ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

Ì Ì Ì

HUỲNH TÂN TRUNG

HỆ THỐNG NHẬN DẠNG VÀ PHÂN LOẠI VĂN BẢN

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

TP.HỒ CHÍ MINH - 2007

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN

HUỲNH TÂN TRUNG

HỆ THỐNG NHẬN DẠNG VÀ PHÂN LOẠI VĂN BẢN

Chuyên ngành: KHOA HỌC MÁY TÍNH
Mã số: 60 48 01

LUẬN VĂN THẠC SĨ CÔNG NGHỆ THÔNG TIN

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. TRẦN THÁI SƠN

Thành phố Hồ Chí Minh - 2007

Lời cảm ơn

Trước tiên, tôi xin gởi lời cảm ơn đến trường Đại Học CNTT đã tạo điều kiện và tổ chức khóa học này để tôi có thể có điều kiện tiếp thu kiến thức mới và có thời gian để hoàn thành luận văn Cao Học này

Tôi cũng xin được cảm ơn TS. Trần Thái Sơn, người đã tận tình chỉ dẫn và động viên để tôi có thể hoàn thành luận văn này.

Tôi xin chân thành cảm ơn các thầy cô đã truyền đạt cho chúng tôi những kiến thức quý báu trong quá trình học Cao học và làm luận văn.

Tôi chân thành cảm ơn các bạn bè cùng lớp đã giúp đỡ và động viên tôi trong quá trình thực hiện luận văn này, đặc biệt tôi xin cảm ơn bạn Nguyễn thị Ngọc Hợp đã giúp tôi rất nhiều để hoàn thành luận văn này.

Cuối cùng, tôi kính gửi thành quả này đến gia đình và người thân của tôi, những người đã hết lòng chăm sóc, dạy bảo và động viên tôi để tôi có được kết quả ngày hôm nay.

Nhận xét Luận văn cao học

HỆ THỐNG NHẬN DẠNG VÀ PHÂN LOẠI VĂN BẢN

của học viên Huỳnh Tân Trung

Trong thời đại bùng nổ thông tin hiện nay, những hướng nghiên cứu về nhận dạng và xử lý văn bản là rất cần thiết, đặc biệt là văn bản tiếng Việt. Luận văn của học viên Huỳnh Tân Trung là một đóng góp nho nhỏ trong lĩnh vực này.

Trong quá trình làm việc với luận văn, học viên đã tỏ ra có kiến thức cơ bản khá tốt về lĩnh vực nghiên cứu của mình, nắm bắt nhanh những vấn đề mới. Điều này có thể thấy qua phần Cơ sở lý thuyết được trình bày trong luận văn. Ngoài ra, học viên đã thể hiện khả năng làm việc độc lập và bước đầu khả năng nghiên cứu. Điều này thể hiện ở phương pháp phân loại văn bản do học viên đề xuất, tuy còn phải làm nhiều việc để đi đến hoàn thiện, nhưng cũng đã phần nào đáp ứng được yêu cầu của một hệ phân loại văn bản tiếng Việt do tính đơn giản và hiệu quả ở mức tương đối khá của nó.

Với một thời gian ngắn (6 tháng), học viên đã phải nỗ lực rất cao để hoàn thành nhiệm vụ đặt ra: tiếp thu một lượng kiến thức lớn, đề xuất được một phương pháp phân loại văn bản trên cơ sở kết hợp và cải tiến các phương pháp đã biết và xây dựng phần mềm thử nghiệm phương pháp này (cho dùng thử ở trung tâm văn thư lưu trữ Bà Rịa - Vũng Tàu). Tôi đánh giá tốt thái độ nghiêm túc và nỗ lực đó của học viên.

Tôi cho rằng học viên Huỳnh Tân Trung đã hoàn thành nhiệm vụ của mình, luận văn cao học "Hệ thống nhận dạng và phân loại văn bản" đáp ứng được yêu cầu của một luận văn cao học ngành Công nghệ thông tin" và đề nghị cho phép bảo vệ trước hội đồng chấm luận văn cao học.

Ngày 5 tháng 6 năm 2007
Giáo viên hướng dẫn

TS. Trần Thái Sơn

NHẬN XÉT CỦA GIÁO VIÊN PHẢN BIỆN

……
……
……
……
……
……
……
……
……
……
……
……
……
……
……
……
……

Ngày…… tháng……năm 2007

Giáo viên phản biện

CHƯƠNG I. MỞ ĐẦU

I.1.Giới thiệu:

Chúng ta hãy cùng nhau xem xét các trường hợp thường hay xảy ra trong thực tế sau:

Trong thời đại bùng nổ công nghệ thông tin hiện nay, hệ thống dữ liệu số hoá trở nên khổng lồ để phục vụ cho việc lưu trữ trao đổi thông tin, Dữ liệu số hoá này rất đa dạng - nó có thể là các dữ liệu dưới dạng tập tin văn bản text, tập tin văn bản MS Word, tập tin văn bản PDF, mail, HTML .v.v. Các tập tin văn bản cũng được lưu trữ trên máy tính cục bộ hoặc đươc truyền tải trên intenet, cùng với thời gian và/hoặc số lượng người dùng tăng nhanh thì các tập tin này ngày càng nhiều và đến một thời điểm nào đó thì số lượng tập tin này sẽ vượt quá tầm kiểm soát, do đó khi muốn tìm kiếm lại 1 văn bản nào đó việc tìm kiếm sẽ rất khó khăn và phức tạp, đặc biệt là trong trường hợp người cần tìm kiếm không nhớ rõ các câu cần tìm chính xác trong văn bản

Các thông tin trên internet có rất nhiều và phong phú gần như đáp ứng được hầu hết các nhu cầu thông tin của con người khi cần tra cứu thông tin. Các thông tin này thường xuyên được cập nhật và thay đổi liên tục, do vậy khi người cần tìm kiếm muốn tìm kiếm thông tin thì lượng thông tin thỏa mãn nhu cầu tìm kiếm sẽ rất nhiều nhưng chưa đủ để trở thành tài liệu phục vụ cho người tìm kiếm; do đó khi người sử dụng muốn sắp xếp các thông tin tìm được theo thể loại (nhóm văn bản) thì thời gian thực hiện sẽ mất rất nhiều (thời gian) và công sức bỏ ra cũng không phải nhỏ

Từ các nhu cầu trên mà yêu cầu về một **Hệ thống nhận dạng và phân loại văn bản** để đáp ứng yêu cầu phân loại văn bản sau đó mới thực hiện tìm kiếm được ra đời nhằm đáp ứng yêu cầu thực tế của người dùng. Đã có rất nhiều công trình nghiên cứu và ứng dụng thực tế dùng để thực hiện việc phân loại văn bản, tuy nhiên các ứng dụng đó cũng chưa thể đáp ứng hoàn toàn nhu cầu của người sử dụng, do vậy mà việc

tìm kiếm, nghiên cứu các giải thuật, các phương pháp phân loại văn bản vẫn được tiếp tục nghiên cứu và hoàn thiện

Với mục tiêu góp phần vào lĩnh vực nghiên cứu và ứng dụng phân loại văn bản vào cuộc sống, luận văn này sẽ thực hiện các công việc sau:

- Nghiên cứu và tổng hợp một số phương pháp phân loại văn bản (tiếng Anh và tiếng Việt) đã làm và sau đó đưa ra 1 số nhận xét đánh giá
- Nghiên cứu và đưa vào ứng dụng trong việc phân loại văn bản tiếng Việt bằng lý thuyết khá mới hiện nay là lý thuyết phân loại văn bản bằng hạt nhân chuỗi (string kernels) và phương pháp hỗ trợ vecto (Support vector Machine - SVM)
- Đưa ra một chương trình máy tính để thử nghiệm và có kết quả đánh giá về phương pháp phân loại văn bản sử dụng Hạt nhân chuỗi (string kernels) kết hợp với Máy hỗ trợ vecto (Support vector Machine - SVM)

I.2. Tổng quan về phân loại văn bản và các nghiên cứu đã thực hiện

Bài toán nhận dạng và phân loại văn bản là một trong những bài toán kinh điển trong lĩnh vực xử lý dữ liệu văn bản. Xử lý dữ liệu văn bản bao gồm:

- Kiểm tra lỗi chính tả (spelling-checker)
- Kiểm tra lỗi văn phạm (grammar checker)
- Từ điển đồng nghĩa (thesaurus)
- Phân tích văn bản (text analyzer)
- **Phân loại văn bản (text classification)**
- Tóm tắt văn bản (text summarization)
- Tổng hợp tiếng nói (voice synthesis)
- Nhận dạng giọng nói (voice recognization)
- Dịch tự động (automatic translation)

- .....

Phân loại văn bản là công việc phân tích nội dung của văn bản và sau đó ra quyết định văn bản này thuộc nhóm nào trong các nhóm văn bản đã cho trước. Do đó để công việc phân loại văn bản chính xác cần phải đáp ứng được các yêu cầu sau:

- Các văn bản trong nhóm đã được phân loại phải có những tiêu chuẩn chung nào đó
- Các văn bản khi phân tích thì phải "hiểu" được nội dung để xác định được các tiêu chuẩn trong văn bản
- Việc xác định loại của văn bản khi so sánh với các nhóm văn bản yêu cầu phải có những định lượng xác định để xác định chính xác văn bản cần phân tích thuộc nhóm văn bản nào

Do đó rõ ràng việc phân loại văn bản chính là công việc khai phá dữ liệu văn bản (text data mining). Trong lĩnh vực khai phá dữ liệu, các phương pháp phân loại văn bản đã dựa trên những phương pháp quyết định như quyết định Bayes, cây quyết định, láng giềng gần nhất, mạng nơron, ... Những phương pháp này đã cho kết quả chấp nhận được và được sử dụng trong thực tế, tuy nhiên việc nghiên cứu việc phân loại văn bản tiếng Việt vẫn chưa được lâu năm và chưa được sâu rộng, nguyên nhân là do tiếng Việt có những đặc trưng khác với tiếng Anh như từ không biến đổi hình thái, ý nghĩa ngữ pháp nằm ở ngoài từ, ranh giới từ không xác định mặc nhiên bằng khoảng trắng .v.v. (xin xem thêm ở phần II.3. Các đặc điểm cơ bản về tiếng Việt), ở đây có thể kể tên khá nhiều nghiên cứu về vấn đề này ở phần tham khảo

I.3.Mục tiêu của luận văn

Do phạm vi bài toán khá lớn và thời gian làm đề tài cũng hạn hẹp nên mục tiêu nghiên cứu của luận văn này sẽ được tập trung ở các điểm sau:

- Nghiên cứu kỹ thuật phân loại văn bản và một số phương pháp phân loại văn bản, mô tả các yêu cầu chính yếu nhất của từng phương pháp và rút ra các ưu/khuyết điểm của từng phương pháp, các phương pháp được nghiên cứu ở đây là các phương pháp được đánh giá tương đối mới, đã được các đề tài nghiên cứu trong nước ứng dụng
- Nghiên cứu và ứng dụng cách xử lý ngôn ngữ tiếng Việt:
    - Phương pháp tách từ ứng dụng trong tiếng Việt (trong luận văn này sử dụng phương pháp Maximum Matching: forward/backward nhưng sẽ có một số cải biến để tăng độ chính xác)
    - Phương pháp phân tích để định dạng văn bản tiếng Việt (trong luận văn sử dụng phương pháp phân tích Support vector machine (SVM) dựa trên lý thuyết về String kernels)
- Xây dựng thử nghiệm phương pháp nhận dạng và phân loại văn bản tiếng Việt dựa trên các nghiên cứu về tách từ, string kernels và SVM đã nêu ở trên
- Đưa ra các kết luận và có thể dùng để so sánh với các phương pháp khác đã được sử dụng, đồng thời cũng sẽ nêu ra phương hướng để giải quyết các vấn đề còn tồn tại

I.4.Nội dung nghiên cứu

Dựa trên các mục tiêu của luận văn việc nghiên cứu trong luận văn này sẽ tiến hành bám sát yêu cầu mục tiêu đòi hỏi:

- Nghiên cứu các phương pháp phân tích văn bản mới được đưa ra hoặc có tính phổ biến được sử dụng nhiều trong thực tế
- Dựa trên các kết quả đã nghiên cứu về phân loại văn bản ở trên thì luận văn sẽ chọn lựa một phương pháp mới trong việc phân loại văn bản đó là phương pháp Hạt nhân chuỗi (String Kernels) kết hợp với Máy Hỗ trợ Vecto (Support vector machine – SVM)
- Luận văn cũng sẽ nghiên cứu các phương pháp phân tích và tách câu-từ trong tiếng Việt, với mỗi phương pháp sẽ đưa ra được các ưu nhược điểm của từng phương pháp

- Dựa trên các nghiên cứu về phân tích câu từ tiếng Việt, luân văn sẽ đề xuất một cách mới để tăng độ chính xác của việc phân tích câu từ tiếng Việt
- Để chứng minh tính chính xác hơn khi phân tích văn bản so với các cách phân tích văn bản cũ; dựa trên các phương pháp phân tích câu-từ tiếng Việt đã đề xuất và với phương pháp Hạt nhân Chuỗi (String Kernels) kết hợp với Máy Hỗ trợ Vecto (Support vector machine – SVM) sẽ xây dựng một chương trình thử nghiệm với các nghiên cứu đã được tổng hợp
- Trong quá trình thực hiện chương trình, để tăng nhanh tốc độ lập trình và hiệu quả của phương pháp làm, sẽ có sử dụng lại các chương trình tính toán được cung cấp ở dạng mã mở (open source code). Cụ thể là việc thực hiện chương trình đã sử dụng cơ sở dữ liệu tiếng Việt của Đinh Điền, chương trình đọc và nhận dạng text cho các file PDF là mã nguồn mở trên http://sourceforge.net/ chương trình tính toán Máy Hỗ trợ Vecto (Support vector machine – SVM) là chương trình của Chih-Jen Lin được cho tại địa chỉ http://www.csie.ntu.edu.tw/~cjlin

Việc kết luận chủ yếu sẽ là đưa ra các kết luận thực nghiệm khi sử dụng, xác định được những thông số để có thể sử dụng các kết quả này nhằm có thể so sánh được với các phương pháp và kết quả nghiên cứu của các công trình khác đã được các tác giả khác nghiên cứu

I.5.Kết quả đạt được

Sau quá trình nghiên cứu và thực hiện luận văn đã đạt được các kết quả như sau:

- Đã nghiên cứu và tiếp thu các kỹ thuật phân loại văn bản đang được sử dụng trong thực tế
- Nắm được phương pháp phân loại văn bản bằng Hạt nhân chuỗi (String Kernels) kết hợp với Máy Hỗ trợ Vecto (Support vector machine – SVM).
- Ứng dụng được các kết quả nghiên cứu xử lý ngôn ngữ tự nhiên vào chương trình phân loại văn bản.

- Đề xuất phương án để phân tích câu tiếng Việt được chính xác và nhanh chóng hơn
- Đã xây dựng thử nghiệm một chương trình phân loại văn bản cho các file văn bản tiếng Việt.
- Có những kết luận và có các khuyến cáo để tăng tốc độ chương trình và hạn chế các sai sót có thể mắc phải

CHƯƠNG II. CƠ SỞ LÝ THUYẾT

II.1.Một số định nghĩa trong vấn đề văn bản và ngôn ngữ:

II.1.1.Các cấp độ trong ngôn ngữ:

Theo trình tự từ nhỏ đến lớn, có thể kể ra các đơn vị ngôn ngữ là:

- Âm vị: đơn vị âm thanh nhỏ nhất để cấu tạo nên ngôn ngữ và khu biệt về mặt biểu hiện vật chất (âm thanh) của các đơn vị khác, ví dụ: k-a-d (card);b-i-g (big)
- Hình vị: đơn vị nhỏ nhất mang nghĩa (nghĩa ngữ pháp hay nghĩa từ vựng) được cấu tạo bởi các âm vị, VD: read-ing;book-s
- Từ: đơn vị mang nghĩa độc lập, được cấu tạo bởi (các) hình vị, có chức năng định danh, VD: I-am-reding-my-books
- Ngữ: gồm 2 hay nhiều từ có quan hệ ngữ pháp hay ngữ nghĩa với nhau, VD:bức thư, mạng máy tính, computer system
- Câu: gồm các từ/ngữ có quan hệ ngữ pháp hay ngữ nghĩa với nhau và có chức năng cơ bản la thông báo, VD: I am reading my books
- Văn bản: hệ thống các câu được liên kết với nhau về mặt hình thức, từ ngữ, ngữ nghĩa và ngữ dụng

II.1.2.Các quan hệ trong ngôn ngữ

Mỗi đơn vị kể trên, đến lượt chúng lại làm thành một tiểu hệ thống trong hệ thống lớn là hệ thống ngôn ngữ. Người ta gọi mỗi tiểu hệ thống (gồm những đơn vị đồng loại) của ngôn ngữ là một cấp độ. Đó là vì các tiểu hệ thống đó có quan hệ chi phối với nhau. Ví dụ: cấp độ câu, cấp độ từ, cấp độ hình vị, cấp độ âm vị. Các đơn vị của ngôn ngữ quan hệ với nhau rất phức tạp và theo nhiều kiểu, tuy nhiên có 3 quan hệ cốt lõi là:

- Quan hệ cấp bậc (hierachical relation): đơn vị cấp cao hơn bao giờ cũng bao hàm đơn vị của cấp độ thấp hơn và ngược lại. Ví dụ: câu bao hàm từ ….
- Quan hệ ngữ đoạn (syntagmatical relation): nối kết các đơn vị ngôn ngữ thành chuỗi khi ngôn ngữ đi vào hoạt động. Đây là tính hình

tuyến của ngôn ngữ. Tính chất này bắt buộc các đơn vị ngôn ngữ phải nối tiếp nhau lần lượt trong ngữ lưu để cho ta những kết hợp gọi là ngữ đoạn (syntagmes). Ví dụ This book, this book is interesting …..

- Quan hệ liên tưởng (associative relation): là quan hệ xâu chuỗi, mọt yếu tố xuất hiện với những yếu tố khiếm diện "đứng sau lưng" nó về nguyên tắc có thể thay thế cho nó. Ví dụ: I read book (newspage, magazine,…) thì các từ newspage, magazine là tương đương với book và có thể thay thế cho book

II.2.Phân loại ngôn ngữ

II.2.1.Phân theo cội nguồn

Căn cứ theo cội nguồn (nghiên cứu lịch đại), ta có các ngữ hệ sau

- Ấn-Âu: dòng Ấn Độ, I-Ran, Bantic, Slave, Roman, Hy Lạp, German, (Gồm Đức, Anh, Hà Lan ….)
- Sê-mít: dòng Sê-mít, Ai Cập, Kusit, Beebe …
- Thổ: Ngôn ngữ Thổ Nhĩ Kỳ, Azecbaizan, Tacta …
- Hán-Tạng: dòng Hán, Tạng, Miến …
- Nam Phương: dòng Nam-Thái, Nam Á. Trong dòng Nam Á có các ngành: Nahali, MunDa, Nicoba và Môn-Khmer. Trong ngành Mon-Khmer có nhóm Việt-Mường và trong nhóm này có ngôn ngữ Tiếng Việt của chúng ta

II.2.2.Phân theo loại hình

Căn cứ theo đặc điểm hiện nay của các ngôn ngữ (nghiên cứu đồng đại), người ta phân các ngôn ngữ thành các loại hình sau (một cách gần đúng)

- Ngôn ngữ hòa kết (flexional): loại hình này bao gồm các ngôn ngữ: Đức, Latin, Hi lạp, Anh, Pháp, Nga, A-rập …
- Ngôn ngữ chắp dính (agglutinate): có hiện tượng cứ nối tiếp thêm một cách máy móc, cơ giới vào căn tố nào đó một hay nhiều phụ tố, mà mỗi

phụ tố đó lại chỉ luôn mang lại một ý nghĩa ngữ pháp nhất định. Ví dụ: Thổ Nhĩ Kỳ, Mông Cổ, Nhật Bản, Triều Tiên ….

- Ngôn ngữ đơn lập (isolate): còn gọi là ngôn ngữ phi hình thái, không biến hình, đơn âm tiết, phân tiết …. Loại hình này bao gồm các ngôn ngữ: tiếng Việt, Hán, Êvê, vùng Đông Nam Á ….
- Ngôn ngữ đa tổng hợp (polysynthetic): còn gọi là ngôn ngữ hỗn nhập hay lập khuôn. Đây là loại mang những đặc điểm của các loại hình nói trên

II.2.3.Phân theo trật tự từ của ngôn ngữ

Xét về loại hình trật tự ở cấp độ câu, thì tiếng Anh và tiếng Việt có cùng chung loại hình, đó là loại hình **S V O**, có nghĩa là trong một câu bình thường (không đánh dấu), thứ tự các thành phần được sắp xếp như sau:

S (subject: chủ ngữ) – V (Verb: động từ) – O (Object: Bổ Ngữ)

Ví dụ :

Tôi nhìn anh ấy và I see Him

S V O S V O

Theo thống kê, thì :

- Loại hình SVO chiếm 32,4 - 41,8 %, bao gồm các tiếng như: tiếng Anh, Pháp, Việt,….
- Loại hình SOV chiếm 41 – 51,8 %, như tiếng Nhật
- Loại hình VSO chiếm 2 – 4 %
- Loại hình VOS chiếm 9 – 18 %
- Loại hình OSV chiếm cỡ 1%

Trật tự từ (word – order) là sự thể hiện hình tuyến của ngôn ngữ. Trật tự từ được hiểu theo nghĩa hẹp là: trật tự các thành phần S-V-O như trên, còn nếu hiểu theo nghĩa rộng, thì là trật tự các thành tố ở ba cấp độ đơn vị ngôn ngữ:

- Từ: trật tự các tiếng, hình vị, từ tố trong từ ghép. Ví dụ: Cha-Mẹ/Mẹ-Cha
- Ngữ: trật tự các từ trong cụm từ hay ngữ, như: trật tự định tố trong danh ngữ, trật tự bỏ ngữ trong động ngữ …
- Câu: trật tự các thành phần S, V, O trong câu

Có một số ngôn ngữ tuy cùng loại hình trật tự từ ở câp độ câu (như tiếng Anh và tiếng Việt cùng loại hình SVO), nhưng trật tự từ bên trong các ngữ có thể khác nhau. Chẳng hạn: trong tiếng Anh tính từ đứng trước danh từ, còn trong tiếng Việt thì ngược lại

II.3.Các đặc điểm của tiếng Anh

Tiếng Anh được xếp vào loại hình biến cách (flexion) hay còn gọi là loại hình khuất chiết với những đặc điểm sau:

- Trong hoạt động ngôn ngữ, từ có biến đổi hình thái. Ý nghĩa ngữ pháp nằm trong từ. Ví dụ: I see him và he see me
- Phương pháp ngữ pháp chủ yếu là: phụ tố. Ví dụ: learning và learned.
- Hiện tượng cấu tạo từ bằng cách ghép thêm phụ tố (affix) vào gốc từ là rất phổ biến. Ví dụ: anticomputerizational (anti-compute-er-ize-action-al)
- Kết hợp giữa các hình vị là chặt chẽ. Ranh giới giữa các hình vị là khó xác định
- Ranh giới từ được nhận diện bằng khoảng trắng hoặc dấu câu

II.4.Tóm tắt các phương pháp phân loại văn bản bằng tiếng Anh

Tiếng Anh là ngôn ngữ hiện đang được sử dụng khá thông dụng trên thế giới do vậy các phương pháp phân loại văn bản tiếng Anh cũng được nghiên cứu khá nhiều, ở đây chỉ nêu 1 vài phương pháp đang sử dụng và tỏ ra có hiệu quả khá cao:

II.4.1.Naïve Bayes (NB)

NB là phương pháp phân loại dựa vào xác suất được sử dụng rộng rãi trong lĩnh vực máy học (Mitchell trình bày năm 1996, Joachims

trình bày năm 1997 và Jason năm 2001) được sử dụng lần đầu tiên trong lĩnh vực phân loại bởi Maron vào năm 1961, sau đó trở nên phổ biến dùng trong nhiều lĩnh vực như trong các công cụ tìm kiếm (được mô tả năm 1970 bởi Rijsbergen), các bộ lọc mail (mô tả năm 1998 bởi Sahami)...

*** Ý tưởng**

Ý tưởng cơ bản của cách tiếp cận Naïve Bayes là sử dụng xác suất có điều kiện giữa 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. Với giả định này NB không sử dụng sự phụ thuộc của nhiều từ vào một chủ đề, không sử dụng việc kết hợp các từ để đưa ra phán đoán chủ đề và do đó việc tính toán NB chạy nhanh hơn các phương pháp khác với độ phức tạp theo hàm số mũ.

*** Công thức**

Mục đích chính là tính được xác suất *Pr(Cj,d')*, xác suất để văn bản *d'* nằm trong lớp *Cj*. Theo luật Bayes, văn bản *d'* sẽ được gán vào lớp *Cj* nào có xác suất *Pr(Cj, d')* cao nhất. Công thức sau dùng để tính *Pr(Cj,d')* (do Joachims đề xuất năm 1997)

$$H_{BAYES} = \arg\max_{C_j \in C}\left(\frac{\Pr(C_j).\prod_{i=1}^{d'}\Pr(w_i \mid C_j)}{\sum_{C' \in C}\Pr(C').\prod_{i=1}^{|d'|}\Pr(w_i \mid C')}\right) = \arg\max_{C_j \in C}\left(\frac{\Pr(C_j).\prod_{i=1}^{d'}\Pr(w \mid C_j)^{IF(w,d')}}{\sum_{C' \in C}\Pr(C').\prod_{w \in F}\Pr(w \mid C')^{IF(w,d')}}\right)$$

Với:

- *(TF,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 trong tập dữ liệu huấn luyện : $$\Pr(C_j) = \frac{\| C_j \|}{\| C \|} = \frac{\| C_j \|}{\sum_{C' \in C} \| C' \|}$$
- $Pr(w_i|C_j)$ được tính sử dụng phép ước lượng Laplace (do Napnik trình bày năm 1982) $$\Pr(w_i \mid C_j) = \frac{1 + TF(w_i, C_j)}{|F| + \sum_{w' \in |F|} TF(w', C_j)}$$
-

Ngoài ra còn có các phương pháp NB khác có thể kể ra như sau ML Naive Bayes, MAP Naive Bayes, Expected Naive Bayes, Bayesian Naive Bayes (Jason mô tả năm 2001). Naive Bayes là một công cụ rất hiệu quả trong một số trường hợp. Kết quả có thể rất tồi 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 nhanh, dễ dàng cập nhật dữ liệu huấn luyện mới và có tính độc lập cao với tập huấn luyện, có thể sử dụng kết hợp nhiều tập huấn luyện khác nhau. Tuy nhiên NB ngoài giả định tính độc lập giữa các từ còn phải cần đến một ngưỡng tối ưu để cho kết quả khả quan. Nhằm mục đích cải thiện hiệu năng của NB, các phương pháp như multiclass-boosting, ECOC (do Berger trình bày năm 1999 và Ghani mô tả lại năm 2000) có thể được dùng kết hợp.

II.4.2.Phương pháp K–Nearest Neighbor (kNN)

Đây là phương pháp truyền thống khá nổi tiếng về hướng tiếp cận dựa trên thống kê đã được nghiên cứu trong nhận dạng mẫu hơn bốn thập kỷ qua (theo tài liệu của Dasarathy năm 1991). kNN được đánh giá là một trong những phương pháp tốt nhất (áp dụng trên tập dữ liệu Reuters phiên bản 21450), được sử dụng từ những thời kỳ đầu của việc

phân loại văn bản (được trình bày bởi Marsand năm 1992, Yang năm 1994, Iwayama năm 1995)

*** Ý tưởng**

Khi cần phân loại một văn bản mới, thuật toán sẽ tính khoảng cách (khoảng cách Euclide, Cosine ...) 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* "láng giềng"), sau đó dùng các khoảng cách này đánh trọng số cho tất cả chủ đề. Trọng số của một chủ đề chính là tổng tất cả 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 mức độ trọng số giảm dần và các chủ đề có trọng số cao sẽ được chọn là chủ đề của văn bản cần phân loại.

*** Công thức**

Trọng số của chủ đề c_j đối với văn bản $\vec{x}$:

$$W(\vec{x}, c_j) = \sum sim(\vec{x}, \vec{d}_i).y(\vec{d}_i, c_j) - b_j$$

Trong đó

$y(\vec{d}_i, c_j) \in \{0,1\}$, với *y = 0*: văn bản $\vec{d}_i$ không thuộc về chủ đề c_j, *y = 1*: văn bản $\vec{d}_i$ thuộc về chủ đề c_j.

$sim(\vec{x}, \vec{d}_i)$: độ giống nhau giữa văn bản cần phân loại $\vec{x}$ và văn bản $\vec{d}_i$. Có thể sử dụng độ đo cosine để tính $sim(\vec{x}, \vec{d}_i)$

$$sim(\vec{x}, \vec{d}_i) = \cos(\vec{x}, \vec{d}_i) = \frac{\vec{x}.\vec{d}_i}{\| \vec{x} \| . \| \vec{d}_i \|}$$

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 việc phân loại, thuật toán phải 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 (theo Yang trình bày năm 1997). Giá trị tốt nhất được sử dụng tương ứng trên hai bộ dữ liệu Reuter và Oshumed là *k = 45*.

II.4.3.Support vector Machine (SVM)

Support vector Machine (SVM) là phương pháp tiếp cận phân loại rất hiệu quả được Vapnik giới thiệu năm 1995 để giải quyết vấn đề nhận dạng mẫu 2 lớp sử dụng nguyên lý Cực tiểu hóa Rủi ro có Cấu trúc (Structural Risk Minimization) (thep Vapnik).

*** Ý tưởng**

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

*** Công thức**

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

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

$$\vec{d}_i.\vec{w}+b=0$$

Đặt

$$h(\vec{d}_i)=sign(\vec{d}_i\cdot\vec{w}+b)=\begin{cases}+1, \vec{d}_i\cdot\vec{w}+b>0\\-1, \vec{d}_i\cdot\vec{w}+b<0\end{cases}$$

Như thế h($\vec{d}_i$) biểu diễn sự phân lớp của $\vec{d}_i$ vào hai lớp như đã nói. Gọi $y_i=\{\pm 1\}$, văn bản $\vec{d}_i \in$ lớp +; $y_i=-1$, văn bản $\vec{d}_i$ lớp -. Lúc này để có siêu mặt phẳng h ta sẽ phải giải bài toán sau :

Tìm Min $\|\vec{w}\|$ với $\vec{w}$ và b thoả điều kiên sau :

$$\forall i\in\overline{1,n}: y_i(\sin g(\vec{d}_i.\vec{w}))\geq 1$$

Bài toán SVM có thể giải bằng kỹ thuật sử dụng toán tử Lagrange để biến đổi thành dạng đẳng thức.

Điểm thú vị ở SVM là mặt phẳng quyết định chỉ phụ thuộc vào các vector hỗ trợ (Support Vector) có khoảng cách đến mặt phẳng quyết định là $\frac{1}{\|\vec{w}\|}$. 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 và NB vì tất cả dữ liệu trong tập huấn luyện đều được dùng để tối ưu hóa kết quả. Các phiên bản SVM tốt có thể kể

đến là SVMLight (Joachims trình bày năm 1998) và Sequential Minimal Optimization (SMO) (Platt trình bày năm 1998)

II.4.4.Neural Network (NNet)

Nnet được nghiên cứu mạnh trong hướng trí tuệ nhân tạo. Wiener là người đã sử dụng Nnet để phân loại văn bản, sử dụng 2 hướng tiếp cận : kiến trúc phẳng (không sử dụng lớp ẩn) và mạng nơron 3 lớp (bao gồm một lớp ẩn)(theo Wiener trình bày năm 1995)

Cả hai hệ thống trên đều sử dụng một mạng nơron riêng rẽ cho từng chủ đề, NNet học cách ánh xạ phi tuyến tính những yếu tố đầu vào như từ, hay mô hình vector của một văn bản vào một chủ đề cụ thể.

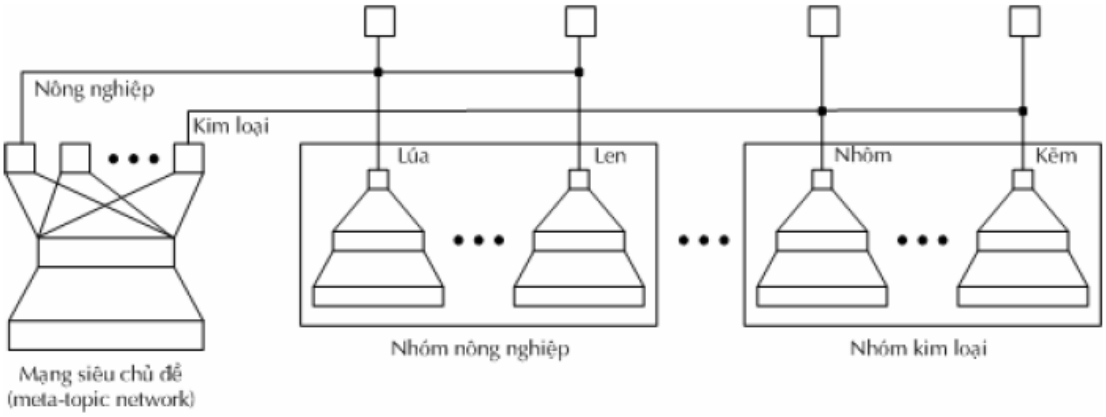
Khuyết điểm của phương pháp NNet là tiêu tốn nhiều thời gian dành cho việc huấn luyện mạng nơron.

*** Ý tưởng**

Mô hình mạng neural gồm có ba thành phần chính như sau: kiến trúc (architecture), hàm chi phí (cost function), và thuật toán tìm kiếm (search algorithm). Kiến trúc định nghĩa dạng chức năng (functional form) liên quan giá trị nhập (inputs) đến giá trị xuất (outputs).

Kiến trúc phẳng (flat architecture) : Mạng phân loại đơn giản nhất (còn gọi là mạng logic) có một đơn vị xuất là kích hoạt kết quả (logistic activation) và không có lớp ẩn, kết quả trả về ở dạng hàm (functional form) tương đương với mô hình hồi quy logic. Thuật toán tìm kiếm chia nhỏ mô hình mạng để thích hợp với việc điều chỉnh mô hình ứng với tập huấn luyện. Ví dụ, chúng ta có thể học trọng số trong mạng kết quả (logistic network) bằng cách sử dụng không gian trọng số giảm dần (gradient descent in weight space) hoặc sử dụng thuật toán interated-reweighted least squares là thuật toán truyền thống trong hồi quy (logistic regression).

Kiến trúc mô dun (modular architecture): Việc sử dụng một hay nhiều lớp ẩn của những hàm kích hoạt phi tuyến tính cho phép mạng thiết lập các mối quan hệ giữa những biến nhập và biến xuất. Mỗi lớp ẩn học để biểu diễn lại dữ liệu đầu vào bằng cách khám phá ra những đặc trưng ở mức cao hơn từ sự kết hợp đặc trưng ở mức trước.



Hình Kiến trúc mô đun (Modular Architecture) . Các kết quả của từng mạng con sẽ là giá trị đầu vào cho mạng siêu chủ đề và được nhân lại với nhau để dự đoán chủ đề cuối cùng.

*** Công thức**

Trong công trình của Wiener et al (1995) dựa theo khung của mô hình hồi quy, liên quan từ đặc trưng đầu vào cho đến kết quả gán chủ đề tương ứng được học từ tập dữ liệu. Do vậy, để phân tích một cách tuyến tính, tác giả dùng hàm sigmoid sau làm hàm truyền trong mạng neural:

$$p = \frac{1}{1+e^{-n}}$$

Trong đó, $\eta = \beta^t x$ là sự kết hợp của những đặc trưng đầu vào và p phải thỏa điều kiện $p \hat{I} (0,1)$.

II.4.5.Linear Least Square Fit (LLSF)

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

*** Ý tưởng**

LLSF 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 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, ta sẽ được ma trận đồng hiện của hệ số hồi quy của từ và chủ đề(matrix of word-category regression coefficients)

*** Công thức**

$$F_{LS} = \arg\min_{F} = \| FA - B \|^2$$

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, 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 ta vẫn tìm thấy điểm chung ở hoạt động của hai phương pháp là việc học ngưỡng tối ưu.

II.4.6.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) (được Han trình bày năm 2000)

*** Ý tưởng**

Mỗi lớp trong dữ liệu luyện sẽ được biểu diễn bởi một vector trọng tâm. Việc xác định lớp của một văn bản thử bất kì sẽ thông qua viêc tìm vector trọng tâm nào gần với vector biểu diễn văn bản thử nhất. Lớp của văn bản thử chính là lớp mà vector trọng tâm đại diện. Khoảng cách được tính theo độ đo cosine.

*** Công thức**

Công thức tính vector trọng tâm của lớp i

$$\overrightarrow{C_i} = \frac{1}{\left\|\{i\}\right\|} \sum_{d_j \in \{i\}} \overrightarrow{d_j}$$

Độ đo khoảng cách giữa vector $\vec{x}$ và $\vec{C}_i$

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

Trong đó :

$\vec{x}$ là vector văn bản cần phân loại

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

Chủ đề của $\vec{x}$ là Cx thoả cos($\vec{x}$, $\vec{C}_i$) argmax(cos($\vec{x}$, $\vec{C}_i$))

II.5.Các đặc điểm cơ bản về tiếng Việt

Tiếng Việt được xếp vào loại hình đơn lập (isolate) hay còn gọi là loại hình phi hình thái, không biến hình, đơn tiết với những đặc điểm chính sau:

- Trong hoạt động ngôn ngữ, từ không biến đổi hình thái. Ý nghĩa ngữ pháp nằm ở ngoài từ. Ví dụ: Tôi nhìn anh ấy và anh ấy nhìn tôi
- Phương thức ngữ pháp chủ yếu là: Trật tự từ và từ hư. Ví dụ: Gạo xay và xay gạo
- Tồn tại một loại đơn vị đặc biệt, đó là “hình tiết” mà vỏ ngữ âm của chúng trùng khít với âm tiết, và đơn vị đó cũng chính là “hình vị tiếng Việt” hay còn gọi là tiếng (theo tác giả Đinh Điền thì có khoảng 10.000 tiếng, nhưng theo khảo sát của hội người mù Việt Nam khi làm chương trình sách nói thì chỉ có khoảng 3000 từ)
- Ranh giới từ không xác định mặc nhiên bằng khoảng trắng như các thứ tiếng biến hình khác. Ví dụ: “học sinh học sinh học”. Điều này khiến cho việc phân tích hình thái (tách từ) tiếng Việt trở nên khó khăn. Việc nhận diện ranh giới từ là quan trọng làm tiền đề cho các xử lý tiếp theo sau đó như: kiểm tra lỗi chính tả, gán nhãn từ, thống kê tần xuất từ ….

- Tồn tại loại từ đặc biệt “từ chỉ loại” (classsifier) hay còn gọi là phó danh từ chỉ loại đi kèm với danh từ như: cái bàn, cuốn sách, bức thư, …..
- Về mặt âm học, các âm tiết tiếng Việt đều mang 1 trong 6 thanh điệu (ngang, sắc, huyền, hỏi, ngã, nặng). Đây là âm vị siêu đoạn tính
- Có hiện tượng láy trong từ tiếng Việt như: lấp lánh, lung ling …. Ngoài ra còn có hiện tượng nói lái (do mối liên kết giữa phụ âm đầu và phần vần trong âm tiết là lỏng lẻo) như: hiện đại **à** hại điện

II.6.So sánh đối chiếu tiếng Anh-Việt

Qua sự phân tích đặc điểm của tiếng Anh và tiếng Việt như trên, ta thấy tiếng Anh và tiếng Việt có nhiều điểm khác biệt (do loại hình ngôn ngữ, do nền văn hóa) chẳng hạn: khác biệt về ngữ âm học, hình vị, ranh giới từ, sự từ vựng hóa (như ox – bò đực, anh – elder brother, …); từ loại; trật tự từ (tính từ và danh từ), kết cấu câu (chủ đề và cụm chủ vị), …

Vì vậy chúng ta không thể áp dụng y nguyên các mô hình xử lý ngôn ngữ của tiếng Anh sang cho tiếng Việt được mà phải có sự điều chỉnh nhất định dựa trên các kết quả so sánh đối chiếu giữa tiếng Anh và tiếng Việt.

II.7.Tóm tắt các phương pháp phân loại văn bản bằng tiếng Việt

II.7.1.Phương pháp khớp tối đa Maximum Matching: forward/backward

*** Nội dung**

Phương pháp khớp tối đa (Maximum Matching) còn gọi là Left Right Maximum Matching (LRMM). Theo phương pháp này, ta sẽ duyệt một ngữ hoặc câu từ trái sang phải và chọn từ có nhiều âm tiết nhất có mặt trong từ điển, rồi cứ thể tiếp tục cho từ kế tiếp cho đến hết câu. Thuật toán được trình bày bởi Chih-Hao Tsai năm 2000

Dạng đơn giản được dùng giải quyết nhập nhằng từ đơn. Giả sử có một chuỗi ký tự (tương đương với chuỗi tiếng trong tiếng Việt) *C1, C2, ..., C_n*. Ta bắt đầu từ đầu chuỗi. Đầu tiên kiểm tra xem *C1*, có phải là từ hay không, sau đó kiểm tra xem *C1C2* có phải là từ hay không. Tiếp tục tìm cho đến khi tìm được từ dài nhất. Từ có vẻ hợp lý nhất sẽ là từ dài nhất. Chọn từ đó, sau đó tìm tiếp như trên cho những từ còn lại cho đến khi xác định được toàn bộ chuỗi từ.

Dạng phức tạp: Quy tắc của dạng này là phân đoạn có vẻ hợp lý nhất là đoạn ba từ với chiều dài tối đa. Thuật toán bắt đầu như dạng đơn giản. Nếu phát hiện ra những cách tách từ gây nhập nhằng (ví dụ, *C1* là từ và *C1C2* cũng là từ), ta xem các chữ kế tiếp để tìm tất cả các đoạn ba từ có thể có bắt đầu với *C1* hoặc *C1C2*. Ví dụ ta được những đoạn sau:

C1 C2 C3 C4

C1C2 C3 C4 C5

C1C2 C3 C4 C5 C6

Chuỗi dài nhất sẽ là chuỗi thứ ba. Vậy từ đầu tiên của chuỗi thứ ba (*C1C2*) sẽ được chọn. Thực hiện lại các bước cho đến khi được chuỗi từ hoàn chỉnh.

**** Ưu điểm***

Với cách này, ta dễ dàng tách được chính xác các ngữ/câu như " hợp tác xã || mua bán", "thành lập || nước || Việt Nam || dân chủ || cộng hòa"

- Cách tách từ đơn giản, nhanh, chỉ cần dựa vào từ điển
- Trong tiếng Hoa, cách này đạt được độ chính xác 98,41% (theo Chih-Hao Tsai trình bày năm 2000).

**** Hạn chế***

- Độ chính xác của phương pháp phụ thuộc hoàn toàn vào tính đủ và tính chính xác của từ điển
- Phương pháp này sẽ tách từ sai trong các trường hợp " học sinh || học sinh|| học", "một || ông || quan tài || giỏi", "trước || bàn là || một || ly || nước"…

II.7.2.Phương pháp giải thuật học cải biến (Transformation-based Learning, TBL)

*** Nội dung**

Đây là cách tiếp cận dựa trên ngữ liệu đã đánh dấu. Theo cách tiếp cận này, để huấn luyện cho máy tính biết cách nhận diện ranh giới từ tiếng Việt, ta có thể cho máy "học" trên ngữ liệu hàng vạn câu tiếng Việt đã được đánh dấu ranh giới từ đúng.

Sau khi học xong, máy sẽ xác định được các tham số (các xác suất) cần thiết cho mô hình nhận diện từ.

**** Ưu điểm***

- Đặc điểm của phương pháp này là khả năng tự rút ra quy luật của ngôn ngữ
- Nó có những ưu điểm của cách tiếp cận dựa trên luật (vì cuối cùng nó cũng dựa trên luật được rút ra) nhưng nó khắc phục được khuyết điểm của việc xây dựng các luật một cách thủ công bởi các chuyên gia.
- Các luật được thử nghiệm tại chỗ để đánh giá độ chính xác và hiệu quả của luật (dựa trên ngữ liệu huấn luyện)

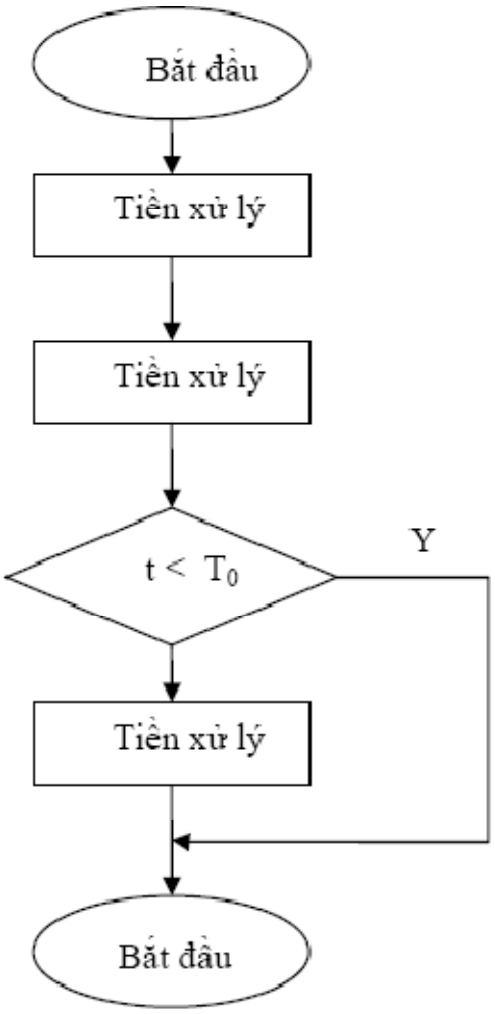
*** Hạn chế**

- Phương pháp này “dùng ngữ liệu có gán nhãn ngôn ngữ để học tự động các qui luật đó” (theo Đinh Điền năm 2004). Nhưng có thể nhận thấy rõ là việc xây dựng một tập ngữ liệu đạt được đầy đủ các tiêu chí của tập ngữ liệu trong tiếng Việt là một điều rất khó, tốn kém nhiều về mặt thời gian và công sức.
- Hệ phải trải qua một thời gian huấn luyện khá lâu để có thể rút ra các luật tương đối đầy đủ
- Cài đặt phức tạp

II.7.3.Mô hình tách từ bằng WFST và mạng Neural

**** Nội dung***

Mô hình mạng chuyển dịch trạng thái hữu hạn có trọng số WFST (Weighted finit–state Transducer) đã được Richard áp dụng để tách từ tiếng Trung Quốc. Ý tưởng cơ bản là áp dụng WFST kết hợp với trọng số là xác suất xuất hiện của mỗi từ trong ngữ liệu. Dùng WFST để duyệt qua câu cần xét. Cách duyệt có trọng số lớn nhất sẽ là cách tách từ được chọn. Giải pháp này cũng đã đượng áp dụng bởi tác giả Đinh Điền (năm 2001) kèm với mạng neutral để khử nhập nhằng. Hệ thống tách từ tiếng Việt gồm hai tầng: tầng WFST ngoài việc tách từ còn xử lý thêm các vấn đề liên quan đến đặc thù của tiếng Việt như từ láy, tên riêng… và tầng mạng neural dùng để khử nhập nhằng nếu có.



Sơ đồ hệ thống WFST

- Tầng WFST :gồm có ba bước
- Xây dựng từ điển trọng số : theo mô hình WFST, việc phân đoạn từ được xem như là một sự chuyển dịch trạng thái có xác suất (Stochastic Transduction). 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 (POS: Part – Of – Speech).

Mỗi cung của D có thể là:

- ◊ Từ một phần tử của H tới một phần tử của H, hoặc
- ◊ Từ ε (ký hiệu kết thúc từ) tới một phần tử của P

Các nhãn trong D biểu thị một chi phí ước lượng (estimated cost) bằng công thức : Cost = - log(f/N)

- ◊ Với f: tần số của từ, N: kích thước tập mẫu.

Đối với các trường hợp từ mới chưa gặp, tác giả áp dụng xác suất có điều kiện Goog-Turning (Baayen) để tính toán trọng số.

- Xây dựng các khả năng phân đoạn từ : Để giảm sự bùng nổ tổ hợp khi sinh ra các dãy các từ có thể từ một dãy các tiếng trong câu, tác giả đề xuất một phương pháp mới là kết hợp dùng từ điển để hạn chế sinh ra các bùng nổ tổ hợp. Khi phát hiện thấy một cách phân đoạn từ nào đó không phù hợp (không có trong từ điển, không phải là từ láy, không phải là danh từ riêng…) thì tác giả loại bỏ các nhánh xuất phát từ cách phân đoạn từ đó.
- Lựa chọn khả năng phân đoạn từ tối ưu : Sau khi được một danh sách các cách phân đoạn từ có thể có của câu, tác giả chọn trường hợp phân đoạn từ có trọng số bé nhất như sau:
- Ví dụ: input = “Tốc độ truyền thông tin sẽ tăng cao”

  o Dictionary

  “tốc độ” 8.68

  “truyền” 12.31

“truyền thông”	1231
“thông tin”	7.24
“tin”	7.33
“sẽ”	6.09
“tăng”	7.43
“cao”	6.95

Id(D)*D* = “Tốc độ # truyền thông # tin # sẽ # tăng # cao.” 48.79

(8.68 +12.31 + 7.33 + 6.09 + 7.43 +6.95 = 48.79)

Id(D)*D* = “Tốc độ # truyền # thông tin # sẽ # tăng # cao.” 48.70

(8.68 +12.31 + 7.24 + 6.09 + 7.43 +6.95 = 48.70)

Do đó, ta có được phân đoạn tối ưu là “Tốc độ # truyền # thông tin # sẽ # tăng # cao.”

- Tầng mạng neural : Mô hình mạng neural mà tác giả đề xuất được dùng để lượng giá 3 dãy từ loại: NNV,NVN, VNN (N: Noun, V: Verb). Mô hình này được học bằng chính các câu mà cách phân đoạn từ vẫn còn nhập nhằng sau khi qua mô hình thứ nhất.

**** Ưu điểm***

- Độ chính xác trên 97% (theo Đinh Điền trình bày băn 2001)

- Mô hình cho kết quả phân đoạn từ với độ tin cậy (xác suất) kèm theo.
- Nhờ có tầng mạng neural nên mô hình có thể khử nhập nhằng các trường hợp tầng WFST cho ra nhiều ứng viên có kết quả ngang nhau
- Phương pháp này cho kết quả với độ chính xác khá cao vì mục đích của tác giả muốn nhắm đến việc tách từ thật chính xác để là nền tảng cho việc dịch máy.

**** Hạn chế***

- Cũng tương tự như phương pháp TBL, việc xây dựng tập ngữ liệu là rất công phu, nhưng thật sự rất cần thiết để phục vụ cho mục đích dịch máy sau này của tác giả.

II.7.4.Phương pháp quy hoạch động (dynamic programming)

**** Nội dung***

Phương pháp quy hoạch động do Le An Ha trình bày năm 2003 chỉ sử dụng tập ngữ liệu thô để lấy thông tin về tần số thống kê của từ , làm tăng độ tin cậy cho việc tính toán. Việc tính toán bắt đầu với những đơn vị chắc chắn như câu, các ngữ (chunk) được phân cách bởi dấu câu (như dấu phẩy, gạch nối, chấm phẩy…) vì những thành phần này không có tính nhập nhằng ngay cả trong văn viết cũng như nói. Sau đó, tác giả cố gắng tối đa hoá xác suất của ngữ bằng cách tìm ra nhiều cách tách ngữ đó. Cách tách cuối cùng là cách tách là cho ngữ đó có xác suất cao nhất. Ý tưởng của cách tách từ này cho một ngữ cần tách từ, ta phải tìm ra các tổ hợp từ tạo nên ngữ đó sao cho tổ hợp đó đạt được xác suất tối đa. Tuy nhiên trong phương pháp tính toán này, tác giả gặp phải vấn đề bùng nổ tổ hợp và phân tích ngữ liệu thô. Để giải quyết vấn đề trên, tác

giả đã sử dụng phương pháp quy hoạch động (dynamic programming) vì lúc đó, xác suất cực đại của một ngữ nhỏ hơn chỉ phải tính toán một lần và sử dụng lại trong các lần sau.

**** Ưu điểm***

- Không cần sử dụng tập ngữ liệu đã đánh dấu chính xác

**** Hạn chế***

- Trong thí nghiệm, tác giả chỉ dừng lại ở việc tách các từ có ba tiếng bởi vì tập ngữ liệu đầu vào vẫn còn khá nhỏ.
- Xác suất từ đúng là 51%, xác suất từ chấp nhận được 65% (theo Le An Ha). Xác suất này tương đối thấp so với các phương pháp tách từ khác đã đề cập ở trên.

II.8.Mô tả phương pháp sử dụng trong đề cương

II.8.1.Chọn phương án thực hiện luận văn

Sau khi nghiên cứu, xem xét các phương pháp dùng để nhân dạng và phân loại văn bản, chúng ta thấy rõ là các phương pháp đều có những ưu, nhược điểm khác nhau, tất cả các phương pháp đều chưa đạt được kết quả tuyệt đối, do vậy mà việc tìm một phương pháp khác có thể có khả năng tốt hơn là một việc cần làm

Tác giả đề tài quyết định chọn kết hợp hai phương pháp đó là phương pháp Hỗ Trợ Véc To (**SVM**- **Support Vector Machine**) và phương pháp Hạt nhân chuỗi (**String kernels**).

Việc chọn Hạt nhân chuỗi (String kernels) là vì:

- Đây là một phương pháp mới và cho đến thời điểm làm luận văn này chưa có nhiều đề tài làm về hat nhân chuỗi

- Việc sử dụng phương pháp phân tích của hạt nhân chuỗi khá gần với tiếng Việt, do trong tiếng Việt từ không biến đổi hình thái, ý nghĩa ngữ pháp nằm ở ngoài từ và phụ thuộc vào việc sắp xếp thứ tự các từ, và hạt nhân chuỗi (String kernels) thì dựa trên sự so sánh khoảng cách của các từ trong câu. Mô tả chi tiết về lý thuyết hạt nhân chuỗi (String kernels) sẽ được nói kỹ ở phần sau

Việc chọn phương pháp Hỗ Trợ Véc To (SVM- Support Vector Machine) là do các thử nghiệm thực tế cho thấy, phương pháp SVM có khả năng phân loại khá tốt đối với bài toán phân loại văn bản cũng như trong nhiều ứng dụng khác (như nhận dạng chữ viết tay, phát hiện mặt người trong các ảnh, ước lượng hồi quy, ...). So sánh với các phương pháp phân loại khác, khả năng phân loại của SVM là tương đương hoặc tốt hơn đáng kể

Do vậy việc sử dụng kết hợp cả hai phương pháp có thể sẽ đem lại kết quả tốt nhất cho việc phân loại văn bản tiếng Việt

II.8.2.Hạt nhân cho các chuỗi Text

Trong phần này ta mô tả một hạt nhân giữa hai văn bản. Ý tưởng là so sánh ý nghĩa các chuỗi con trong hai văn bản: càng có nhiều chuỗi con chung thì chúng càng giống nhau. Điều quan trọng là các chuỗi con này không cần phải nằm liền kề nhau và mức độ kề nhau của một chuỗi con trong văn bản được xác định bằng so sánh trọng lượng. Ví dụ: chuỗi con "**c-a-r**" hiện diện trong cả hai từ "**car**d" và từ "**c**ust**ar**d", nhưng trọng lượng của chúng khác nhau. Mỗi chuỗi con là một chiều trong không gian đặc trưng, và giá trị của toạ độ phụ thuộc vào mức độ xuất hiện thường xuyên, chặt chẽ của chuỗi con đó trong văn bản. Để đối phó với các chuỗi con không liền kề, cần phải sử dụng một nhân tố phân rã $\lambda \in (0, 1)$ để đo lường sự hiện diện của một đặc trưng nào đó trong văn bản (Xem Định nghĩa 1 để biết thêm chi tiết).

Vì dụ: Xét các văn bản đơn giản bao gồm các từ *cat, car, bat, bar*. Nếu chúng ta chỉ xem xét *k=2*, chúng ta sẽ có một không gian đặc trưng 8 chiều, với các từ được ánh xạ như sau:

	c-a	c-t	a-t	b-a	b-t	c-r	a-r	b-r
φ(cat)	λ^2	λ^3	λ^2	0	0	0	0	0
φ(car)	λ^2	0	0	0	0	λ^3	λ^2	0
φ(bat)	0	0	λ^2	λ^2	λ^3	0	0	0
φ(bar)	0	0	0	λ^2	0	0	λ^2	λ^3

Vì vậy, hạt nhân không được chuẩn hóa giữa *car* và *cat* là K(car, cat) = λ^4, ngược lại một phiên bản được chuẩn hoá có được như sau:

K(car,car) = K(cat,cat) = $2\lambda^4 + \lambda^6$ do đó K(car,cat) = $\lambda^4/(2\lambda^4 + \lambda^6) = 1/(2 + \lambda^2)$. Lưu ý thông thường một văn bản sẽ có nhiều hơn một từ, do đó việc ánh xạ toàn bộ văn bản vào một không gian đặc trưng là kết tất cả các từ và các khoảng trắng (bỏ qua dấu chấm câu) thành một dãy sự kiện duy nhất.

Ví dụ: Chúng ta có thể tính toán điểm tương đồng giữa hai phần của một câu nổi tiếng bằng Kant.

K(“science is organized knowledge”,“wisdom is organized life”)

Các giá trị của hạt nhân, với các giá trị k = 1, 2, 3, 4, 5, 6 là: K_1 = 0.580, K_2 = 0.580,K_3 = 0.478, K_4 = 0.439, K_5 = 0.406, K_6 = 0.370.

Tuy nhiên, đối với chuỗi con có kích thước *k>4* và các văn bản đã chuẩn hóa kích thước thì việc ước lượng trực tiếp các đặc trưng liên quan có thể không thực tế (thậm chí đối với các văn bản có kích cỡ vừa phải), vì vậy rõ ràng là việc sử dụng phương pháp biểu diễn là không khả thi. Nhưng cũng nhờ đó có thể định nghĩa và tính toán các đặc trưng như thế cho hạt nhân một cách rất hiệu quả bằng việc sử dụng các kỹ thuật

lập trình động. Để chuyển hoá sang hạt nhân ta bắt đầu bằng các đặc trưng và tính toán "inner product" của chúng. Trong trường hợp này không cần phải chứng minh nó thoả mãn các điều kiện của Mercer (symmetry and positive semi-definiteness) vì chúng sẽ tự động phát sinh một "inner product". Hạt nhân này gọi là một hạt nhân sự kiện con của chuỗi (SSK- string subsequence kernel) là cơ sở cho hoạt động của các ứng dụng sinh học. Nó ánh xạ các chuỗi vào một vectơ đặc trưng được chỉ mục bằng tất cả các *k-tupe* ký tự. Một *k*-tupe sẽ có một phần tử khác 0 nếu có một chuỗi con sự kiện hiện diện ở bất kỳ vị trí nào (không nhất thiết phải liên tục) trong chuỗi. Trọng lượng của đặc trưng sẽ là tổng số lần xuất hiện của k-tupple nhân tố phân rã trong chuỗi sự kiện.

*** Các định nghĩa**

Định nghĩa 1: (Hạt nhân chuỗi sự kiện con của chuỗi – String subsequence kernel - SSK). Xét Σ là một mẫu tự xác định. Một chuỗi là một dãy hữu hạn sự kiện của các ký tự từ Σ, bao gồm cả dãy sự kiện rỗng. Đối với chuỗi s, t, chúng ta biểu thị bởi |s| chiều dài của chuỗi $s = s_1 \ldots s_{|s|}$, và st chuỗi có được bằng cách ghép chuỗi s và t. Chuỗi s[i : j] là chuỗi con $s_i \ldots s_j$ của s. Ta nói u là dãy con sự kiện của s, nếu tồn tại các chỉ mục **i** = (i1, . . . , i|u|), với $1 \le i_1 < \ldots < i_{|u|} \le |s|$, như vậy $u_j = s_{ij}$, với j = 1, . . . , |u|, hoặc u = s[**i**]. Chiều dài l(**i**) của dãy con sự kiện trong s là $i_{|u|} - i1 + 1$. Ta biểu thị bằng Σ^n tập tất cả cả các chuỗi hữu hạn có chiều dài n, và bằng Σ^* tập tất cả các chuỗi

$$\Sigma^* = \bigcup_{1}^{a} \Sigma^n \qquad (1)$$

Bây giờ chúng ta định nghĩa các không gian đặc trưng $F_n = \Re^{\Sigma^n}$. Việc ánh xạ đặc trưng *φ* cho chuỗi *s* được tạo ra bởi việc định nghĩa *u* kết hợp *φu*(*s*) cho mỗi chuỗi *u* Î Σ*n*. Ta định nghĩa

$$\Phi_u(s) = \sum_{i:u=s[i]} l^{l(i)} \qquad (2)$$

với $\lambda \leq 1$. Những đặc trưng này được đo lường bằng số lần xuất hiện của dãy con sự kiện trong chuỗi *s*. Vì vậy, "inner product" của các không gian đặc trưng cho hai chuỗi *s* và *t* là kết cộng tất cả các dãy con được định lượng thông qua tần số xuất hiện và chiều dài của theo công thức sau:

$$K_n(s,t) = \sum_{u \in \sum^n} \langle K_u(s).\Phi_u(t) \rangle = \sum_{u \in \sum^n} \sum_{i:u=s[i]^n} l^{l(i)} \sum_{j:u=t[j]^n} l^{l(j)} = \sum_{u \in \sum^n} \sum_{i:u=s[i]^n} \sum_{j:u=t[j]^n} l^{l(i)+l(j)}$$

Việc tính toán trực tiếp các đặc trưng này mất một đơn vị là $O(|\sum|n)$ thời gian và không gian, vì đó là số đặc trưng liên quan. Rõ ràng là hầu hết các đặc trưng sẽ có các thành phần khác 0 đối các văn bản lớn. Để việc tính toán hạt nhân được hiệu quả, chúng ta giới thiệu một hàm bổ sung để hạn chế phép toán đệ quy cho hạt nhân này.

Xét $K'_i(s, t) = \sum_{u \in \sum^i} \sum_{i:u=s[i]^n} \sum_{j:u=t[j]} l^{|s|+|t|-i_1-j_1+2}$

$i = 1, \ldots, n-1$, dùng để đo chiều dài bắt đầu từ một chuỗi sự kiện nào đó đến cuối chuỗi *s* và *t* thay vì chỉ *l*(**i**) và *l*(**j**).Bây giờ chúng ta có thể định nghĩa một phép đệ quy cho K'_i và từ đó tính K_n,

Định nghĩa 2: Phép tính đệ quy cho hạt nhân dãy con sự kiện

$K'_0(s, t) = 1$, *đối với cả s, t,*

$K'_i(s, t) = 0$, *nếu* min $(|s|, |t|) < i$,

$K_i(s, t) = 0$, *nếu* $\min(|s|, |t|) < i$,

$$\mathrm{K'_i}(s,\ t) = l\,K'_i(s,t) + \sum_{j:t_j = x} K'_{i-1}(s, t[1:\ j-1])\,l^{|t|-j+2}$$

$i = 1, \ldots, n-1$,

$$\mathrm{K_n}(s,\ t) = l\,K_n(s,t) + \sum_{j:t_j = x} K'_{n-1}(s, t[1:\ j-1])\,l^{2}$$

Lưu ý chúng ta cần hàm hỗ trợ ***K'*** vì những kẽ hở bên trong của dãy con cần phải được xử lý. Tính đúng đắn của phép đệ quy được xác nhận khi quan sát chiều dài các chuỗi gia tăng, và việc gánh chịu nhân tố λ cho mỗi đơn vị chiều dài dư thừa. Vì thế, công thức cho *K'i*(*sx, t*), giới hạn đầu tiên có ít ký tự hơn, vì thế đòi hỏi một nhân tố λ đơn, trong khi giới hạn thứ 2 có khoảng $|t| - j + 2$ ký tự. Đối với công thức cuối cùng có giới hạn thứ 2 yêu cầu thêm chỉ 2 ký tự, một đối với *s* và một đối với *t*[1 : $j - 1$],vì x là ký tự cuối cùng của dãy n. Nếu chúng ta muốn tính toán $K_n(s, t)$ trong khoảng giá trị n, chúng ta chỉ cần tính $K'_i(s, t)$, và sau đó áp dụng bước đệ quy cuối cùng cho mỗi $K_n(s, t)$ cần sử dụng tới các giá trị lưu trữ của $K'_i(s, t)$. Dĩ nhiên chúng ta có thể tạo hạt nhân $K(s, t)$ bằng cách kết hợp các trọng lượng (dương) khác nhau của các $K_n(s, t)$ khác nhau cho mỗi *n*.

Việc tạo ra hạt nhân là cách chuẩn hoá nhằm loại bỏ bất kỳ sự lệch lạc nào trong văn bản. Ta có thể tận dụng đặc tính này một cách hiệu quả bằng cách chuẩn hoá các vectơ đặc trưng trong không gian đặc trưng. Vậy chúng ta tạo một phép kết gắn mới $\hat{f}(s) = \frac{f(s)}{\| f(s) \|}$ để tìm hạt nhân

$$\hat{K}(s,t) = \langle \hat{f}(s) \cdot \hat{f}(t) \rangle = \langle \frac{f(s)}{\| f(s) \|} \cdot \frac{f(t)}{\| f(t) \|} \rangle = \frac{1}{\| f(s) \| \times \| f(t) |} \langle f(s) \cdot f(t) \rangle = \frac{K(s,t)}{\sqrt{K(s,s)K(t,t)}}$$

Hiệu quả tính toán của SSK (string subsequence kernel)

SSK tính điểm giống nhau giữa văn bản *s* và *t* trong thời gian $n|s||t|^2$ với *n* là chiều dài của chuỗi sự kiện. Điều này được mô tả rõ ràng trong phép đệ quy của Định nghĩa 2, vòng lặp đệ quy phía ngoài cùng thực hiện với chiều dài của chuỗi và đối với mỗi chiều dài và mỗi ký tự thêm vào trong s và t thì tổng chuỗi sự kiện phải được tính toán ước lượng. Tuy nhiên, hoàn toàn có thể tăng tốc cho việc tính toán của SSK. Bây giờ, chúng ta trình bày tính hiệu quả của phép toán đệ quy SSK để làm giảm độ phức tạp của phép toán *O(n|s||t|)*, trước tiên bằng cách đánh giá

$$\mathrm{K}''_{\mathrm{i}}(\mathrm{sx},\ \mathrm{t}) = \sum_{j:t_j = x} K'_{i-1}(s, t[1:\ j-1])\, l^{|t|-j+2}$$

và để ý rằng chúng ta có thể ước lượng *K'i*(*s, t*) với phép đệ quy *O*(|*s*||*t*|),

$$\mathrm{K}'_{\mathrm{i}}(\mathrm{sx},\ \mathrm{t}) = l\,\mathrm{K}'_{\mathrm{i}}(\mathrm{sx},\ \mathrm{t}) + \mathrm{K}''_{\mathrm{i}}(\mathrm{sx},\ \mathrm{t})$$

Ta thấy $\mathrm{K}''_{\mathrm{i}}(\mathrm{sx},\ \mathrm{tu}) = l^{|u|}\mathrm{K}''_{\mathrm{i}}(\mathrm{sx},\ \mathrm{t})$, với điều kiện *x* không xuất hiện trong *u*, trong khi đó

$$\mathrm{K}''_{\mathrm{i}}(\mathrm{sx},\ \mathrm{tx}) = l\,\mathrm{K}''_{\mathrm{i}}(\mathrm{sx},\ \mathrm{t}) + l\,\mathrm{K}'_{\mathrm{i\text{-}1}}(\mathrm{s},\ \mathrm{t})$$

Tổng hợp các quan sát ta thấy khi tính toán K''_i (*s, t*) có thực hiện đệ quy *O*(|*s*||*t*|). Vì thế, chúng ta có thể ước lượng tất cả hạt nhân trong khoảng thời gian *O*(*n*|*s*||*t*|).

II.8.3.Cơ sở lý thuyết của Support vector Machine (SVM):

Đặc trưng cơ bản quyết định khả năng phân loại của một bộ phân loại là hiệu suất tổng quát hóa, hay là khả năng phân loại những dữ liệu mới dựa vào những tri thức đã tích lũy được trong quá trình huấn luyện. Thuật toán huấn luyện được đánh giá là tốt nếu sau quá trình huấn luyện, hiệu suất tổng quát hóa của bộ phân loại nhận được cao. Hiệu suất tổng quát hóa phụ thuộc vào hai tham số là *sai số huấn luyện* và *năng lực* của máy học. Trong đó sai số huấn luyện là tỷ lệ lỗi phân loại trên tập dữ liệu huấn luyện. Còn năng lực của máy học được xác định bằng *kích thước Vapnik- Chervonenkis (kích thước VC).* Kích thước VC là một khái niệm quan trọng đối với một họ hàm phân tách (hay là bộ phân loại). Đại lượng này được xác định bằng số điểm cực đại mà họ hàm có thể phân tách hoàn toàn trong không gian đối tượng. Một bộ phân loại tốt là bộ phân loại có năng lực thấp nhất (có nghĩa là đơn giản nhất) và đảm bảo sai số huấn luyện nhỏ.Phương pháp SVM được xây dựng dựa trên ý tưởng này.

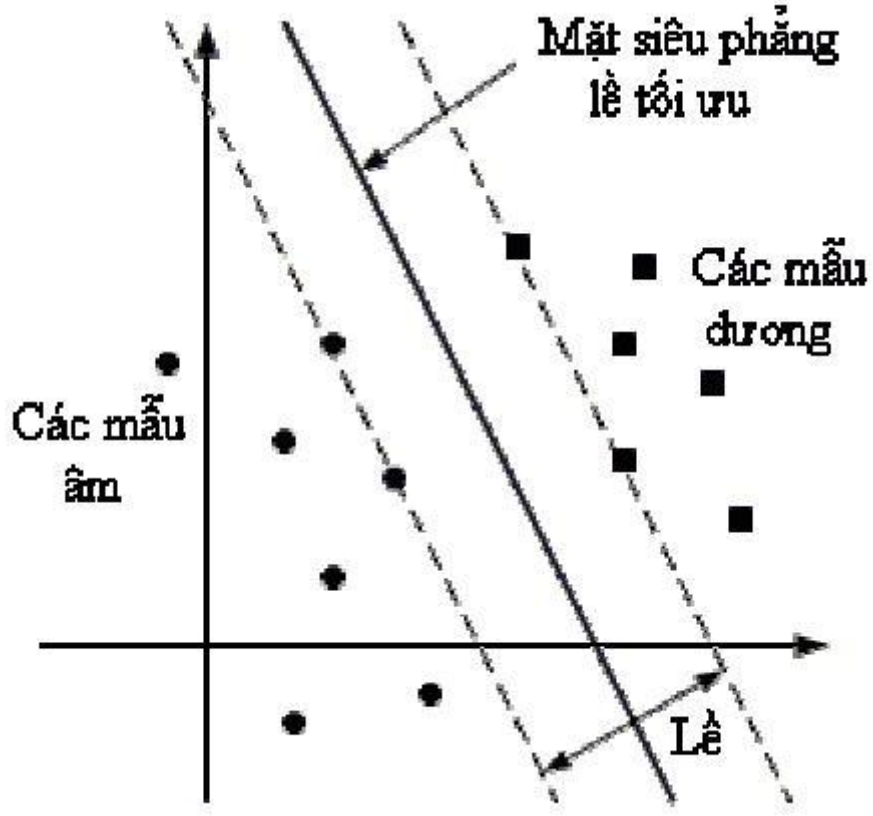
Xét bài toán phân loại đơn giản nhất - phân loại hai phân lớp với tập dữ liệu mẫu:

*{(**x**i, yi)| i = 1, 2, ..., N, **x**i □ Rm }*

Trong đó mẫu là các vector đối tượng được phân loại thành các mẫu dương và mẫu âm:

− Các mẫu dương là các mẫu **x***i* thuộc lĩnh vực quan tâm và được gán nhãn *yi* = 1;

− Các mẫu âm là các mẫu **x***i* không thuộc lĩnh vực quan tâm và được gán nhãn *yi* = −1;



Mặt siêu phẳng tách các mẫu dương khỏi các mẫu âm.

Trong trường hợp này, bộ phân loại *SVM là mặt siêu phẳng phân tách các mẫu dương khỏi các mẫu âm với độ chênh lệch cực đại*, trong đó *độ chênh lệch – còn gọi là* ***lề*** (margin) xác định bằng khoảng cách giữa các mẫu dương và các mẫu âm gần mặt siêu phẳng nhất (hình 1). Mặt siêu phẳng này được gọi là *mặt siêu phẳng lề tối ưu*.

Các mặt siêu phẳng trong không gian đối tượng có phương trình là **w***T***x** + *b* = 0, trong đó **w** là vector trọng số, *b* là độ dịch. Khi thay đổi **w** và

b, hướng và khoảng cách từ gốc tọa độ đến mặt siêu phẳng thay đổi. Bộ phân loại SVM được định nghĩa như sau:

$f(\mathbf{x}) = \text{sign}(\mathbf{w}^T\mathbf{x} + b)$ (1)

Trong đó

$\text{sign}(z) = +1$ nếu $z \geq 0$,

$\text{sign}(z) = -1$ nếu $z < 0$.

Nếu $f(\mathbf{x}) = +1$ thì **x** thuộc về lớp dương (lĩnh vực được quan tâm), và ngược lại, nếu $f(\mathbf{x}) = -1$ thì **x** thuộc về lớp âm (các lĩnh vực khác).

Máy học SVM là một họ các mặt siêu phẳng phụ thuộc vào các tham số **w** và *b*. Mục tiêu của phương pháp SVM là ước lượng **w** và *b* để cực đại hóa lề giữa các lớp dữ liệu dương và âm. Các giá trị khác nhau của lề cho ta các họ mặt siêu phẳng khác nhau, và lề càng lớn thì năng lực của máy học càng giảm. Như vậy, cực đại hóa lề thực chất là việc tìm một máy học có năng lực nhỏ nhất. Quá trình phân loại là tối ưu khi sai số phân loại là cực tiểu.

Nếu tập dữ liệu huấn luyện là *khả tách tuyến tính*, ta có các ràng buộc sau:

$\mathbf{w}^T\mathbf{x}_i + b \geq +1$ nếu $y_i = +1$ (2)

$\mathbf{w}^T\mathbf{x}_i + b \leq -1$ nếu $y_i = -1$ (3)

Hai mặt siêu phẳng có phương trình là $\mathbf{w}^T\mathbf{x} + b = \pm 1$ được gọi là các mặt siêu phẳng hỗ trợ (các đường nét đứt trên hình 1).

Để xây dựng một mặt siêu phẳng lề tối ưu, ta phải giải bài toán quy hoạch toàn phương sau:

Cực đại hóa:

$$\sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j y_i y_j x_i^T x_j$$

với các ràng buộc:

$$a_i \geq 0$$

$$\sum_{i=1}^{N} a_i y_i = 0$$

trong đó các hệ số Lagrange α *i*, *i* = 1, 2, ..., *N*, là các biến cần được tối ưu hóa. Vector **w** sẽ được tính từ các nghiệm của bài toán toàn phương nói trên như sau:

$$\mathrm{w} = \sum_{i=1}^{N} a_i y_i x_i$$

Để xác định độ dịch *b*, ta chọn một mẫu **x***i* sao cho với α*i* > 0, sau đó sử dụng điều kiện Karush–Kuhn–Tucker (KKT) như sau:

$$a_i[y_i(\mathrm{w}^{\mathrm{T}} x_i + b) - 1] = 0$$

Các mẫu $\mathbf{x}_i$ tương ứng với α*i* > 0 là những mẫu nằm gần mặt siêu phẳng quyết định nhất (thỏa mãn dấu đẳng thức trong (2), (3)) và được gọi là các *vector hỗ trợ*. Những vector hỗ trợ là những thành phần quan trọng nhất của tập dữ liệu huấn luyện. Bởi vì nếu chỉ có các vector hỗ trợ, ta vẫn có thể xây dựng mặt siêu phẳng lề tối ưu như khi có một tập dữ liệu huấn luyện đầy đủ.

Nếu tập dữ liệu huấn luyện không khả tách tuyến tính thì ta có thể giải quyết theo hai cách.

Cách thứ nhất sử dụng một *mặt siêu phẳng lề mềm*, nghĩa là cho phép một số mẫu huấn luyện nằm về phía sai của mặt siêu phẳng phân tách hoặc vẫn ở vị trí đúng nhưng rơi vào vùng giữa mặt siêu phẳng phân tách và mặt siêu phẳng hỗ trợ tương ứng. Trong trường hợp này, các hệ số Lagrange của bài toán quy hoạch toàn phương có thêm một cận trên *C* dương - tham số do người sử dụng lựa chọn. Tham số này tương ứng với giá trị phạt đối với các mẫu bị phân loại sai.

Cách thứ hai sử dụng một ánh xạ phi tuyến Φ để ánh xạ các điểm dữ liệu đầu vào sang một không gian mới có số chiều cao hơn. Trong không gian này, các điểm dữ liệu trở thành khả tách tuyến tính, hoặc có thể phân tách với ít lỗi hơn so với trường hợp sử dụng không gian ban đầu. Một mặt quyết định tuyến tính trong không gian mới sẽ tương ứng với một mặt quyết định phi tuyến trong không gian ban đầu. Khi đó, bài toán quy hoạch toàn phương ban đầu sẽ trở thành:

Cực đại hóa:

$$\sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} a_i a_j y_i y_j k(x_j, x_j)$$

với các ràng buộc:

$$0 \le a_i \le C$$

$$\sum_{i=1}^{N} a_i y_i = 0$$

trong đó *k* là một *hàm nhân* thỏa mãn:

$$k(x_i, x_j) = f(x_i)^T . f(x_j)$$

Với việc dùng một hàm nhân, ta không cần biết rõ về ánh xạ Φ. Hơn nữa, bằng cách chọn một nhân phù hợp, ta có thể xây dựng được

nhiều bộ phân loại khác nhau. Chẳng hạn, nhân đa thức $k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j + 1)^p$ dẫn đến bộ phân loại đa thức, nhân Gaussian $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma||\mathbf{x}_i - \mathbf{x}_j||2)$ dẫn đến bộ phân loại RBF (Radial Basis Functions), và nhân sigmoid $k(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\kappa \mathbf{x}_i^T \mathbf{x}_j + \delta)$, trong đó tanh là hàm tang hyperbol, dẫn tới mạng nơron sigmoid hai lớp (một lớp nơron ẩn và một nơron đầu ra). Tuy nhiên, một ưu điểm của cách huấn luyện SVM so với các cách huấn luyện khác là hầu hết các tham số của máy học được xác định một cách tự động trong quá trình huấn luyện. Để giải quyết vấn đề này thì có nhiều tác giả dùng các phương pháp khác nhau, nhưng trong giai đoạn gần đây thì thấy đa phần các tác giả sử dụng phương pháp tối ưu hóa tuần tự cực tiểu (*Sequential Minimal Optimization* - SMO) Thuật toán này sử dụng tập dữ liệu huấn luyện (còn gọi là *tập làm việc*) có kích thước nhỏ nhất bao gồm hai hệ số Lagrange. Bài toán quy hoạch toàn phương nhỏ nhất phải gồm hai hệ số Lagrange vì các hệ số Lagrange phải thỏa mãn ràng buộc đẳng thức (11). Phương pháp SMO cũng có một số heuristic cho việc chọn hai hệ số Lagrange để tối ưu hóa ở mỗi bước.

II.8.4.Huấn luyện SVM

Huấn luyện SVM là việc giải bài toán quy hoạch toàn phương SVM. Các phương pháp số giải bài toán quy hoạch này yêu cầu phải lưu trữ một ma trận có kích thước bằng bình phương của số lượng mẫu huấn luyện. Trong những bài toán thực tế, điều này là không khả thi vì thông thường kích thước của tập dữ liệu huấn luyện thường rất lớn (có thể lên tới hàng chục nghìn mẫu). Nhiều thuật toán khác nhau được phát triển để giải quyết vấn đề nêu trên. Những thuật toán này dựa trên việc phân rã tập dữ liệu huấn luyện thành những nhóm dữ liệu. Điều đó có nghĩa là bài toán quy hoạch toàn phương lớn được phân rã thành các bài toán quy hoạch toàn phương với kích thước nhỏ hơn. Sau đó, những thuật toán này kiểm tra các điều kiện KKT để xác định phương án tối ưu.

II.8.5.Phân loại văn bản

Để thực hiện quá trình phân loại, các phương pháp huấn luyện được sử dụng để xây dựng bộ phân loại từ các tài liệu mẫu, sau đó dùng bộ phân loại này để dự đoán lớp của những tài liệu mới (chưa biết chủ đề). Một số phương pháp *định trọng số từ* thông dụng:

1. **Tần suất từ** (term frequency - *TF*): Trọng số từ là tần suất xuất hiện của từ đó trong tài liệu. Cách định trọng số này nói rằng một từ là quan trọng cho một tài liệu nếu nó xuất hiện nhiều lần trong tài liệu đó.
2. **TFIDF**: Trọng số từ là tích của tần suất từ *TF* và tần suất tài liệu nghịch đảo của từ đó và được xác định bằng công thức

IDF = log(N / DF) + 1 (13)

trong đó:

N là kích thước của tập tài liệu huấn luyện;

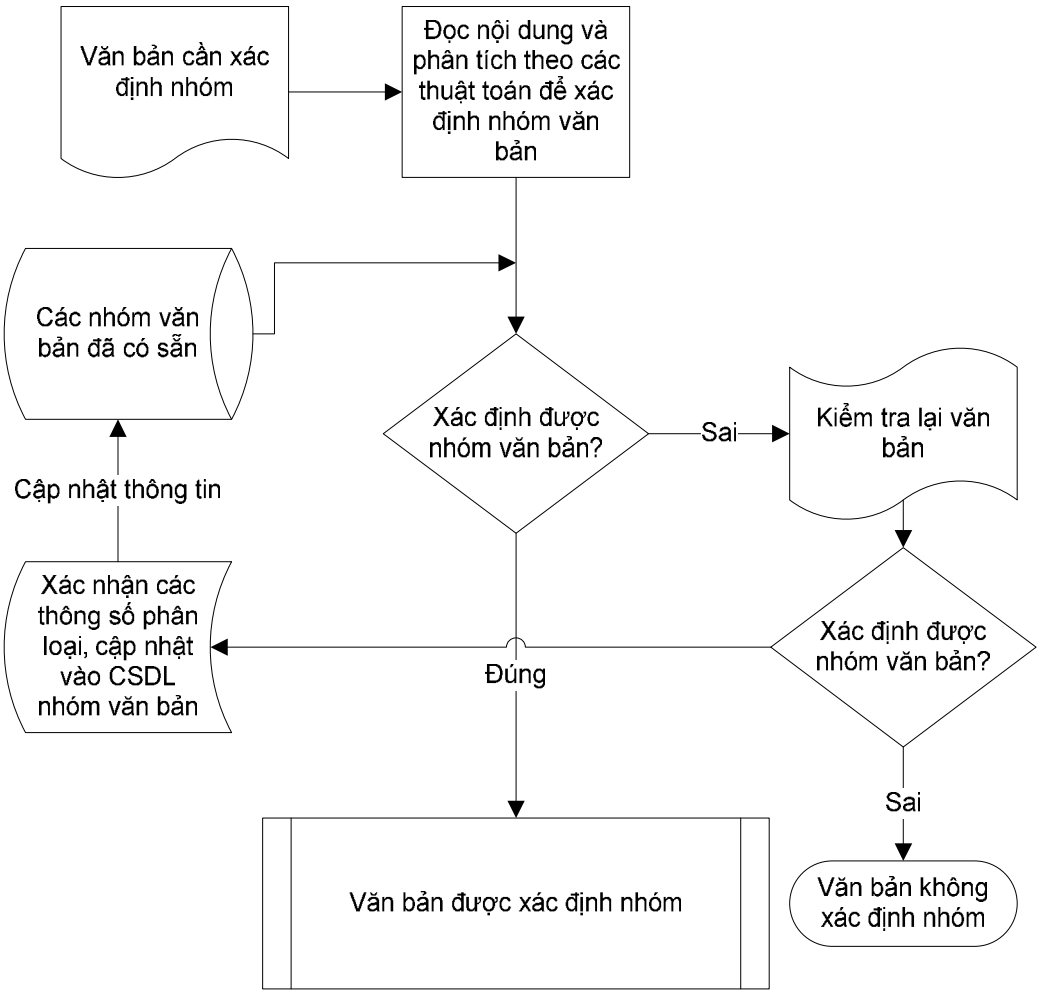
DF là tần suất tài liệu: là số tài liệu mà một từ xuất hiện trong đó.

Trọng số *TFIDF* kết hợp thêm giá trị tần suất tài liệu *DF* vào trọng số *TF*. Khi một từ xuất hiện trong càng ít tài liệu (tương ứng với giá trị *DF* nhỏ) thì khả năng phân biệt các tài liệu dựa trên từ đó càng cao.

CHƯƠNG III. MÔ TẢ BÀI TOÁN và XỬ LÝ BÀI TOÁN

III.1.Các yêu cầu đối với việc phân loại văn bản

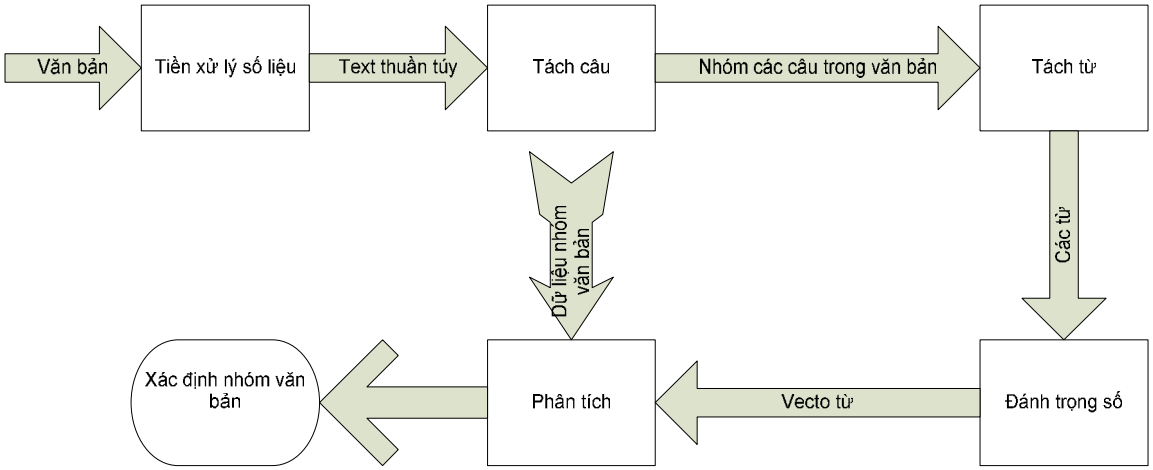
Yêu cầu chính của việc phân loại văn bản đó là việc xác định một văn bản sau khi xử lý sẽ xác định được văn bản đó thuộc nhóm văn bản nào trong các văn bản đã được xác định trước. Đối với các văn bản không thể xác định được hoặc văn bản có tính "nhập nhèm" thì chương trình cần phải chỉ ra và cho phép người sử dụng có thể xác định bằng tay văn bản này thuộc vào nhóm văn bản nào. Sau khi xác định thì kết quả này phải được cập nhật vào hệ thống nhận dạng để chương trình có thể nhận dạng được các văn bản tương tự lần sau



Vấn đề quan trọng ở đây đó là việc đọc nội dung và phân tích ngữ nghĩa để xác định loại văn bản, nếu phần việc này làm tốt thì việc phải xác định lại văn bản bằng nhân công sẽ giảm đi khá nhiều. Do đó có thể coi như bài toán phải giải quyết là công việc đọc nội dung và phân tích nội dung đọc được sau đó chọn thuật toán để đưa ra quyết định nhóm của văn bản được đọc

III.2.Cấu trúc chương trình

Dựa vào các nghiên cứu đã mô tả ở chương II và theo yêu cầu của bài toán như mô tả ở phần III.1, ta có thể thấy rằng để việc nhận dạng một văn bản được chính xác, cần phải thực hiện các bước sau:



III.2.1.Bước 1: Tiền xử lý số liệu

Mục đích của bước này là xử lý tương đối sạch dữ liệu đọc vào để các bước sau sẽ xử lý tốt hơn, do đó công việc của bước này sẽ chỉ là chuyển thành chuỗi ký tự thuần túy (text), do đó nó sẽ có yêu cầu như sau:

- Đầu vào: Tệp văn bản cần phải phân tích (File PDF, TXT, DOC, HTML, HTM)

- Đầu ra: chuỗi ký tự thuần túy (text only)

III.2.2.Bước 2: Tách câu:

Mục đích của bước này là tách một văn bản text thuần túy thành các câu

- Đầu vào: Chuỗi ký tự văn bản thuần túy
- Đầu ra: Vecto chứa các câu được tách trong văn bản

III.2.3.Bước 3: Tách từ:

Tách các từ từ các câu đã được lấy ra, từ ở đây là từ tiếng Việt do đó đây là điều phải lưu ý

- Đầu vào: Câu văn bản
- Đầu ra: Vecto chứa các từ có nghĩa trong câu

III.2.4.Bước 4: Gán nhãn từ loại – Đánh trọng số

Gán nhãn từ loại là định lượng các từ trong văn bản

- Đầu vào: Vecto các từ
- Đầu ra: Vecto chứa các từ đã được gán nhãn

III.2.5.Bước 5: Sử dụng thuật toán để phân loại văn bản cần đọc

Đây là bước chính yếu của chương trình

- Đầu vào: Vecto các từ, dữ liệu chuẩn của các nhóm văn bản
- Đầu ra: Xác định nhóm của văn bản

III.3.Các bước thực hiện trong chương trình

III.3.1.Tiền xử lý số liệu:

Nhiệm vụ: Đọc nội dung các tập tin số liệu cần đọc, chuyển các văn bản cần phải kiểm tra thành dạng text thuần túy, nghĩa là loại bỏ các

thành phần như ảnh, các tag (trong trường hợp trang web), các thông tin định dạng …

Để thống nhất khuôn dạng của văn bản thì tất cả các văn bản phải có cùng một phông chữ duy nhất, phông chữ được chọn là font Unicode, do đó trước khi thực hiện việc chuyển thành chuỗi ký tự (text) thì việc đầu tiên phải làm là chuyển tất cả các văn bản có font chữ khác với font chữ Unicode về thành font chữ Unicode. Do đa phần các tệp văn bản hiện nay đều đã sử dụng font unicode và việc nhận dạng font tiếng Việt sử dụng trong tệp văn bản là khá khó khăn do đó phần chuyển đổi này sẽ được làm bằng tay, có nghĩa là do người sử dụng tự quyết định, chương trình dùng để chuyển đổi từ các dạng font khác nhau sang font Unicode được đi kèm trong chương trình, tuy nhiên có thể dùng các chương trình chuyển dạng font khác như Unikey, Vietkey để chuyển từ các font khác về font Unicode

Các số liệu phải đươc làm sạch các thông tin không phải là text, các thông tin này có thể là hình ảnh, âm thanh, định dạng văn bản .v.v.. Việc tách này phụ thuộc vào từng kiểu tập tin dữ liệu đầu vào

- Nếu dữ liệu đầu vào là tệp văn bản dạng text (txt) thì lấy tất cả số liệu
- Nếu dữ liệu đầu vào là tệp văn bản dạng rich-text-box (rtf) thì số liệu lấy ra sẽ là dạng text do sử dụng control rft trong chương trình, control này sẽ có đầu vào là tên tệp .rft (có chứa đường dẫn) và đầu ra là dạng text thông thường
- Nếu dữ liệu đầu vào là tệp văn bản dạng MS word (doc) thì sẽ sử dụng Microsoft.Office.Core để chuyển đổi, với công cụ này việc chuyển đổi một file dạng Microsoft word sang text chỉ là một hàm
- Nếu dữ liệu đầu vào là tệp văn bản dạng PDF thì sẽ sử dụng control PDFbox để đọc và loại bỏ các thuộc tính không cần thiết cho chương trình như hình ảnh, âm thanh, định dạng và chỉ lấy giá trị text

- Nếu dữ liệu đầu vào là các tệp văn bản (htm) hay (html) thì việc loại bỏ các dữ liệu là loại bỏ các đoạn tag định dạng, các link liên kết, các link hình ảnh

Loại bỏ các thông tin định dạng trang Web

Các trang Web hiện nay được thiết kế theo chuẩn HTML bao gồm các thẻ (tag) định dạng cho các thành phần nội dung trong trang Web, ta có thể liệt kê theo nhóm như sau:

- Tag định dạng thông tin chung của trang Web: <TITLE>, <!DOCTYPE>, <HEAD>, <HTML>, …
- Tag phân vùng, chia dòng, chia cột, …:
, <DIV>, <BLOCKQUOTE>, <TABLE>, <TR>, <TD>, …
- Tag liệt kê đề mục: <LI>, <MENU>, <OL>, …
- Tag định dạng chữ, hiệu ứng: <FONT>, <B>, <I>, <A>, <MARQUEE>, <STRIKE>, …
- Tag xử lý: <SCRIPT>, <APPLET>, <CODE>, <FRAME>, <EMBED>, <STYLE>, <INPUT>, …

Các thông tin định dạng, xử lý này cần được loại bỏ, chỉ giữ lại những phần thông tin bằng lời mà trang Web muốn thông báo cho người xem.

Loại bỏ các vùng văn bản phụ không cần thiết

Sau khi đã loại bỏ những thông tin định dạng, thông tin xử lý và trích ra phần thông tin bằng lời, trong thông tin này vẫn còn những thông tin phụ, không cần thiết mà ta cần tiếp tục loại bỏ.

Trong trang web, ngoài thông tin chính của trang web, thường chứa nhiều thông tin phụ khác như: thông tin quảng cáo, thông báo phụ, các đề mục, menu, … Do đó, cần có một cách phù hợp để bỏ qua những phần nội dung không cần thiết và chỉ giữ lại phần nội dung chính để tách lấy các câu tạo tóm tắt. Nội dung bước này sẽ được trình bày chi tiết trong các phần sau.

Việc tách lấy các đoạn văn bản “tường thuật” (narrative) trong văn bản, bỏ qua các phần văn bản rời rạc như các đề mục, liên kết, … có thể

được thực hiện theo hai cách sau:

- **Cách 1**. Dùng các heuristic hoặc học máy để rút ra các luật tách lấy những phần văn bản "tường thuật"
- **Cách 2.** Áp dụng dùng đặc tính tính ngữ pháp để loại bỏ các phần văn bản không tạo thành câu, chẳng hạn những đoạn không có chứa động từ hoặc chứa nhiều hơn 4 từ không thể xác định từ loại

Làm sạch số liệu tiếp theo bao gồm:

- Loại bỏ các khoảng trắng nhiều hơn 1 khoảng trắng
- Các dấu xuống dòng
- Cách dòng trống
- Các ký tự lạ
- ….

III.3.2.Tách câu

Đoạn văn bản sẽ được duyệt tuần tự và sẽ được cho ngắt câu khi gặp các ký tự ngắt câu như "." (chấm), "!" (chấm than), "?" (chấm hỏi), với điều kiện: ký tự tiếp theo (có thể có các ký tự "khoảng trắng" ở giữa) là ký tự viết in.

Cách làm trên loại bỏ được các trường hợp không phải ngắt câu như:

- Dấu "." không phải là ngắt câu mà là dấu trong 1 chuỗi số. Có được điều này vì nếu là "dấu chấm" trong chuỗi số thì ký tự tiếp theo phải là số, không phải ký tự viết in.

- Dấu "." trong một loạt "dấu ba chấm" bên trong câu, chưa phải là cuối câu.

Lấy một số ví dụ:

- Đoạn văn bản*"Hôm nay là một ngày đẹp trời. Chúng ta sẽ đi cắm trại ngoài trời"* sẽ được ngắt ở giữa từ "trời" và từ "chúng" để thành hai câu.

- Đoạn văn bản *"Trong vườn có 1.200 cây các loại, trong đó đa số là cây ăn trái như cam, quýt, đào, lê, mận, … và một số cây cảnh như cau, tùng, …"* chỉ thuộc một câu.

Luật trên vẫn chưa đủ để phân biệt hết các trường hợp xuất hiện dấu chấm. Ta xử lý thêm cho các trường hợp có xuất hiện dấu chấm nhưng không tách câu như sau:

- Chuỗi *link*, hay địa chỉ Web (URL).

  - Dấu hiệu nhận diện: có chứa ký tự "." hay "/" và chứa một trong các chuỗi con sau (ở đây chỉ liệt kê một số chuỗi thông dụng trong các địa chỉ Web): "http", ".com", ".net", ".edu", ".vn", ".org", ".htm", ".html", ".asp", ".jsp", ".php", ".gif", ".jpg", ".bmp", ".pdf", ".ps", ".txt", ".exe", ".wav", ".m3u", ".mp3".

  - Ví dụ: http://www.vnuit.edu.vn

- Ký hiệu viết tắt : Danh sách các ký tự viết tắt được xử lý: "GS.", "PGS.", "TS.", "VS.", "TSKH.", "NCS.", "ThS.", "BS.", "NS.", "DS.", "YS.", "LS.", "KS.", "CN.", "GĐ.", "PGĐ.", "TP.", "Tp.", "KCN.".

- Các chuỗi có chứa nhiều dấu chấm liên tục, chẳng hạn

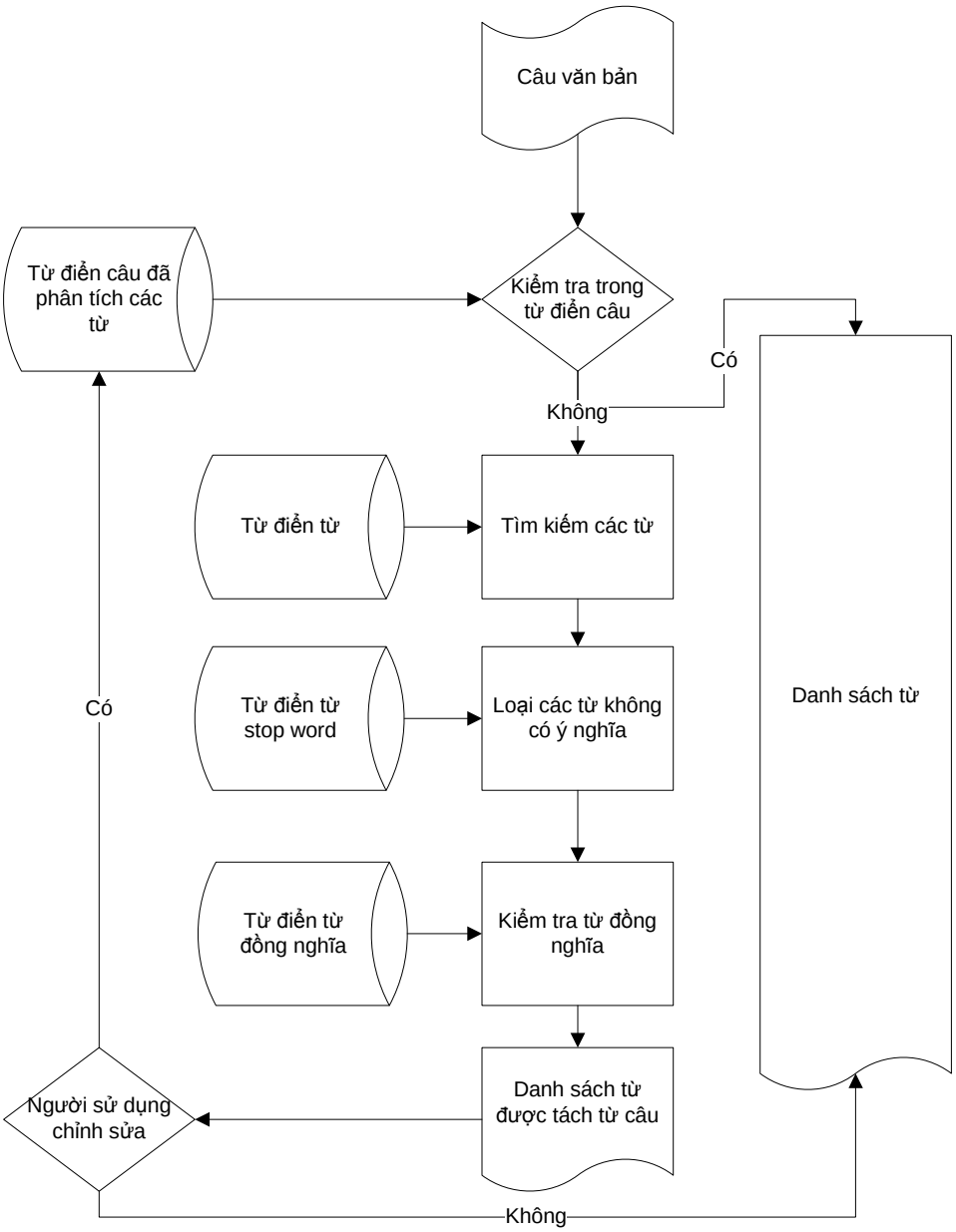
- Ø Chuỗi version (ví dụ: version 1.2.1). Chuỗi dạng này có chứa nhiều ký tự số.

- Ø Địa chỉ IP (ví dụ: 172.9.10.1). Chuỗi dạng này cũng chứa nhiều ký tự số.

- Ø Chuỗi định dạng cho một kiểu ghi nào đó (ví dụ: “version của chương trình này phải được ghi theo dạng Vx.x.x.x”).

III.3.3.Tách từ

Tách từ là vấn đề quan trọng nhất của chương trình, nó quyết định chương trình có thể thực hiện đúng và chính xác việc phân loại hay không là nhờ kết quả của việc tách từ đúng hay sai. Do đặc điểm tiếng Việt như đã trình bày ở trên (phần II.3) trong đó đặc biệt chú ý việc tiếng Việt không thể tách từ bằng khoảng trắng nên việc chọn phương pháp để tách từ cũng khá khó khăn. Như đã phân tích ở trên (phần II.4) việc chọn một phương pháp duy nhất có những cái hay và cái dở khác nhau do đó trong luận văn này sẽ chọn một phương pháp hỗn hợp để việc tách từ được tốt hơn. Phương pháp đó được trình bày như sau:



(a) Đối với một câu văn bản đưa vào sẽ kiểm tra trong dữ liệu có sẵn đã có mẫu câu này chưa, nếu đã có sẽ lấy các mẫu tách từ của mẫu câu này (sử dụng *Phương pháp giải thuật học cải biến - Transformation-based Learning- TBL*)

(b) Nếu chưa có mẫu câu này thì chương trình sẽ đọc chữ đầu tiên và xem tiếp chữ kế tiếp, nếu chữ đầu tiên và chữ kế tiếp có trong cơ sở dữ liệu

thì chương trình sẽ đọc chữ tiếp theo, cứ như vậy cho đến khi đọc chữ tiếp theo mà dãy chữ đó không có trong dữ liệu thì sẽ dừng lại và lấy từ là dãy chữ đã đọc được, tức là chương trình sẽ duyệt một ngữ hoặc câu từ trái sang phải và chọn từ có nhiều âm tiết nhất có mặt trong từ điển, rồi cứ thể tiếp tục cho từ kế tiếp cho đến hết câu (sử dụng *Phương pháp khớp tối đa (Maximum Matching) còn gọi là Left Right Maximum Matching (LRMM)*)

(c) Sau khi thực hiện xong bước (b) chương trình sẽ kiểm tra và loại bỏ các từ có tính chất kết nối, mô tả không có ý nghĩa trong câu (từ stop words) và chỉ giữ lại những từ có ý nghĩa nhất

(d) Các từ này trước khi được đưa vào phân tích cần phải qua bước kiểm tra từ đồng nghĩa nhưng khác âm như: túc cầu-bóng đá, địa cầu-trái đất .v.v. để qui tất cả các từ này về chung một mẫu thống nhất

(e) Trong trường hợp đang lấy mẫu thì sau khi thực hiện bước (b) chương trình sẽ đưa kết quả để người sử dụng có thể tự xác định xem việc tách từ là đúng hay sai, trong trường hợp người sử dụng phải xác định lại thì mẫu câu này sẽ được lưu trong dữ liệu để lần sau khi gặp mẫu câu này thì chương trình sẽ tự động tách như ở bước (a)

Đây là qui trình chung nhưng khi thực hiện thì để tăng tốc độ (khi kiểm tra từ theo Phương pháp khớp tối đa (Maximum Matching) thì tốc độ tìm kiếm khá chậm) thì sẽ có hai qui trình khác nhau:

- Qui trình khi thực hiện mẫu thử: thực hiện đúng như qui trình đã nêu ở trên trong đó:
    - Từ điển tiếng Việt là kết hợp từ hai từ điển: từ điển 78.000 từ của Đinh Điền và từ điển của chương trình Vikass (chương trình phân loại tin điện tử)
    - Từ điển các từ hư (stop word) cũng sử dụng của chương trình ViKass
    - Từ điển từ đồng nghĩa tự xây dựng
- Qui trình khi thực hiện chương trình phân loại văn bản thì để tăng tốc độ các bước sẽ hơi khác một chút như sau:

- Từ điển tiếng Việt sẽ là những từ được dùng để đánh giá phân loại văn bản kết hợp với từ điển đồng nghĩa
- Bỏ qua bước lại bỏ các từ hư (stop word) vì các hư từ thực ra là đã bị loại bỏ khi lấy các từ chỉ thuộc nhóm các từ mẫu (ở bước trên)

Phương pháp này thực ra không phải là phương pháp tốt nhất (ví dụ như so sánh với phương pháp giải thuật học cải biến hay mô hình tách từ bằng WFST và mạng Neural) nhưng nó có ưu điểm là nhanh, việc tách từ là do theo quan điểm của việc xử lý hạt nhân chuỗi (string kernels) thì "càng có nhiều chuỗi con chung thì chúng càng giống nhau" và các chuỗi con này là tập hợp từ các từ có nghĩa trong một câu do đó phương pháp hỗn hợp này tạm thời được sử dụng trong luận văn này

III.3.4.Gán nhãn – đánh trọng số

Việc gán nhãn – đánh trọng số là để lượng hóa các từ trong văn bản, nhờ việc lượng hóa này mà chương trình có thể xác định được văn bản đang chọn thuộc nhóm văn bản nào. Việc đánh nhãn cũng có tính chất quyết định đến kết quả phân loại văn bản. Việc gán nhãn – đánh trọng số sẽ được thực hiện hư sau: Các từ của văn bản sau khi đọc vào sẽ được sắp xếp vào một bange có các thông tin như sau:

- Từ đặc trưng
- Số lần xuất hiện trong văn bản: đây là số đếm số lần xuất hiện trong văn bản
- Khoảng cách lớn nhất: đây là khoảng cách tính bình quân gia quyền lớn nhất của khoảng cách của từ đặc trưng đến đầu câu (xem ví dụ ở dưới)
- Khoảng cách nhỏ nhất: đây là khoảng cách tính bình quân gia quyền nhỏ nhất của khoảng cách của từ đặc trưng đến đầu câu (xem ví dụ ở dưới)

- Khoảng cách trung bình: đây là khoảng cách tính bình quân gia quyền trung bình của khoảng cách của từ đặc trưng đến đầu câu (xem ví dụ ở dưới)

Ví dụ: Trong một văn bản có từ đặc trưng là “đặc trưng”

Trong một câu từ “đặc trưng” xuất hiện tại vị trí thứ 3

Từ đặc trưng	Số lần xuất hiện	Khoảng cách lớn nhất	Khoảng cách nhỏ nhất	Khoảng cách trung bình
Đặc trưng	1	3	3	3

Tại một câu khác từ “đặc trưng” xuất hiện ở vị trí thứ 5 khi đó công thức tính sẽ như sau:

Do khoảng cách này lớn hơn khoảng cách lớn nhất đã lưu do đó sẽ tính lại khoảng cách lớn nhất và khoảng cách trung bình

Khoảng cách lớn nhất (mới)=(1x3+1x5)/2=4

Khoảng cách trung bình (mới)=(1x3+1x5)/2=4

Số lần xuất hiện (mới)=1+1=2

Từ đặc trưng	Số lần xuất hiện	Khoảng cách lớn nhất	Khoảng cách nhỏ nhất	Khoảng cách trung bình
Đặc trưng	2	4	3	4

Tại một câu khác từ “đặc trưng” xuất hiện ở vị trí thứ 2 khi đó công thức tính sẽ như sau:

Do khoảng cách này nhỏ hơn khoảng cách nhỏ nhất đã lưu do đó sẽ tính lại khoảng cách nhỏ nhất và khoảng cách trung bình

Khoảng cách nhỏ nhất (mới)=(2x3+1x2)/3=2.66

Khoảng cách trung bình (mới)=(2x4+1x2)/3=3.33

Số lần xuất hiện (mới)=2+1=2

Từ đặc trưng	Số lần xuất hiện	Khoảng cách lớn nhất	Khoảng cách nhỏ nhất	Khoảng cách trung bình
Đặc trưng	3	4	2.66	3.33

Cứ như vậy cho đến khi hết văn bản, khi đó sẽ tính lại là khoảng cách lớn nhất và khoảng cách nhỏ nhất sẽ là giá trị trung bình cộng của hai khoảng cách này đến khoảng cách trung bình, như trong trường hợp trên thì khoảng cách trung bình sẽ là:

((4-3.33)+(3.33-2.66))/2= (4-2.66)/2=0.67

Khoảng cách lớn nhất = 3.33+0.67=4

Khoảng cách nhỏ nhất=3.33-0.67=2.66

Từ đặc trưng	Khoảng cách lớn nhất	Khoảng cách nhỏ nhất	Khoảng cách trung bình
Đặc trưng	4	2.66	3.33

Nhưng do khoảng cách của từ trong văn bản là một số nguyên dương nên khi đó chương trình sẽ lấy

- Giá trị lớn nhất là số nguyên dương nhỏ nhất lớn hơn hay bằng số khoảng cách lớn nhất
- Giá trị nhỏ nhất là số nguyên dương lớn nhất lớn hơn hay bằng khoảng cách nhỏ nhất

Từ đặc trưng	Khoảng cách lớn nhất	Khoảng cách nhỏ nhất	Khoảng cách trung bình
Đặc trưng	4	2	3.33

Như vậy là ta đã có bảng đánh trọng số của một từ trong văn bản, tuy nhiên ta cũng thấy rõ việc gán nhãn – đánh trọng số nếu thực hiện trên tất cả các từ có nghĩa trong văn bản thì sẽ dẫn đến việc vecto từ phổ biến trong văn bản sẽ có chiều rất lớn và điều này sẽ làm cho việc tính toán cần phải có máy tính rất mạnh (trong thực tê thì việc thực hiện trên các máy tính thông thường sẽ rất chậm). Để tăng tốc độ của chương trình thì phải giảm khối lượng của các vecto này, nhưng việc giảm các vecto này thì lại làm cho việc nhận dạng văn bản không được chính xác. Do đó để thực hiện được công việc đánh giá chính xác và giảm chiều của các vecto văn bản này thì tạm thời trong luận văn xác định như sau:

- Đối với việc huấn luyện thì với văn bản lấy mẫu sẽ lấy:
    - Môt văn bản chỉ lấy các từ có nghĩa được xuất hiện tương đối phổ biến nhất, nghĩa là các từ có nghĩa xuất hiện nhiều nhất (số lần xuất hiện nhiều nhất) do vậy khi lấy các từ làm mẫu thì chương trình sắp xếp số lần xuất hiện của các từ đặc trưng theo mức độ từ nhiều đến ít và chỉ lấy 1/3 tổng số từ (xét từ nhiều đến ít) và cách lấy là lấnhiều nhất
    - Nếu trong những từ đạc trưng được lấy ra như trên mà những từ có số lần xuất hiện ít hơn 1/3 so với từ có số lần xuất hiện nhiều nhất thì từ đó sẽ được loại bỏ
    - Chương trình chấp nhận sai số là nếu hai từ có số lần xuất hiện như nhau nhưng phải loại bỏ một thì chương trình sẽ bỏ ngẫu nhiên một từ và giữ lại một từ mà không quan tâm đến ngữ nghĩa của từ bị bỏ và từ giữ lại
- Đối với văn bản phân loại thì:
    - Chỉ lấy các từ có nghĩa trong danh sách các từ đặc trưng được chọn lựa của văn bản mẫu
    - Các từ có khoảng cách không giao với khoảng cách của từ trong danh sách mẫu từ đặc trưng sẽ bị loại bỏ, có nghĩa là các từ có khoảng cách lớn nhất và nhỏ nhất không nằm trong khoảng cách lớn nhất và nhỏ nhất của bất cứ từ mẫu đặc trưng nào thì sẽ bị loại bỏ khỏi danh sách từ dùng để phân loại

Với việc gán nhãn – đánh trọng số này thì các số liệu văn bản mẫu sẽ trở thành các vecto

$VM_i(M_0(x_0,y_0), M_1(x_1,y_1), M_2(x_2,y_2)\ldots\ldots M_m(x_m,y_m))$

và các văn bản cần phân nhóm sẽ chuyển thành các vecto:

$V_i(C_0(x_0,y_0), C_1(x_1,y_1), C_2(x_2,y_2)\ldots\ldots C_n(x_n,y_n))$

Trong đó:

- VM là vecto văn bản mẫu
- V_i là vecto văn bản cần phân loại
- M_i là ký tự đặc trưng thứ i của văn bản mẫu
- C_i là ký tự đặc trưng thứ i của văn bản cần phân loại, lưu ý $C_i \sqsubset M_j$ với {j=0,m}
- x_i là giá trị khoảng cách nhỏ nhất của từ đặc trưng thứ i trong văn bản
- y_i là giá trị khoảng cách lớn nhất của từ đặc trưng thứ i trong văn bản

Việc so sánh các vecto này bằng phương pháp Máy hỗ trợ vecto (Support vector Machine – SVM) sẽ cho kết quả là nhóm của văn bản cần phân loại gần với nhóm nào trong các mẫu nhất và từ kết quả đó sẽ biết văn bản cần phân loại thuộc nhóm văn bản nào

III.3.5.Huấn luyện

Tập tài liệu dùng làm mẫu được lấy là các trang web htm từ địa chỉ http://vietnamnet.vn/, để công việc được đơn giản và không mất quá nhiều thời gian thì luận văn coi như là các trang web này đã được phân loại chính xác và luận văn giả thiết như sau:

- Các tài liệu được phân lớp thành những phân nhóm tách biệt. *Trên thực tế, các tài liệu trên http://vietnamnet.vn/ được phân loại không chính xác. Các phân lớp tài liệu có sự giao thoa và do đó một tài*

liệu thuộc một phân lớp có thể có những đặc trưng thuộc một phân lớp khác

- Sự phân bố tài liệu trong một phân nhóm không ảnh hưởng tới sự phân bố tài liệu trong phân nhóm khác. *Giả thiết này được đặt ra để có thể chuyển bài toán phân loại nhiều phân lớp giao thoa thành các bài toán phân loại phân lớp tách biệt. .*

Các từ được dùng để biểu diễn các tài liệu cũng thường được gọi là các *đặc trưng*. Để nâng cao tốc độ và độ chính xác phân loại, tại bước tiền xử lý văn bản, ta loại bỏ các từ không có ý nghĩa cho phân loại văn bản. Thông thường những từ này là những từ có số lần xuất hiện quá ít. Tuy vậy việc loại bỏ những từ này có thể không làm giảm đáng kể số lượng các đặc trưng. Với số lượng các đặc trưng lớn bộ phân loại sẽ học chính xác tập tài liệu huấn luyện, tuy vậy nhiều trường hợp cho kết quả dự đoán kém chính xác đối với các tài liệu mới. Để tránh hiện tượng này, ta phải có một tập tài liệu mẫu đủ lớn để huấn luyện bộ phân loại. Tuy vậy, thu thập được tập mẫu đủ lớn tương ứng với số lượng đặc trưng thường khó thực hiện được trong thực tế. Do đó để cho bài toán phân loại có hiệu quả thực tiễn, cần thiết phải làm giảm số lượng đặc trưng.

Có nhiều phương pháp chọn đặc trưng hiệu quả. Ở đây, luận văn sử dụng phương pháp *lượng tin tương hỗ* . Phương pháp này sử dụng độ đo lượng tin tương hỗ giữa mỗi từ và mỗi lớp tài liệu để chọn các từ tốt nhất. Lượng tin tương hỗ giữa từ *t* và lớp *c* được tính như sau:

$$MI_{(t,c)} = \sum_{t\in\{0,1)} \sum_{c\in\{0,1)} P_{(t,c)} \log \frac{P_{(t,c)}}{P_{(t)}P_{(c)}}$$

trong đó:

P(t, c) là xác suất xuất hiện đồng thời của từ *t* trong lớp *c*;

P(t) là xác suất xuất hiện của từ *t* và

P(c) là xác suất xuất hiện của lớp *c*.

Độ đo MI toàn cục (tính trên toàn bộ tập tài liệu huấn luyện) cho từ *t* được tính như sau:

$$MI_{avg(t)} = \sum_i P_{(c_i)} MI(t, c_i)$$

Khi sử dụng các phương pháp chọn đặc trưng, ta có thể loại bỏ đi nhiều từ quan trọng, dẫn đến mất mát nhiều thông tin, điều đó làm cho độ chính xác phân loại sẽ giảm đi đáng kể. Trong thực tế, theo thí nghiệm của Joachims, rất ít đặc trưng không có liên quan, và hầu hết đều mang một thông tin nào đó, vì vậy một bộ phân loại tốt nên được huấn luyện với nhiều đặc trưng nhất nếu có thể. Tuy nhiên giải thuật SVM có khả năng điều chỉnh năng lực phân loại tự động đảm bảo hiệu suất tổng quát hóa tốt, thậm chí cả trong không gian dữ liệu có số chiều cao (số đặc trưng rất lớn) và lượng tài liệu mẫu là có hạn, chính vì vậy mà trong vấn đề chọn đặc trưng ta có thể quyết định cách chọn là lấy 1/3 số mẫu từ lấy được trong danh sách mẫu của tất cả các văn bản được chọn làm mẫu với điều kiện là số lần xuất hiện của từ ít nhất trong danh sách được chọn không được ít hơn quá 1/3 lần so với từ nhiều nhất được chọn. Với cách chọn như vậy thì số lượng mẫu sẽ ít để đảm bảo không gian vecto không quá lớn và vẫn đủ để phân loại văn bản một cách chính xác

III.3.6.Phân loại văn bản

Các số liệu kiểm thử cũng là các tệp htm và html được lấy trên địa chỉ http://vnexpress.net/Vietnam/Home/ và được nạp về (download) thành các tệp trên đĩa cứng, các phần được lấy ở các phần chỉ mục khá gần với số liệu mẫu

Như đã nói ở trên, việc huấn luyện được coi là các số liệu đã được phân loại tạm coi như là chính xác, các file đã được phân loại này cũng được coi như là đã được chuẩn hóa, các số liệu của các file này sẽ là các véc tơ mẫu để phân loại các văn bản sẽ được sử dụng sau

Văn bản sẽ thực hiện các qui trình như tiền xử lý-tách câu-tách từ như đã mô tả ở phần III.2, sau khi được tách từ sẽ được biểu diễn được biểu diễn dưới dạng vector với các thành phần (chiều) của vector này là các trọng số của các từ.

$V(C_0(x_0,y_0), C_1(x_1,y_1), C_2(x_2,y_2)\ldots\ldots C_n(x_n,y_n))$

Trong đó:

- V là vecto văn bản cần phân loại
- C_i là ký tự đặc trưng thứ i của văn bản cần phân loại
- x_i là giá trị khoảng cách nhỏ nhất của từ đặc trưng thứ i trong văn bản
- y_i là giá trị khoảng cách lớn nhất của từ đặc trưng thứ i trong văn bản

Ở đây, luận văn bỏ qua thứ tự giữa các từ cũng như các vấn đề ngữ pháp khác (theo lý thuyết về string kernels). Với mỗi số đặc trưng được chọn, các tài liệu được biểu diễn dưới dạng các vector thưa dùng cách định trọng số từ TFIDF. Mỗi vector thưa gồm hai mảng:

- Một mảng số nguyên lưu chỉ số của các giá trị khác 0 của ký tự đặc trưng, số này được lấy là từ số chỉ mục của từ đặc trưng mẫu trong cơ sở dữ liệu văn bản mẫu
- Một mảng số thực lưu các giá trị khác 0 tương ứng với ký tự này, nó là khoảng cách lớn nhất và nhỏ nhất của ký tự đặc trưng xuất hiện trong văn bản.

Sở dĩ dùng các vector thưa là do số từ xuất hiện trong mỗi tài liệu là rất nhỏ so với tổng số từ được sử dụng; điều này một mặt tiết kiệm bộ nhớ, mặt khác làm tăng tốc độ tính toán lên đáng kể.

Để thực hiện phân loại văn bản bằng phương pháp SVM, trong luận văn đã sử dụng các hàm trong thư viện http://www.csie.ntu.edu.tw/~cjlin/libsvm/, các hàm ở đây giúp cho việc tính toán các vecto của số liệu đọc được và các vecto của tập số liệu huấn luyện, các kết quả đưa ra sẽ ở dạng ma trận kết quả và chương

trình sẽ nhận dạng ma trận kết quả này để phân loại văn bản thành các nhóm văn bản như đã mô tả ở trên, đối với các kết quả cho kết quả không xác định rõ ràng thì chương trình sẽ đánh dấu là không xác định được và đưa ra tất cả các kết quả gần giống, người sử dụng sẽ tự xác định nhóm loại văn bản, nếu người sử dụng muốn sử dụng kết quả này dùng để huấn luyện thì các đặc trưng của văn bản này sẽ chuyển thành các mẫu lưu vào dữ liệu huấn luyện và làm cho dữ liệu huấn luyện đầy đủ hơn, giúp cho lần xử lý sau có kết quả khả quan hơn.

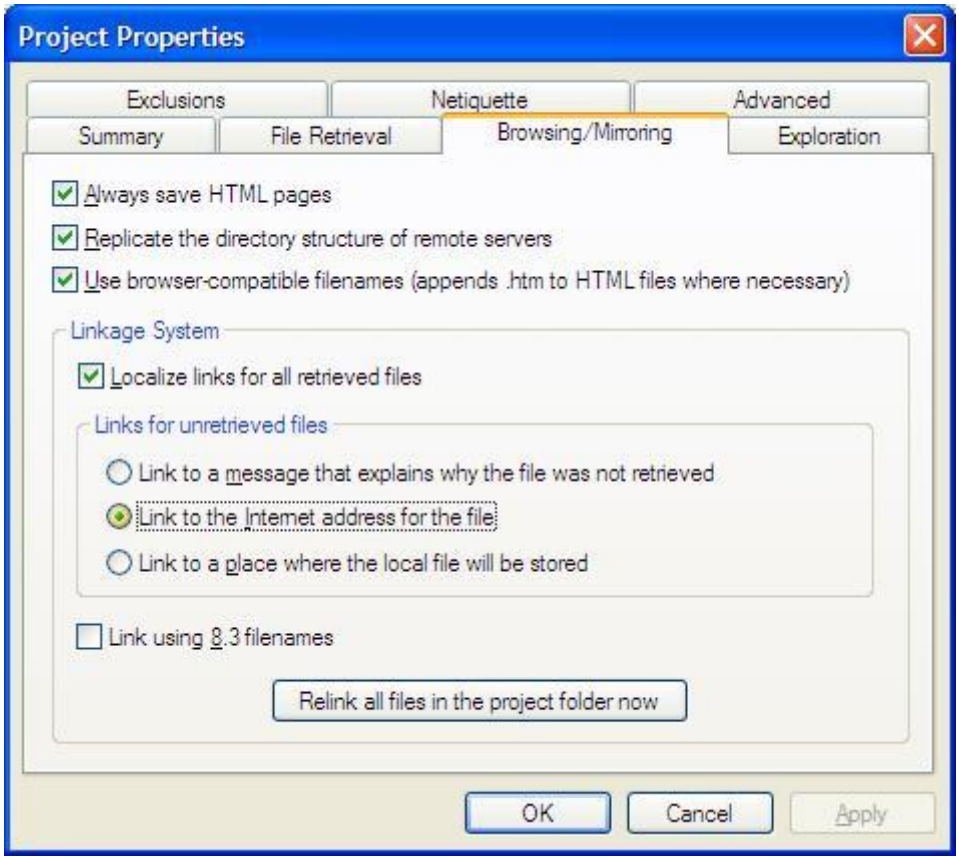
CHƯƠNG IV. CHƯƠNG TRÌNH THỬ NGHIỆM

Phương pháp sử dụng trong chương trình đã được mô tả ở chương II, cách thức thực hiện cũng được mô tả ở chương III, nên thuật toán sử dụng trong chương trình cũng tuân theo các cách thức và phương pháp đã mô tả

IV.1.1.Chuẩn bị số liệu

Các số liệu lấy mẫu sẽ được lấy về (download) từ trang web http://vietnamnet.vn/ bằng cách dùng chương trình Teleport pro được cung cấp ở địa chỉ http://tenmax.com/.

Khi lấy các trang web này về cần để ở chế độ “Replicate the directory structure of remote server” để cương trình có thể lấy tất cả các cấu trúc ở trên trang cần lấy về (download) và lặp lại cấu trúc đó trên máy lấy về, việc làm này là để việc phân loại văn bản được dễ dàng hơn



Sau khi lấy về các số liệu được tách thành các mục sau:

Mục	Số Tập tin	Tổng dung lượng (chỉ tính văn bản text) (bytes)
ChinhTri	1.374	28.383.744
CNTT	13.439	204.711.680
GiaoDuc	4.271	89.755.136
KhoaHoc	1.807	30.454.784
KinhTe	3.337	61.110.272
Cộng	**24.228**	**414.415.616**

Các số liệu đều là font Unicode nên không cần phải chuyển font

Các số liệu kiểm thử được lấy từ trang http://vnexpress.net/Vietnam/Home/ và cũng sử dụng chương trình Teleport pro và cũng để chế độ "Replicate the directory structure of remote server" để lấy cấu trúc giống như khi lấy mẫu

Từ điển tiếng Việt là một tệp trên đĩa cứng ở dạng văn bản (text), từ điển này được rút ra từ hai từ điển của Đinh Điền (77107 từ) và từ điển của chương trình Vikass (73901 từ) loại bỏ các từ trùng nhau giữa ai từ điển và được một từ điển sử dụng trong chương trình có 107773 từ

Từ điển từ không có nghĩa (từ hư - stop word) được lấy từ chương trình Vikass và có 805 từ

IV.1.2.Mô tả chương trình:

Chương trình sử dụng công nghệ .NET của Microsoft do vậy hỗ trợ hoàn toàn Unicode tiếng Việt và được viết trên nền Window (winform)

Các cơ sở dữ liệu sử dụng trong chương trình bao gồm: bộ từ điển cho chương trình, bộ từ điển của số liệu huấn luyện được dùng là XML để chương trình có thể thực hiện mà không phụ thuộc quá nhiều vào hệ thống thử nghiệm

Tất cả các số liệu được đính kèm vào trong thực thi (.exe) và các tập tin thư viện (.dll) do đó khi lần đầu sử dụng thì các số liệu liên quan sẽ được bung ra (ví dụ như các từ điển đã nói ở trên)

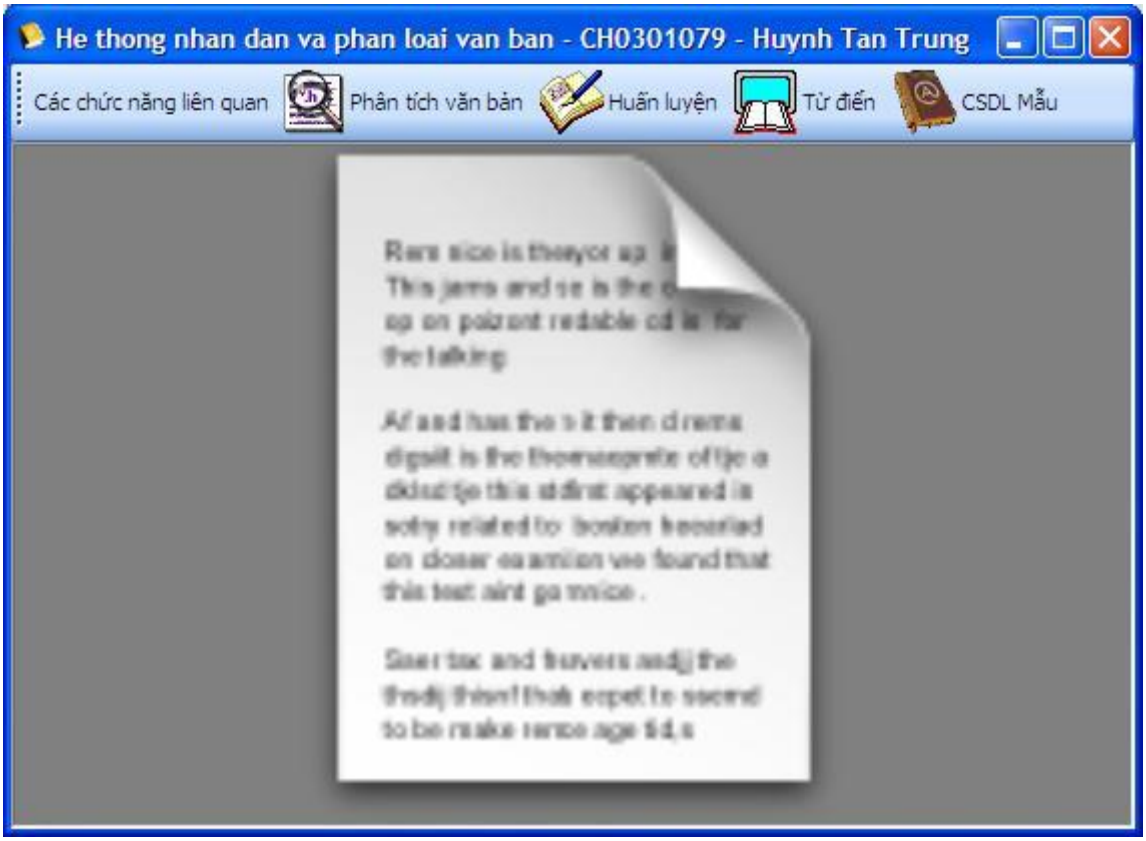
IV.1.1.Cài đặt

Để chương trình có thể thực hiện được thì yêu cầu tối thiểu để chương trình có thể thực hiện được là: máy tính phải được cài bộ .net framwork 1.1 và .net framwork 2.0 (được Microsoft cho sử dụng free)

Để có thể dùng được Unicode thì nên sử dụng các hệ điều hành hỗ trợ Unicode như Window 2000 hay WindowXP trở lên

Các số liệu kiểm thử cần phải tạo thành tập tin để trên máy hoặc trên mạng, chương trình chưa thực hiện kiểm tra online hoặc tạo thành một mô-đun đính lèm (plugin) để gắn vào chương trình khác nhưng việc sử dụng thuật toán này hoàn toàn có thể ứng dụng vào được các chương trình khác

IV.1.2.Một số giao diện của chương trình

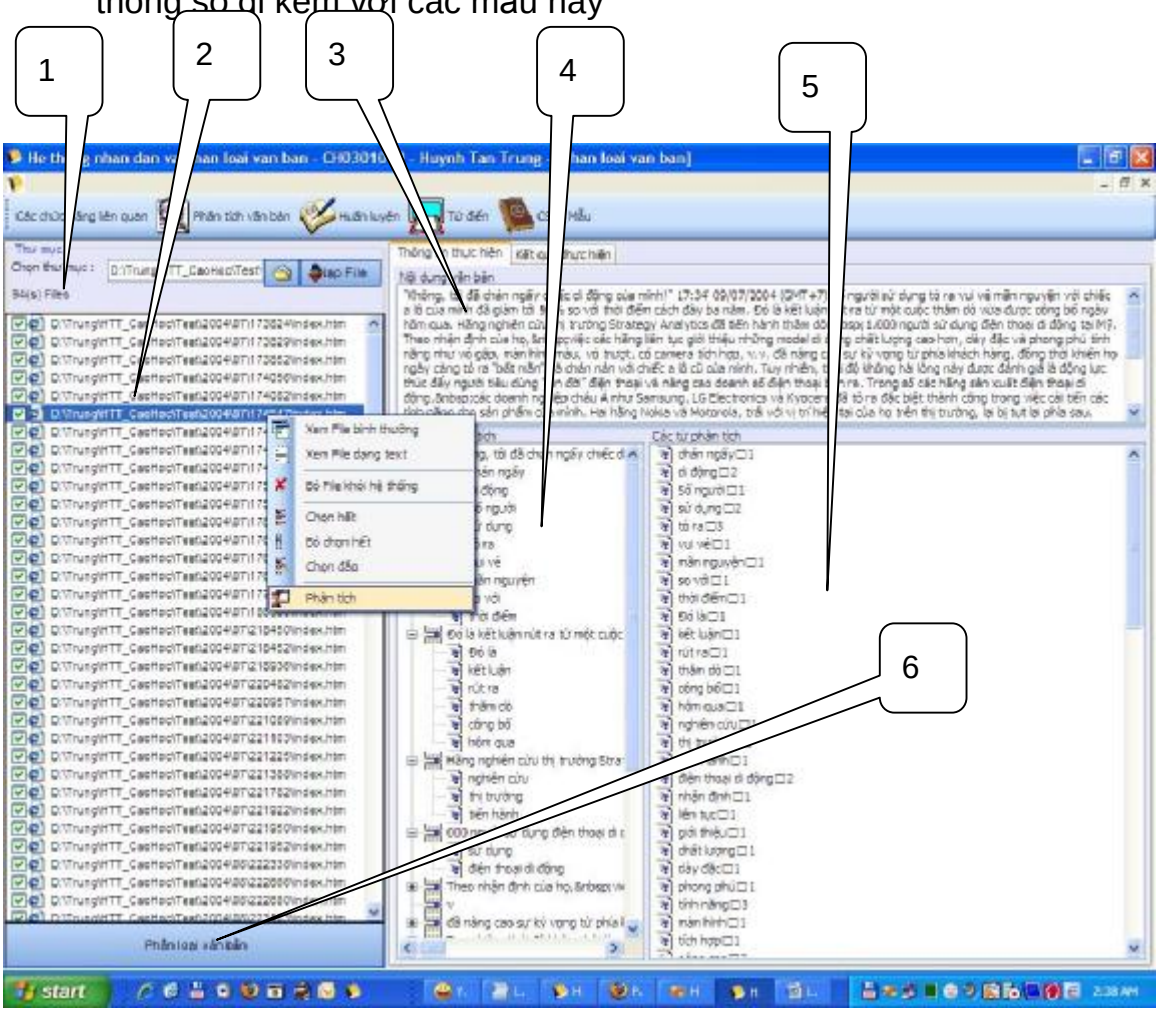


Hình 4.1.2.1.Màn hình làm việc chính gồm các chức năng:

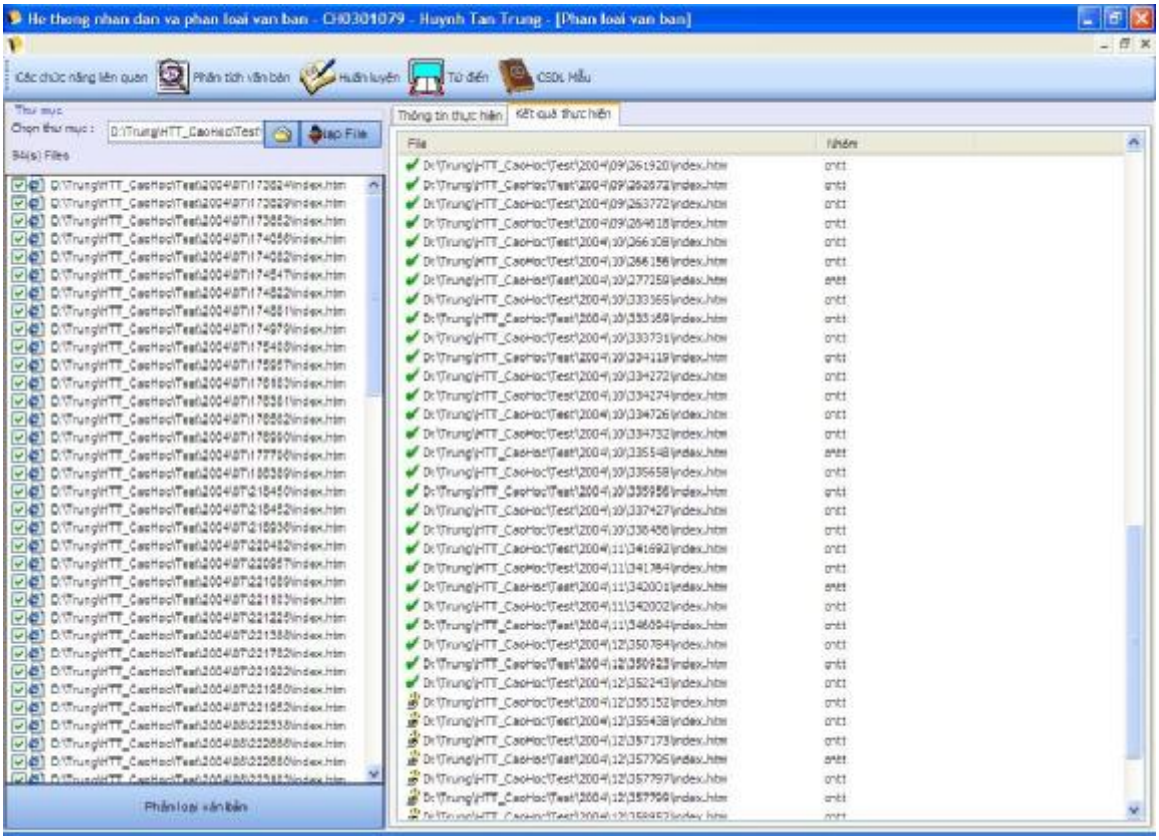
- Phân tích văn bản: đọc một văn bản hoặc nhiều văn bản và đưa ra kết quả là nhóm của văn bản (hoặc văn bản không thể xác định được nhóm)
- Huấn luyện: dùng để chạy tập tin mẫu và phân tích các đặc trưng, rút kết các đặc trưng cơ bản nhất đưa kết quả vào danh sách dùng

làm mẫu để phân tích các văn bản số liệu, đưa vào các nhóm văn bản theo mẫu

- Từ điển: từ điển tiếng Việt, từ điển từ không quan trọng (từ hư – stop word) và từ điển từ đồng nghĩa
- CSDL Mẫu: Cơ sở dữ liệu các mẫu cuẩn dùng để phan loại và các thông số đi kèm với các mẫu này



Hình 4.1.2.2.Màn hình phân tích số liệu: Phân tích số liệu bằng nhân công và có thể dùng để kiểm tra tính đúng của chương trình phân nhóm văn bản



Hình 4.1.2.3.Màn hình phân tích số liệu: Kết quả phân tích

Đây là phần làm việc chính của chương trình, phần này gồm hai phần: phần phân tích số liệu và phần kết quả phân nhóm. Hình 4.1.2.1 là màn hình phân tích gồm các phần sau:

Phần (1) là phần chọn thư mục, nơi chứa các tệp dữ liệu cần phân nhóm, sau khi chọn thư mục thì chọn phần nạp số liệu để nạp các tệp văn bản vào bộ nhớ, chuẩn bị cho việc phân tích. Ở đây việc chọn các tệp là lấy tất cả các tệp trong thư mục hiện hành và cả các tệp trong các thư mục con-cháu của thư mục hiện hành

Phần (2) là danh sách các tệp đã được chọn trong thư mục, trước khi phân tích người sử dụng có thể chọn lại/bỏ chọn và có thể loại bỏ các tệp văn bản không cần thiết, người sử dụng cũng có thể đọc nguyên dạng các tệp này hoặc chỉ đọc những đoạn văn bản (text only) của các

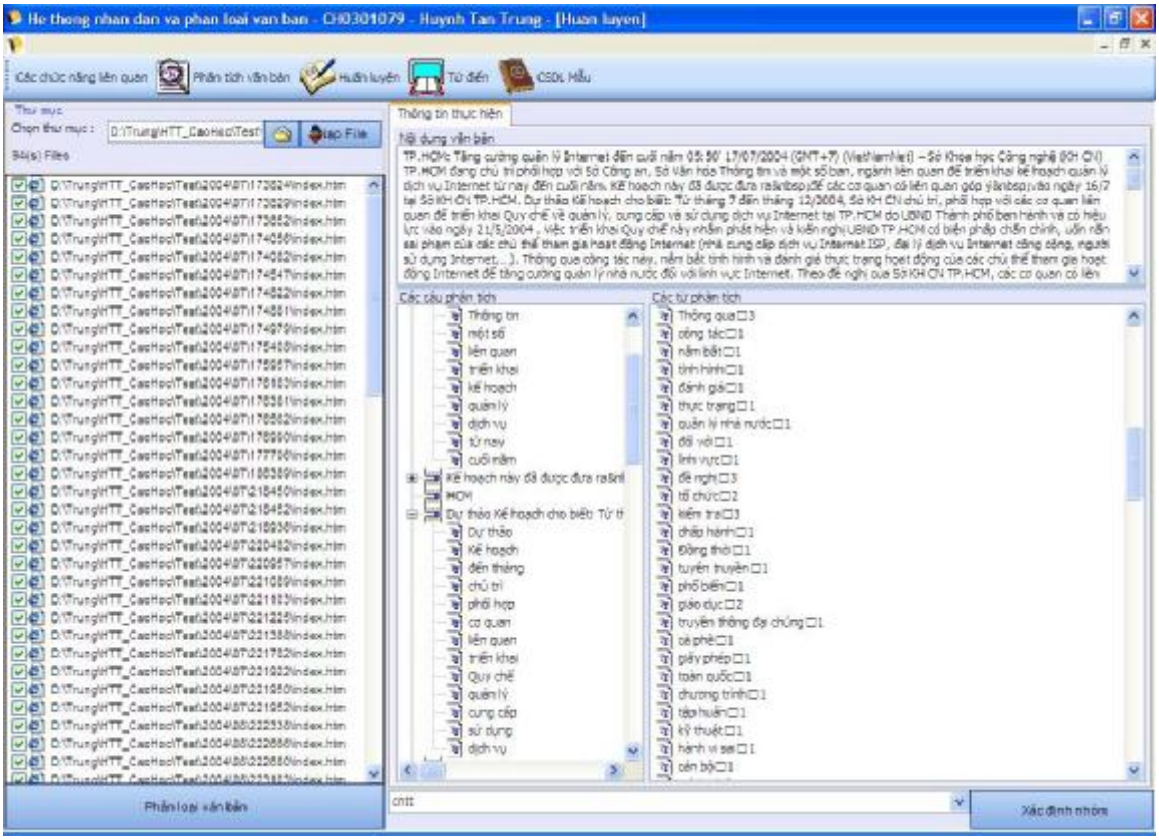
tệp này, người sử dụng cũng có thể phân tích một tệp ở được chọn để xem việc phân tích câu, tách từ và các thông số đặc trưng (các từ đặc trưng cho văn bản cần phân tích)

Phần (3) là nội dung của đoạn văn bản đọc được từ tệp cần đọc, đoạn này chỉ hiển thị là văn bản

Phần (4) là hiển thị danh sách các câu trong văn bản và ứng với mỗi câu là các từ được tách từ câu qua bước tách từ, phần này cũng giúp cho người sử dụng có thể xem để biết từ được tách đã chính xác và đúng ngữ nghĩa hay không

Phần (5) là danh sách tổng cộng các từ được tách từ văn bản, các từ này sẽ là các từ đặc trưng của văn bản cần tách, các từ này cũng có kèm theo các thông số vecto để người sử dụng có thể nhận xét và quyết định nhóm văn bản bằng tay (trong trường hợp chương trình không nhận diện được)

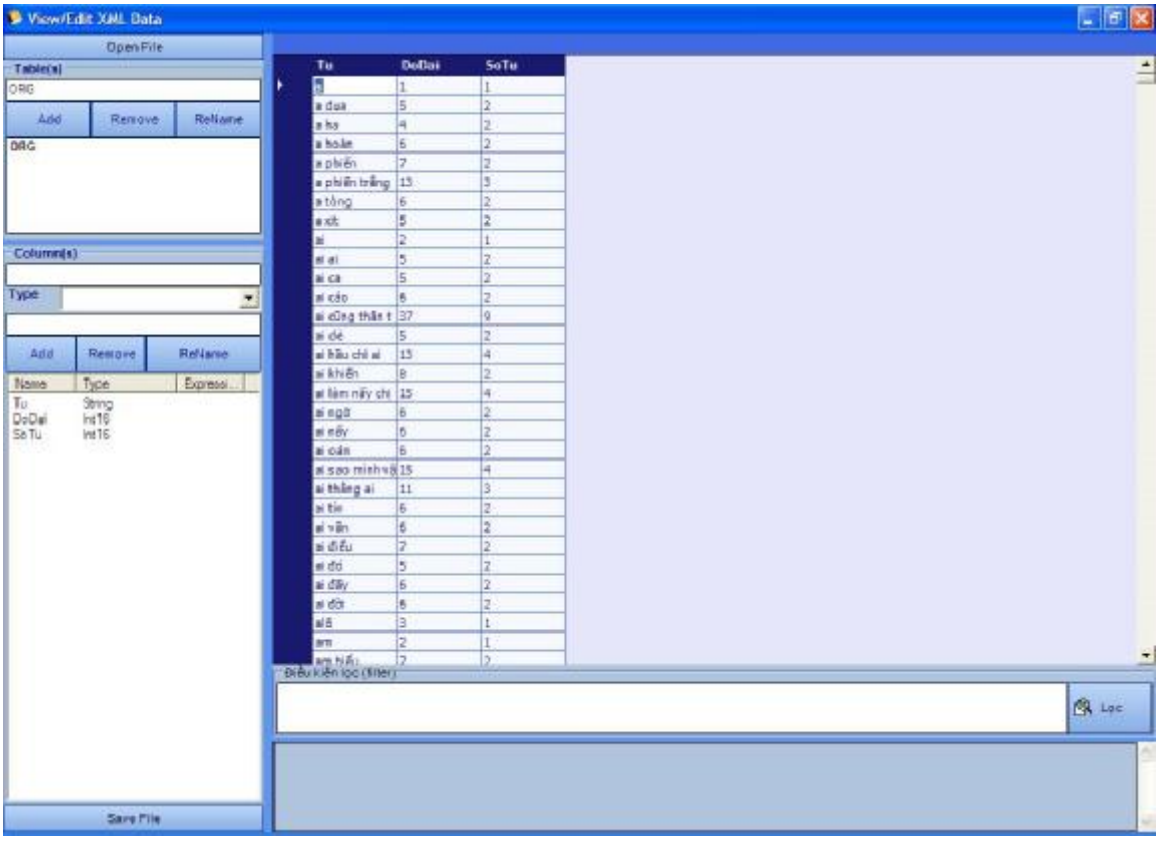
Phần (6) là nút dùng để thực hiện việc phân tích tất cả các tệp văn bản được đánh dấu chọn lựa, khi đó chương trình sẽ tự động thực hiên việc phân tích tất cả các tệp được chọn và sẽ hiển thị két quả phân tích ở hình 4.1.2.3



Hinh 4.1.2.4. Màn hình huấn luyện

Về cơ bản thì màn hình huấn luyện cũng gần giống như màn hình phân tích, tuy nhiên ở màn hình huấn luyện thì người sử dụng phải chọn lựa kiểu loại văn bản để cập nhật vào bảng lưu mẫu văn bản chuẩn

Lưu ý là ở phần huấn luyện thì từ điển được dùng sẽ là từ điển tiếng Việt còn trong pần phân tích thì từ điển được dùng chính là từ điển các từ mẫu trong nhóm phân loại



Hinh 4.1.2.5. Màn hình từ điển

Màn hình từ điển ở đây là màn hình từ điển tiếng Việt, cho phép người sử dụng có thế thêm bớt sửa xóa các thông tin trong từ điển

Các màn hình tương tự là màn hình từ điển từ đồng nghĩa và màn hình từ điển từ hư (stop word)

IV.1.3.Cài đặt

Để chương trình có thể thực hiện được thì yêu cầu tối thiểu để chương trình có thể thực hiện được là: máy tính phải được cài bộ .net framework 1.1 và .net framework 2.0 (được Microsoft cho sử dụng free)

Để có thể dùng được Unicode thì nên sử dụng các hệ điều hành hỗ trợ Unicode như Window 2000 hay WindowXP trở lên

Các số liệu kiểm thử cần phải tạo thành tập tin để trên máy hoặc trên mạng, chương trình chưa thực hiện kiểm tra online

IV.1.4.Các lưu ý khi chuẩn bị số liệu

Khi chuẩn bị các số liệu lấy mẫu thì việc quan trọng nhất là phải xem xét các tệp văn bản sẽ dùng làm mẫu, mặc dù luận văn đã coi như các số liệu được sắp xếp ở trang web làm mẫu là đã được sắp xếp đúng nhưng thực sự thì việc loại bỏ những bài không đúng là một việc cần phải làm và việc này về cơ bản vẫn phải do người thực hiện tiến hành.

Ví dụ như trường hợp phân loại CNTT thì xét các bài được phân tích thì bài ngắn nhất có nội dung

Mười điều không muốn nghe từ nhân viên hỗ trợ kỹ thuật

21:24' 09/07/2004 (GMT+7)

Nghề hỗ trợ kỹ thuật, hướng dẫn người sử dụng máy tính qua điện thoại thường gặp phải những tình huống "tréo ngoe", buộc họ phải có... quái chiêu khi gặp khách hàng... khó hỗ trợ.

10. Ông có cái... búa tạ hay cục gạch nào ở gần đó không ạ?

9. ... Vâng! Đúng rồi, thậm chí tôi cũng... không thể khắc phục được!

8. Thế... ông đang mặc cái gì vậy?

7. Gừ... Đồ vô công rồi nghề!

6. Chúng tôi có thể giúp ông khắc phục, nhưng ông cần phải có một con dao phết bơ, một cuộn băng dính và một bình ắc-quy xe hơi.

5. Tôi xin lỗi, Dave. Tôi sợ là tôi không thể làm điều đó.

3. Ông giữ máy một giây nhé... Ối mẹ ơi! Thằng Timmy đang đánh con!

2. OK, hãy lật tới trang 523 trong cuốn Cẩm nang Khoa học học (Dianetics) của ông.

1. Làm ơn giữ máy để gặp... luật sư của Bill Gates.

Địa chỉ thực của bài này là http://www.vnn.vn/cntt/itpark/2004/07/174595/, đọc bài ta có nhận xét là bài này không phải là bài về công nghệ thông tin mà chỉ là một câu chuyện cười do đó cần phải loại bỏ khỏi chuyên mục CNTT

Bài dài nhất là bài sau:

Nhiều ý kiến khác bày tỏ sự bất bình và đưa ra các ý kiến về việc, không chỉ FPT, mà hiện tại còn có nhiều nhà cung cấp dịch vụ ADSL khác tại Việt Nam cũng có tình trạng tương tự. Chất lượng dịch vụ thực tế cách quá xa với quảng cáo và không đảm bảo các thông số của "băng thông rộng"!

Để bạn đọc tiện theo dõi và có các ý kiến phản hồi để chúng tôi có thêm căn cứ tiến hành tìm hiểu thêm thông tin. Xin tạm chia các ý kiến phản hồi làm hai hướng: Phần bạn đọc đưa ra ý kiến về chất lương dịch vụ của ADSL FPT, phần thứ hai là ý kiến về chất lượng của các nhà cung cấp dịch vụ khác tại Việt Nam.

Không chỉ FPT - Các ISP khác cũng "có vấn đề"?

Ho ten: Dương Đức Thành

Dia chi: 2D22 TT Đại học Thủy Lợi

Email: thanhlan7479@yahoo.com

Tieu de: Không chỉ là ADSL của FPT

Noi dung:Tôi là người kinh doanh dịch vụ internet. Hiện nay tôi đang dùng đường truyền 2MB (MegaVNN - Maxi- Tốc độ tối đa 2Mbps/640Kbps) của VDC nhưng cũng bị hiện tượng như đường truyền của FPT. Lúc đầu download khoảng 250-300kb/s, nhưng hiện nay có thời điểm download là 15-60kb/s. Tôi có thắc mắc đến 800126 thì được trả lời là hãy kiểm tra xem may có bị nhiễm Vius không? Và chắc là do nhà tôi nối nhiều máy tính quá ?(10 máy). Điều này

là không đúng vì trước kia tôi vẫn sử dụng 10 máy tính để truy cập Internet. Tôi đã kiểm tra Virus thì đường truyền vẫn chậm, đấu thử vào duy nhất 1 máy cũng vẫn chậm như thường. Tôi nghi chắc VDC cũng giống như những gì mà ông Đình Anh nói " Không ai đưa ra các thông số tối thiểu hoặc các thông số tốc độ ổn định trong hợp đồng cả". Ngay như nhà cung cấp còn nói vậy thì chúng tôi biết nói gì? Thắc mắc với ai? Để được sử dụng theo đúng như quyền lợi của chúng tôi?!

Ho ten: thanhloi

Dia chi:

Email: thanhloi2003gl@yahoo.com

Tieu de:

Noi dung: Ở Gia lai chỉ có 1 đwờng truyền ADSL thuê bao của VNN, thuê qua bưu điện Gia lai, không riêng gì FPT, đường truyền VNN cũng vậy, chỉ có từ 10 đêm đến 6 h sáng hôm sau thì tương đối ổn định, bắt đầu từ 6 sáng trở đi rất chậm, thậm chí có lúc còn thua Dial up 1269. Vì mình là kinh doanh quán Net mà, hỏi mấy quán Net khác cũng chậm vậy. Không hiểu Bưu điện Gia lai làm ăn kiểu gì? Nếu như Gia lai có dịch vụ khác thì có lẽ bỏ đường truyền này rồi. Hỏi Bưu điện thì họ nói rằng không biết. Không biết quyền lợi khách hàng họ có đảm bảo không?

Ho ten: Hoang Cong Xuong

Dia chi: 115 nguyen Thien thuat,thi xa Hung yen,tinh Hung yen

Email: urethane@hn.vnn.vn

Tieu de: ve toc do truy cap ADSL

Noi dung: Toi thi khong xai FPT, ma xai VNPT o co quan va o nha rieng nhung so fan cung khong hon gi. Toi dong y voi cac ban la nha cung cap chi quang cao, con thuc te toc do chi dat 1/5-1/10 tham chi nhieu thoi diem khong ket noi duoc. Xem phim thuong bi dung hinh nhieu. Toi nghi quang cao phai di doi voi nang luc, do do tien thue bao thuc te cung khong re!

……*Bài còn dài nhưng chỉ đưa lên một đoạn để mô tả*…….

Địa chỉ thực của bài này: http://www.vnn.vn/cntt/2005/11/512827/

Bài này thực sự dài hơn bài đưa lên đây nhưng rõ ràng bài này có mấy vấn đề:

- Tuy cũng đề cập đến CNTT nhưng là sự trả lời (FAQ) về mạng ADSL và không có nhiều đặc trưng về một bài viết CNTT
- Trong bài có rất nhiều đoạn văn bản sử dụng tiếng Việt không dấu do đó việc phân tích sẽ loại bỏ hoàn toàn các từ tiếng Việt này do đó thực tế là số lượng từ đặc trưng rất ít

Do đó những bài này cũng cần phải loại bỏ

Ví dụ một bài ở khoảng giữa (sắp xếp theo thứ tự kích cỡ tệp từ cao đến thấp)

Microsoft hoãn phát hành Windows Vista đến 1/2007

10:38' 22/03/2006 (GMT+7)

Gã khổng lồ phần mềm dự định hoãn phát hành Windows Vista phiên bản người dùng đến tận đầu năm sau, thay vì mục tiêu nửa cuối năm 2006 như trước đây.

Nguồn: CNET

Tuy nhiên, Microsoft vẫn cam kết sẽ phát hành phiên bản Vista dành cho khách hàng doanh nghiệp ngay trong tháng 11 tới đây. Giá cổ phiếu của Microsoft sau thông tin này đã lập tức sụt gần 3%.

Ban đầu, Vista, bản nâng cấp Windows đáng kể nhất kể từ sau Windows XP (ra mắt cách đây 5 năm), được kỳ vọng sẽ kịp ra mắt ngay trong năm 2005. Thế nhưng Microsoft đã nhiều lần hoãn lên hoãn xuống, từ đầu năm 2006 sang cuối năm 2006 và giờ là sang hẳn năm 2007.

Việc hoãn phát hành Vista từ 8-10 tuần này, theo phân tích của hãng nghiên cứu Gartner, có thể ảnh hưởng đến toàn bộ ngành công nghệ, từ các hãng sản xuất máy tính cho đến hãng chip, các hệ thống phân phối, giới nghiên cứu và cả các nhà đầu tư.

Thị trường chứng khoán, do lo ngại về tác động của sự kiện này lên tình hình tiêu thụ máy tính, đã chứng kiến sự hạ giá dây chuyền của Intel, HP và Dell.

Trong suốt thời gian qua, Windows Vista luôn chiếm giữ vị trí "minh tinh", đi trước dẫn đường cho hàng loạt sản phẩm và dịch vụ (vừa hoặc sắp công bố) của Microsoft, từ thiết bị chơi game thế hệ mới Xbox 360 cho đến phần mềm Office mới.

Sự thành công của những phần mềm mới này có ý nghĩa sống còn đối với Microsoft trong việc duy trì và củng cố địa vị của hãng trên thị trường phần mềm, khi mà cổ phiếu Microsoft gần như dậm chân tại chỗ trong suốt 5 năm qua.

Windows, hệ điều hành nắm giữ tới 90% máy tính để bàn toàn cầu, chính là con gà đẻ trứng vàng số một của Microsoft. Chính vì lẽ đó, việc Microsoft buộc phải dời lại ngày phát hành Vista đã khiến không chỉ hãng này, mà hàng loạt công ty công nghệ khác rơi vào tình trạng hoang mang. "Chúng tôi không nghĩ ra được nên làm việc gì vào lúc này nữa. Thôi đành mặc đến đâu thì đến vậy".

Về phần mình, Microsoft lý giải nguyên nhân của sự chậm trễ này là do họ muốn nâng cao chất lượng Vista lên hơn nữa, nhất là ở lĩnh vực bảo mật, vả lại bản thân các hãng chế tạo máy tính cũng không muốn, do một phiên bản hệ điều hành mới tung ra ngay mùa mua sắm sẽ gây nên tình trạng bất ổn định trên thị trường.

Những "mông má" mới cho Windows

Bên cạnh tính năng bảo mật được siết chặt, Hệ điều hành Windows mới sẽ có một giao diện mới với scrolling 3 chiều giữa các cửa sổ. Những cửa sổ này có thể hiển thị trong suốt để người dùng xem được các thông tin bên dưới.

Chưa hết, nó còn có thể phát cũng như thu truyền hình phân giải cao ngay trên máy tính, và cho phép người dùng tìm kiếm các tài liệu lưu trong ổ cứng cũng

như trên mạng Internet một cách dễ dàng, hiệu quả hơn.

Cũng theo các thông tin trước đây, thì Microsoft dự định sẽ phát hành tới 8 phiên bản Vista khác nhau, nhắm đến những đối tượng sử dụng máy tính khác nhau, thay vì phân chia theo thông số phần cứng như cách làm truyền thống.

Địa chỉ gốc http://www.vnn.vn/cntt/2006/03/552699/, bài này cho thấy là một trang thông tin về CNTT thông tin đáp ứng nhu cầu.

Cũng tương tự như vậy với các trang về GiaoDuc, ChinhTri .v.v. do vậy phương pháp lấy mẫu sẽ có hai cách:

- Do người sử dụng chọn trước và xác định kiểu
- Chương trình sẽ lấy các tệp trong thư mục định sẵn là thuộc nhóm nào, nhưng sẽ không sử dụng các tệp quá lớn hoặc quá nhỏ

Trên thực tế thì chương trình dùng các lấy các tệp trong cùng thư mục đã chọn và chỉ lấy số lượng tệp khoảng ½ số tệp trong thư mục, loại bỏ hoàn toàn (không lấy làm văn bản mẫu) các tệp mà dung lượng quá lớn và/hoặc quá nhỏ hoặc cách khoảng giữa (của hai dung lượng) quá ½ khoảng cách đến hai đầu (lớn/nhỏ), như vậy với cách lấy này thì tuy số lượng tệp làm mẫu ít đi nhưng mức độ đặc trưng của mẫu lại tăng lên

Về các từ lấy làm mẫu đặc trưng thì về mặt lý thuyết thì các từ có nghĩa có số lượng xuất hiện nhiều nhất sẽ có ý nghĩa hơn các từ xuất hiện ít hơn, tuy nhiên khi tiến hành chương trình thì các từ đặc trưng mà đơn âm tiết (chỉ có một từ nói duy nhất) thì xuất hiện khá nhiều nhưng không tạo được đặc trưng cần thiết do vậy về mặt thực tế chương trình sẽ ưu tiên lấy theo thứ tự sau:

- Các từ có nhiều hơn một từ nói sẽ được ưu tiên cao hơn
- Các từ có số lần xuất hiện nhiều hơn sẽ được ưu tiên

Tuy nhiên số lượng từ đặc trưng được lấy không quá 1/3 toàn bộ số từ đặc trưng lấy được từ văn bản và trong các từ đặc trưng được lấy thì số lần xuất hiện của từ đặc trưng ít nhất không được ít hơn số lần xuất hiện của từ đặc trưng xuất hiện nhiều nhất quá 1/3 lần

Do chương trình không sử dụng các cơ sở dữ liệu chuyên biệt như MS SQL Server hay MS Access để làm nơi lưu trữ các số liệu tạm thời mà dùng việc lưu trữ XML làm cơ sở dữ liệu nên các công việc tìm kiếm, lọc dữ liệu sẽ có tốc độ chậm hơn các cơ sở dữ liệu chuyên biệt do vậy mà thuật toán cần phải cố gắng rút gọn và hạn chế đến mức thấp nhất việc liên tục tìm kiếm, lọc dữ liệu; trong chương trình thì có ba lần tìm kiếm để phân tách từ có nghĩa trong một câu, đó là:

- Lọc từ có nghĩa trong từ điển
- Lọc từ đồng nghĩa nhưng khác âm
- Loại bỏ các từ vô nghĩa (không có giá trị đặc trưng – từ hư – stop word)

Thì chương trình sẽ rút gon ba bước này trước khi phân tách từ trong câu bằng cách:

- Gán các từ đồng nghĩa nhưng khác âm vào trong từ điển từ để tìm từ đúng
- Loại bỏ các từ trong từ điển nếu từ đó tồn tại trong từ điển từ hư (stop word)
- Các từ trong từ điển cần phải tạo chỉ mục (index) để tăng tốc độ tìm kiếm
- Các từ tìm kiếm sẽ được tìm từ hai từ trở lên trước với phương pháp khớp tối đa Maximum Matching: forward/backward

IV.1.5.Kết quả thử nghiệm

Chương trình thử nghiệm sau khi đã huấn luyện sẽ thử nghiệm nhận dạng các tệp văn bản html được lấy về từ trang http://vnexpress.net/ trong các thư mục như sau

Mục	Số Tập tin	Tổng dung lượng (chỉ tính văn bản text) (bytes)
The-Gioi	137	5.130.498
Vi-tinh	297	25.539.072
Xa-Hoi	134	6.181.187
Kinh-Doanh	38	1.290.156

Để có thể tổng kết được tạm thời coi như số liệu kiểm tra cũng đã được sắp xếp chuẩn hóa, do đó khi kiểm tra mục (Xã hội) tạm so sánh với mục ChinhTri trên mẫu chuẩn.

Kết quả như sau:

Nội dung	Số lượng tập tin
Số tệp văn bản phải phân tích	134
Số tệp nhận dạng là chính trị	87
Số tệp không xác định được kiểu là	10
Số tệp nhận dạng là khác kiểu chính trị	37

Sau khi có kết quả lần 1 kiểm tra lại các tệp nhận dạng khác kiểu và không nhận dạng được kiểu thì xác định được:

Nội dung	
Số tệp văn bản không nhận dạng được nhưng có nội dung là chính trị	2
Số tệp nhận dạng kiểu khác nhưng xác định là chính trị	15

Như vậy tổng cộng có số tệp đúng loại là 87+2+15=104 tệp

Nhận dạng được 87 tệp/104 =83,65%

Cũng tương tự như vậy đối với các dạng văn bản khác

Tổng kết tất cả các dạng văn bản thì có được kết quả khoảng 82,63%

Nhận xét về kết quả:

Kết quả như khi thực nghiêm là chưa cao đặc biệt nhưng nó cũng không kém các phương pháp khác quá nhiều, đặc biệt là trong việc phân loại văn bản tiếng Việt, điều này có thể hoàn toàn giải thích được do các nguyên nhân sau:

- Các mẫu chuẩn lấy từ một trang web và được coi như sự phân loại sắp xếp đó là chính xác, trong khi thực sự thì các văn bản này chắc chắn có sự sắp xếp không chuẩn xác như mong đợi
- Các mẫu dùng làm chuẩn chỉ được coi như là chuẩn và cố gắng chuẩn ở mức độ nào đó (chỉ lấy các tệp văn bản ở khoảng giữa của tất cả các văn bản được coi là chuẩn) nên số liệu chuẩn cũng không đúng đắn
- Các số liệu đặc trưng của văn bản chủ yếu là dùng phương pháp thống kê do đó chắc chắn sẽ có trường hợp số liệu là đặc trưng nhưng không thật sự là đặc trưng cho thể loại văn bản, do đó nếu có phương pháp nào khác để xác định độ đặc trưng chính xác hơn thì kết quả chắc chắn sẽ cao hơn

CHƯƠNG V. KẾT LUẬN

Tác giá đã xây dựng được một chương trình phân loại văn bản, tuy thời gian còn hạn chế nên các tính năng tiện dụng của chương trình chưa cao nhưng chương trình đã được sử dụng các lý thuyết mới nhất và hiện đang được áp dụng khá nhiều trong thực tế, đó là các lý thuyết về hạt nhân chuỗi – string kernels, hỗ trợ vecto (Support vector Machine - SVM) do đó về mặt lý thuyết chương trình đã có những bước tiến nhất định, mặc dù kết quả không thật sự là nổi trội hơn các chương trình khác tương tự nhưng với số liệu dùng để huấn luyện chưa được chuẩn hóa nên vẫn còn có sự nhập nhèm thì kết quả này là hoàn toàn chấp nhận được. Đối với các loại văn bản có sự tách biệt rõ ràng như văn bản về CNTT hay văn bản về kinh tế thì mức độ nổi trội rõ ràng.

Qua chương trình này ta cũng thấy để tăng hiệu quả của một chương trình thì việc ứng dụng lý thuyết là phải ứng dụng nhiều lý thuyết kết hợp với nhau để tăng hiệu quả

Các hướng cải tiến chương trình:

- Sử dụng phương pháp phân tích câu có mức độ chính xác hơn
- Số liệu số liệu mẫu nhiều hơn thì có thể đảm bảo độ chính xác cao hơn
- Số liệu mẫu khi lấy vào cần phải có chọn lọc chính xác, tránh sự nhập nhèm giữa các mẫu thử làm kết quả bị hạn chế

Các hướng nghiên cứu trong tương lai

- Bổ xung thêm bộ phân tích ngữ nghĩa tiếng Việt để tăng mức độ chính xác
- Nghiên cứu thêm các thuật toán để bổ xung cho CSDL phân tích
- Nghiên cứu thêm cơ chế dùng kiểm tra được các trang web trên mạng để hỗ trợ cơ chế tìm kiếm và phân loại trực tuyến

Ứng dụng trong thực tế

Chương trình đang bắt đầu được triển khai để sử dụng cho việc tìm kiếm và phân loại của Trung Tâm Văn Thư lưu trữ của UBND tỉnh Bà Rịa – Vũng Tàu

CHƯƠNG VI. TÀI LIỆU THAM KHẢO

Tiếng Anh

[1]. Amitay E. and C. Paris (2000), “Automatically summarising web sites - is there a way around it?”, *ACM 9th International Conference on Information and Knowledge Management*.

[2]. Aone C., M. E. Okurowski, J. Gorlinsky, and B. Larsen (1997), “A scalable summarization system using robust nlp”, *Proceeding of the ACL'97/EACL'97 Workshop on Intelligent Scalable Text Summarization*, p.66-73.

[3]. Barzilay R., and M. Elhadad (1997), “Using lexical chains for text summarization”, *Proceedings of the Intelligent Scalable Text Summarization Workshop (ISTS'97),* ACL, Madrid, Spain.

[4]. Buyukkokten O., H. Garcia-Molina, and A. Paepcke (2001), “Seeing the whole in parts: Text summarization for web browsing on handheld devices”, *Proceedings of 10th International World-Wide Web Conference*.

[5]. Cavnar William B. (1994), “Using An N-Gram-Based Document Representation With A Vector Processing Retrieval Model”, *NIST Special Publication 500-225: Overview of the Third Text Retrieval Conference (TREC-3)*, p. 269-278, NIST.

[6]. Delort -Y. J., B. Bouchon-Meunier, and M. Rifqi (2003), “Enhanced Web Document Summarization Using Hyperlinks”, *under submission*.

[7]. Dinh Dien, Hoang Kiem, Nguyen Van Toan (2001), “Vietnamese Word Segmentation”, *Proceedings of the Sixth Natural Language Processing Pacific Rim Symposium (NLPR2001)*, p. 749-756, Tokyo.

[8]. Goldstein J., M. Kantrowitz, V. Mittal, and J. Carbonell (1999), “Summarizing text documents: Sentence selection and evaluation metrics”, *Proceedings of SIGIR*, p. 121-128.

[9]. Hassel Martin, Automatic text summarization evaluation, *Term Paper*, Royal Institute of Technology.

[10]. Jr. Santos Eugen, Ahmed A. Mohamed, and Qunhua Zhao (2004), “Automatic Evaluation of Summaries Using Document Graphs”, *ACL*.

[11]. Luhn H. P. (1958), “The Automatic Creation of Literature Abstracts”, *IBM Journal of Research Development*, 2(2), p. 159-165.

[12]. Mallet Daniel (2003), *Text Summarization: An Annotated Bibliography*, (Last compiled June 24).

[13]. Mani I. (2001), “Recent developments in text summarization”, *CIKM'01*, p. 529-531.

[14]. Nguyen Thi Minh Huyen, Laurent Romany , Xuan Luong Vu (2003), “A Case Study in POS Tagging of Vietnamese Texts”, *TALN 2003*, Batz-sur-Mer.

[15]. Oard Douglas W. (2001), “The Vector Space Model”, *LBSC 708A/CMSC*, 838L,Session 3. (http://www.cse.lehigh.edu/~brian/course/2002/searchengines/notes/notes-08-29.pdf)

[16]. Radev D. R., H. Jing, and M. Budzikowska (2000), “Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation, and user studies”, *Summa rization Workshop*.

[17]. Radev Dragomir R., Eduard Hovy and Kathleen McKeown (2002), “Introduction to the special issue on summarization”, *Computational Linguistics*, 28(4), p.399-408.

[18]. Ruiz Miguel, “Automatic Indexing & Text Categorization” (http://informatics.buffalo.edu/faculty/ruiz/teaching/Seminars/Automatic_Indexing.ppt)

[19]. Zha H. (2002), “Generic Summarization and Keyphrase Extraction Using Mutual Reinforcement Principle and Sentence Clustering”, *SIGIR’02*, p. 113-120.

[20]. Zhang Y., N. Zincir-Heywood, Evangelos Milios (2002), “World Wide Web Site Summarization”, *Technical Report CS-2002-08*, Faculty of Computer Science, Dalhousie University.

Tiếng Việt

[1]. Nguyễn Ngọc Bình (2004), *"Dùng lý thuyết tập thô và các kỹ thuật khác để phân loại, phân cụm văn bản tiếng Việt"*, Kỷ yếu hội thảo ICT.rda'04. Hà nội

[2]. Đỗ Bích Diệp (2004), *"Phân loại văn bản dựa trên mô hình đồ thị",* Luận văn cao học. Trường Đại học Tổng hợp New South Wales - Australia.

[3]. Đỗ Thành Dũng (2003), *Rút trích thông tin từ các tóm tắt của các bài báo khoa học về trí tuệ nhân tạo dùng độ thị khái niệm,* Luận văn Thạc sĩ Công nghệ thông tin, Khoa Công nghệ thông tin, Trường Đại học Bách khoa, ĐH Quốc gia TPHCM.

[4]. Lại Thị Hạnh (2002), *Trích cụm danh từ tiếng Việt nhằm phục vụ cho các hệ thống tra cứu thông tin đa ngôn ngữ,* Luận văn thạc sĩ Tin học, Thư viện Cao học, Khoa Công nghệ thông tin, Trường Đại Học Khoa học Tự nhiên, ĐH Quốc gia TPHCM.

[5]. Nguyễn Thị Minh Huyền, Vũ Xuân Lương, Lê Hồng Phương (2003), "Sử dụng bộ gán nhãn từ loại xác suất QTAG cho văn bản tiếng Việt*"*, *Kỷ yếu Hội thảo ICT.rda'03*, Hà Nội.

[6]. Võ Lý Hòa (2004), *Tìm hiểu văn bản tóm tắt và phương pháp tóm tắt văn bản*, Luận án tiến sĩ Ngữ văn, Đại học Khoa học Xã hội và Nhân văn, TP.HCM.

[7]. Đỗ Phúc, Hoàng Kiếm (2004), "Rút trích ý chính từ văn bản tiếng Việt hỗ trợ tạo tóm tắt nội dung", *Tạp chí Các công trình nghiên cứu - triển khai viễn thông và công nghệ thông tin*, số 13, trang 59-63.Đinh Thị Phương Thu, Hoàng Vĩnh Sơn, Huỳnh Quyết Thắng (2005), *"Phương án xây dựng tập mẫu cho bài toán phân lớp văn bản tiếng Việt: nguyên lý, giải thuật, thử nghiệm và đánh giá kết quả"*, Bài báo đã gửi đăng tại Tạp chí khoa học và công nghệ

[8]. Trần Ngọc Thêm (2000), *Hệ thống liên kết văn bản tiếng Việt*, NXB Giáo dục, TP. HCM.

[9]. Đồng Thị Bích Thủy, Hồ Bảo Quốc (2001), Ứng dụng xử lý ngôn ngữ tự nhiên trong hệ thống tìm kiếm thông tin trên văn bản tiếng Việt. Khoa Công nghệ thông tin, Trường Đại học Khoa học Tự nhiên, TPHCM.

CHƯƠNG VII. PHỤ LỤC

VII.1.Cấu trúc CSDL của chương trình

Bảng từ điển và từ hư (stop word) chỉ có một cột duy nhất là cột TỪ

Bảng từ đồng nghĩa khác từ:

Tu	Chứa từ đồng nghĩa
Nghia	Chứa từ có nghĩa tương đương trong từ điển

Bảng chứa dữ liệu văn bản mẫu:

Tu	Chứa từ đặc trưng
Nhom	Chứa nhóm văn bản
TrungBinh	Khoảng cách trung bình của từ trong câu
LonNhat	Khoảng cách lớn nhất của từ trong câu
NhoNhat	Khoảng cách nhỏ nhất của từ trong câu

VII.2.Kết quả nhận dạng văn bản

Cần nhận dạng	Nhận là	Số File	Nhận dạng được	Không nhận dạng được	Nhận dạng kiểu khác	Nhân dạng lại kiểu E	Nhận dạng lại kiểu F	Tổng kết
A	B	C	D	E	F	G	H	I
The Gioi	Chinhtri	137	74	11	52	14	16	71.15
Vi tinh	cntt	297	203	21	73	5	27	86.38
Xa hoi	Chinhtri	134	87	10	37	2	15	83.65385
Kinh Doanh	Kinh te	38	31	1	6	1	3	88.57
TỔNG CỘNG		606	395	43	168	22	61	82.63

VII.3.Các đặc trưng của mẫu phân loại văn bản (trích)

ChinhTri

Từ	Giá trị trung bình	Giá trị lớn nhất	Giá trị bé nhất
Chính phủ	34.63	103.09	7.53
Chủ tịch	26.10	59.11	3.42
đại biểu	19.83	52.93	4.84
đầu tư	48.99	113.46	9.81
phát triển	158.85	340.31	10.61
Quốc hội	29.61	91.98	6.06
Thủ tướng	17.95	56.88	2.69
thực hiện	41.01	73.86	4.80
Tổ chức	43.13	87.67	0.00
vấn đề	136.62	243.02	6.76
Việt Nam	35.87	87.38	6.44

CNTT

Từ	Giá trị trung bình	Giá trị lớn nhất	Giá trị bé nhất
có thể	16.27	33.22	3.06
công nghệ	16.17	32.06	3.88
công ty	17.12	34.76	3.72
cung cấp	16.76	34.70	4.76
di động	16.10	31.75	6.02
dịch vụ	17.23	70.20	5.52
điện thoại	20.25	35.59	4.13
khách hàng	18.07	53.42	7.12
máy tính	18.18	32.26	0.92

phản hồi	11.44	21.92	3.00
phần mềm	15.54	38.51	4.67
phát triển	19.54	36.87	4.74
Sản phẩm	20.14	35.87	0.00
sử dụng	18.37	37.14	2.79
thế giới	18.40	54.74	8.61
Thị trường	16.61	33.45	2.48
thông tin	19.91	35.67	0.77
viễn thông	19.27	37.45	6.10
Việt Nam	18.96	39.76	5.68

GiaoDuc

Từ	Giá trị trung bình	Giá trị lớn nhất	Giá trị bé nhất
chương trình	21.67	60.09	6.42
có thể	18.68	47.72	4.80
đào tạo	24.80	59.40	2.78
giáo dục	29.17	84.64	6.12
giáo viên	20.25	56.53	2.89
học sinh	19.96	44.96	0.93
sinh viên	23.11	50.43	5.11
Thí sinh	16.76	75.20	4.54
tổ chức	21.99	57.79	7.25
tốt nghiệp	20.87	62.82	7.87
Việt Nam	25.80	72.97	3.90

KinhTe

Từ	Giá trị trung bình	Giá trị lớn nhất	Giá trị bé nhất

Chứng khoán	17.04	42.47	0.00
cổ phiếu	11.88	40.08	2.77
có thể	17.59	114.29	5.33
công ty	17.79	45.37	3.65
đầu tư	19.35	43.81	5.76
doanh nghiệp	19.25	54.05	5.89
giao dịch	13.70	37.97	4.56
Ngân hàng	18.44	41.70	4.94
phát triển	22.75	44.22	2.55
thị trường	18.24	42.01	6.66
Việt Nam	20.11	44.25	1.77

KhoaHoc

Từ	Giá trị trung bình	Giá trị lớn nhất	Giá trị bé nhất
bệnh nhân	18.84	42.59	3.02
Bệnh viện	33.04	362.59	4.72
có thể	74.13	609.13	4.80
điều trị	21.00	40.76	3.21
gia cầm	23.36	55.00	9.01
khoa học	42.68	198.87	3.72
môi trường	19.61	67.66	6.81
nghiên cứu	22.54	125.17	3.09
nguy cơ	24.42	137.85	2.68
phản hồi	21.26	120.38	4.96
phát hiện	15.46	67.70	3.60
phát triển	41.54	201.24	3.03
Phẫu thuật	23.68	73.74	2.70
sản xuất	32.64	292.88	3.03
sử dụng	20.62	35.10	2.59

tế bào	19.07	70.85	8.04
thế giới	19.42	37.36	6.90
trường hợp	15.72	45.35	4.57
tử vong	21.96	48.83	4.75
Việt Nam	35.70	235.59	3.17
Y tế	26.65	483.37	4.77