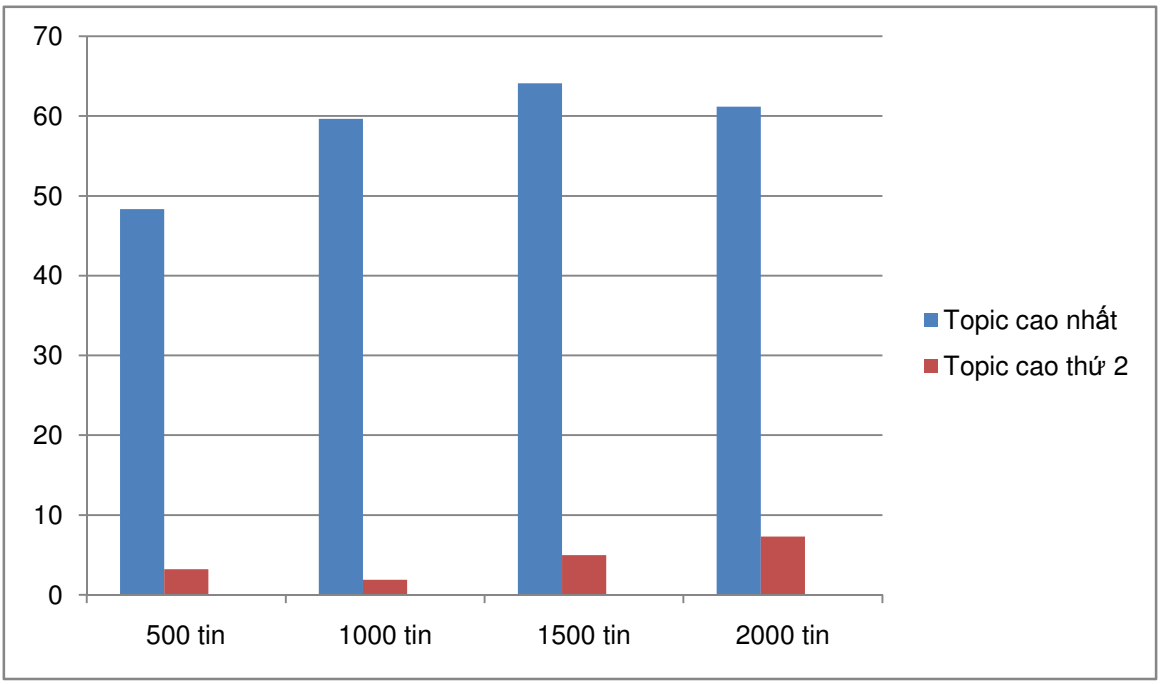
Học với 2000 tin, được kết quả :



Topic	Count
8	16840
73	2010
65	807
35	427
26	377
9	376
78	321
53	307
5	236

8: 61.187413705399315%
73: 7.303248310442554%
65: 2.932199694789623%
35: 1.5514860838601847%
26: 1.3698132403168373%
9: 1.3661797834459704%
78: 1.1663396555482886%
53: 1.1154712593561515%
5: 0.8574958215245985%
51: 0.6467553230143158%

Hình 3.12: Topic có tỉ lệ cao thuộc thể loại kinh doanh với 2000 tin.



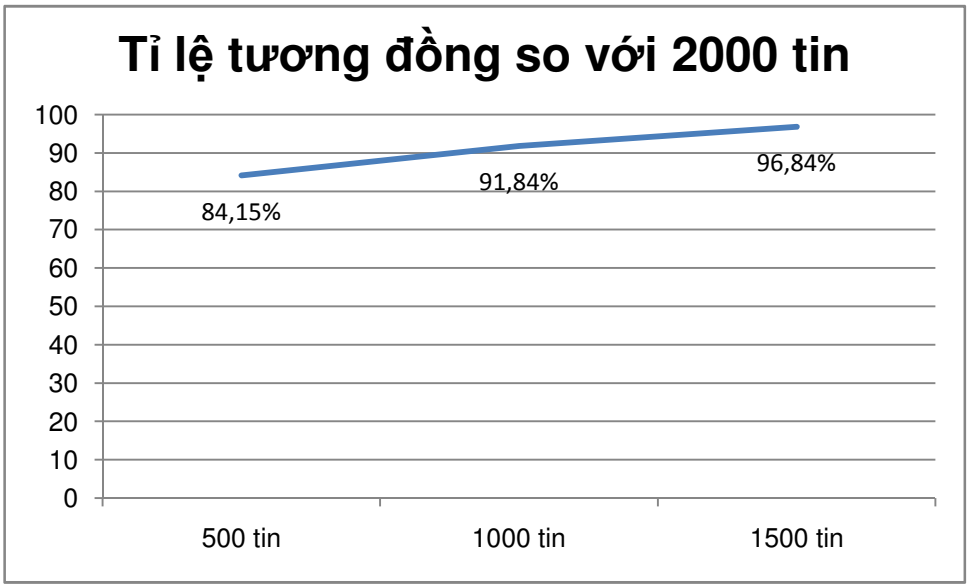
Hình 3.13: Biểu đồ tỉ lệ số lượng tin tức học máy thể loại kinh doanh.

Nhận thấy, với thể loại kinh doanh khi học lần lượt với 500, 1000, 1500, 2000 tin thì số lượng từ tăng nhiều và đều ở chủ đề 8, trong khi ở các chủ đề còn lại tăng rất ít. Như vậy có học thêm bao nhiêu thì xác suất số lượng từ tăng nhiều ở chủ đề số 8. Khẳng định các từ trong chủ đề 8 là đặc trưng của thể loại kinh doanh.

Bảng 3.5: 30/100 đặc trưng sau mỗi lần suy luận.

500 tin	1000 tin	1500 tin	2000 tin
ngân_hàng	ngân_hàng	ngân_hàng	ngân_hàng
tiền	tiền	doanh_nghiệp	doanh_nghiệp
doanh_nghiệp	doanh_nghiệp	tiền	mức
nhà_nước	công_ty	mức	tiền
chính_sách	nhà_nước	công_ty	công_ty
nợ	hàng	usd	usd
công_ty	chính_sách	thị_trường	nhà_nước
cổ_phần	nợ	nhà_nước	thị_trường
lãi_suất	mức	nợ	hàng

hàng	usd	hàng	kinh_tế
mức	thị_trường	chính_sách	lãi_suất
thẻ	lãi_suất	lãi_suất	nợ
tp_hcm	hà_nội	thuế	thuế
chính_phủ	khách_hàng	hà_nội	chính_sách
đà_nẵng	cổ_phần	lượng	nước
hoạt_động	hoạt_động	kinh_tế	hoạt_động
tín_dụng	xăng	khách_hàng	lương
đơn_vị	tín_dụng	nước	khách_hàng
tài_sản	nhân_viên	hoạt_động	tài_chính
nhân_viên	tp_hcm	lương	hà_nội
thị_trường	vietcombank	cổ_phần	chính_phủ
thuế	tiền_tệ	chính_phủ	cổ_phần
khách	khách	tín_dụng	việt_nam
sim	nước	xăng_dầu	lượng
lượng	nhà_băng	tp_hcm	tín_dụng
chi_nhánh	tài_sản	tài_chính	tập_đoàn
miếng	tài_chính	thu_nhập	tp_hcm
nước	lãnh_đạo	nhân_viên	thu_nhập
thuế_thu_nhập	thẻ	chuyên_gia	nhà_băng
công_thương	đơn_vị	vietcombank	dự_án



Hình 3.14: Biểu đồ độ tương đồng số lượng học máy của thể loại kinh doanh.

Một số đặc trưng của thể loại kinh doanh, hiển thị dưới dạng từ và trọng số của từ đó:

Bảng 3.6: 25/100 đặc trưng của thể loại kinh doanh.

ngân_hàng	0.022099737098371555
doanh_nghiệp	0.017216884188108458
mức	0.009575357317244747
tiền	0.00940983687960871
công_ty	0.00896844904591261
usd	0.0084718877330045
nhà_nước	0.008140846857732425
thị_trường	0.007147724231916203
hàng	0.006513229220978059
kinh_tế	0.006127014866493973
lãi_suất	0.005547693334767843
nợ	0.0051338922406777504
thuế	0.004830438105011683
chính_sách	0.00477526462579967
nước	0.003920075698013478
hoạt_động	0.0038097287395894536
lương	0.003671795041559423
khách_hàng	0.003589034822741404
tài_chính	0.003395927645499361
hà_nội	0.0033407541662873485
chính_phủ	0.003120060249439299
cổ_phần	0.0030648867702272866
việt_nam	0.002954539811803262
lượng	0.0028993663325912497
tín_dụng	0.0027614326345612185

Các thể loại còn lại cũng tương tự, sau mỗi lẫn học thì tỉ lệ các từ tập trung vào 1 topic tăng lên đáng kể, và dung topic đó làm đặc trưng từng thể loại. Các đặc trưng của các thể loại thu được:

Bảng 3.7: 25/100 đặc trưng của các thể loại.

Đời sống	Khoa học	Ô tô - xe máy
bác_sĩ	khu	xe
bệnh_viện	con	chiếc
con	loài	thuế
tuổi	môi_trường	động_cơ
bệnh_nhân	công_nghiệp	mẫu
bệnh	nước	ôtô
gia_đình	xây_dựng	hãng
em	việt_nam	hệ_thống
bé	hoạt_động	km
cháu	thành_phố	phiên_bản
sản_phụ	rừng	mã_lực
mẹ	cơ_sở	cá_nhân
tp_hcm	người_dân	thu_nhập
tình_trạng	trái_đất	công_suất
thai	vùng	usd
trường_hợp	động_vật	lít
y_tế	năng_lượng	lexus
vợ_chồng	biển	mức
mặt	điện	màu
khoa	rác	mm
ca	nhà_máy	thể_thao
chị	nhà_khoa_học	sang
đứa	mặt_trời	tiêu_chuẩn
phòng	nhóm	đường
chiếc	khu_vực	nhiên_liệu
Pháp luật	**Thế giới**	**Thể thao**
công_an	nước	giải
cảnh_sát	trung_quốc	trận
tuổi	máy_bay	đội
huyện	triều_tiên	cầu_thủ
tỉnh	chủ_tịch	bóng

cơ_quan	tàu	mùa
vụ	chiếc	chelsea
quận	mỹ	vô_địch
xe	tên_lửa	vòng
tiền	philippines	chung_kết
hành_vi	vụ	champions_league
điều_tra	hạt_nhân	sân
thanh_niên	chuyến	bàn
ma_túy	khu_vực	hạng
giao_thông	tin	barca
xã	quan_chức	liverpool
tội_phạm	hải_quân	châu
hình_sự	hoạt_động	bayern
phường	vệ_tinh	cup
tài_sản	iran	đội_tuyển
đường	lời	bóng_đá
lực_lượng	bình_nhưỡng	tiền_vệ
thành_phố	indonesia	chức
cán_bộ	bắc_kinh	điểm
tp_hcm	núi	ngoại_hạng
Văn hóa	**Vi tính**	**Xã hội**
phim	máy	xe
diễn_viên	sản_phẩm	ôtô
khán_giả	màn_hình	đường
ca_sĩ	máy_tính	thuế
điện_ảnh	chip	chiếc
đạo_diễn	thiết_bị	giao_thông
đêm	hãng	cá_nhân
bộ	inch	xe_máy
tuổi	điện_thoại	tp_hcm
váy	tay	hàng
ca_khúc	đầu_tiên	người_dân
tình_yêu	camera	hà_nội

cuộc_thi	tốc_độ	khu_vực
nam	apple	thành_phố
chiếc	bộ_nhớ	thu_nhập
chương_trình	ảnh	cầu
vai	dòng	phương_tiện
thí_sinh	dữ_liệu	nước
buổi	laptop	chiều
nữ	model	tai_nạn
việt_nam	phone	vụ
giám_khảo	usd	nguyên_nhân
ngôi_sao	video	thời_gian
tài_tử	samsung	lực_lượng
hoa_hậu	thị_trường	quận

3.2.2.3 Phân loại văn bản

Với 10 bộ dữ liệu thuộc 10 thể loại thu thập từ Vnexpress.net mới nhất (theo đúng thể loại do báo điện tử Vnxpress.net đưa ra) đưa vào hệ thống phân loại với 2 phương pháp: sử dụng tần suất chủ đề và sử dụng hệ số Cosine. Lấy thể loại từ trang Vnexpress.net làm chuẩn, kết quả phân loại so với chuẩn:

Bảng 3.8: Kết quả phân loại dùng tần suất chủ đề và hệ số Cosine.

Thể loại	Dùng tần suất	Hệ số Cosine
Đời sống	65	73
Khoa học	58	66
Kinh doanh	82	89
Ô tô – xe máy	89	90
Pháp luật	81	81
Thế giới	70	66
Thể thao	88	91

Văn hóa	91	89
Vi tính	80	86
Xã hội	50	40
Trung bình	**75,4%**	**77,1%**

Xét toàn hệ thống, kết quả phân loại dùng hệ số Cosine tốt hơn dùng tần suất chủ đề.

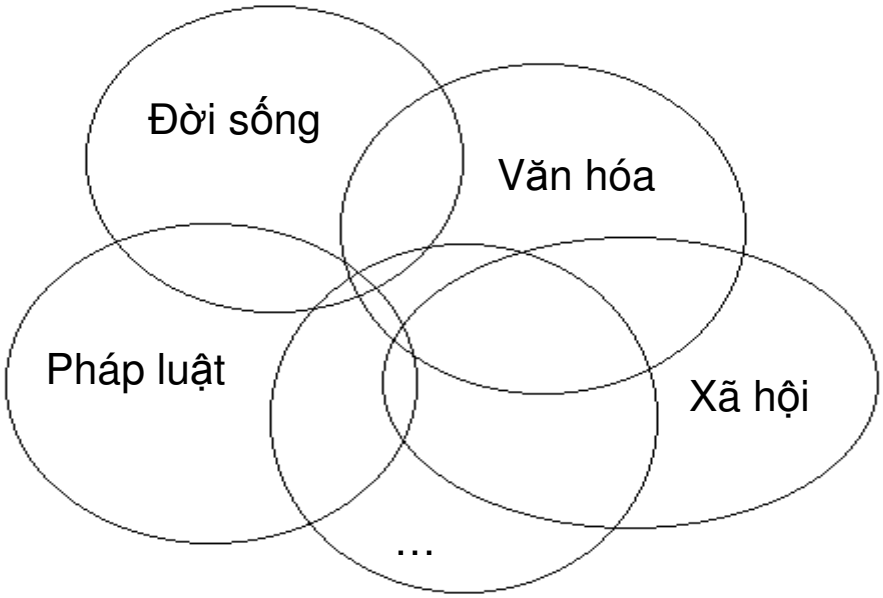
Chọn phương pháp dùng hệ số Cosine làm phương pháp chính cho hệ thống phân loại, tiếp tục xét chi tiết từng thể loại cho kết quả khác so với báo đã đưa. Thu được kết quả:

Bảng 3.9: Kết quả phân loại hệ thống so với báo.

Thể loại	**Số tài liệu sai so với báo**	**Báo đưa sai**	**Hệ thống sai**
Đời sống	27	18	9
Khoa học	34	21	13
Kinh doanh	11	6	5
Ô tô – xe máy	10	5	5
Pháp luật	19	8	11
Thế giới	34	22	12
Thể thao	9	4	5
Văn hóa	11	4	7
Vi tính	14	9	5
Xã hội	60	32	28
Tổng	229	129	100

Như vậy, khi sử dụng hệ số Cosine để tính độ tương đồng trong phân loại văn bản sẽ cho kết quả tốt hơn sử dụng tần suất chủ đề trên toàn bộ dữ liệu. Kết quả phân loại đạt độ chính xác so với dữ liệu mẫu là 77,1%, trong số 22,9% còn lại thì dữ liệu mẫu đưa sai là 12,9% và hệ thống phân loại sai là 10%, như

vậy, tỉ lệ trung bình độ chính xác của hệ thống đạt 90%. Kết quả 90% là khả quan, trong khi các tập đặc trưng là các dữ liệu có dạng liên kết với nhau, và vì thế, có nhiều khả năng một tài liệu văn bản có thể thuộc 1 thể loại, 2 hai thể hoặc nhiều thể loại, hệ thống sẽ gán tài liệu vào thể loại có hệ số cao nhất. Biểu diễn các tập đặc trưng như sau:



Hình 3.15: Các tập đặc trưng liên kết với nhau.

PHẦN KẾT LUẬN

Kết quả đạt được của luận văn

Luận văn tiến hành nghiên cứu giải quyết bài toán phân loại văn bản tiếng Việt dựa vào đặc trưng. Bài toán là nền tảng cho nhiều ứng dụng quan trọng thực tế như lọc thư spam, rút trích văn bản, hệ thông khuyến cáo người dùng, …

Phương pháp giải quyết của luận văn tập trung vào quá trình phân tích đặc trưng văn bản cho cả dữ liệu học máy và dữ liệu cần phân loại dựa vào các nghiên cứu về chủ đề ẩn, và biểu diễn văn bản dưới dạng vector. Xây dựng được bộ dữ liệu đặc trưng cho từng thể loại.

Đưa ra và sử dụng độ đo tương đồng để đánh giá phân loại.

Một mô hình phân loại cũng được đưa ra từ các bước tiền xử lý cho tới khi đưa giá trị cuối cùng đạt kết quả khả quan, cho thấy tính đúng đắn của việc lựa chọn cũng như kết hợp các phương pháp.

Tuy bước đầu đạt một số kết quả khả quan, nhưng vẫn tồn tại một số vấn đề cần khắc phục:

+ Một văn bản đầu vào cần phân loại sau quá trình cho kết thuộc vào một thể loại duy nhất.

+ Hạn chế số lượng và chất lượng của kho dữ liệu tin tức ảnh hưởng đến chất lượng phân loại của hệ thống.

+ Cần xác định giá trị chuẩn để một văn bản thuộc vào 1 hoặc nhiều thể loại, hoặc không thuộc thể loại nào.

Hướng phát triển của luận văn

Phát triển mở rộng mô hình phân loại văn bản cho các văn bản khác ngoài văn bản dạng tin tức.

Cải tiến các quy trình xử lý để tăng tốc cho hệ thống.

TÀI LIỆU THAM KHẢO

Tiếng Việt:

[1] Nguyễn Việt Cường (2006), "*Sử dụng các khái niệm mờ trong biểu diễn văn bản và áp dụng vào bài toán phân lớp văn bản*", luận văn tốt nghiệp đại học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[2] Nguyễn Song Hà (2009), "*Hệ thống tư vấn Website cho máy tìm kiếm dự trên khai phá Query log*", luận văn tốt nghiệp đại học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[3] Nguyễn Thị Thu Hằng (2007), "*Phương pháp phân cụm tài liệu Web và áp dụng vào máy tìm kiếm*", luận văn cao học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[4] Hoàng Minh Hiền (2008), "*Độ tương đồng ngữ nghĩa giữa hai câu và ứng dụng trong tóm tắt văn bản*", luận văn tốt nghiệp đại học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[5] JVnTagger-Manual, Công cụ gán nhãn từ loại tiếng Việt dựa trên Conditional Random Fields và Maximum Entropy.

[6] Nguyễn Thị Thùy Linh (2006), *"Phân lớp tài liệu Web độc lập ngôn ngữ",* luận văn tốt nghiệp đại học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[7] Uông Huy Long (2010), "*Giải pháp mở rộng thông tin ngữ cảnh phiên duyệt Web người dùng nhằm nâng cao chất lượng tư vấn trong hệ thống tư vấn tin tức*", luận văn tốt nghiệp đại học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[8] Trần Thị Oanh (2008), "*Mô hình tách từ, gán nhãn từ loại và hướng tiếp cận tích hợp cho tiếng Việt*", luận văn cao học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[9] Nguyễn Hữu Phương (2009), "*Quảng cáo trực tuyến hướng câu truy vấn với sự giúp đỡ của phân tích chủ đề ẩn và kỹ thuật tính hạng*", luận văn tốt nghiệp đại học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[10] Nguyễn Thế Quang, "*Phát triển thuật toán gom cụm văn bản HTML và ứng dụng".*

[11] Trình Quốc Sơn, "*Phân loại văn bản*", khóa luận môn Datamining.

[12] Nguyễn Phương Thái, "*Phát triển bộ công cụ hỗ trợ xây dựng kho ngữ liệu cho phân tích văn bản tiếng Việt*", luận văn cao học, trường Đại học Khoa học tự nhiên.

[13] Nguyễn Cẩm Tú (2008), "*Hidden Topic Discovery toward Classification and Clustering in Vietnamese Web Documents*", luận văn cao học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[14] Trần Mai Vũ (2009), "*Tóm tắt văn bản dựa vào trích xuất câu*", luận văn cao học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

[15] Nguyễn Thị Hải Yến (2007), "*Phân lớp bán giám sát và ứng dụng thuật toán SVM vào phân lớp trang Web*", luận văn tốt nghiệp đại học, trường Đại học Công Nghệ, Đại học Quốc gia Hà Nội.

Tiếng Anh:

[16] Arturo Montejo-Rasez (2005), "*Automatic Text Categorization of document in the High Energy Physics domain*", thesis.

[17] Fabrizio Sebastiani, "*Text Categorization*", Dipartimento di Matematica Pura e Applicata, Universit`a di Padova.

[18] Hiroya Takamura (2003), "*Clustering Approaches to Text Categorization*", Doctor's thesis, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology.

[19] Rong Hu (2011) "*Active Learning for Text Classification*", Doctoral Thesis, Dublin Institute of Technology.

[20] T. Hofmann (1999), "*Probabilistic Latent Semantic Analysis*", To appear in: Uncertainity in Articial Intelligence, UAI'99, Stockholm.

[21] Thorsten Joachims, "*Text Categorization with Support Vector Machines: Learning with Many Relevant Features*", University Dortmund.

[22] Tong Zhang and Frank J. Oles, "*Text Categorization Based on Regularized Linear Classifiacation Methods*", Mathematical Sciences Department IBM.

[23] Tran Vu Pham, Le Nguyen Thach (2011) , "*Social-Aware Document Similarity Computation for Recommender Systems*", Proceedings of the 2011 IEEE Ninth International Conference on Dependable, Autonomic and Secure Computing.

[24] Tran Vu Pham (2011), **"***Dynamic Profile Representation and Matching in Distributed Scientific Networks"*, in Journal of Science and Technology Development, Vol. 14, No. K2

Internet:

[25] Công cụ phân tích chủ đề ẩn, http://jgibblda.sourceforge.net/

[26] Hệ tách từ tiếng Việt, http://vlsp.vietlp.org:8080/demo/?page=resources

[27] Thông tin chi tiết: đề tài – dự án, http://vpct.gov.vn/News.aspx?ctl=projectdetail&ID=29