

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

LỜI CẢM ƠN

Đầu tiên, em xin chân thành cảm ơn thầy giáo ThS. Lương Mạnh Bá - Bộ môn CNPM, Khoa CNTT - đã gợi ý hướng dẫn và tận tình giúp đỡ em hoàn thành đồ án này.

Em xin chân thành cảm ơn các thầy cô giáo trong khoa Công nghệ thông tin cũng như các thầy cô giảng dạy tại trường Đại học Bách khoa Hà Nội đã truyền đạt cho em những kiến thức bổ ích trong suốt thời gian em học tập và nghiên cứu tại trường.

Cuối cùng, em xin nói lời cảm ơn đến gia đình và bạn bè, những người đã giúp đỡ, động viên em rất nhiều trong suốt quá trình học tập và làm đồ án tốt nghiệp.

Trong quá trình thực hiện đồ án, do thời gian và kiến thức có hạn nên em không thể tránh khỏi những thiếu sót nhất định. Vì vậy em mong nhận được sự giúp đỡ và góp ý kiến từ phía thầy cô giáo và các bạn.

Một lần nữa em xin chân thành cảm ơn !

Hà nội ngày 15 tháng 05 năm 2005

Sinh viên

Vũ Hải Tùng

MỤC LỤC

MỤC LỤC	2
DANH MỤC CÁC HÌNH VẼ	6
DANH MỤC CÁC BẢNG	8
DANH MỤC CÁC TỪ VIẾT TẮT	9
CHƯƠNG I - MỞ ĐẦU	10
1.1 Khai thác văn bản	11
1.1.1 Khai thác văn bản là gì?	11
1.1.2 Một số bài toán tiêu biểu trong Khai thác văn bản	11
1.2 Bài toán TTVB - Automatic Text Summarization (ATS)	13
1.2.1 Tóm tắt văn bản (TTVB)	13
1.2.2 Ứng dụng của TTVB	13
1.2.3 Giải quyết bài toán TTVB	14
1.3 Mục đích lựa chọn đề tài	15
1.4 Các mục tiêu cụ thể trong đồ án	15
CHƯƠNG II - CÁC PHƯƠNG ÁN GIẢI QUYẾT BÀI TOÁN TÓM TẮT VĂN BẢN..	16
2.1 Một số khái niệm cơ bản về TTVB	17
2.1.1 Mô hình một hệ thống TTVB	17
2.1.1.1 Các loại TTVB	17
2.1.1.2 Các tiêu chí khi thực hiện tóm tắt	18
2.1.1.3 Mô hình bên ngoài của một hệ thống Tóm tắt	18
2.1.2 Qui trình thực hiện TTVB	19
2.1.2.1 Quá trình tiền xử lý	20
2.1.2.2 Quá trình xử lý	21
2.1.2.3 Quá trình sinh kết quả	21
2.2 Các giải thuật TTVB.	23
2.2.1 Giải thuật dựa trên giá trị trọng số của thuật ngữ (Determining Term Weights)	23
2.2.1.1 Một số định nghĩa	23
2.2.1.2 Giải thuật lựa chọn câu có trị trung bình tàn số cao nhất	24
2.2.2 Giải thuật dựa trên phân nhóm các đoạn văn trong văn bản (Paragraphs Clustering for Summarization).....	25
2.2.2.1 Định nghĩa phân nhóm	25
2.2.2.2 Giải thuật cho bài toán phân nhóm	26
2.2.2.3 Áp dụng phân nhóm văn bản cho bài toán TTVB	27
2.2.2.4 Đánh giá	27

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

2.2.3 Giải thuật sử dụng các đặc trưng tóm tắt kết hợp thuật toán học máy (Summarization using Machine Learning Algorithm)	28
2.2.3.1 Các đặc trưng của tóm tắt (<i>Summarized Features</i>).....	28
2.2.3.2 Kết hợp các đặc trưng (<i>Features Combination</i>) để tạo tóm tắt... <td>29</td>	29
2.2.3.3 Áp dụng giải thuật học máy (<i>Machine Learning Algorithm</i>)	30
2.2.3.4 Đánh giá	31
2.2.4 Giải thuật áp dụng các đặc trưng liên kết ngữ nghĩa trong văn bản (Summarization using Cohesion Features)	32
2.2.4.1 Các định nghĩa cơ bản.....	32
2.2.4.2 Liên kết ngữ nghĩa ứng dụng trong TTVB	33
2.4.2.3 Giải thuật áp dụng chuỗi từ vựng để TTVB (<i>Summarization using Lexical Chains</i>).....	34
2.4.2.3 Đánh giá	35
2.2.5 Giải thuật áp dụng các đặc trưng liên kết cấu trúc trong văn bản (Summarization using Coherence Features)	35
2.2.5.1 Khái niệm về liên kết cấu trúc (<i>Coherence</i>).	35
2.2.5.2 Áp dụng liên kết cấu trúc cho TTVB.....	35
2.2.6 Kết luận	36
CHƯƠNG III - TIỀN XỬ LÝ VĂN BẢN TIẾNG VIỆT	37
3.1 Phương pháp tách thuật ngữ tiếng Việt	38
3.2 Xây dựng từ điển.....	41
3.2.1 Tổ chức cấu trúc bản ghi trong từ điển	41
3.2.2 Tổ chức kết cấu.....	45
3.2.2.1 Lưu trữ theo danh sách sắp xếp	45
3.2.2.2 Lưu trữ sử dụng bảng băm	46
3.3 Loại bỏ từ dừng (stop word)	48
3.4 Biểu diễn văn bản theo mô hình không gian vec tơ	49
3.1.1 Mô hình Boolean.....	49
3.1.2 Mô hình tần suất TF	49
3.1.3 Mô hình nghịch đảo tần số văn bản – IDF.....	49
3.1.4 Mô hình kết hợp TF-IDF	50
3.1.5 Mô hình vec tơ thura.....	50
3.1.6 Các công thức tính toán trên mô hình không gian vec tơ.....	50
CHƯƠNG IV - THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG.....	52
4.1 Mô hình hệ thống	53
4.2 Module xử lý văn bản.....	55
4.2.1 Nhiệm vụ	55

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

4.2.2 Mô hình chức năng	55
4.3.2 Thực hiện.....	55
4.3.2.1 Chuẩn hoá văn bản	55
4.3.2.2 Tách thuật ngữ	56
4.3.2.3 Loại bỏ từ dừng.....	59
4.3.2.4 Thống kê từ khoá, tạo kết quả	59
4.3 Module thực hiện giải thuật 1	61
4.3.1 Một số nhận định quan trọng.....	61
4.3.2 Mô hình chức năng	62
4.3.3 Thực hiện.....	62
4.3.3.1 Hệ số ghi điểm.....	62
4.3.3.2 Tính trọng số các câu	63
4.3.3.3 Sắp xếp, tính ngưỡng và đưa ra kết quả.....	63
4.4 Module thực hiện giải thuật 2	65
4.4.1 Mô hình của giải thuật	65
4.4.2 Tách thuật ngữ đại diện.....	65
4.4.3 Véc tơ hoá đoạn văn.....	66
4.4.4 Phân nhóm đoạn văn.....	67
4.4.5 Trích rút Tóm tắt.....	67
4.5 Module thực hiện giải thuật 3	71
4.5.1 Mô hình giải thuật.....	72
4.5.2 Trích rút theo đặc trưng.....	72
4.5.3 Giải thuật học máy.....	76
4.5.4 Áp dụng kết hợp	77
4.6 Module tạo kết quả.	78
4.7 Cài đặt hệ thống.	79
4.7.1 Môi trường và công cụ cài đặt.....	79
4.7.2 Mô tả chương trình.	79
4.7.2.1 Các lớp chính được thiết cho chương trình:.....	79
4.7.2.2 Giao diện chính chương trình	80
4.7.2.3 Giao diện giải thuật 1	81
4.7.2.4 Giao diện giải thuật 2.....	82
4.7.2.5 Giao diện giải thuật 3.....	83
4.8 Minh họa một số thực nghiệm và đánh giá.....	84
4.8.1 Đại lượng đánh giá độ chính xác.....	84
4.8.2 Cơ sở dữ liệu thực nghiệm	85
4.8.3 Thực nghiệm trên modul Tiền xử lý văn bản.....	87

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê	
4.8.4 Thực nghiệm trên các module Tóm tắt.	87
TỔNG KẾT	89
TÀI LIỆU THAM KHẢO.....	90

DANH MỤC CÁC HÌNH VẼ

Hình 1: Định nghĩa bài toán TTVB	13
Hình 2: Mô hình bên ngoài một hệ thống Tóm tắt	19
Hình 3: Ba bước qui trình thực hiện TTVB	20
Hình 4: Giải thuật tóm tắt dựa trên trung bình trọng số cao nhất.....	24
Hình 5: Các quả bóng được đánh dấu theo thứ tự bất kỳ.	25
Hình 6: Đã phân nhóm	25
Hình 7: Thuật toán K-Means.....	26
Hình 8: Thuật toán cây phân cấp dưới lên	26
Hình 9: Áp dụng phân nhóm văn bản để thực hiện tóm tắt	27
Hình 10: Ví dụ về cây nhị phân	29
Hình 11: Vào - ra với mỗi đặc trưng tóm tắt.....	30
Hình 12: Mô hình kết hợp các đặc trưng tóm tắt.....	30
Hình 13: Vào - ra kết hợp các đặc trưng tóm tắt	30
Hình 14: Giải thuật TTVB dựa theo chuỗi từ vựng.....	34
Hình 15. Hoạt động của từ điển.....	41
Hình 19: Mô hình hệ thống	54
Hình 20: Module Tiền xử lý	55
Hình 21: Một đoạn các thuật ngữ trong từ điển	57
Hình 22: Tổ chức dữ liệu có cấu trúc cho văn bản.....	60
Hình 23: Module giải thuật 1.....	62
Hình 24: Đồ thị trọng số câu	64
Hình 25: Module thực hiện giải thuật 2	65
Hình 26: Ví dụ cây phân cấp theo giải thuật phân cấp dưới lên	68
Hình 27: Module thực hiện giải thuật 3	72

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê	
Hình 28: Giải thuật tạo cây nhị phân	75
Hình 29: Giao diện chính của chương trình	80
Hình 30: Giao diện giải thuật 1.	81
Hình 31: Giao diện giải thuật 2	82
Hình 33: Precision và Recall	84

DANH MỤC CÁC BẢNG

Bảng 1: Các cụm phụ âm đầu.....	42
Bảng 2: Các cụm phụ âm cuối.....	43
Bảng 3: Các cụm nguyên âm.....	44
Bảng 4: Một số từ dừng trong tiếng Việt	48
Bảng 5: Minh họa các giá trị Precision và Recall	85
Bảng 6: Tập tóm tắt mẫu	86
Bảng 7: Kết quả tách thuật ngữ.	87
Bảng 8. Đánh giá độ chính xác các giải thuật.....	88

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

DANH MỤC CÁC TỪ VIẾT TẮT

STT	Từ viết tắt	Giải nghĩa
1	ATS	Automatic Text Summarization
2	CSDL	Cơ Sở Dữ Liệu
3	DM	Data Mining
4	DTW	Determining Term Weights
5	FS	Fuzzy Set
6	hoÆc	Hierachical Clustering
7	IDF	Inverse Document Frequency
8	IPF	Inverse Paragraph Frequency
9	ISF	Inverse Sentence Frequency
10	IR	Information Retrieval
11	KDT	Knowledge-Discovery in Text
12	MDS	Multi Documents Summarization
13	PCS	Paragraphs Clustering for Summarization
14	SDS	Single Document Sumarization
15	SF	Summaried Feature
16	SMLA	Summarization using Machine Learning Algorithm
17	TF	Term Frequency
18	TM	Text Mining
19	TRSM	Tolerance Rough Set Model
20	TTVB	Tóm Tắt Văn Bản
21	VSP	Vector Space Model

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

CHƯƠNG I

MỞ ĐẦU

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

1.1 Khai thác văn bản.

1.1.1 Khai thác văn bản là gì?

Với sự phát triển vượt bậc của khoa học công nghệ đặc biệt là CNTT, ngày nay lượng thông tin tồn tại trên các phương tiện truyền thông (internet, TV, news, email,...) phát triển một cách nhanh chóng. Mỗi một ngày lại có vô số thông tin mới được tạo ra từ nhiều nguồn khác nhau. Chúng đòi hỏi phải được lưu trữ để truy cập và sử dụng khi cần thiết. Từ nhu cầu thực tế đó, lĩnh vực khai thác dữ liệu (Data Mining - DM) mà cụ thể là khai thác văn bản (Text Mining - TM) đặt ra nhiều yêu cầu nghiên cứu khác nhau liên quan phục vụ cho việc quản lý và khai thác nguồn dữ liệu khổng lồ này.

Vậy thế nào là khai thác dữ liệu văn bản?

Khai thác dữ liệu là các phương pháp trích chọn, sàng lọc để tìm ra các thông tin cần thiết từ một kho dữ liệu ban đầu. Các thông tin này chưa được biết trước, có giá trị và tiềm năng sử dụng.

Văn bản (Text) là một kiểu dữ liệu, cụ thể: là một tập hợp các từ đi liền nhau nhằm diễn đạt một nội dung nào đó. Do vậy văn bản là loại dữ liệu không có cấu trúc hoặc bán cấu trúc.

Khai thác văn bản, còn được biết đến như phân tích văn bản thông minh (intelligent text analysis), khai thác dữ liệu văn bản (text data mining) hoặc khám phá tri thức văn bản (knowledge-discovery in text - KDT) liên quan đến quá trình trích lọc các thông tin, tri thức cần thiết chưa được khai phá và có giá trị sử dụng từ các kho văn bản.

Khai thác văn bản là một lĩnh vực kết hợp nhiều lĩnh vực nghiên cứu khác liên quan: tìm kiếm thông tin (information retrieval), khai thác dữ liệu (data mining), học máy (machine learning), ngôn ngữ học máy tính (computer linguistics). Với hơn 80% thông tin dữ liệu đang được lưu trữ dưới dạng văn bản (theo thống kê của Bách khoa toàn thư WIKIPEDIA), khai thác văn bản có tiềm năng ứng dụng rất lớn và ngày càng trở nên quan trọng hơn.

1.1.2 Một số bài toán tiêu biểu trong Khai thác văn bản

Có thể nêu ra một số bài toán có ứng dụng quan trọng trong lĩnh vực khai thác văn bản sau:

- Phân loại văn bản (Text Categorization - Text Classification): Cho một tập các văn bản đã được phân loại theo các chủ đề cho trước (VD: kinh tế, triết học, thể thao, văn hoá,). Xuất hiện một văn bản mới chưa được phân loại, vấn đề đặt ra là xác định văn bản đó thuộc loại - chủ đề nào.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

- Lập nhóm văn bản (Text Clustering): Từ một tập hợp văn bản bất kỳ, cần lập ra các nhóm văn bản căn cứ theo độ tương tự về nội dung của chúng. Số nhóm này có thể do người dùng chỉ định hoặc hệ thống lựa chọn số nhóm thích hợp.

- Tóm tắt văn bản (Text Summarization): Cho một văn bản bất kỳ, cần đưa ra một thể hiện nội dung ngắn gọn cho văn bản đó.

- Tìm kiếm thông tin (Information Retrieval): Từ một tập hợp dữ liệu (ở đây, dữ liệu được hiểu là các văn bản) ban đầu, người dùng đưa ra một truy vấn về thông tin cần tìm kiếm. Hệ thống sẽ cung cấp một danh sách dữ liệu được xếp loại thỏa mãn yêu cầu thông tin đó.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

1.2 Bài toán TTVB - Automatic Text Summarization (ATS)

Trước tiên phải hiểu định nghĩa cụ thể cho bài toán TTVB.

1.2.1 Tóm tắt văn bản (TTVB)

TTVB là quá trình thực hiện giảm đi độ dài, sự phức tạp của một văn bản trong khi vẫn giữ lại được các nội dung có giá trị của nó. TTVB nhằm đưa ra thể hiện về nội dung một cách ngắn gọn của văn bản.

Có thể phát biểu bài toán TTVB như sau:

Đầu vào:	Một văn bản hoặc một tập hợp văn bản
Đầu ra:	Nội dung ngắn gọn(tóm tắt) hoặc một tập các nội dung ngắn gọn của chúng.

Hình 1: Định nghĩa bài toán TTVB

Thực ra TTVB đã xuất hiện từ rất lâu, nhưng chúng thường được thực hiện một cách truyền thống do con người. Tác dụng chính của những tóm tắt kiểu này là để giúp đỡ cho người đọc có cái nhìn tổng quát về nội dung chính sẽ được trình bày trong tài liệu. Trong hầu hết các trường hợp, người đọc trước khi quyết định xem có nên đọc một văn bản nào đó không thường thích nhìn vào tóm tắt của văn bản đó để xem nội dung của nó có thoả mãn nhu cầu về thông tin của mình hay không.

1.2.2 Ứng dụng của TTVB

TTVB có rất nhiều ứng dụng thực tế. Có thể nêu ra một số ứng dụng chính như:

Tóm tắt phục vụ máy tìm kiếm (Search engine hits): tóm tắt các thư viện dữ liệu không lồ để phục vụ cho mục đích tìm kiếm thông tin. Với tài nguyên dữ liệu lớn, mỗi lần thực hiện tìm kiếm nếu chỉ rà soát thông tin trên danh mục các tóm tắt của dữ liệu sẽ tiết kiệm thời gian và giảm độ phức tạp của bài toán tìm kiếm. Hiện một số địa chỉ tìm kiếm nổi tiếng như Google, Altavista,.. đều đã ứng dụng rất tốt TTVB vào hệ thống của mình.

Tóm tắt tin tức (Multimedia news summaries): có ứng dụng rất lớn trong thương mại. Giá trị của thông tin trong thương mại là rất quan trọng. Song với lượng thông tin lớn được xuất bản mỗi ngày, doanh nghiệp không thể tiếp nhận và xử lý hết chúng. Tóm tắt tin tức có thể giúp cho thu thập đủ các thông tin cần thiết từ nguồn dữ liệu này. Đã có nhiều công ty (kể cả ở Việt Nam) khai thác giá trị thương mại này, bằng cách cung cấp cho khách hàng những thông tin được xuất

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê
bản trong ngày có nội dung liên quan đến một lĩnh vực được “đặt hàng” trước nào
đó.

Hỗ trợ tìm kiếm đa ngôn ngữ: Giả sử người dùng cần tìm các tài liệu về một
vấn đề nào đó. Nhưng các tài liệu này lại tồn tại dưới dạng các ngôn ngữ khác
nhau. Trước hết tóm tắt nội dung của tài liệu, sau đó áp dụng hệ thống dịch tự động
đưa chúng về ngôn ngữ của người đọc. Nếu tài liệu này thoả mãn yêu cầu người
dùng, nó sẽ được người dùng tìm cách dịch và sử dụng.

Tóm tắt còn có thể sử dụng để xây dựng thông tin cho các thiết bị cầm tay
(máy tính bỏ túi, điện thoại di động) . Với khả năng hiển thị hạn chế của các thiết bị
này, việc cô đọng thông tin để phù hợp với kích thước sử dụng là cần thiết.

Một số ứng dụng khác của TTVB như: hỗ trợ người khiếm thị; cô đọng nội
dung và đọc lại cho người dùng; giúp đỡ điều trị bệnh nhân; tóm tắt và so sánh sự
điều trị cần thiết cho mỗi bệnh nhân; thu thập thông minh; tự động xây dựng một
tiểu sử 500 từ về chủ tịch Hồ Chí Minh;

1.2.3 Giải quyết bài toán TTVB

Trên thế giới, bài toán TTVB đã xuất hiện từ rất lâu. Những kỹ thuật đầu
tiên áp dụng để TTVB xuất hiện từ những năm 50 của thế kỷ trước (như nghiên cứu
của Luhn năm 1959,...). Sau đó, chúng tiếp tục được nghiên cứu và đạt nhiều kết
quả ngày càng tốt hơn, cho nhiều loại ngôn ngữ như tiếng Anh, tiếng Pháp, tiếng
Nhật, tiếng Trung... (các nghiên cứu này sẽ được trình bày trong chương tiếp theo
của báo cáo). Ở Việt Nam bước đầu cũng đã có một số nghiên cứu giải quyết bài
toán cho ngôn ngữ tiếng Việt nhưng số lượng cũng như chất lượng con thấp do đây
là một vấn đề còn khá mới mẻ.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

1.3 Mục đích lựa chọn đề tài

Những năm gần đây là khoảng thời gian Internet có sự phát triển mạnh mẽ tại Việt Nam. Cách đây khoảng 7,8 năm nếu như Internet còn khá xa lạ thì hiện nay hiện tượng người dùng truy nhập và sử dụng các thông tin tiếng Việt trên Internet đã trở nên phổ biến. Xuất phát từ sự thay đổi đó rất nhiều các bài toán thuộc lĩnh vực khai thác văn bản cho tiếng Việt đã được nghiên cứu và ban đầu có một số ứng dụng thực tế (ví dụ ứng dụng trong hệ thống tìm kiếm thông tin trang Web tiếng Việt như Vinaseek, Panvietnam, ..).

Bài toán TTVB rõ ràng có một vai trò khá quan trọng trong lĩnh vực khai thác dữ liệu nói chung và khai thác văn bản nói riêng. Nhưng đáng ngạc nhiên là số lượng các nghiên cứu giải quyết bài toán đối với tiếng Việt lại rất ít. Bởi vậy tác giả đã mạnh dạn chọn TTVB tiếng Việt làm nội dung nghiên cứu cho đề tài tốt nghiệp. Qua việc nghiên cứu các phương pháp, kỹ thuật có thể ứng dụng để giải quyết bài toán, tác giả hy vọng có thể tiếp cận với nhiều kỹ thuật tiên tiến và mở rộng kiến thức của mình, đặc biệt trong lĩnh vực Khai thác dữ liệu.

1.4 Các mục tiêu cụ thể trong đồ án

Khi lựa chọn đề tài này, tác giả mong rằng có thể đưa ra và thực hiện phương án giải quyết cụ thể cho bài toán TTVB tiếng Việt. Vì đây là vấn đề còn khá mới mẻ ở Việt Nam, tác giả đặt mục tiêu nghiên cứu nền tảng cơ sở của bài toán và hy vọng nó có thể làm cơ sở để nghiên cứu phát triển cao hơn sau này. Chính vì vậy, các mục tiêu cụ thể được đưa ra trong đồ án:

- Nghiên cứu tổng quan bài toán TTVB.
- Nghiên cứu và trình bày các phương pháp đã có trên thế giới cho kết quả tốt đối với bài toán TTVB.
- Áp dụng các phương pháp đã nghiên cứu để thực hiện xây dựng cụ thể một hệ thống TTVB tiếng Việt. Cụ thể trong đồ án này phương pháp được lựa chọn là các kỹ thuật lượng giá, thống kê.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

CHƯƠNG II

CÁC PHƯƠNG ÁN GIẢI QUYẾT BÀI TOÁN TÓM TẮT VĂN BẢN

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Trước khi đi vào phân tích cụ thể một số phương pháp thực hiện TTVB, cần tìm hiểu qua một số khái niệm cơ bản, ví dụ như: giải quyết bài toán TTVB nhằm thực hiện mục đích gì, thực hiện thế nào, bao gồm các bước nào...

2.1 Một số khái niệm cơ bản về TTVB

2.1.1 Mô hình một hệ thống TTVB.

2.1.1.1 Các loại TTVB

Tóm tắt của một văn bản là một thể hiện ngắn gọn nội dung của văn bản đó. Tuy vậy không phải mỗi văn bản đều chỉ có thể có một tóm tắt duy nhất cho nó. Về cơ bản, có thể phân ra hai loại tóm tắt cho văn bản dựa trên cách xây dựng chúng như sau:

- Tóm tắt trích rút (Extract Summarization): là các tóm tắt được xây dựng bằng cách rút ra *y nguyên, không thay đổi* những câu chứa nội dung quan trọng trong văn bản gốc.
- Tóm tắt trừu tượng (Abstract Summarization): là các tóm tắt mà một số thành phần của nó không xuất hiện trong văn bản gốc mà do tác giả đưa vào, ví dụ như các câu, các thành ngữ, các chú giải...

Tóm tắt Abstract (ở đây xin gọi hai loại tóm tắt là Extract và Abstract cho sát với nghĩa gốc) thường do con người tạo ra. Mục đích của chúng nhằm tạo ra nên sự diễn đạt một cách **ngắn gọn** và **liền mạch** về nội dung của văn bản. Tuy rằng nó không rút ra một cách nguyên bản các câu trong văn bản gốc nhưng đa phần các từ, các ngữ và thành ngữ câu thành nên nó đều được lấy từ văn bản gốc.

Tóm tắt Extract có thể được tạo ra bởi con người hoặc máy, cũng nhằm mục đích tạo ra một sự diễn đạt về nội dung cho văn bản gốc. Tuy nhiên mục tiêu **liền mạch** khó có thể thoả mãn được đối với các tóm tắt kiểu này. Bởi mỗi câu trong văn bản chỉ tạo được sự kết dính trong ngữ cảnh của văn bản gốc với các câu ngay trước và sau chúng. Vì vậy nếu trích rút, cũng có nghĩa là loại bỏ một số câu trong văn bản gốc sẽ làm mất đi sự kết dính này.

Có một số nghiên cứu đã được thực hiện theo hướng xây dựng nên Tóm tắt Abstract, tuy vậy hầu hết các nghiên cứu còn lại cho TTVB đều thực hiện theo hướng xây dựng Tóm tắt Extract. Bởi vì để xây dựng một hệ thống thực hiện Tóm tắt Abstract giống như con người có thể làm, hệ thống đó không chỉ có khả năng đọc-hiểu văn bản gốc mà còn phải có khả năng tự “xây dựng văn bản” từ những từ khoá, thành ngữ, khái niệm cho trước. Một hệ thống như vậy đòi hỏi phải có cơ sở tri thức cũng như khả năng tính toán không lồ, khó có thể thực hiện hoàn hảo được trong hoàn cảnh hiện nay.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

Trong giới hạn nghiên cứu đồ án này, tác giả sẽ chỉ nghiên cứu theo hướng tạo Tóm tắt Extract đối với bài toán TTVB tiếng Việt. Mọi khả năng phát triển để xây dựng Tóm tắt Abstract cũng như mở rộng hệ thống sẽ được trình bày trong chương cuối.

2.1.1.2 Các tiêu chí khi thực hiện tóm tắt

Tóm tắt cho một văn bản được thực hiện phải thoả mãn các tiêu chí định trước sau:

- **Hệ số rút gọn thông tin:** còn được gọi là hệ số cô đặc thông tin, đặc trưng cho độ cô đọng thông tin của tóm tắt. Hệ số rút gọn được tính bằng chiều dài của tóm tắt trên chiều dài của văn bản gốc. Độ cô đọng càng cao, có nghĩa là văn bản càng được cô đọng đi nhiều thì tóm tắt của nó càng ngắn gọn => hệ số rút gọn càng nhỏ. Hệ số này (tính theo %) có thể được tính bằng:

- + Độ dài (từ hoặc ký tự) của văn bản gốc trên độ dài của tóm tắt.

$$c = \frac{\text{length}(Sum)}{\text{length}(Text)} \times 100\%$$

- + Số câu của tóm tắt trên số câu của văn bản gốc (đối với tóm tắt Extract).

$$c = \frac{\text{SentenceCount}(Sum)}{\text{SentenceCount}(Text)} \times 100\%$$

- **Tiêu chí về nội dung thông tin:** dựa trên các yếu tố sau

- + Tính đúng đắn so với văn bản gốc.
 - + Tính thích hợp với nhu cầu của người dùng.

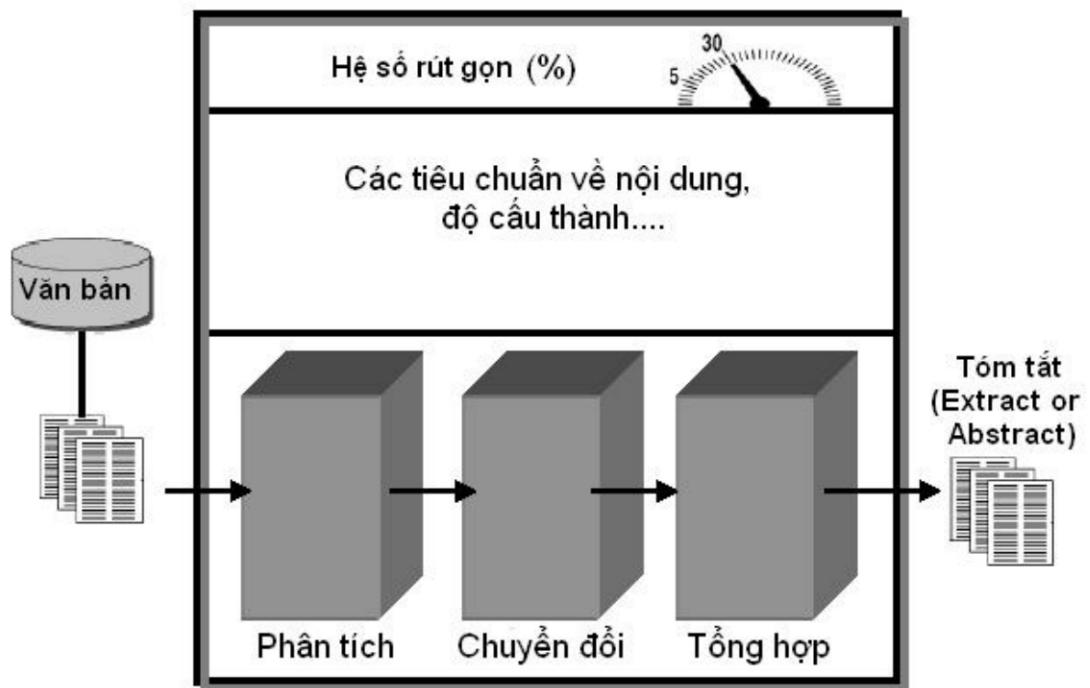
Tính thích hợp với nhu cầu của người dùng ở đây có thể hiểu là Tóm tắt được tạo ra là Tóm tắt chung (generic summarization) hay Tóm tắt theo yêu cầu (user focused summarization). Tóm tắt chung bao gồm toàn bộ các thông tin quan trọng trong văn bản gốc còn Tóm tắt theo yêu cầu chỉ chứa những nội dung liên quan tới yêu cầu thông tin (information query) mà người dùng đưa vào.

- **Tiêu chí về tính cấu thành của tóm tắt:** Đối với tóm tắt Extract thì phải tránh được sự đứt mạch, sự lặp lại, tránh các danh sách liệt kê... Đối với tóm tắt Abstract thì cần có sự liền mạch về nội dung, ngữ pháp chính xác...

2.1.1.3 Mô hình bên ngoài của một hệ thống Tóm tắt

Như vậy, một hệ thống Tóm tắt có thể có mô hình bên ngoài như sau:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê



Hình 2: Mô hình bên ngoài một hệ thống Tóm tắt

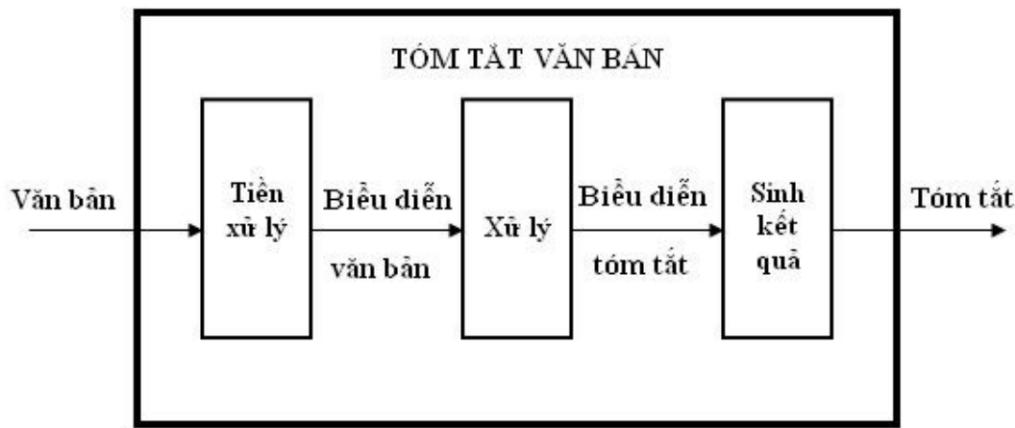
Đây là mô hình hệ thống tóm tắt nhìn từ phía bên ngoài dựa theo các đặc điểm phân loại và tiêu chí thực hiện tóm tắt. Dưới đây sẽ trình bày tổng quát qui trình thực hiện bên trong của một hệ thống (trong mô hình bên ngoài được hiểu như một quá trình Phân tích - Chuyển đổi - Tổng hợp).

2.1.2 Qui trình thực hiện TTVB

Một hệ thống TTVB tổng quát bao gồm 3 quá trình:

- Quá trình tiền xử lý (phân tích): xây dựng một biểu diễn có cấu trúc của văn bản.
- Quá trình xử lý (chuyển đổi): bao gồm một giải thuật nào đó chuyển đổi biểu diễn văn bản có cấu trúc sang một dạng biểu diễn có cấu trúc khác: biểu diễn cho tóm tắt.
- Quá trình sinh kết quả (tổng hợp): Tóm tắt được tạo ra bằng cách dựa vào biểu diễn cho tóm tắt.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê



Hình 3: Ba bước qui trình thực hiện TTVB

2.1.2.1 Quá trình tiền xử lý

Tiền xử lý văn bản nói chung là quá trình thực hiện đọc văn bản và chuyển đổi văn bản đó sang một dạng biểu diễn có cấu trúc.

Biểu diễn có cấu trúc là gì? Đó là một dạng mô hình biểu diễn để có thể biến đổi định dạng không có cấu trúc và tính chất nguyên bản của văn bản - vốn gây rất nhiều khó khăn cho bài toán Khai thác văn bản - về dạng dữ liệu có cấu trúc. Mô hình biểu diễn này có vai trò rất quan trọng, hiệu quả và hiệu xuất của phương án giải quyết mỗi bài toán phụ thuộc rất nhiều vào việc lựa chọn mô hình này.

Một số mô hình để biểu diễn văn bản:

- **Mô hình không gian véc tơ** (Vector Space Model - VSM). Bản chất của mô hình này là mỗi văn bản hoặc mỗi thành phần của văn bản được biểu diễn thành một véc tơ. Mỗi thành phần của véc tơ là một thuật ngữ riêng biệt trong tập văn bản gốc và được gán một giá trị trọng số w được tính theo tần suất xuất hiện của thuật ngữ trong văn bản/thành phần của văn bản. Các biến thể của mô hình không gian véc tơ thưa dưa trên sự khác nhau về hàm đánh giá giá trị trọng số này.

Đặc điểm quan trọng của mô hình không gian véc tơ chính là ở chổ độ tương tự của 2 văn bản/thành phần văn bản có thể được tính qua độ tương tự giữa 2 véc tơ đại diện của chúng. Mô hình không gian véc tơ được sử dụng rất rộng rãi vì tính đơn giản và hiệu quả của nó.

- **Mô hình dựa trên tập mờ** (Fuzzy Set - FS). Chủ yếu xoay bài toán biểu diễn văn bản về việc lưu trữ trên tập mờ, có nghĩa là lưu trữ và xử lý các khái niệm thay vì làm việc trên các thuật ngữ.
- **Mô hình tập thô dung sai** (Tolerance Rough Set Model - TRSM).

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Tiền xử lý văn bản đóng vai trò khá quan trọng trong các bài toán khai thác văn bản. Nó làm giảm thiểu phần dữ liệu thừa phải tính toán, làm giảm kích thước của bài toán. Có một số phương pháp có thể áp dụng trong tiền xử lý văn bản: **Case Folding, Loại bỏ từ dừng (stop word)**.

- **Case Folding** thực hiện chuyển đổi tất cả các ký tự trong văn bản về cùng một dạng format, chỉ là ký tự hoa hoặc thường. VD: các từ “anH”, “Anh”, “ANh”.. đều được chuyển về thành từ “anh”.

- **Stopword** là các từ xuất hiện rất thường xuyên trong văn bản. Và đó cũng xuất hiện rất phổ biến trong các văn bản khác. Chúng mang ít thông tin về nội dung văn bản mà chúng xuất hiện. Do đó, cần thiết loại bỏ chúng. Ví dụ, đó là các từ “áy”, “cái”, “nó”,..

Thường thì quá trình tiền xử lý thường được tiến hành: đầu tiên thực hiện Case Folder, sau đó Loại bỏ từ dừng , thu được các thuật ngữ và biến đổi chúng về dạng biểu diễn phù hợp.

2.1.2.2 Quá trình xử lý

Đây là quá trình áp dụng các giải thuật để biến các giá trị biểu diễn của văn bản đã đạt được sau quá trình tiền xử lý thành các giá trị biểu diễn khả năng xây dựng tóm tắt. Các giá trị sau khi biến đổi được dùng làm đầu vào cho quá trình sinh kết quả. Không có một mô hình biểu diễn chung nào cho các giá trị này như ở giai đoạn trên mà chúng được xây dựng phụ thuộc vào giải thuật chuyển đổi và vào cách đánh giá để sinh kết quả trong giai đoạn sau.

Đây là giai đoạn thực hiện quan trọng nhất của một hệ thống Tóm tắt. Độ mạnh/yếu của hệ thống được đánh giá dựa trên độ mạnh/yếu của giải thuật thực hiện xử lý này. Một số giải thuật cụ thể sẽ được trình bày trong phần dưới.

2.1.2.3 Quá trình sinh kết quả

Bước cuối cùng hệ thống nhằm đưa ra một tóm tắt cho văn bản gốc. Đây thường là bước đơn giản nhất, tuy nhiên độ phức tạp của nó cũng phụ thuộc vào quá trình xử lý ở trên.

Lấy một ví dụ đơn giản cho ba quá trình thực hiện trong một hệ thống tóm tắt extract chỉ đánh giá độ quan trọng (khả năng trích rút để tham gia vào tóm tắt) của mỗi câu trên số lần xuất hiện của các thuật ngữ trong câu.

Quá trình 1 - tiền xử lý:

- Loại bỏ các từ dừng, đưa các từ về cùng một dạng format chuẩn

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

- Biểu diễn văn bản theo mô hình véc tơ thưa, theo đó mỗi câu được biểu diễn dưới dạng véc tơ, mỗi thành phần của véc tơ là một thuật ngữ xuất hiện trong văn bản.

Quá trình 2 - xử lý:

- Mỗi véc tơ được đánh giá bởi một hàm f , tính số lần các thuật ngữ quan trọng xuất hiện trong câu đó.

Quá trình 3 - đưa ra kết quả:

- Các câu được sắp xếp theo thứ tự từ cao đến thấp với giá trị f . Một số câu có thứ tự cao nhất (tuỳ thuộc vào hệ số rút gọn đã trình bày trong phần trước) được rút ra và tạo thành tóm tắt với thứ tự như trong văn bản gốc.

Tất nhiên trên đây chỉ là một ví dụ đơn giản cho các bước trong qui trình thực hiện tóm tắt. Hiệu quả của hệ thống nếu được xây dựng như vậy sẽ rất thấp. Trong phần dưới đây xin trình bày một số giải thuật có hiệu quả cho TTVB.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

2.2 Các giải thuật TTVB.

Có rất nhiều giải thuật cho TTVB được nghiên cứu và phát triển, đặc biệt trong khoảng thời gian gần đây. Có thể phân loại chúng dựa trên nền tảng cơ sở phát triển, từ đơn giản tới phức tạp.

Các giải thuật được trình bày dưới đây là các giải thuật xây dựng TTVB bằng cách trích rút ra những câu/đoạn quan trọng nhất trong văn bản gốc, các giải thuật xây dựng tóm tắt extract.

2.2.1 Giải thuật dựa trên giá trị trọng số của thuật ngữ (Determining Term Weights).

Các giải thuật dựa trên giá trị trọng số của thuật ngữ (DTS) là các giải thuật đơn giản nhất, xong đến nay vẫn chứng minh được tính hiệu quả của chúng. Giải thuật này được thừa kế từ giải thuật đánh giá trọng số trong lĩnh vực tìm kiếm thông tin (Information Retrieval). Nội dung cơ bản của giải thuật này là dựa vào việc tính toán giá trị trọng số cho mỗi thuật ngữ xuất hiện trong câu, từ đó tính toán giá trị trọng số cho mỗi câu trong văn bản và cuối cùng trích rút các câu có giá trị trọng số cao nhất. Thực hiện TTVB trên nền tảng giải thuật này, gần đây nhất là nhóm các tác giả J Larroca Neto, AD Santos, CAA Kaestner và AA Freitas (2000) [4].

Trước khi phân tích cụ thể giải thuật, cần hiểu một số định nghĩa cơ bản sau:

2.2.1.1 Một số định nghĩa.

- **Tần suất thuật ngữ** (*term frequency*) của một từ w trong một văn bản d , ký hiệu $TF(w,d)$ là số lần xuất hiện của từ w trong văn bản d .
- **Tần suất văn bản** (*document frequency*) của một từ w , ký hiệu $DF(w)$ là số lượng văn bản mà từ w có xuất hiện. Nghịch đảo của tần suất văn bản (*inverse document frequency*) của một từ w , ký hiệu $IDF(w)$ được cho bởi công thức:

$$IDF(w) = 1 + \log(|D| / DF(w))$$

trong đó $|D|$ là số lượng văn bản trong tập văn bản nguồn.

- **Tần suất TF-IDF** (*term document frequency*) là kết hợp của hai loại tần suất nói trên:

$$TF-IDF(w,d) = TF(w,d) * IDF(w)$$

Như vậy, chỉ số $TF(w)$ của một từ w cao khi từ đó xuất hiện nhiều trong văn bản, chỉ ra rằng nó có giá trị nội dung trong văn bản đó cao, còn chỉ số $IDF(w)$ của một từ w cao nếu từ đó xuất hiện trong ít văn bản, chỉ ra rằng từ đó có giá trị phân

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê biệt văn bản cao. Do vậy, các từ có giá trị $TF-IDF(w,d)$ sẽ đặc trưng cho một văn bản.

- **Tần suất TF-ISF** (*term sentence frequency*) cũng tương tự như tần suất $TF-IDF(w,d)$ nhưng khác nhau ở chỗ $TF-IDF$ đại diện cho giá trị từ w trong câu s chứ không phải trong văn bản d , ký hiệu $TF-ISF(w,s)$, được tính bởi công thức:

$$TF-ISF(w,s) = TF(w,s) * ISF(w)$$

trong đó $TF(w,s)$ là số lần xuất hiện của từ w trong câu s , và nghịch đảo $ISF(w)$ được cho bởi công thức:

$$ISF(w) = 1 + \log(|S| / SF(w)),$$

với tần suất câu $SF(w)$ là số lượng câu có chứa từ w , $|S|$ là số câu trong văn bản.

- **Tần suất trung bình của câu.** Với mỗi câu s , tần suất trung bình $TF-ISF$ của câu, ký hiệu $Avg-TF-ISF(s)$ được tính bằng trung bình số học $TF-ISF(w,s)$ của tất cả các từ w trong câu. Đó là:

$$Avg-TF-ISF(s) = \sum_{i=1}^{W(s)} TF-ISF(i,s) / W(s)$$

trong đó $W(s)$ là số lượng các từ trong câu.

2.2.1.2 Giải thuật lựa chọn câu có trị trung bình tần số cao nhất

Mô hình minh họa giải thuật như sau:

Bước 1:	Tách các thuật ngữ khỏi văn bản gốc.
Bước 2:	Đưa các thuật ngữ về cùng một dạng format, loại bỏ từ dừng.
Bước 3:	Duyệt từ đầu tới cuối văn bản, với mỗi thuật ngữ xuất hiện, lập ma trận trọng số wij tính tần số xuất hiện của thuật ngữ i trong câu j .
Bước 4:	Dựa vào ma trận wij , tính tần suất trung bình $Avg-TF-ISF(s)$ cho mỗi câu s trong văn bản.
Bước 5:	Tìm câu có giá trị $Avg-TF-ISF$ cao nhất.
Bước 6:	Trích rút những câu s có giá trị $Avg-TF-ISF(s) > Max Avg-TF-ISF * k$ với k là hệ số cho trước.

Hình 4: Giải thuật tóm tắt dựa trên trung bình trọng số cao nhất

Độ phức tạp của giải thuật là không lớn. Trong trường hợp xấu nhất là tích của số thuật ngữ và số câu trong văn bản. Neto và các đồng sự[4] khi áp dụng giải

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê thuật này cho hệ thống của mình đã so sánh kết quả của hệ thống với một hệ thống tóm tắt khác được đánh giá cao (CGI/CMU). Kết quả cho thấy hệ thống tuy đơn giản nhưng tóm tắt được xây dựng có tính khái quát nội dung rất cao (chưa kiểm chứng với tập mẫu).

2.2.2 Giải thuật dựa trên phân nhóm các đoạn văn trong văn bản (Paragraphs Clustering for Summarization)

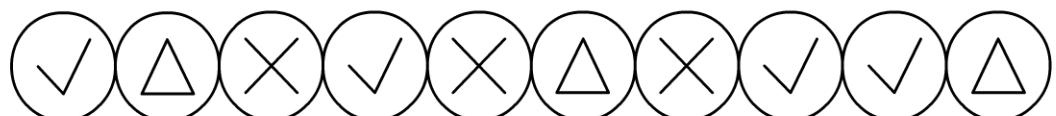
Giải thuật dựa trên phân nhóm đoạn văn (PCS) là phương pháp xây dựng tóm tắt bằng cách áp dụng bài toán phân nhóm văn bản (Text Clustering, xem chương I).

2.2.2.1 Định nghĩa phân nhóm.

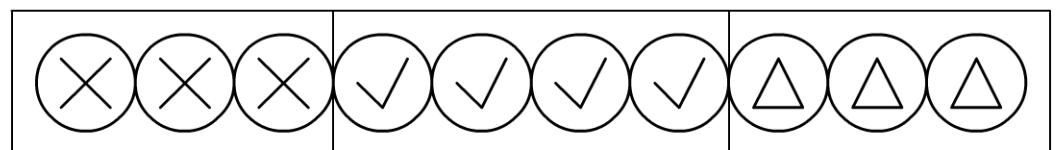
Phân nhóm là một hoạt động quan trọng của con người và nó thường hình thành cơ sở cho học tập và tri thức. Chẳng hạn, một đứa trẻ học cách phân biệt giữa động vật và thực vật hay giữa chim và cá bằng cách không ngừng cải thiện lược đồ phân loại tiềm thức. Cơ bản, lược đồ đó được rèn luyện bằng cách quan sát các đặc điểm hay tính chất của đối tượng.

Ví dụ mô tả việc phân loại các quả bóng có cùng dấu.

Cho 10 quả bóng với 3 loại dấu khác nhau (hình 5). Chúng ta phân các quả bóng thành 3 nhóm (3 cụm) bằng những dấu của chúng (hình 6).



Hình 5: Các quả bóng được đánh dấu theo thứ tự bất kỳ.



Hình 6: Đã phân nhóm

Bài toán Phân nhóm văn bản là bài toán thực hiện gom các văn bản từ một tập hợp văn bản ban đầu thành k nhóm (k cho trước hoặc tự chọn) nhằm cực đại hoá sự tương đồng giữa các văn bản trong cùng một nhóm và cực tiểu hoá sự tương đồng giữa các văn bản khác nhau.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

2.2.2.2 Giải thuật cho bài toán phân nhóm

Có rất nhiều các giải thuật khác nhau áp dụng cho bài toán Phân nhóm văn bản. Độ phức tạp của giải thuật tỷ lệ với độ lớn dữ liệu đầu vào mà nó có thể giải quyết. Ở đây chỉ xin giới thiệu hai giải thuật đơn giản nhưng cho độ chính xác cao bởi vì ứng dụng cho bài toán phân nhóm đoạn văn trong một văn bản là bài toán có điều kiện dữ liệu đầu vào nhỏ.

- Thuật toán K-Means

Đây là một trong những thuật toán kinh điển của Phân nhóm văn bản. Thuật toán này thực hiện phân hoạch tập các văn bản ban đầu thành các K nhóm không giao nhau, có nghĩa mỗi văn bản chỉ thuộc vào một nhóm duy nhất.

Bước 1:	Chọn K điểm trọng tâm của các nhóm một cách ngẫu nhiên
Bước 2:	Gắn tất cả các điểm dữ liệu tới trọng tâm gần nhất (có độ tương tự cao nhất). Lúc này đã hình thành k nhóm
Bước 3:	Gắn lại trọng tâm cho mỗi nhóm
Bước 4:	Lặp lại bước 2 và bước 3 cho đến khi các trọng tâm không còn thay đổi hoặc sau một số bước lặp nhất định

Hình 7: Thuật toán K-Means

Trong thuật toán K-means, để biểu diễn văn bản và tính độ tương tự giữa các văn bản với nhau, mô hình véc tơ thưa được ưa chuộng sử dụng nhất (sẽ trình bày cụ thể mô hình VSP trong chương sau).

- Thuật toán lập nhóm theo cây phân cấp (Hierarchical Clustering - HC)

Thuật toán lập nhóm theo cây phân cấp tạo ra các phân hoạch với các nhóm lồng nhau, nhóm ở mức dưới là một tập con của nhóm ở mức trên. Có hai giải thuật phân cấp phục vụ cho phân nhóm văn bản:

Bước 1:	Ban đầu mỗi văn bản được coi như một nhóm
Bước 2:	Tính độ tương tự giữa tất cả các nhóm với nhau
Bước 3:	Chọn ra 2 nhóm có độ tương tự cao nhất, kết hợp chúng lại thành một nhóm mới đồng thời loại bỏ 2 nhóm đó
Bước 4:	Lặp lại bước 2 và bước 3 cho đến khi chỉ còn 1 nhóm duy nhất chứa toàn bộ các văn bản

Hình 8: Thuật toán cây phân cấp dưới lên

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

Thuật toán cây phân cấp trên xuống cũng tương tự như thuật toán cây phân cấp dưới lên, nhưng bước ban đầu quy tập tất cả các văn bản vào một nhom, sau đó mỗi bước chọn một nhom trong các nhom để phân chia thành hai nhom con theo một điều kiện nào đó. Quá trình kết thúc khi mỗi văn bản đã thuộc một nhom khác nhau.

2.2.2.3 Áp dụng phân nhom văn bản cho bài toán TTVB

Điểm cốt yếu của giải thuật này nằm ở chỗ coi văn bản như là một tập hợp văn bản và các đoạn văn như những văn bản con nằm trong tập hợp văn bản đó. Mô hình đơn giản của hệ thống có thể được thực hiện như sau:

Bước 1: Tiền xử lý văn bản

Đầu vào: văn bản gốc

Đầu ra: biểu diễn của các đoạn văn trong văn bản theo mô hình véc tơ thưa. Mỗi đoạn văn được biểu diễn dưới dạng một véc tơ.

Bước 2: Áp dụng phân nhom văn bản để phân nhom các đoạn văn.

Đầu vào: biểu diễn véc tơ thưa của m đoạn văn trong văn bản gốc

Đầu ra: m đoạn văn được phân thành k nhom ($0 < k < m$)

Bước 3: Trích rút câu tạo tóm tắt

Đầu vào: k nhom đoạn văn

Đầu ra: k câu được trích rút từ k nhom trên.

Hình 9: Áp dụng phân nhom văn bản để thực hiện tóm tắt

Đối với bước 3, phương pháp trích câu có thể là sử dụng là

- Rút ra câu đầu tiên xuất hiện trong một đoạn văn.
- Rút ra câu chính giữa trong một đoạn văn.
- Rút ra câu có độ tương tự lớn nhất với véc tơ đặc trưng của nhom.

2.2.2.4 Đánh giá

Giải thuật này được Kathleen R. McKeown và đồng sự ứng dụng trong hệ thống tóm tắt SIMFINDER được thực hiện năm 2001[5]. Các tác giả còn áp dụng một số phương pháp phân nhom khác nhằm cho kết quả tốt hơn so với hai phương pháp cơ bản trình bày ở trên. Các tác giả cho rằng kết quả của hệ thống tóm tắt phụ thuộc nhiều vào kết quả phân nhom

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Tuy nhiên giải thuật này gặp phải một số hạn chế có thể dễ thấy như:

- Chỉ tóm tắt được các văn bản có cấu tạo gồm nhiều đoạn văn.
- Nếu số đoạn văn nhỏ hơn so với số câu cần có trong tóm tắt => phải chọn nhiều hơn một câu trong một nhóm: kết quả thường không chính xác.

Có một số phương hướng giải quyết hạn chế này, ví dụ như: thực hiện phân nhóm trên các câu chứ không trên các đoạn văn. Tuy nhiên hướng giải quyết này chưa được chứng minh tính đúng đắn và có vẻ nó cũng có độ chính xác không cao như khi phân nhóm trên các đoạn văn.

2.2.3 Giải thuật sử dụng các đặc trưng tóm tắt kết hợp thuật toán học máy (Summarization using Machine Learning Algorithm)

Giải thuật sử dụng thuật toán học máy (SMLA) là giải thuật khá phổ biến, và đã có nhiều nghiên cứu phát triển dựa trên nền tảng này. Bởi vì nó thể hiện rất rõ các đặc trưng, tính chất của công việc TTVB thực sự. Nó được coi như là một phương pháp “vét nõn” để tìm ra kết quả tốt nhất có thể cho tóm tắt Extract.

Một trong những người nghiên cứu đầu tiên về giải thuật này phải kể đến là Julian Kupiec (1995)[13]. Phương pháp mà Kupiec đưa ra tuy kết hợp chưa nhiều các đặc trưng tóm tắt xong nó là cơ sở giải thuật để các nghiên cứu khác có thể phát triển thêm sau này. Dưới đây xin trình bày những điểm mấu chốt của giải thuật phát triển theo hướng này.

2.2.3.1 Các đặc trưng của tóm tắt (Summarized Features)

Đặc trưng của tóm tắt (SF) là một đặc điểm nào đó của một thành phần trong văn bản cho thấy nó có giá trị về nội dung cao và có nhiều khả năng được sử dụng để tạo nên TTVB.

Ví dụ trong giải thuật dựa vào tính giá trị trung bình tần suất ở trên, ta chọn những câu có giá trị *Avg-TF-ISF* cao để đưa vào tóm tắt. Suy ra *Avg-TF-ISF* cũng là một đặc trưng của TTVB.

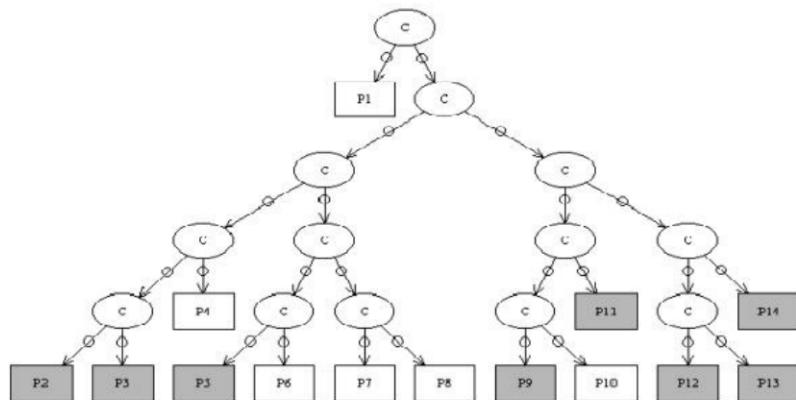
Có rất nhiều đặc trưng tóm tắt, có thể nêu ra cơ bản một số đặc trưng sau:

- ❖ **Độ dài câu** (Sentence Length feature) Đặc trưng này chỉ ra rằng những câu có độ dài quá ngắn (có số từ hoặc số ký tự ngắn hơn một độ dài cho trước nào đó) khó có thể được sử dụng để tạo Tóm tắt.
- ❖ **Vị trí câu** (Sentence Position feature) Đặc trưng này liên quan tới khả năng câu chứa ý chính có vị trí đặc biệt nào đó trong văn bản, hay trong đoạn văn thuộc văn bản. Ví dụ: Một hoặc hai câu đầu tiên của mỗi văn bản,

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

mỗi đoạn văn có khả năng cao để tạo tóm tắt. Một vài câu gần cuối cùng của văn bản, đoạn văn cũng có giá trị tương tự. Tuy nhiên câu cuối cùng thì không bao giờ được sử dụng để tạo tóm tắt.

- ❖ **Chứa nội dung tiêu đề** (Title feature). Nếu câu nào đó chứa các thuật ngữ xuất hiện trong tiêu đề thì nó có nhiều khả năng được sử dụng để tóm tắt
- ❖ **Chứa các thuật ngữ đặc biệt** (Fixed-phrases feature). Đặc trưng này chỉ ra rằng nếu các câu có chứa các *thuật ngữ tóm lược* (Cue phrases) như “tóm lại”, “tổng quát”, “tổng hợp”,... hoặc các *thuật ngữ nhấn mạnh* (emphasizer) như “quan trọng”, “riêng biệt”,... thì chúng đều có khả năng rất cao được sử dụng để tạo tóm tắt.
- ❖ **Từ viết hoa** (Uppercase word feature). Từ viết hoa thường là viết tắt cho một thuật ngữ dài hoặc một tên riêng nào đó. Ví dụ VCB là viết tắt của VietCom Bank. Thực tế cho thấy các câu chứa các định nghĩa viết hoa cũng hay chứa những nội dung quan trọng có thể được sử dụng trong tóm tắt.
- ❖ **Dựa trên cây nhị phân** (Binary Tree). Cây nhị phân được sử dụng để tính độ tương tự giữa các thành phần liền kề nhau trong một văn . Vị trí của một câu trong cây nhị phân xác định độ tương quan về nội dung với các thành phần liền kề nó, qua đó có thể xác định khả năng nó có được sử dụng để tóm tắt hay không



Hình 10: Ví dụ về cây nhị phân

Còn rất nhiều đặc trưng của văn bản có thể sử dụng để hỗ trợ tóm tắt. Vấn đề đặt ra ở chỗ kết hợp các đặc trưng này để xây dựng tóm tắt như thế nào.

2.2.3.2 Kết hợp các đặc trưng (Features Combination) để tạo tóm tắt

Với mỗi đặc trưng tóm tắt được liệt kê sử dụng, mỗi văn bản đầu vào sẽ cho ra kết quả theo mô hình sau:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

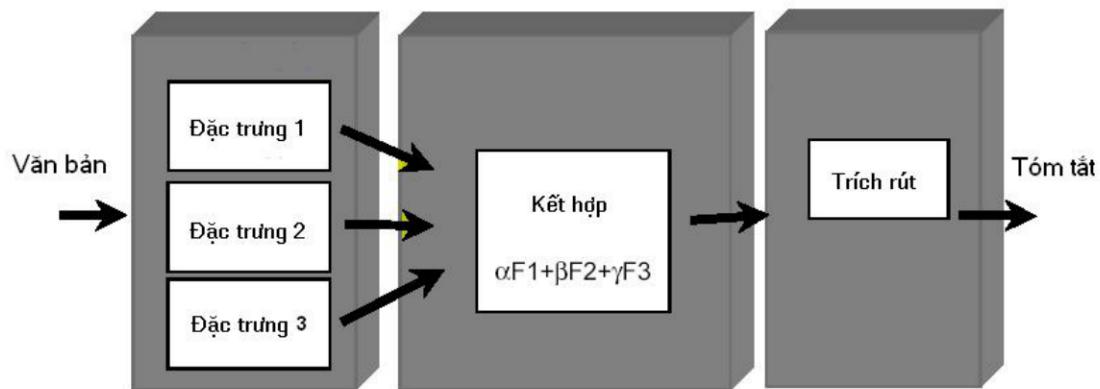
Đặc trưng F*

Đầu vào: Văn bản d $\{s_1, s_2, \dots, s_n\}$.

Đầu ra: Dãy các trọng số $w(s_1), w(s_2), \dots, w(s_n)$ đánh giá giá trị của câu/thành phần văn bản theo đặc trưng F^*

Hình 11: Vào - ra với mỗi đặc trưng tóm tắt.

Để kết hợp các đặc trưng lại với nhau thực hiện TTVB, sử dụng mô hình kết hợp:



Hình 12: Mô hình kết hợp các đặc trưng tóm tắt

Vậy có thể mô tả giải thuật kết hợp như sau:

Kết hợp các đặc trưng

Đầu vào: Văn bản d $\{s_1, s_2, \dots, s_n\}$ cùng các đặc trưng F_1, F_2, \dots, F_m và các hệ số k_1, k_2, \dots, k_m .

Đầu ra: Dãy các trọng số $W(s_1), W(s_2), \dots, W(s_n)$ đánh giá giá trị của câu/thành phần văn bản tham gia tóm tắt:

$$W(s_i) = k_1 w_1(s_i) + k_2 w_2(s_i) + \dots + k_m w_m(s_i)$$

Hình 13: Vào - ra kết hợp các đặc trưng tóm tắt.

Ở bước trích rút, các câu có giá trị trọng số cao nhất được rút ra theo một tỷ lệ định trước.

2.2.3.3 Áp dụng giải thuật học máy (Machine Learning Algorithm)

Vẫn đề đặt ra đối với mô hình kết hợp đặc trưng ở chỗ không thể biết trước được sự kết hợp những đặc trưng nào sẽ cho kết quả tóm tắt tốt. Điều này có thể

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê được giải quyết bằng cách sử dụng một tập mẫu các văn bản đã được tóm tắt sẵn và áp dụng các giải thuật học máy để rút ra một sự kết hợp tốt nhất các đặc trưng có thể. Mục đích của giải thuật học máy là để tìm ra các hệ số k_i cho mỗi đặc trưng F_i .

Có thể kể ra một vài giải thuật học máy phổ biến nhất như: giải thuật sử dụng các luật thống kê Naïve Bayes; giải thuật C4.5; giải thuật SCDF; giải thuật AQ. Trong số này, giải thuật áp dụng các luật Bayes được sử dụng rộng rãi nhất vì hiệu quả cao của nó.

Giả sử các giá trị k_i chỉ là 0 hoặc 1, có thể sử dụng luật xác suất Bayes để quyết định k_i . Luật xác suất Bayes :

Ký hiệu $P(A)$ là xác suất xảy ra A; $P(A|B)$ là xác suất xảy ra A khi đã có B. Ta có:

$$P(A|B) = \frac{P(B|A) \times P(A)}{P(B)}$$

Vì vậy, ta có thể tính xác suất một câu s thuộc văn bản gốc có nằm trong tóm tắt S của văn bản sử dụng các đặc trưng F_1, F_2, \dots, F_k đó không:

$$P(s \in S | F_1, F_2, \dots, F_k) = \frac{P(F_1, F_2, \dots, F_k | s \in S) \times P(s \in S)}{P(F_1, F_2, \dots, F_k)}$$

$$P(s \in S | F_1, F_2, \dots, F_k) = \frac{\prod_{j=1}^k P(F_j | s \in S) \times P(s \in S)}{\prod_{j=1}^k P(F_j)}$$

Giá trị $P(s \in S)$ là một hằng số (bằng hệ số rút gọn)

Giá trị $P(F_j | s \in S)$ và $P(F_j)$ có thể được tính theo các tập mẫu văn bản đã được tóm tắt.

2.2.3.4 Đánh giá

Có thể thấy rõ giải thuật áp dụng thuật toán học máy là một phương pháp rất tổng quát để giải quyết bài toán TTVB. Tuỳ vào mỗi đặc điểm riêng của từng loại ngôn ngữ, từng loại văn bản mà phương pháp này sẽ đưa ra một kết hợp các đặc trưng tóm tắt có hiệu quả tốt nhất. Có thể nhận thấy hai giải thuật đã trình bày ở trên cũng được coi như là một “đặc trưng” tóm tắt phức tạp. Do đó chúng cũng có thể được áp dụng và kết hợp với các đặc trưng đơn giản nhất. Xong thực tế chúng minh không phải càng nhiều đặc trưng kết hợp với nhau thì kết quả cho ra càng tốt.

Để đánh giá kết quả của hệ thống xây dựng trên phương pháp này, Kupiec đã thực hiện các thử nghiệm đánh giá kết quả khi kết hợp 5 đặc trưng: *Độ dài câu, Vị trí câu, Thuật ngữ đặc biệt, Từ viết hoa và Trọng số câu*. Kết quả cho thấy hệ thống chính xác nhất khi kết hợp 2 đặc trưng *Độ dài câu* và *Thuật ngữ đặc biệt*. Với tỷ lệ rút gọn 10%, độ chính xác của tóm tắt là vào khoảng trên dưới 40%.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Các giải thuật được trình bày ở trên có thể được ghép vào thành một nhóm: các giải thuật dựa trên thông kê các từ trong văn bản và lượng giá ý nghĩa của thông kê này. Bởi vì chỉ “đọc” mà không “hiểu” văn bản nên các giải thuật này được xếp vào loại “giải thuật nông” (shallow approaches).

Dưới đây xin trình bày một số phương pháp tóm tắt dựa trên việc phân tích ngữ nghĩa và các đặc tính ngôn ngữ học của văn bản (discourse features), có thể xếp vào loại “giải thuật sâu” (deep approaches).

2.2.4 Giải thuật áp dụng các đặc trưng liên kết ngữ nghĩa trong văn bản (Summarization using Cohesion Features)

2.2.4.1 Các định nghĩa cơ bản

- **Cohesion:** trong văn bản có các liên kết giữa các thành phần của văn bản để biểu hiện quan hệ về mặt ngữ nghĩa. Chúng được gọi là Cohesion. Có hai loại liên kết Cohesion trong văn bản: liên kết về mặt ngữ pháp (Gramatical Cohesion) và liên kết về mặt từ vựng (Lexical Cohesion)
- **Gramatical Cohesion:** là các liên kết về nội dung trong văn bản được tạo ra trong ngữ cảnh cụ thể với cấu trúc ngữ pháp của các câu.

Ví dụ: Hùng có một chiếc ô tô. **Nó** rất đẹp.

Ở đây giữa “ô tô” và “nó” có một liên kết. Liên kết này được phát hiện và chỉ tồn tại trong ngữ cảnh cụ thể này.

- **Lexical Cohesion:** là các liên kết về nội dung trong văn bản được tạo ra bởi sự đồng nhất về ý nghĩa của các từ vựng.

Ví dụ: Hùng rất thích ô tô. Anh ấy đã mua một chiếc xe hơi riêng.

Liên kết tồn tại trong tình huống này “ô tô” và “xe hơi” là do chúng mang ý nghĩa tương đương nhau.

- **Lexical Chain:** chuỗi từ vựng.

Khái niệm của các chuỗi từ vựng được giới thiệu đầu tiên bởi Morris và Hirst. Các chuỗi từ vựng cơ bản khai thác sự kết dính giữa một số từ có liên hệ với nhau (Morris và Hirst 1991). Chuỗi các từ vựng có thể được thực hiện trong một tài liệu nguồn bằng cách nhóm những tập hợp những từ có liên hệ với nhau về nghĩa. Sự đồng nhất, đồng nghĩa và sự khái quát là những mối tương quan giữa các từ, chúng có thể nhóm các từ đó vào cùng một chuỗi từ vựng. Đặc biệt, các từ có thể nhóm lại khi:

- Hai danh từ giống nhau và được dùng cùng hướng như nhau:
(Ngôi nhà này rất đẹp. **Ngôi nhà** được làm từ gỗ)

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

- Hai danh từ được dùng với cùng hướng như nhau:

(**Con chó** chạy nhanh. **Chiếc ô tô** của tôi nhanh hơn)

- Hướng sử dụng của hai danh từ có mối liên hệ cao thấp giữa chúng. (Tôi có một chiếc xe **Honda**. Nó là một chiếc **Future**)
- Hướng sử dụng của hai danh từ là anh em ruột trong mối quan hệ cao thấp thuộc dạng cây. (**Cái xe ba gác** chạy rất nhanh. **Chiếc ô tô** chạy nhanh hơn).

Trong việc thực hiện các chuỗi từ vựng, các cá thể danh từ phải được nhóm theo những mối liên hệ trên, nhưng mỗi danh từ phải chỉ thuộc về một chuỗi từ vựng. Có một vài khó khăn trong việc xác định một danh từ nên thuộc vào chuỗi từ vựng nào. Chẳng hạn, một danh từ có thể tương ứng với vài hướng sử dụng từ khác nhau, và vì thế hệ thống phải quyết định hướng nào để sử dụng (ví dụ: một trường hợp cụ thể của “nhà” phải được hiểu theo hướng sử dụng 1, tức nơi để ở, hay hướng sử dụng 2, tức cơ quan lập pháp).Thêm vào đó, ngay cả nếu hướng sử dụng từ của một cá thể từ nào đó có thể được xác định, chúng ta cũng có thể nhóm các cá thể từ đó vào những chuỗi từ vựng khác nhau bởi vì nó có thể có liên quan đến những từ trong những chuỗi khác. Ví dụ, hướng sử dụng của một từ có thể giống của từ khác trong một nhóm trong khi có thể có mối liên hệ cao thấp với hướng sử dụng của một từ trong một nhóm khác. Điều quan trọng phải đạt được là những từ phải được nhóm lại sao cho sự nhóm nói chung là tối ưu trong việc tạo thành những chuỗi từ vựng dài nhất/mạnh nhất có thể. Vì vậy có thể định nghĩa: những từ được nhóm vào cùng một chuỗi khi chúng là “sắp sửa” có cùng khái niệm cơ bản.

2.2.4.2 Liên kết ngữ nghĩa ứng dụng trong TTVB

Phương pháp áp dụng liên kết ngữ nghĩa trong văn bản có thể được mô tả tổng quát gồm hai giai đoạn như sau:

Giai đoạn 1: Biểu diễn văn bản dưới dạng đồ thị trong đó:

- Nút: là các từ vựng, thuật ngữ; các câu hoặc các đoạn văn.
- Cạnh giữa các nút: Cạnh có trọng số hoặc không có trọng số biểu thị mối tương quan liên kết về mặt ý nghĩa nội dung của các nút với nhau.

Giai đoạn 2: Từ biểu diễn bằng đồ thị, lựa chọn và lấy ra các thành phần có liên kết nhiều nhất tương đương với việc nó mang nội dung chính của văn bản.

Như vậy có thể thấy giải thuật TTVB dựa trên phân nhóm văn bản đã trình bày ở phần trên thực chất là xuất phát từ mô tả tổng quát này. Tuy vậy nó chỉ xét

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê tới các thành phần văn bản (cụ thể là thuật ngữ) được lặp lại chứ không xét trên các liên kết khác nếu có giữa chúng.

Trong các giải thuật áp dụng liên kết ngữ nghĩa để TTVB, phương pháp sử dụng các chuỗi từ vựng được nghiên cứu nhiều nhất.

2.4.2.3 Giải thuật áp dụng chuỗi từ vựng để TTVB (Summarization using Lexical Chains)

Giải thuật này được trình bày đầu tiên bởi Regina Barzilay và Michael Elhadad (Using Lexical Chains for Text Summarization - 1997). Điểm mấu chốt của giải thuật này là xây dựng các chuỗi từ vựng từ văn bản gốc sao cho độ dài các chuỗi này là lớn nhất, sau đó ghi điểm và chọn ra các chuỗi mạnh. Tóm tắt được trích rút từ văn bản gốc bằng cách với mỗi chuỗi mạnh, tìm một câu chứa nội dung liên quan tới chuỗi từ vựng đó. Trong giải thuật của mình, Barzilay có đề cập tới việc sử dụng thư viện WordNet (mỗi từ được giải nghĩa theo nhiều hướng sử dụng. Mỗi hướng sử dụng được biểu thị bởi một tập hợp các từ đồng nghĩa. Tập hợp đó gọi là *synset*).

Cụ thể, Barzilay[7] đưa ra giải thuật:

Bước 1: Đọc văn bản và lọc ra một tập hợp các thuật ngữ là các danh từ.

Bước 2: Với mỗi thuật ngữ tìm được ở bước 1 thực hiện:

(a). Dựa vào WordNet tìm xem các chuỗi từ vựng với hướng sử dụng cụ thể đã có có liên quan tới thuật ngữ không. Nếu có sang (b), nếu không sang (c).

(b). Nếu có nhiều hơn một chuỗi từ vựng đã có liên quan tới thuật ngữ, chọn các liên kết mạnh nhất để đưa thuật ngữ này vào chuỗi từ vựng đó. Cập nhật lại chuỗi từ vựng và hướng sử dụng.

(c). Nếu không có, thêm một chuỗi từ vựng mới chỉ bao gồm thuật ngữ này và tất cả các hướng sử dụng có thể của nó.

Bước 3: Tính điểm cho mỗi chuỗi từ vựng:

$$Score(chain) = Length * HI$$

Bước 4: Chọn ra các chuỗi có điểm cao nhất. Với mỗi chuỗi này, thực hiện tìm và rút trong văn bản câu đầu tiên chứa một thành phần của chuỗi.

Hình 14: Giải thuật TTVB dựa theo chuỗi từ vựng

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

2.4.2.3 Đánh giá

Trong các nghiên áp dụng chuỗi từ vựng để TTVB sau này đều có áp dụng một số kỹ thuật khác để tăng hiệu quả và giảm tốc độ tính toán các chuỗi từ vựng. Kết quả của phương pháp này đối với TTVB được đánh giá cao xong khả năng áp dụng đối với bài toán Tóm tắt tiếng Việt gấp nhiều hạn chế bởi hai vấn đề:

- Chưa có một thư viện WordNet tiếng Việt.
- Sự phân biệt giữa các danh từ, động từ, trợ từ,... trong ngữ pháp tiếng Việt là rất phức tạp chứ không được thực hiện đơn giản như tiếng Anh.

2.2.5 Giải thuật áp dụng các đặc trưng liên kết cấu trúc trong văn bản (Summarization using Coherence Features)

2.2.5.1 Khái niệm về liên kết cấu trúc (Coherence).

Coherence: Trong văn bản có các liên kết giữa các thành phần của văn bản để biểu hiện quan hệ về mặt cấu trúc nội dung. Chúng được gọi là các liên kết coherence. Có thể phân ra các loại liên kết coherence sau:

- **Liên kết theo cấu trúc định dạng tài liệu** (Document format)
Ví dụ: cấu trúc một văn bản gồm nhiều chương, nhiều phần => các chương, các phần có mối quan hệ liên kết cấu trúc định dạng tài liệu với nhau.
- **Liên kết theo cấu trúc chủ đề** (Topic structure)
- **Liên kết theo cấu trúc tu từ** (Rhetorical structure). Đây là khái niệm liên kết cấu trúc quan trọng nhất. Có thể hiểu liên kết tu từ là loại liên kết giữa các thành phần văn bản có liên hệ bổ trợ cho nhau về mặt nội dung.

Ví dụ: Anh ấy làm việc rất chăm chỉ. Vì vậy anh ấy được thăng chức.

Rõ ràng mệnh đề sau là kết quả của mệnh đề trước, được phát hiện qua từ “vì vậy”. Hai mệnh đề này có mối liên kết theo cấu trúc tu từ với nhau.

- **Liên kết theo cấu trúc kể** (narrative structure). Các thành phần liên kết về mặt nội dung tiếp diễn nhau.

2.2.5.2 Áp dụng liên kết cấu trúc cho TTVB.

Để áp dụng liên kết cấu trúc vào TTVB, trước hết cần phải thực hiện giải quyết bài toán phân tích cú pháp văn bản. Đây là một bài toán có độ tính toán cao và đòi hỏi những phân tích ngôn ngữ học rất phức tạp.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê
Áp dụng các phương pháp này cho bài toán TTVB tiếng Việt cần những
nghiên cứu rộng và xa hơn nữa.

2.2.6 Kết luận

Như vậy, có thể thấy bài Toán TTVB có rất nhiều hướng giải quyết từ đơn giản đến phức tạp. Trên đây chỉ là trình bày ngắn gọn về một số hướng giải quyết khác nhau này. Tác giả sẽ áp dụng chúng để xây dựng một hệ thống TTVB tiếng Việt trong các phần sau.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

CHƯƠNG III

TIỀN XỬ LÝ VĂN BẢN TIẾNG VIỆT

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

Như đã đề cập, tiền xử lý văn bản chiếm một vai trò rất quan trọng trong Khai thác văn bản. Nó là bước mở đầu cho mỗi hướng giải quyết một bài toán nào đó. Các bước tuần tự cho quá trình tiền xử lý văn bản tiếng Việt được trình bày dưới đây.

3.1 Phương pháp tách thuật ngữ tiếng Việt

Từ một văn bản ban đầu, các từ phải được tách ra thành các thuật ngữ theo từ điển. Mỗi thuật ngữ là một từ hoặc một cụm từ (ngữ) có nghĩa.

Về từ, tiếng Việt ta có các từ loại sau:

1. Danh từ : nhà cửa, ...
2. Động từ : nhìn, ...
3. Tính từ : xinh đẹp, ...
4. Đại từ : tôi, ...
5. Số từ : một, hai, ...
6. Loại từ : con, cái, ...
7. Quán từ : các, những, ...
8. Trạng từ : trên, dưới, ...
9. Liên từ : và, hay, ...
10. Giới từ : cùng, với, ...
11. Phó từ : đã, sẽ, ...
12. Trợ từ : nhỉ, nhé, ...
13. Lai từ : súp văng tơ, gi đông, ...

Các loại từ này lại được phân loại theo cách biểu diễn:

- Từ đơn : là từ một tiếng.
- Từ phức : là từ gồm hai tiếng trở lên.
 - Từ ghép :
 - Từ ghép chính phụ : hoa hồng, bài học,
 - Từ ghép đồng lập : nhà cửa, đường sá,
 - Từ láy : sạch sành sanh, linh tinh,
 - Từ phức ngẫu kết : tắc kè, bù nhìn,

Về ngữ, có các loại cơ bản sau:

- Ngữ danh từ : ngữ có danh từ là trọng tâm như ‘lớp mót’.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

- Ngữ vị từ : ngữ có động từ hoặc tính từ là trọng tâm như '*nóng như lửa*'.
- Ngữ giới từ : ngữ bắt đầu là giới từ như '*trong nhà*'.

Ngoài ra tiếng Việt còn có một loại ngữ đặc biệt gọi là thành ngữ như '*con ông cháu cha*'.

Trong tiếng Anh hầu như không có những từ ghép mà các thành phần của từ đó không làm nên ý nghĩa của từ đó, tức ý nghĩa của từ ghép do ý nghĩa của những từ đơn tạo thành. Nhưng tiếng Việt thì khác, từ ghép có rất nhiều trong đó có rất nhiều từ ghép kết hợp ngẫu nhiên, ý nghĩa không phải do ý nghĩa của các từ đơn hợp thành ví dụ như '*bồ câu*'.

Như vậy, ý nghĩa cơ bản của việc tách thuật ngữ là xác định được trong văn bản đâu là các từ, đâu là các ngữ chính xác, phân chia ra để từ đó biểu diễn văn bản.

Thuật toán 1:

- 1) Vị trí hiện tại bắt đầu từ đầu văn bản.
- 2) Từ vị trí hiện tại, đọc vào một mảng tạm có độ dài bằng từ dài nhất có trong từ điển.
- 3) Hiệu chỉnh lại mảng tạm để mảng chứa một số nguyên tử đơn.
- 4) Kiểm tra mảng có đang chứa một từ thuộc từ điển không, nếu đúng thì ta tìm được một từ.
- 5) Dịch vị trí hiện tại đi một khoảng bằng chiều dài của từ vừa tìm được.
- 6) Quay lại bước 2 đến hết văn bản.

Thuật toán này nếu áp dụng cho tiếng Việt sẽ phân tích được thiểu thuật ngữ.

Ví dụ: '*Quần áo may rất đẹp*' sẽ tách được những thuật ngữ sau '*quần áo, may, rất, đẹp*' nhưng ta thấy rằng hai thuật ngữ '*quần, áo*' là có ý nghĩa trong câu trên. Vậy suy ra thiểu thuật ngữ. Trên thực tế những từ ghép gây mất thuật ngữ như '*quần áo*' có rất nhiều trong tiếng Việt. Để tránh việc mất thuật ngữ người ta đã đưa ra thuật toán 2.

Thuật toán 2:

- 1) Vị trí hiện tại bắt đầu từ đầu văn bản.
- 2) Từ vị trí hiện tại, đọc vào một mảng tạm có độ dài bằng từ dài nhất có trong từ điển.
- 3) Hiệu chỉnh lại mảng tạm để mảng chứa một số nguyên tử đơn.
- 4) Kiểm tra mảng có đang chứa một từ thuộc từ điển không, nếu đúng thì ta tìm được một từ.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

- 5) Thực hiện loại bớt một từ đơn ở cuối mảng nếu mảng còn chứa nhiều hơn một từ đơn, nếu mảng chỉ còn chứa một từ đơn thì nhảy tới bước 7.
- 6) Quay lại bước 4.
- 7) Dịch vị trí hiện tại đi một khoảng bằng chiều dài của từ vừa tìm được.
- 8) Quay lại bước 2 đến hết văn bản.

Thuật toán này sẽ tìm thửa thuật ngữ, tức nó sẽ chấp nhận cả những thuật ngữ không mang ý nghĩa trong câu.

Ví dụ: ‘*Bồ câu là biểu tượng cho hòa bình*’ theo thuật toán phân tích thuật ngữ trên ta sẽ thu được những thuật ngữ sau ‘*bồ*, *câu*, *là*, *biểu*, *tượng*, *cho*, *hòa*, *bình*’, chúng ta có thể thấy rằng khá nhiều thuật ngữ thu được không có ý nghĩa trong câu trên như ‘*bồ*, *câu*, *biểu*, *tượng*, *hòa*, *bình*’.

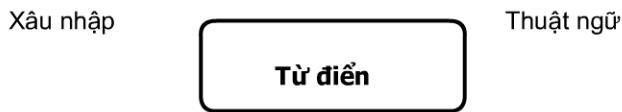
Có một số thuật toán cài tiến từ thuật toán 2 để giải quyết sai sót này, xong chúng đều khó khả thi vì cần các công thức tính toán phức tạp hoặc phải sử dụng từ điển đồng nghĩa để tách thuật ngữ. Do vậy tác giả không đề cập đến bởi vẫn chưa có từ điển đồng nghĩa cho tiếng Việt. Mặt khác các thuật toán 1,2 cũng giải quyết cơ bản nhu cầu tách thuật ngữ với độ chính xác chấp nhận được.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

3.2 Xây dựng từ điển

Rõ ràng để thực hiện tốt việc tách thuật ngữ tiếng Việt, cần một cách lưu trữ tập mẫu các thuật ngữ hợp lý nhất. Cách lưu trữ này vừa phải tiết kiệm được bộ nhớ đồng thời khi thực hiện so sánh với các thuật ngữ lấy từ văn bản phải cho thời gian ngắn. Cách lưu trữ và xử lý này được gói gọn trong một cấu trúc từ điển.

Hoạt động chính của từ điển là khi đưa một xâu đầu vào, cần phải xác định xem đó có phải là một thuật ngữ không. hỏi từ điển một xâu có phải là một thuật ngữ hay không, nếu có thì đưa ra thông tin của thuật ngữ đó.



Hình 15. Hoạt động của từ điển.

Hoạt động của từ điển phụ thuộc rất nhiều vào cách tổ chức dữ liệu trong chúng. Nếu dữ liệu được tổ chức hiệu quả, khả năng xử lý của các bài toán có cơ sở dữ liệu lớn được tăng lên đáng kể.

Tổ chức dữ liệu của một từ điển có thể phân chia ra hai nhiệm vụ:

- Thứ nhất, định nghĩa cấu trúc bản ghi của dữ liệu. Ví dụ một thuật ngữ “máy tính” được có thể được lưu trữ đơn giản dưới dạng một chuỗi chứa từ “máy tính” hoặc một cấu trúc nào khác cũng có khả năng giúp hệ thống ánh xạ tới từ “máy tính”.
- Thứ hai, tổ chức kết cấu các bản ghi này theo hệ thống nhằm giảm bớt các bước so sánh khi tìm kiếm. Do đó, còn có thể gọi nó là tổ chức tìm kiếm. Thường thì hệ thống được tổ chức theo kết cấu danh sách, cây nhị phân hoặc theo bảng băm để giảm thiểu thời gian tìm kiếm.

3.2.1 Tổ chức cấu trúc bản ghi trong từ điển

Mỗi bản ghi dùng để lưu trữ một thuật ngữ trong từ điển. Đơn giản nhất là trực tiếp lưu luôn thuật ngữ đó như là một trường trong bản ghi. Tuy nhiên đối với các hệ thống lớn lưu trữ một lượng dữ liệu khổng lồ, khi cần đòi hỏi phải tổ chức cấu trúc bản ghi dưới dạng nhỏ nhất có thể. Vì vậy các thuật ngữ phải được mã hoá để lưu trữ sao cho có hiệu quả nhất. Dưới đây là một cách tổ chức ánh xạ dữ liệu để lưu trữ các thuật ngữ tiếng Việt.

Đối với đặc trưng tiếng Việt, có thể thấy mỗi một từ đơn đều có cấu trúc sau:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Cụm phụ âm đầu + Cụm nguyên âm + Cụm phụ âm cuối.

Phân tích cho thấy có tất cả 26 cụm phụ âm đầu và 9 cụm phụ âm cuối.

STT	Cụm phụ âm đầu
1	ϕ
2	b
3	c
4	ch
5	d
6	đ
7	g
8	gh
9	h
10	kh
11	l
12	m
13	n
14	nh
15	ng
16	ngh
17	p
18	ph
19	qu
20	r
21	s
22	t
23	th
24	v
25	x

Bảng 1: Các cụm phụ âm đầu

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

STT	Cụm phụ âm cuối
1	ϕ
2	c
3	ch
4	m
5	n
6	ng
7	nh
8	p
9	t

Bảng 2: Các cụm phụ âm cuối

Đối với cụm nguyên âm, lại được chia ra làm 3 loại: loại không đi kèm với cụm phụ âm cuối; loại có đi kèm với cụm phụ âm cuối và loại trung gian (có thể đi kèm hoặc không)

Không đi kèm	Trung gian	Có đi kèm
au	a	iê
ai	ă	oă
ao	â	oâ
ay	e	oê
âu	ê	oo
ây	o	uô
eo	oa	uyê
êu	oe	uơ
ia	ô	
iêu	ơ	
iu	u	
oi	uê	

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

oai	uâ	
oay	ư	
ôi	y	
ơi		
ua		
uôi		
ui		
uy		
uroi		
ura		
urou		
uri		
uru		
yêu		

Bảng 3: Các cụm nguyên âm

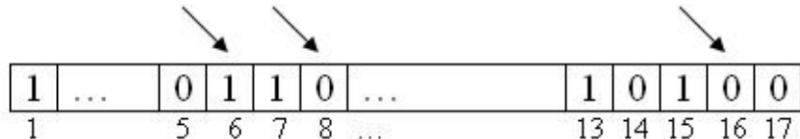
Như vậy có tất cả 26 cụm nguyên âm không đi kèm phụ âm (26×6 nếu tính cả dấu), 23 cụm nguyên âm có thể đi kèm phụ âm (23×6 nếu tính cả dấu). Theo cách phân tích này, có thể dùng 17 bit để lưu giữ bất kỳ một từ đơn tiếng Việt nào, trong đó:

- 5 bit đầu để lưu trữ cụm phụ âm đầu (cần 26 giá trị).
- 8 bit để lưu giữ cụm nguyên âm trong trường hợp đây là cụm nguyên âm không đi kèm (cần 156 giá trị).
- 8 bit để lưu giữ cụm nguyên âm trong trường hợp đây là cụm nguyên âm có đi kèm (cần 138 giá trị).
- 4 bit để lưu giữ cụm phụ âm cuối trong trường hợp cụm nguyên âm có đi kèm (cần 9 giá trị).

Minh họa cho cách lưu trữ một từ đơn như sau:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

+)*Cụm nguyên âm giữa không đi kèm phụ âm*



Hình 16: Cấu trúc không đi kèm phụ âm cuối

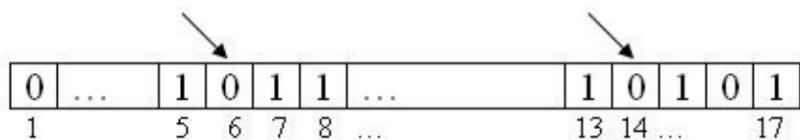
5 bit đầu lưu trữ cụm 26 phụ âm đầu.

Bit 6 và 7 đều bằng 1 cho biết cụm nguyên âm không đi kèm với phụ âm cuối.

8 bit tiếp theo: từ bít 8 đến 15 lưu trữ 162 giá trị của cụm nguyên âm giữa.

2 bít cuối cùng có giá trị bằng 0.

+)*Cụm nguyên âm giữa có thể đi kèm phụ âm:*



Hình 17: Cấu trúc có đi kèm phụ âm cuối

5 bit đầu lưu trữ cụm 26 phụ âm đầu.

8 bit tiếp theo: từ bít 6 đến 13 lưu trữ 138 giá trị của cụm nguyên âm giữa. Lưu ý rằng giá trị lớn nhất của 8 bit này tương ứng với 138 là 10001001 do vậy 2 bít 6 và 7 không cùng có giá trị 1. Đây là điểm phân biệt với từ được cấu trúc ở mẫu trên.

4 bít cuối cùng lưu trữ cụm phụ âm cuối đi kèm nguyên âm.

3.2.2 Tổ chức kết cấu

Có nhiều cách tổ chức kết cấu từ điển. Mục đích chính của chúng là để phục vụ tốt nhất cho quá trình tìm kiếm thuật ngữ. Hai cách tổ chức kết cấu thông dụng nhất là lưu trữ theo danh sách sắp xếp và lưu trữ sử dụng bảng băm:

3.2.2.1 Lưu trữ theo danh sách sắp xếp

Các thuật ngữ được lưu lại dưới dạng một danh sách. Danh sách này được sắp xếp theo thứ tự từ điển. Sau đó mỗi lần so sánh thuật ngữ sẽ áp dụng các phương pháp tìm kiếm để chọn ra đúng thuật ngữ cần tìm. Thông thường phương pháp tìm kiếm được sử dụng sẽ là phương pháp tìm kiếm theo mốc nghĩa là đặt các mốc dữ liệu và so sánh thuật ngữ với các mốc đó.

Ví dụ: Danh sách các thuật ngữ với các mốc: thuật ngữ bắt đầu bằng ký tự “a”; thuật ngữ bắt đầu bằng ký tự “b”.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Với phương pháp lưu trữ này, tốc độ tìm kiếm đạt được tốt nhất khi sử dụng cây tìm kiếm nhị phân để đặt mốc trong danh mục.

3.2.2.2 Lưu trữ sử dụng bảng băm

a. Khái niệm về hàm băm.

Hàm băm là một ánh xạ từ một tập L sang một tập M, tập M thường là tập số nguyên từ $1..m$, còn tập L thì có thể là tập bất kì nào đó như tập các xâu, tập các số ngẫu nhiên...

Một trong những hàm băm phổ biến là hàm **mod**:

$h(i) = i \bmod M$; trong đó M là giá trị băm

Giá trị $\alpha = |L| / |M|$ được gọi là *hệ số tải* của hàm băm.

Một hàm băm được gọi là hoàn hảo

Nếu $\forall i_1, i_2 \in L$ và $i_1 \neq i_2$ thì $h(i_1) \neq h(i_2)$

Điều đó có nghĩa có sự tương ứng 1-1 giữa tập L và tập M. Rõ ràng để hàm băm hoàn hảo thì trước hết $\alpha \leq 1$, và nếu $\alpha = 1$ thì hàm băm đạt tính chất tối ưu. Việc xây dựng một hàm băm hoàn hảo là rất phức tạp và không khả chuyền, khi tập L thay đổi thì hàm băm đó cũng phải thay đổi, quả thực đó là điều không thể chấp nhận. Vậy ở đây người ta phải đặt vấn đề là có xung đột (collision), xung đột là hiện tượng hai phần tử phân biệt thuộc L lại có cùng một giá trị băm

$\exists i_1, i_2 \in L$ và $i_1 \neq i_2$ sao cho $h(i_1) = h(i_2)$

Nếu *hệ số tải* càng nhỏ thì khả năng xảy ra xung đột càng nhỏ.

b. Sử dụng bảng băm

Bảng băm là một cơ cấu lưu trữ trong đó có sử dụng hàm băm để đánh số phục vụ tìm kiếm nhanh. Ban đầu, bảng băm được để trống. Sau đó, mỗi phần tử dữ liệu với một khoá so sánh K được ánh xạ tới vị trí trên bảng băm và được đưa vào bảng. Mỗi lần tìm kiếm một phần tử, sử dụng khoá so sánh K của phần tử đó để tìm vị trí trên bảng băm. Trong trường hợp có xung đột, bảng băm sẽ làm giảm không gian tìm kiếm.

Ví dụ có một tập $L = \{1, 5, 57, 42, 79, 93, 26, 12\}$ và hàm băm $h(i) = i \bmod 5$ ta có

$$h(1) = 1 \bmod 5 = 1$$

$$h(5) = 5 \bmod 5 = 0$$

$$h(57) = 34 \bmod 5 = 2$$

$$h(42) = 25 \bmod 5 = 2$$

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

$$h(79) = 79 \bmod 5 = 4$$

$$h(93) = 35 \bmod 5 = 3$$

$$h(26) = 6 \bmod 5 = 1$$

$$h(12) = 13 \bmod 5 = 2$$

Sau khi L được phân hoạch bởi hàm băm thành các L_i thì với mỗi khoá K thuộc L bây giờ chỉ thực hiện tìm kiếm trên L_i chứ không phải tìm kiếm trên cả tập L. Ví dụ với K = 12, kết quả băm bằng 2 và tập $L_2 = \{25, 42, 12\}$, áp dụng tìm kiếm nhanh chóng hơn nhiều so với tìm kiếm trên cả tập L.

Sử dụng bảng băm cho lưu trữ từ điển bằng cách xây dựng hàm băm từ các thuật ngữ đầu vào để ánh xạ đến vị trí trên bảng sao cho xung đột là nhỏ nhất.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

3.3 Loại bỏ từ dừng (stop word)

Nhắc lại về từ dừng, là các từ xuất hiện thường xuyên trong văn bản nhưng không mang nhiều ý nghĩa về nội dung văn bản. Đó có thể là các loại từ mang tính hỗ trợ cho từ khác hoặc mang ý nghĩa về mặt cấu trúc (lưu ý đối với các hệ thống phân tích cú pháp văn bản thì các từ mang ý nghĩa biểu lộ cấu trúc lại có giá trị cao).

Loại bỏ từ dừng đơn giản chỉ là so sánh các thuật ngữ tìm được và loại bỏ chúng khỏi biểu diễn văn bản. Tuy vậy, nó cũng khá quan trọng bởi các yếu tố:

- Loại bỏ từ dừng có thể làm đơn giản hóa dữ liệu, làm giảm chiều của vector biểu diễn văn bản cũng như độ phức tạp tính toán của chúng.
- Loại bỏ từ dừng để không gây nên “nhiều” dữ liệu (tránh cho các hệ thống đánh giá nhầm mức độ quan trọng của chúng chỉ dựa vào tần suất xuất hiện)

Dưới đây là một bảng ví dụ về từ dừng

Có thể	Nếu	Vì vậy
Sau khi	Thì	Nếu không
Trước khi	Vì thế	Loại trừ
Tất cả	Cho nên	Một số
Nhưng	Nhưng	Rõ ràng
Phản lón	bởi	với
Hầu như	Là	với lại
Bởi vì	Thay vì	Tất cả

Bảng 4: Một số từ dừng trong tiếng Việt

Về cách phát hiện các từ dừng, thông thường người ta đặt một ngưỡng: nếu tần suất xuất hiện của một từ vượt quá một ngưỡng nào đó thì đó là từ dừng.

```
StopWordSet = Ø;  
For  $t_i \in TermSet$  do  
  If  $idf(t_i) > StopWordsThresold$  then  
    StopWordSet =  $t_i \cup StopWordSet$ ;
```

Hình 18: Thuật toán tính tập từ dừng

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

3.4 Biểu diễn văn bản theo mô hình không gian véc tơ

Mô hình không gian véc tơ (VSP) vẫn được sử dụng nhiều nhất cho xử lý văn bản. Mỗi văn bản được biểu diễn một véc tơ. Thành phần của các véc tơ chính là các thuật ngữ xuất hiện trong văn bản. Mỗi thuật ngữ được gán một giá trị trọng được tính bởi hàm f . Công thức tính hàm f lại phân chia ra các mô hình con trong không gian véc tơ.

3.1.1 Mô hình Boolean

Hàm f cho ra giá trị rắc rối với duy nhất hai giá trị đúng và sai (true và false). Hàm f tương ứng với term t_i sẽ cho ra giá trị đúng khi và chỉ khi term t_i xuất hiện trong văn bản đó.

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

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

3.1.2 Mô hình tần suất TF

Các giá trị w_{ij} được tính dựa trên tần số xuất hiện của thuật ngữ trong văn bản. Gọi f_{ij} là số lần xuất hiện của thuật ngữ t_i trong văn bản d_j , khi đó w_{ij} được tính bởi một trong các công thức :

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

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

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

Trọng số w_{ij} tỷ lệ thuận với số lần xuất hiện của thuật ngữ t_i trong văn bản d_j . Khi số lần xuất hiện thuật ngữ t_i trong văn bản d_j càng lớn thì điều đó có nghĩa là văn bản d_j càng phụ thuộc vào thuật ngữ t_i , thuật ngữ t_i mang nhiều thông tin trong văn bản d_j .

3.1.3 Mô hình nghịch đảo tần số văn bản – IDF

Giá trị w_{ij} được tính như sau:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

$$w_{ij} = \begin{cases} \log m/h_i = \log(m) - \log(h_i) & \text{nếu thuật ngữ } t_i \text{ xuất hiện trong tài liệu } d_j \\ 0 & \text{nếu ngược lại.} \end{cases}$$

Trong đó m là số lượng văn bản và h_i là số văn bản mà thuật ngữ t_i xuất hiện.

Như vậy với trọng số w_{ij} tỷ lệ nghịch với h_i . Càng có ít văn bản có chứa t_i thì w_{ij} càng cao. Trọng số w_{ij} có ý nghĩa phân biệt với các văn bản khác.

3.1.4 Mô hình kết hợp TF-IDF

Mô hình này là sự kết hợp của 2 mô hình trên, giá trị của ma trận trọng số được tính như sau:

$$w_{ij} = \begin{cases} [1+\log(f_{ij})]\log(m/h_i) & \text{nếu } h_{ij} \geq 1 \\ 0 & \text{nếu ngược lại.} \end{cases}$$

Với mô hình TF-IDF, trọng số w_{ij} có ý nghĩa kết hợp sự quan trọng của t_i trong văn bản d_j với giá trị phân biệt bởi t_i giữa văn bản d với các văn bản khác.

3.1.5 Mô hình véc tơ thura

Trọng số w_{ij} được tính bằng tần số xuất hiện của thuật ngữ t_i trong văn bản d_j và độ hiếm của thuật ngữ t_i trong toàn bộ cơ sở dữ liệu.

Trong mô hình biểu diễn trên thì việc tính toán sẽ trở nên rất phức tạp và cồng kềnh. Lý do là các văn bản thường có nhiều thuật ngữ và do vậy các vector sẽ có số chiều rất lớn. Hiển nhiên, lưu trữ các vector cũng tốn rất nhiều bộ nhớ. Khắc phục điều đó, người ta dùng mô hình biểu diễn bằng vector thura.

Điểm cơ bản của mô hình này là thay vì biểu diễn toàn bộ thuật ngữ có trong từ điển thì chúng ta chỉ biểu diễn chỉ các thuật ngữ có trong hệ cơ sở dữ liệu. Tuy vậy khi sử dụng mô hình véc tơ thura đặc biệt phải lưu ý tính chính xác của lời giải khi đã loại bỏ thông tin.

3.1.6 Các công thức tính toán trên mô hình không gian véc tơ

Đặc điểm quan trọng nhất của biểu diễn văn bản theo không gian véc tơ là có thể tính toán, cộng trừ hai văn bản dựa trên tính toán hai véc tơ biểu diễn của chúng. Qua đó, dữ liệu văn bản trừu tượng có thể được lượng giá một cách chính xác. Các công thức quan trọng cho biểu diễn văn bản theo mô hình không gian véc tơ:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

***) Độ tương tự giữa hai véc tơ**

Giả sử hai văn bản X, Y được biểu diễn dưới dạng mô hình tần suất bằng hai véc tơ $\{x_1, x_2, \dots, x_n\}$ và $\{y_1, y_2, \dots, y_n\}$. Khi đó, độ tương tự giữa hai văn bản được tính theo công thức Cosin:

$$Sim(X, Y) = \text{Cosin}(X, Y) = \frac{\mathbf{X} \cdot \mathbf{Y}}{|\mathbf{X}| |\mathbf{Y}|} = \frac{\sum_{i=1}^n x_i y_i}{\sqrt{\sum_{i=1}^n x_i^2} \sqrt{\sum_{i=1}^n y_i^2}}$$

***) Véc tơ trọng tâm của nhóm**

Giả sử có một nhóm văn bản $D = \{d_1, d_2, \dots, d_m\}$ có lần lượt các véc tơ biểu diễn là v_1, v_2, \dots, v_n . Khi đó, véc tơ trọng tâm của nhóm văn bản được tính theo công thức:

$$\overrightarrow{V_{cen}} = \frac{\sum_{i=1}^m \vec{v}_i}{m}$$

***) Độ tương tự giữa hai nhóm**

Giả sử có hai nhóm văn bản D_1, D_2 . Khi đó, độ tương tự giữa hai nhóm được tính theo bằng độ tương tự giữa hai véc tơ trọng tâm của nhóm:

$$Sim(D_1, D_2) = Sim(V_{cen1}, V_{cen2})$$

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

CHƯƠNG IV

THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Các chương trước trình bày tổng quan và cơ sở lý thuyết của bài toán TTVB. Trong chương này, xin trình bày về việc thiết kế xây dựng một hệ thống cụ thể thực hiện TTVB tiếng Việt.

Mục tiêu xây dựng hệ thống:

- Hoạt động trên văn bản bằng ngôn ngữ tiếng Việt.
- Tóm tắt đơn lẻ một văn bản (Single Document Summarization). Văn bản có độ dài vừa phải (30 - 50 câu), có hoặc không có tiêu đề.
- Có khả năng thực hiện tính toán đối với tập văn bản vào lớn.

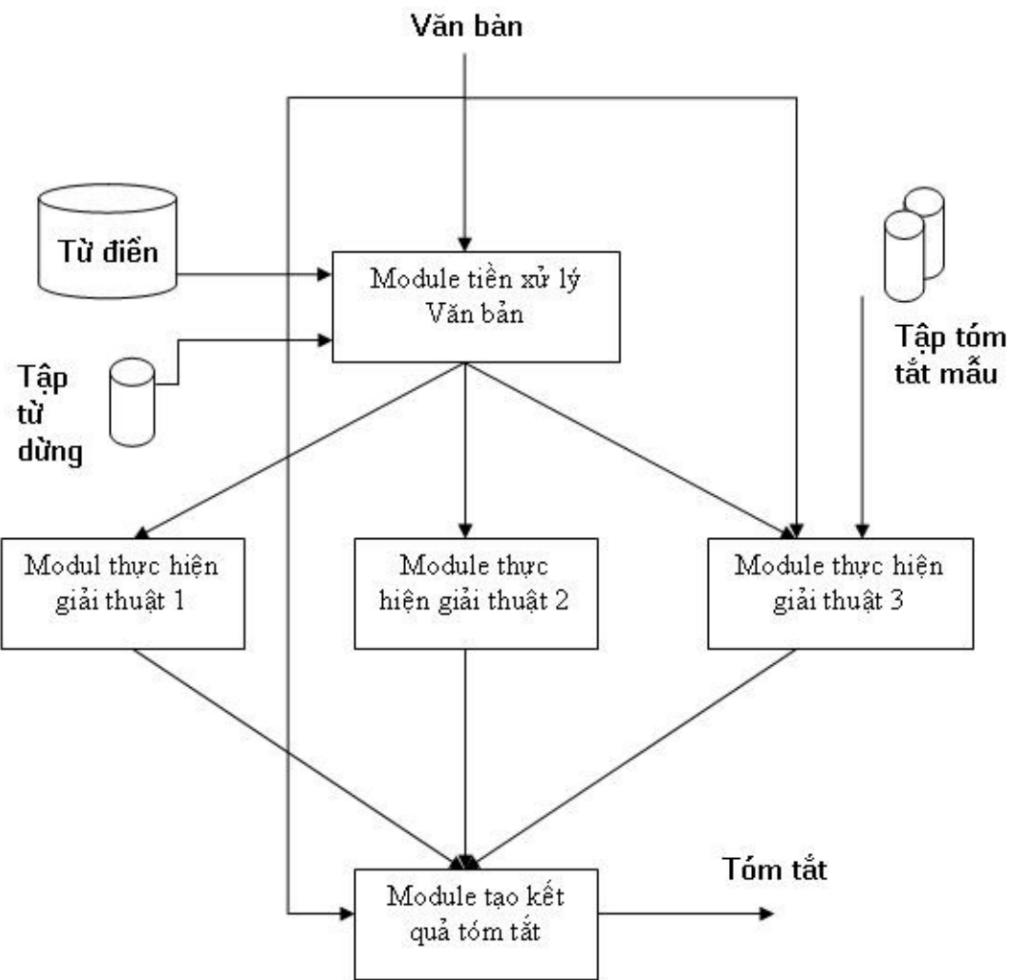
4.1 Mô hình hệ thống

Hệ thống TTVB được xây dựng áp dụng các kỹ thuật lượng giá và thông kê đã trình bày trong chương II. Để có thể đánh giá kết quả dựa trên độ phức tạp của giải thuật, tác giả thực hiện cả 3 phương án đã giới thiệu đối với hệ thống:

- Giải thuật tóm tắt dựa vào trọng số các câu.
- Giải thuật tóm tắt dựa vào phân nhóm các đoạn văn.
- Giải thuật tóm tắt có áp dụng thuật toán học máy trên các đặc trưng đơn giản.

Mô hình hệ thống như sau:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê



Hình 19: Mô hình hệ thống

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.2 Module xử lý văn bản

4.2.1 Nhiệm vụ

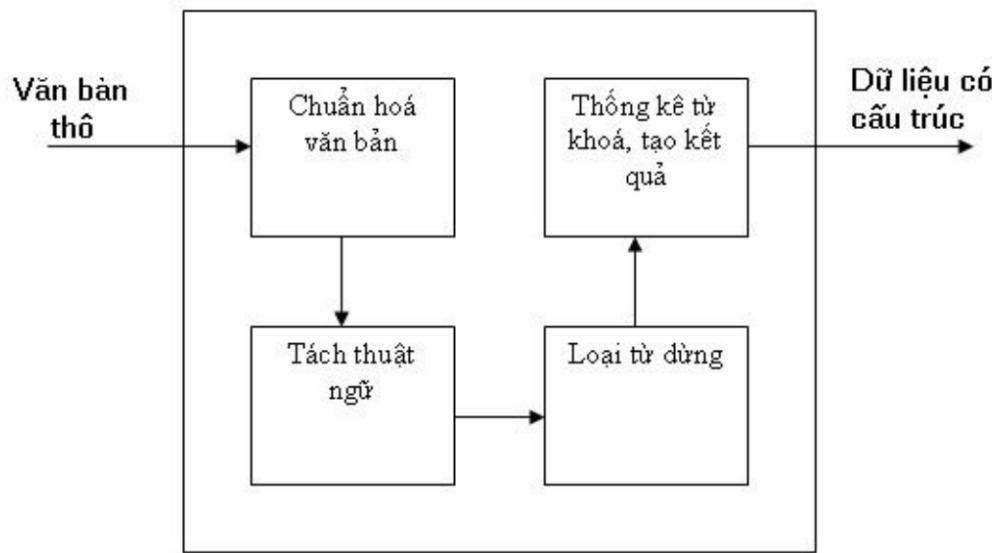
Đây là bước xử lý dữ liệu, chuẩn hoá dữ liệu văn bản sang dạng dữ liệu mong muốn. Như vậy:

Đầu vào: Văn bản/tập văn bản ở dạng text file, được lưu dưới dạng chuẩn Unicode (UNI16_LE).

Đầu ra: Dạng dữ liệu của văn bản được tổ chức có cấu trúc. Mỗi văn bản được biểu diễn thành một danh sách kè các đoạn văn. Mỗi đoạn văn được biểu diễn thành một danh sách kè các câu. Mỗi câu được biểu diễn thành một danh sách kè các thuật ngữ xuất hiện trong câu cùng với tần suất của chúng.

4.2.2 Mô hình chức năng

Sơ đồ chức năng của module được thể hiện như sau:



Hình 20: Module Tiền xử lý

4.3.2 Thực hiện

Các bước tiền xử lý văn bản được thực hiện như sau:

4.3.2.1 Chuẩn hoá văn bản

Đầu vào: Văn bản ở dạng thô (có thể được lấy về từ nhiều nguồn thông tin như báo điện tử, sách,...) với định dạng Unicode

Đầu ra: Văn bản ở dạng chuẩn chỉ gồm các ký tự tiếng Việt, các dấu chấm (“.”), dấu cách (“ ”) và ký tự xuống dòng.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Thực hiện:

Văn bản đầu ra phải có dạng gồm các chuỗi được ngăn cách bởi dấu xuống dòng. Mỗi chuỗi gồm các xâu con được ngăn bởi dấu chấm. Cách tiến hành như sau:

+ Đọc văn bản, với mỗi ký tự không phải là ký tự tiếng Việt hoặc dấu chấm, dấu cách thì xoá bỏ chúng.

+ Duyệt mỗi ký tự với các ký tự liền kề nó:

Nếu chúng là một chuỗi dấu cách thì xoá đi và thay bằng một dấu cách.

Nếu một dấu chấm đi cạnh một dấu cách thì xoá bỏ dấu cách đó.

Nếu chúng là một chuỗi dấu chấm (mang ý nghĩa liệt kê, ví dụ: “bàn, ghế, tủ, …” thì tùy thuộc vào ký tự sau chuỗi dấu chấm này là ký tự hoa hay thường để thay chuỗi bằng một dấu chấm hoặc một dấu cách.

+ Lập một bảng gồm tất cả các ký tự tiếng Việt bằng chữ hoa (103 phần tử) và một bảng khác gồm tất cả các ký tự tiếng Việt bằng chữ thường tương ứng. Khi thực hiện, so sánh các ký tự đọc được trong văn bản với bảng chữ hoa để chuyển về dạng ký tự thường.

4.3.2.2 Tách thuật ngữ

Đầu vào: Văn bản ở dạng chuẩn.

Đầu ra: Văn bản ở dạng số hoá bao gồm các dãy ID của thuật ngữ trong văn bản. Mỗi dãy là một danh sách kè các ID này biểu diễn cho một câu.

Thực hiện: Sử dụng thuật toán tách thuật ngữ số 1 - thuật toán tách thuật ngữ cho thuật ngữ có độ dài lớn nhất (trình bày trong 3.1). Phương pháp này có thể có sai số (thiếu thuật ngữ) nhưng có ưu điểm thực hiện nhanh, không tạo thuật ngữ lồng nhau và không cần dựa vào từ điển đồng nghĩa.

a. Tổ chức từ điển:

Từ điển thuật ngữ tiếng Việt mà hệ thống sử dụng có 70.266 thuật ngữ. Từ điển được hệ thống đọc vào thông qua một file văn bản lưu dưới dạng Unicode, mỗi thuật ngữ nằm trên một dòng.

Tổ chức từ điển là phần quan trọng nhất đối với module Tiền xử lý. Với kích thước số thuật ngữ lớn như vậy, tổ chức từ điển quyết định tốc độ của chức năng tách thuật ngữ. Việc tổ chức từ điển bao gồm 2 vấn đề: lưu trữ bản ghi và tổ chức cấu trúc tìm kiếm.

Về lưu trữ, có thể lưu trữ mỗi thuật ngữ theo một trong hai cách:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

- Lưu dưới dạng một xâu có độ dài n ký tự. Như vậy mỗi thuật ngữ được lưu trữ sẽ chiếm $n*2$ byte.
- Lưu dưới dạng mã hoá các tiếng (đã trình bày trong chương trước): mỗi từ trong thuật ngữ được lưu bằng 3 byte (trong đó chỉ sử dụng 17 bit). Như vậy một thuật ngữ có độ dài m tiếng sẽ chiếm $m*3$ byte.

Qua khảo sát, với phương pháp lưu trữ thứ nhất, hệ thống đọc và lưu các thuật ngữ với thời gian không đáng kể. Bộ nhớ phải sử dụng là hơn 1900K, chấp nhận được. Với phương pháp thứ hai, bộ nhớ phải sử dụng giảm đi còn 700K xong thời gian thực hiện tăng lên do các hàm thực hiện mã hoá và giải mã phải thực hiện nhiều phép so sánh. Như vậy phương pháp thứ nhất hỗ trợ tìm kiếm nhanh trong khi phương pháp thứ hai hỗ trợ tốt cho công tác lưu trữ.

Do đặc điểm của hệ thống tác giả quyết định sử dụng phương pháp thứ hai bởi tính đơn giản và tốc độ, đồng thời bộ nhớ cần thiết 1900K là chấp nhận được.

Về **tổ chức tìm kiếm**, cách tổ chức thông dụng nhất là cây tìm kiếm nhị phân. Theo đó, mỗi một lần tìm kiếm thuật ngữ cần so sánh với các mốc nhị phân trong từ điển. Như vậy, mỗi một lần tìm kiếm một thuật ngữ phải thực hiện $\log_2(70.266) = 16$ lần so sánh. Đây là con số nhỏ. Phương pháp này dễ sử dụng nhất xong có nhược điểm là danh mục thuật ngữ cần phải được sắp xếp theo thứ tự so sánh. Danh sách các thuật ngữ đọc vào đã được sắp xếp theo thứ tự từ điển đối với con người, xong không phải đối với máy. Cụ thể trong ví dụ sau, xét một danh sách các thuật ngữ đã sắp xếp từ điển:

tây ba lô
tây bắc
tay bài
tẩy bỏ
tay búp măng
tay cầm
tay cà tay súng
tay chân
tẩy chay

Hình 21: Một đoạn các thuật ngữ trong từ điển

Các ký tự “a”, “â”, “ả” đều được coi là ký tự “a” và không được sắp xếp trước - sau. Thế nhưng khi lưu trữ trong máy, chúng là các ký tự có mã khác nhau. Bởi vậy nếu muốn áp dụng phương pháp tìm kiếm nhị phân thì buộc phải sắp xếp lại các thuật ngữ khi chúng được nhập vào. Điều này là phi thực tế đối với mọi phương pháp sắp xếp cho 70.266 thuật ngữ. Do vậy, tác giả sử dụng phương pháp tổ chức tìm kiếm theo bảng băm cho hệ thống.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Đối với phương pháp sử dụng bảng băm, tính chất của hàm băm quyết định hiệu quả của bảng. Có rất nhiều cách xây dựng hàm băm, chủ yếu nhằm giảm bớt xung đột cho các phần tử trong bảng. Để áp dụng cho hệ thống, tác giả sử dụng phương pháp băm kép (Double Hashing Method) với việc sử dụng hai hàm băm để tìm vị trí của phần tử trên bảng. Bảng băm được xây dựng như sau:

Bước 1: Ban đầu, bảng băm được khởi tạo là một danh sách kề có M nút. Mỗi nút của bảng băm sẽ có một trường chứa thuật ngữ (sẽ được sử dụng như là khoá của phần tử dữ liệu). Trường này được đặt giá trị NULL.

Bước 2: Với mỗi thuật ngữ nhập vào có khoá K chính là xâu chứa thuật ngữ đó, dùng hai hàm băm để tính $f(K) = i$ và $g(K) = j$. i, j là số nguyên và $i, j < M$.

+ Xét địa chỉ i trong danh sách, nếu trống (khoá của nó là NULL) chèn thuật ngữ vào vị trí này.

+ Nếu không phải, xét tiếp địa chỉ $(i+j) \ mod \ M$. Nếu trống, chèn thuật ngữ vào vị trí này. Nếu không lại xét tiếp địa chỉ $(i+2j) \ mod \ M$ tiếp tục cho đến khi tìm được vị trí trống.

Mỗi khi tìm kiếm một thuật ngữ, quá trình tìm kiếm cũng thực hiện tương tự. Giả sử thuật ngữ có khoá K' , dùng hai hàm băm f và g để tính $i' = f(K')$ và $j' = g(K')$. Xét địa chỉ i' trong danh sách, nếu i' không rỗng tiến hành so sánh K và K' . Nếu chúng bằng nhau \Rightarrow tìm được thuật ngữ. Nếu không lại xét tiếp địa chỉ $(i+j) \ mod \ M$, $(i+2j) \ mod \ M$, Nếu một vị trí nào đó trống \Rightarrow không có thuật ngữ này trong từ điển.

Hệ thống sử dụng hai hàm băm đã được tác giả Phan Thanh Liêm[2] trình bày trước đây:

Hàm f : Giả sử khoá K là một chuỗi các ký tự $K = [c_1, c_2, \dots, c_n]$

$$f = \left(\sum_{i=1}^n \text{code}(c_i) \times i^3 \right) \mod M$$

$$\text{Hàm } g: g = \left(\sum_{i=1}^n \text{code}(c_i) \right) \mod M$$

Trong đó $\text{code}(c)$ là mã của ký tự c trong bảng mã Unicode.

b. *Tách thuật ngữ.*

Các bước tách thuật ngữ từ văn bản gốc được thực hiện như sau:

Bước 1: Rút một câu từ văn bản ở dạng chuẩn. Một câu là một xâu các ký tự và kết thúc bằng dấu chấm. Khởi tạo một danh sách kề lưu ID các thuật ngữ trong câu. Danh sách này ban đầu rỗng.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

Bước 2: Tách lấy một chuỗi chứa nhiều nhất các từ đơn đồng thời có độ dài nhỏ hơn độ dài của thuật ngữ dài nhất trong từ điển. Nếu chuỗi thu được rỗng, kết thúc. Nếu không, sang bước 3.

Bước 3: Tìm kiếm xem chuỗi đó có phải là thuật ngữ trong từ điển không. Nếu đúng, tách chuỗi này khỏi xâu và lấy ID của chuỗi này (chính là vị trí của thuật ngữ tìm được trong bảng băm) thêm vào danh sách kè. Nếu không sang bước 4.

Bước 4: Rút ngắn chuỗi trong bước 2 đi một từ cuối cùng. Nếu chuỗi khác rỗng thực hiện lại bước 3. Nếu chuỗi rỗng, cắt bỏ đi từ đầu của xâu và thực hiện lại bước 2.

4.3.2.3 *Loại bỏ từ dừng*

Đầu vào: Danh sách kè ID của các thuật ngữ trong văn bản.

Đầu ra: Danh sách kè ID của các thuật ngữ trong văn bản trong đó không có ID nào thuộc danh sách từ dừng.

Thực hiện: Lập một danh sách các từ dừng để so sánh mỗi thuật ngữ trong văn bản có thuộc danh sách này hay không.

Để tăng tốc độ thực hiện, danh sách này lưu các từ dừng dưới dạng ID của chúng trong từ điển.

Danh sách này (khoảng 200 đến 300 từ) được sắp xếp lại theo thứ tự của ID để có thể áp dụng tìm kiếm nhị phân.

Cách xây dựng danh sách từ dừng: Duyệt một tập văn bản mẫu, chọn lọc các thuật ngữ có tần suất vượt quá một ngưỡng nào đó.

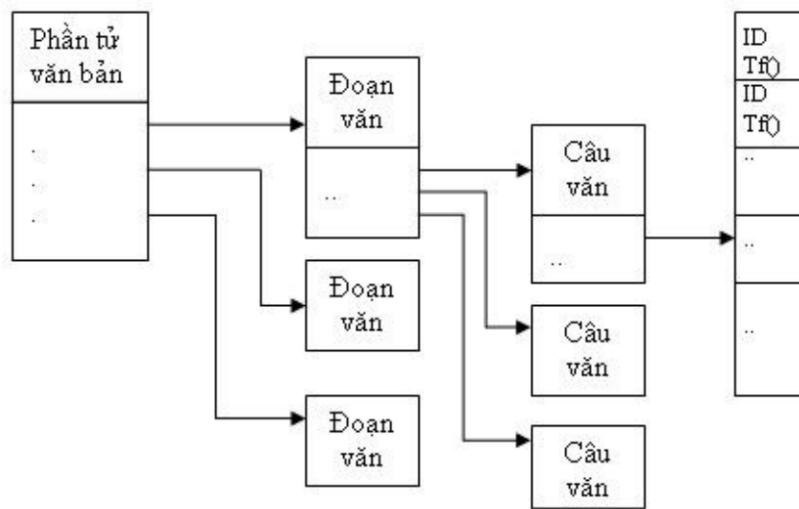
4.3.2.4 *Thống kê từ khoá, tạo kết quả*

Đầu vào: Các dãy ID của thuật ngữ xuất hiện trong văn bản

Đầu ra: Tổ chức dữ liệu có cấu trúc cho văn bản.

Dữ liệu đầu ra được tổ chức theo hướng đối tượng. Mỗi văn bản được tổ chức theo cấu trúc như sau:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê



Hình 22: Tổ chức dữ liệu có cấu trúc cho văn bản

Các lớp cho mỗi phần tử được viết thêm các hàm chức năng như hàm tính tần suất, hàm tìm kiếm, hàm so sánh,..

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.3 Module thực hiện giải thuật 1

Giải thuật 1 là giải thuật đơn giản nhất của hệ thống. Mục đích của nó là tạo ra TTVB bằng cách xây dựng hệ thống ghi điểm cho mỗi câu của văn bản. Sau đó dựa vào hệ số rút gọn để rút ra những câu có điểm cao nhất.

Đầu vào: Văn bản được tổ chức theo dữ liệu có cấu trúc.

Đầu ra: Danh sách các giá trị trọng số tương ứng với mỗi câu.

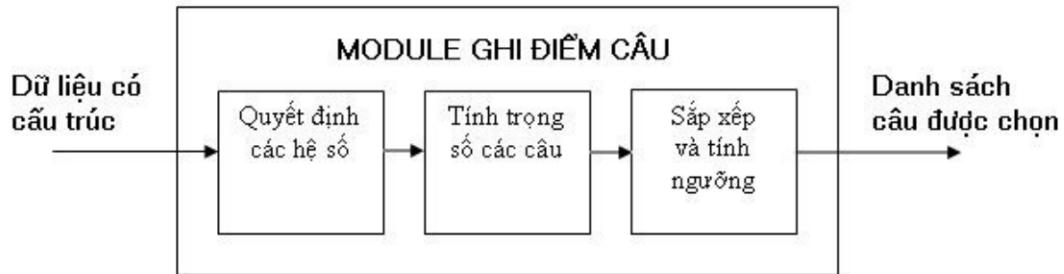
4.3.1 Một số nhận định quan trọng.

Trước khi mô tả việc xây dựng giải thuật, có thể đưa ra một số nhận xét sau:

- Các từ xuất hiện trong tiêu đề thường là các từ rất quan trọng trong văn bản, tuy không thể chỉ dùng chúng để quyết định độ quan trọng của các câu trong văn bản. Có thể áp dụng cho giải thuật bằng cách tăng trọng số của các từ này theo một hệ số nào đó.
- Thông tin đưa ra trong một vài câu đầu (nhiều khi là một đoạn văn đầu) của văn bản trong hầu hết trường hợp có tính biểu lộ cao ý nghĩa của văn bản. Các câu quan trọng cũng có thể xuất hiện ở cuối văn bản, nhưng ít hơn so với đầu văn bản. Vì vậy, với mỗi câu thuộc các vị trí đầu hoặc cuối văn bản, tăng trọng số của chúng theo một hệ số nào đó.
- Bởi vì trọng số của mỗi câu được tính toán không phải trên tổng các trọng số của các thuật ngữ trong câu mà là tính trên độ trung bình các giá trị trọng số thuật ngữ này. Do vậy, sẽ có khả năng một số câu rất ngắn không mang nội dung nhưng chứa những thuật ngữ có trọng số cao vẫn sẽ được đưa vào trong tóm tắt. Có thể hạn chế sai sót này bằng cách chỉ xét những câu có số lượng thuật ngữ lớn hơn một độ dài nhất định nào đó.
- Với những văn bản có mật độ thông tin dày đặc, đặc biệt đối với những văn bản về lĩnh vực thương mại hay tài chính, sẽ rất khó khăn cho hệ thống khi trích rút. Do vậy độ chính xác của tóm tắt sẽ thấp hơn, có nghĩa là hệ thống có thể sẽ bỏ qua nhiều thông tin quan trọng. Điều này hiển nhiên sẽ giới hạn các lĩnh vực nội dung văn bản mà hệ thống có thể thực hiện. Tuy nhiên, cũng phải thừa nhận rằng chính con người khi tóm tắt các văn bản thuộc loại này cũng gặp rất nhiều khó khăn.
- Hệ thống chắc chắn cũng sẽ gặp nhiều khó khăn khi thực hiện tóm tắt các văn bản nhiều nội dung.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.3.2 Mô hình chức năng



Hình 23: Module giải thuật 1.

4.3.3 Thực hiện

Các bước trong module ghi điểm được thực hiện như sau:

4.3.3.1 Hệ số ghi điểm

Các hệ số này phục vụ cho việc tính toán trọng số câu ở bước sau. Chúng được sử dụng để tăng tính chính xác của giải thuật khi tạo tóm tắt. Các hệ số này cung cấp trước cho hệ thống là các hằng số. Tuy nhiên đạt hiệu quả cao nhất khi sử dụng các hệ số này, đòi hỏi phải trải qua quá trình thực nghiệm với kết quả của giải thuật hoặc áp dụng các thuật toán học máy để quyết định giá trị phù hợp cho chúng. Ở đây tạm coi là chúng đã có giá trị phù hợp nhất.

Các hệ số ghi điểm bao gồm:

- Hệ số tiêu đề h_{td} : quyết định trọng số của một thuật ngữ xuất hiện trong tiêu đề được nhân lên bao nhiêu lần. Nó được trả giá trị 1 khi văn bản không có tiêu đề.
- Hệ số vị trí câu: bao gồm các hệ số:
 - + Hệ số h_{vt1} : trọng số một câu được nhân lên bao nhiêu lần khi câu đó ở đầu văn bản.
 - + Hệ số h_{vt2} : áp dụng với 2 câu tiếp theo.
 - + Hệ số h_{vt3} : áp dụng với 2 câu áp chót văn bản.
- Thông thường $h_{vt1} > h_{vt2} > h_{vt3}$
 - + Hệ số h_{dv1} : áp dụng đối với mỗi câu nằm ở đầu đoạn văn.
 - + Hệ số h_{dv2} : áp dụng đối với mỗi câu nằm thứ hai hoặc áp chót đoạn văn.
- Hệ số độ dài câu h_{len} : quyết định những câu không có số thuật ngữ vượt quá con số này không được đưa vào tóm tắt.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.3.3.2 *Tính trọng số các câu*

Đầu vào: Dữ liệu có cấu trúc của văn bản và các hệ số ghi điểm

Đầu ra: Điểm của các câu.

Quá trình thực hiện gồm các bước:

Bước 1. Duyệt toàn bộ văn bản, với mỗi thuật ngữ t trong câu s tính:

+ f_{ts} là số lần xuất hiện thuật ngữ t trong câu s .

+ h_t là số lượng các câu có chứa thuật ngữ t .

Bước 2. Duyệt lại văn bản, với mỗi câu s , thực hiện:

- Tính trọng số cho mỗi thuật ngữ t trong câu s :

$$TF - ISF(t, s) = (1 + \log(f_{ts})) \times \log\left(\frac{m}{h_t}\right) \quad \text{nếu } t \text{ xuất hiện trong tiêu đề văn bản.}$$

$$TF - ISF(t, s) = (1 + \log(f_{ts})) \times \log\left(\frac{m}{h_t}\right) \quad \text{nếu ngược lại.}$$

Trong đó: m là số lượng câu trong văn bản.

h_{td} là hệ số tiêu đề

- Tính điểm của câu:

$$\text{Score}(s) = \frac{\sum_{i=1}^{T(s)} TF - ISF(t_i, s)}{T(s)} \times h_{vt}(s)$$

Trong đó: $T(s)$ là số thuật ngữ có trong câu s .

$h_{vt}(s)$ là hệ số vị trí của câu s trong văn bản.

4.3.3.3 *Sắp xếp, tính ngưỡng và đưa ra kết quả*

Bước cuối cùng trước khi đưa ra kết quả là danh sách các câu được tóm tắt.

Các bước:

Bước 1. Duyệt toàn bộ các câu, nếu câu nào có $T(s)$ nhỏ hơn h_{len} thì đặt lại trọng số cho câu: $\text{Score}(s) = 0$.

Bước 2. Sắp xếp các $\text{Score}(s)$

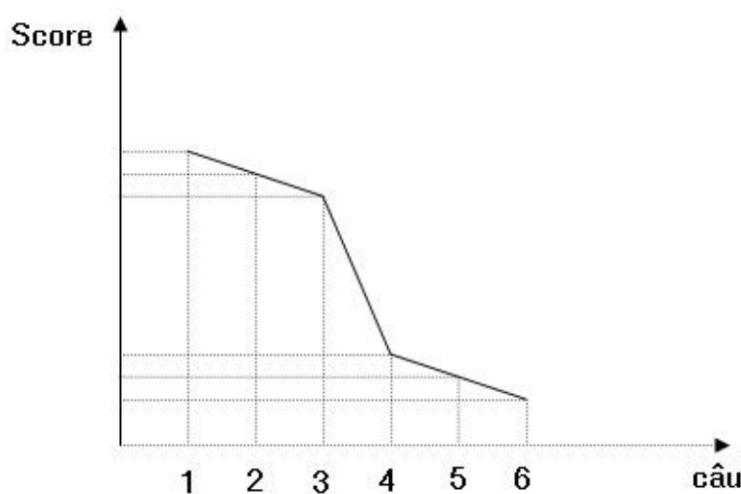
Bước 3. Theo danh sách đã tóm tắt chọn vị trí i trên danh sách để:

$$\frac{i}{m} \times 100\% = h_{co}$$

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê với h_{co} là tỷ lệ rút gọn tóm tắt.

Bước 4. Kiểm tra dịch i lên/xuống 1 vị trí nếu i là vị trí mà tại đó có sự thay đổi đột ngột về độ lớn trọng số của câu.

Ví dụ:



Hình 24: Đồ thị trọng số câu

Trong ví dụ trên, tại các vị trí $i=3$, $i=4$ có sự thay đổi đột ngột về giá trị trọng số của câu.

Vai trò của bước cuối cùng này nhằm tăng độ chính xác cho giải thuật khi được kiểm thử.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

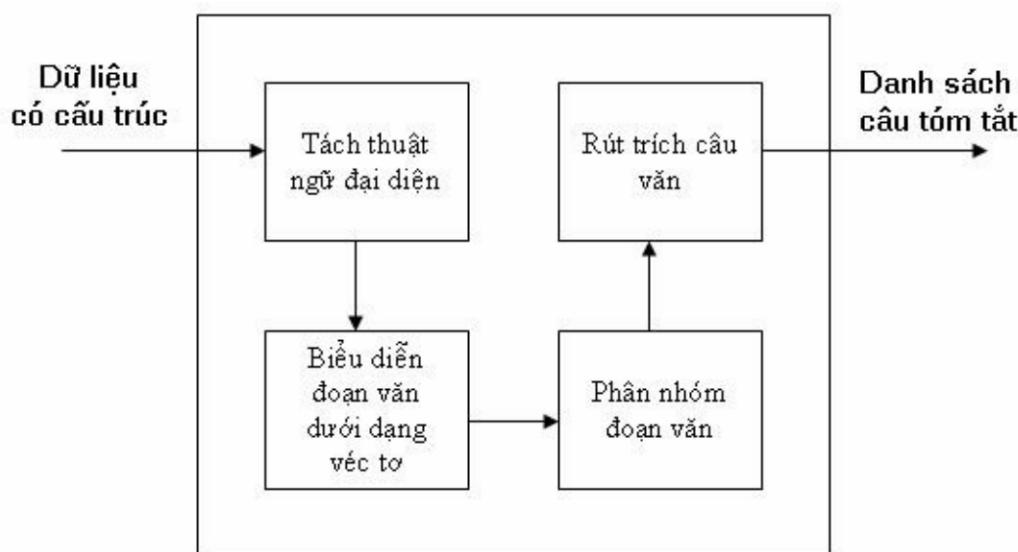
4.4 Module thực hiện giải thuật 2

Giải thuật 2 áp dụng phương pháp phân nhóm để nhóm các câu có cùng nội dung vào một nhóm. Sau đó đưa ra tóm tắt bằng cách chọn ở mỗi nhóm một câu đại diện tốt nhất.

Đầu vào: Văn bản được biểu diễn dưới dạng dữ liệu có cấu trúc

Đầu ra: Danh sách nhóm các câu trong văn bản.

4.4.1 Mô hình của giải thuật



Hình 25: Module thực hiện giải thuật 2

4.4.2 Tách thuật ngữ đại diện

Ý tưởng cơ bản của giải thuật 2 là biểu diễn mỗi đoạn văn trong văn bản bằng một vec tơ (tương tự với cách biểu diễn mỗi văn bản bằng một vec tơ) chứa tần suất của các thuật ngữ xuất hiện trong đoạn văn. Tuy vậy, với một văn bản có nhiều các thuật ngữ khác nhau xuất hiện thì độ phức tạp tính toán sẽ cao, đồng thời độ chính xác khi phân nhóm cũng thấp. Bởi vậy, hướng giải quyết là chỉ chọn lọc các thuật ngữ có giá trị nội dung cao trong văn bản, gọi là các thuật ngữ đại diện của văn bản.

Đầu vào: Văn bản được biểu diễn dưới dạng dữ liệu có cấu trúc với đầy đủ thuật ngữ.

Đầu ra: Danh sách các thuật ngữ đại diện cho mỗi đoạn văn.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

Các bước thực hiện như sau :

Bước 1. Duyệt toàn bộ văn bản, với mỗi thuật ngữ t trong đoạn văn p tính:

+ f_{tp} là số lần xuất hiện thuật ngữ t trong đoạn văn p .

+ h_t là số lượng các đoạn văn có chứa thuật ngữ t .

Bước 2. Duyệt lại văn bản, với mỗi đoạn văn p , thực hiện:

- Tính trọng số cho mỗi thuật ngữ t trong đoạn văn p :

$$TF - IPF(t, p) = (1 + \log(f_{tp})) \times \log\left(\frac{m}{h_t}\right)$$

Trong đó: m là số lượng các đoạn văn trong văn bản.

- Chuẩn hoá các trọng số này theo công thức:

$$w_{tp} = \frac{w_{tp}}{\sqrt{\sum_{t \in p} (w_{tp})^2}}$$

Trong đó: $w_{tp} = TF - IPF(t, p)$

- Chọn ra các thuật ngữ có trọng số lớn hơn một ngưỡng cho trước và coi chúng là các thuật ngữ đại diện cho đoạn văn.

4.4.3 Véc tơ hóa đoạn văn.

Phân nhóm đoạn văn cũng tức là gom các đoạn văn có sự tương tự về nội dung lại chung một nhóm với nhau. Như vậy cần có công thức đánh giá độ tương tự về nội dung giữa các đoạn văn. Độ tương tự này có thể được tính bằng công thức Cosine đã đề cập trong chương III.

Đầu vào: Văn bản cùng danh sách các thuật ngữ đại diện cho mỗi đoạn văn.

Đầu ra: Danh sách các véc tơ có cùng số chiều (nằm trong cùng một hệ toạ độ), mỗi véc tơ biểu diễn một đoạn văn.

Thực hiện:

Bước 1. Duyệt toàn bộ văn bản, xây dựng một tập thuật ngữ đại diện cho văn bản là hợp của tất cả các tập thuật ngữ đại diện cho từng đoạn văn trong văn bản.

$$T = t(p_1) \cup t(p_2) \cup \dots \cup t(p_m)$$

Trong đó: m là số đoạn văn.

$t(p_i)$ là tập các thuật ngữ đại diện cho đoạn văn i .

Giả sử T có n thành phần:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

$$T = \{t_1, t_2, \dots, t_n\}$$

Bước 2. Duyệt lại văn bản, với mỗi đoạn văn p xây dựng véc tơ biểu diễn:

$$V_p = (w_1, w_2, \dots, w_n)$$

Trong đó, w_i bằng trọng số của thuật ngữ t_i trong đoạn văn p nếu nó là thuật ngữ đại diện cho p và bằng 0 nếu không phải.

4.4.4 Phân nhóm đoạn văn

Tác giả sử dụng thuật toán lập nhóm theo cây phân cấp (HC) để phân nhóm các đoạn văn trong văn bản.

Đầu vào: Danh sách các đoạn văn cùng với véc tơ biểu diễn.

Đầu ra: Cây phân cấp dưới lên phân nhóm các đoạn văn.

Các bước thực hiện:

Bước 1: Lập danh sách m nhóm, mỗi nhóm chứa 1 đoạn văn thuộc văn bản. Véc tơ trọng tâm của nhóm chính là véc tơ biểu diễn cho mỗi đoạn văn đó.

Bước 2: Tính độ tương tự giữa các nhóm với nhau theo công thức Cosin:

$$\text{Sim}(X, Y) = \frac{\sum_{i=1}^n x_i \cdot y_i}{\sqrt{\sum_{i=1}^n x_i^2} \cdot \sqrt{\sum_{i=1}^n y_i^2}}$$

Trong đó: n là số chiều của các véc tơ.

(x_1, x_2, \dots, x_n) là véc tơ trọng tâm của nhóm X .

(y_1, y_2, \dots, y_n) là véc tơ trọng tâm của nhóm Y .

Bước 3: Chọn 2 nhóm có độ tương tự lớn nhất, gộp chung lại một nhóm và tính lại véc tơ trọng tâm theo công thức:

$$\overrightarrow{V_{cen}} = \frac{\sum_{i=1}^k \vec{v}_i}{k}$$

với k là số phần tử có trong một nhóm.

Bước 4: Lặp lại các bước 2 và 3 cho đến khi chỉ còn một nhóm.

4.4.5 Trích rút Tóm tắt.

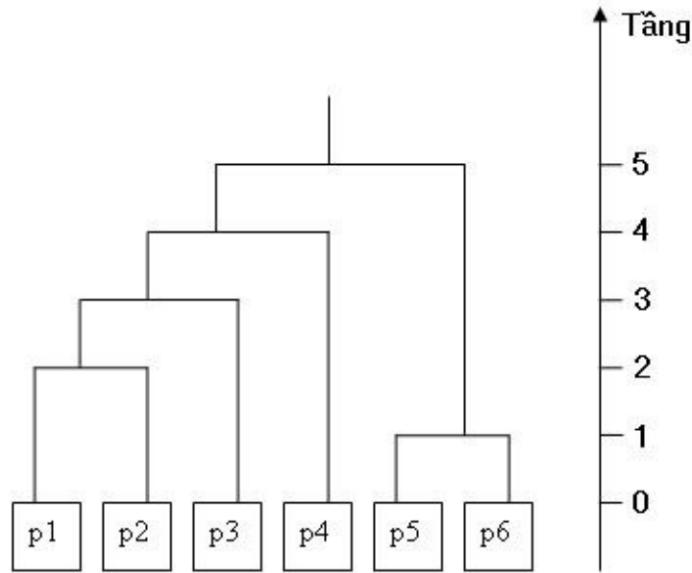
Đầu vào: Cây phân cấp xây dựng được từ giai đoạn trước.

Đầu ra: Danh sách các câu được trích rút để sử dụng cho tóm tắt.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

Đây là bước quan trọng để tạo ra kết quả và nó quyết định độ chính xác khi thực hiện tóm tắt. Nội dung của nó thực hiện hai mục đích quan trọng:

- Quyết định số nhom sẽ phân chia các đoạn văn (quyết định tầng kết quả của cây phân cấp)
- Quyết định lựa chọn câu/các câu nào trong mỗi nhom.



Hình 26: Ví dụ cây phân cấp theo giải thuật phân cấp dưới lên

* Quyết định số nhom:

Thông thường đối với các bài toán phân nhom văn bản, nhiệm vụ phân nhom được cho là tối ưu khi sự giống nhau giữa các văn bản cùng một nhom được cực đại hóa và sự giống nhau giữa các văn bản không cùng nhom được cực tiểu hóa. Chính vì lẽ đó, để quyết định số nhom được phân chia trong bài toán phân nhom đoạn văn, có thể được quyết định thông qua việc tối ưu hóa để tìm giá trị nhỏ nhất của hàm mục tiêu ϕ :

$$\phi = \frac{D}{S}$$

trong đó: D là đại diện cho sự giống nhau giữa các đoạn văn không cùng nhom, được tính bằng:

$D = \max Sim(x,y)$ với x,y là 2 đoạn văn bất kỳ không thuộc cùng một nhom với nhau

S là đại diện cho sự giống nhau giữa các đoạn văn cùng nhom, được tính bằng:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

$S = \min Sim(x,y)$ với x,y là 2 đoạn văn bất kỳ thuộc cùng một nhóm với nhau

Cách thực hiện: với mỗi một bước lặp k ($k = 0..n-1$) trong giải thuật phân nhóm ở trên (tương ứng với tầng k trong cây phân cấp), sau khi gom hai nhóm có độ tương tự lớn nhất lại với nhau, tính và lưu giá trị ϕ_k tương ứng.

Tìm giá trị ϕ_k nhỏ nhất, tương ứng có số nhóm cần phân chia là:

$$c = n - k$$

Quyết định số nhóm được phân chia trong bài toán TTVB tuy vậy còn liên quan vào một yếu tố khác: số lượng các câu tóm tắt phải có (hay hệ số rút gọn tóm tắt). Cụ thể, sự liên quan được trình bày dưới đây.

* Lựa chọn câu trong nhóm:

Lựa chọn câu trong nhóm có nghĩa là đối với mỗi nhóm đoạn văn được phân chia, phải rút ra một hoặc hơn một câu có giá trị nội dung cao nhất trong nhóm để đưa vào tóm tắt. Tỷ lệ tốt nhất là 1/1, có nghĩa là cứ 1 nhóm đoạn văn thì rút ra 1 câu. Tuy vậy, có thể số câu cần trích rút tạo tóm tắt lớn hơn hoặc nhỏ hơn nhiều lần so với số nhóm đã quyết định ở mục trên.

Bước 1: Nếu ký hiệu số câu cần trích rút là a , số đoạn văn là n và số nhóm quyết định ở mục trên là c , trong trường hợp:

- $a < c$: đặt lại số nhóm được phân chia $c = a$.
- $c < a < n$: đặt lại số nhóm được phân chia $c = a$.
- $n < a$: tìm giá trị l nhỏ nhất sao cho

$$a.2^{-l} < n$$

Khi đó, đặt lại số nhóm được phân chia $c = a.2^{-l}$ đồng thời ở mỗi nhóm thay vì trích rút 1 câu, thực hiện rút $l+1$ câu. Đây là trường hợp không mong muốn bởi các câu được rút nằm trong một nhóm, có thể trùng nhau về nội dung.

Bước 2: Đây là bước cuối cùng của giải thuật: Duyệt toàn bộ các nhóm, với mỗi nhóm rút l câu có giá trị nội dung cao nhất.

Có rất nhiều cách để lấy ra các câu văn từ mỗi nhóm này, có thể chỉ đơn giản bằng cách lấy ra l câu đầu tiên hoặc l câu dài nhất, hoặc áp dụng giải thuật 1 để ghi điểm cho từng câu trong mỗi nhóm đoạn văn và rút ra câu có điểm cao nhất. Khi phân tích khả năng bao chứa nội dung của các câu trong mỗi đoạn văn sau khi được phân nhóm, có thể đưa ra nhận xét sau:

- Hệ thống ghi điểm ở giải thuật 1 dựa trên tính toán giá trị nội dung của tất cả các thuật ngữ có trong văn bản.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

- Việc trích rút các câu trong mỗi nhóm đoạn văn cần rút ra các câu có nội dung đại diện cho nhóm đoạn văn đó nhất chứ không phải cho toàn bộ văn bản.
- Giá trị nội dung của mỗi câu khi đó cũng không tính trên trung bình các thuật ngữ xuất hiện trong câu mà tính theo tổng các thuật ngữ đại diện của nhóm đoạn văn có trong câu.

Vì vậy, hệ thống ghi điểm cho mỗi câu trong nhóm đoạn văn sẽ chỉ xét trên các thuật ngữ đại diện cho nhóm đoạn văn đó. Công thức ghi điểm được trình bày đơn giản như sau (các bước thực hiện cũng giống như ở giải thuật 1):

$$Score(s) = (\sum_{t \in s} TF - IPF(t, p)) \times h_{vt}$$

trong đó h_{vt} là hệ số vị trí của câu s trong đoạn văn p , hoặc trong văn bản gốc.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.5 Module thực hiện giải thuật 3

Giải thuật 3 là giải thuật thực hiện TTVB có độ phức tạp cao nhất được xây dựng trong hệ thống. Nội dung cơ bản của giải thuật là áp dụng các đặc trưng để tạo tóm tắt. Với mỗi đặc trưng, sẽ có một tóm tắt cho văn bản được tạo ra bằng cách sử dụng đặc trưng đó.

Các đặc trưng này sau đó được kết hợp với nhau và dựa vào thực nghiệm trên các tập CSDL mẫu để tìm ra sự kết hợp cho kết quả tốt nhất. Có thể nói đây là một mô hình tổng quát để giải quyết bài toán tạo tóm tắt bằng cách trích rút câu. Bởi bất cứ một kỹ thuật để tạo tóm tắt nào từ đơn giản đến phức tạp nhất cuối cùng cũng đều cho ra một tóm tắt cho văn bản, và như vậy đều có thể coi là một “đặc trưng tóm tắt”.

Giải thuật 3 được xây dựng ở đây sử dụng các đặc trưng tóm tắt đơn giản nhất, bởi vì quá trình tối ưu hoá trên tập CSDL mẫu có độ phức tạp tính toán cao. Do vậy, nếu lại sử dụng các đặc trưng phức tạp, hiệu quả về chất lượng có thể được nâng lên nhưng hiệu quả thời gian tính toán rất thấp.

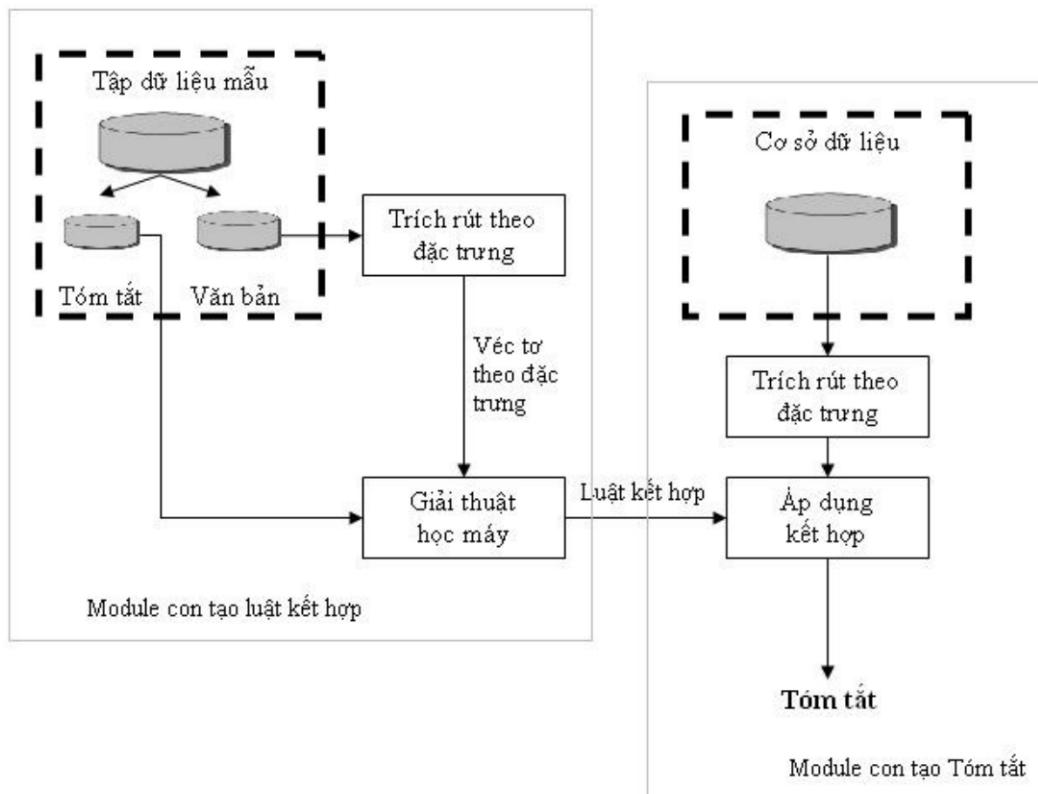
Mô tả giải thuật:

Đầu vào: Văn bản ở dạng biểu diễn có cấu trúc.

Đầu ra: Danh sách các câu được tóm tắt.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.5.1 Mô hình giải thuật.



Hình 27: Module thực hiện giải thuật 3.

Module thực hiện giải thuật 3 lại được chia thành 2 module con:

- Module áp dụng giải thuật học máy để tìm luật kết hợp các đặc trưng tóm tắt.
- Module sử dụng luật kết hợp để xây dựng tóm tắt.

Trong 2 module này đều sử dụng chức năng “Trích rút theo đặc trưng” để tạo ra tóm tắt từ văn bản gốc theo các đặc trưng định trước.

4.5.2 Trích rút theo đặc trưng

Chức năng này có thể coi như là một hệ thống TTVB “con”, có nghĩa là nó có khả năng đưa ra một tóm tắt cụ thể. Tuy vậy, mục đích chính của nó để tạo ra các véc tơ đặc trưng cho mỗi một thành phần văn bản (trong trường hợp này là một câu).

Đầu vào: Văn bản ở dạng dữ liệu có cấu trúc cùng với k đặc trưng tóm tắt.

Đầu ra: Các véc tơ k chiều đặc trưng cho mỗi câu trong văn bản ban đầu.

Giả sử có k đặc trưng: $F_1, F_2, F_3, \dots, F_k$ và văn bản gốc có n câu.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

=> đầu ra của chức năng là n vec tơ:

$$v_i = (w_{i1}, w_{i2}, w_{i3}, \dots, w_{ik}) \quad (i = 1..n)$$

Véc tơ đặc trưng cho câu chính là một dãy các trọng số của câu ứng với các đặc trưng để TTVB (để đơn giản hóa hệ thống, ta sử dụng mô hình Boolean: các trọng số này chỉ là 0 hoặc 1 có nghĩa w_{ij} chỉ có giá trị 0,1).

Ví dụ: Với đặc trưng “**Các câu có chứa tiêu đề sẽ được rút ra để xây dựng Tóm tắt**”, nếu câu có chứa tiêu đề giá trị trọng số ứng với đặc trưng này sẽ bằng 1, ngược lại bằng 0.

Các đặc trưng tóm tắt được phân tích để áp dụng trong giải thuật này:

- (a) **Đánh giá trị trọng số và ghi điểm cho mỗi câu trong văn bản gốc.** Đây là đặc trưng được sử dụng trong giải thuật 1, tuy nhiên trong trường hợp này công thức ghi điểm cho câu được đưa về công thức nguyên bản:

$$\text{Score}(s) = \frac{\sum_{i=1}^{T(s)} TF - ISF(t_i, s)}{T(s)}$$

Trong đó: $T(s)$ là số thuật ngữ có trong câu s .

Các câu có điểm cao nhất theo một ngưỡng cho trước (phụ thuộc vào hệ số rút gọn tóm tắt) sẽ có giá trị trọng số 1 đối với đặc trưng này.

- (b) **Độ dài câu.** Tương tự đặc trưng trên, sử dụng độ dài câu lớn hơn một hằng số cho trước cũng đã được sử dụng trong giải thuật 1.

$$w_i = \begin{cases} 1 & \text{nếu câu } i \text{ có số thuật ngữ lớn hơn hằng số cho trước} \\ 0 & \text{nếu ngược lại} \end{cases}$$

- (c) **Vị trí câu.** Có rất nhiều phương pháp khác nhau khai thác vị trí của câu trong văn bản để thực hiện tóm tắt. Trong giải thuật 1, một cách khai thác vị trí câu cũng đã được sử dụng: đó là sử dụng các hệ số nhân điểm cho câu theo vị trí trong văn bản.

Ở đây, đặc trưng vị trí câu được thực hiện bằng cách trước hết ghi điểm khởi đầu cho mỗi câu. Cụ thể:

- Ba câu đầu và hai câu áp chót văn bản có điểm là a .
- Câu đầu mỗi đoạn văn có điểm là b
- Câu thứ hai và câu cuối mỗi đoạn văn có điểm c

Các câu còn lại có điểm là 1. Trong đó $a > b > c > 1$. (Các hệ số này được quyết định bằng thực nghiệm khi chỉ sử dụng riêng một đặc trưng vị trí)

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

Sau đó, điểm cho mỗi câu được đặt lại:

$$mark(i) = mark(i) + \frac{n-i+1}{n} \times h$$

trong đó $mark(i)$ là điểm của câu thứ i trong văn bản

n là số câu trong văn bản

h là hằng số được quyết định bằng thực nghiệm

Cuối cùng, các câu được sắp xếp theo điểm vị trí của chúng, các câu có điểm vượt quá ngưỡng cho trước được xem như thoả mãn đặc trưng vị trí câu.

- (d) **Độ tương tự với tiêu đề.** Các câu có chứa thông tin liên quan đến tiêu đề hiển nhiên mang giá trị nội dung cao. Để tính toán độ tương tự với tiêu đề, có thể sử dụng nhiều cách. Ở đây, tác giả sử dụng công thức tính độ tương tự Cosin, coi tiêu đề như một truy vấn và tính độ tương tự của mỗi câu với truy vấn này (phương pháp thường được sử dụng trong các hệ tìm kiếm thông tin - IR). Các câu có độ tương tự với tiêu đề vượt một ngưỡng cho trước được xem như thoả mãn đặc trưng này.

- (e) **Độ tương tự với từ khoá.** Từ khoá (key word) là các từ đặc trưng về nội dung cho văn bản. Bởi vậy chúng cũng có giá trị nội dung tương đương với các thuật ngữ xuất hiện trong tiêu đề. Độ tương tự của mỗi câu với dãy các từ khoá cũng được tính theo công thức như trên.

Các từ khoá được phát hiện sử dụng phương pháp đánh giá trọng số. Các thuật ngữ có tần số IF-TDF cao nhất vượt quá ngưỡng cho trước chính là các từ khoá của một văn bản.

- (f) **Độ tương tự với các câu khác trong văn bản.** Các câu trong văn bản có nội dung liên kết nhiều nhất với các câu khác có thể coi là câu đại diện cho văn bản, vì vậy cũng có khả năng tham gia tóm tắt cao. Độ tương tự này được tính bằng cách:

$$Sum(s) = \sum_{\forall s' \in d; s' \neq s} sim(s, s')$$

trong đó $sim(s, s')$ là độ tương tự giữa hai câu trong văn bản được tính theo công thức Cosin (đã trình bày trong giải thuật 2). Các giá trị này được sắp xếp và chọn ra các câu cao nhất vượt quá ngưỡng.

- (g) **Độ tương tự với véc tơ trọng tâm của văn bản.** Để tính giá trị đặc trưng này cho mỗi câu, trước hết tính véc tơ trọng tâm của văn bản:

$$\overrightarrow{V}_{cen} = \frac{\sum_{i=1}^n \vec{v}_i}{n}$$

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê trong đó v_i là các véc tơ biểu diễn câu theo tần suất TS-ISF.

Sau khi xây dựng véc tơ trọng tâm, các véc tơ biểu diễn câu nào trong văn bản có độ tương tự với véc tơ trọng tâm lớn nhất sẽ được chọn để thỏa mãn đặc trưng này.

- (h) **Phân nhóm các câu có cùng nội dung trong văn bản.** Đặc trưng tóm tắt này tương tự với giải thuật 2 đã thực hiện. Xong các thành phần được phân nhóm không phải là các đoạn văn mà là các câu. Do vậy khả năng áp dụng là lớn hơn so với giải thuật 2 (chỉ áp dụng đối với các văn bản được phân chia ra thành các đoạn văn).
- (i) **Xuất hiện tên riêng trong câu.** Đặc trưng này đã được trình bày trong chương II, phần giới thiệu các phương pháp TTVB. Nó chỉ ra rằng các câu có xuất hiện tên riêng (thường viết tắt bằng chữ hoa) có giá trị tóm tắt cao.
- (j) **Xuất hiện các thuật ngữ đặc biệt.** Các câu có chứa các thuật ngữ như “tổng quát”, “tóm tắt”, “nói chung”, “cụ thể”, có nhiều khả năng được sử dụng để tạo tóm tắt.

Xây dựng danh sách các thuật ngữ đặc biệt, sau đó duyệt toàn bộ văn bản, những câu có chứa thuật ngữ đặc biệt này xem như thỏa mãn đặc trưng.

- (k) **Vị trí của câu trong cây nhị phân.** Cây nhị phân được xây dựng cho mỗi văn bản để đánh giá sự liên kết về nội dung giữa các thành phần văn bản liền kề (ở đây là câu).

Giải thuật xây dựng cây nhị phân tương tự với giải thuật gom cụm để tạo cây phân cấp. Điểm khác nhau duy nhất là các thành phần được gộp lại với nhau phải là các thành phần liền kề.

Có thể trình bày đơn giản giải thuật như sau:

Bước 1:	Ban đầu coi mỗi câu như một nhóm
Bước 2:	Tính độ tương tự giữa tất cả các cặp 2 nhóm liền kề với nhau
Bước 3:	Chọn ra 2 nhóm có độ tương tự cao nhất, kết hợp chúng lại thành một nhóm mới thay vào vị trí 2 nhóm đó
Bước 4:	Lặp lại bước 2 và bước 3 cho đến khi chỉ còn 1 nhóm duy nhất chứa toàn bộ các câu trong văn bản

Hình 28: Giải thuật tạo cây nhị phân

Từ cây nhị phân được tạo thành, có thể rút ra các đặc trưng nhỏ:

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

+ Các câu gần với gốc (chỉ qua từ 1 đến 4 nút) không mang nhiều giá trị nội dung cho văn bản.

+ Mỗi nhóm các câu xa gốc nhất thường có chung một giá trị nội dung và có thể trích rút một trong chúng để xây dựng tóm tắt.

Đặc trưng nhỏ thứ nhất phù hợp bởi các tính chất không mang nội dung được chứng minh, trong khi đặc trưng thứ hai có giá trị tương tự với đặc trưng (h).

4.5.3 Giải thuật học máy

Mục đích của chức năng này nhằm đưa ra một sự kết hợp các đặc trưng tốt nhất có thể để xây dựng tóm tắt. Như đã trình bày trong chương II, mục đích của giải thuật là tìm ra các hệ số thực hiện k_i cho mỗi đặc trưng F_i . Để đơn giản hệ thống, các hệ số k_i được coi là chỉ có giá trị 0 hoặc 1. Với mỗi đặc trưng F , hệ số $k=0$ có nghĩa là nó không được sử dụng để tạo tóm tắt và $k=1$ có nghĩa là nó được sử dụng trong kết hợp.

Đầu vào: Tập các đặc trưng F_1, F_2, \dots, F_k và tập văn bản mẫu đã được véc tơ đặc trưng hóa.

Đầu ra: Một kết hợp các đặc trưng F'_1, F'_2, \dots, F'_m cho kết quả tóm tắt tốt nhất.

Thực hiện:

Nhắc lại về luật xác suất Bayes đã trình bày trong phần trước:

Xác suất một câu s thuộc văn bản gốc có nằm trong tóm tắt S của văn bản sử dụng các đặc trưng F_1, F_2, \dots, F_k đó hay không là:

$$P(s \in S | F_1, F_2, \dots, F_k) = \frac{P(F_1, F_2, \dots, F_k | s \in S) \times P(s \in S)}{P(F_1, F_2, \dots, F_k)}$$
$$P(s \in S | F_1, F_2, \dots, F_k) = \frac{\prod_{j=1}^k P(F_j | s \in S) \times P(s \in S)}{\prod_{j=1}^k P(F_j)}$$

Giá trị $P(s \in S)$ là một hằng số (bằng hệ số rút gọn).

Giá trị $P(F_j | s \in S)$ và $P(F_j)$ có thể được tính theo các tập mẫu văn bản đã được tóm tắt.

=> với bất cứ một tập hợp u đặc trưng bất kỳ, đều có thể tính toán xác suất một câu s trong văn bản d đã thoả mãn u đặc trưng đó có nằm trong tóm tắt hay không.

Nếu áp dụng các luật Bayes với một tập hợp nhiều đặc trưng, độ phức tạp tính toán sẽ tỷ lệ với tổng số tất cả các đặc trưng đó. Hệ thống khi đó có hiệu

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê suất thời gian rất thấp. Bởi vậy trong giải thuật này, tác giả đề xuất phương hướng như sau:

- Chỉ áp dụng luật Bayes với tổ hợp chap 3 hoặc 4 phần tử đặc trưng trở xuống.
- Không thực hiện trên toàn bộ các tổ hợp mà phân tích mỗi liên kết giữa các đặc trưng để xét các tổ hợp hợp lý.

4.5.4 Áp dụng kết hợp

Đầu vào: Các véc tơ đặc trưng cho mỗi câu của văn bản cần tóm tắt.

Luật kết hợp các đặc trưng tối ưu.

Đầu ra: Danh sách các câu được tóm tắt.

Đây là bước cuối cùng đơn giản nhất của giải thuật. Thực hiện:

Duyệt toàn bộ các câu trong văn bản. Nếu véc tơ đặc trưng của câu thoả mãn toàn bộ các đặc trưng tối ưu, đưa câu và danh sách các câu được tóm tắt.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.6 Module tạo kết quả.

Đầu vào: Danh sách các câu cùng đã được sắp xếp thự tự để thực hiện tóm tắt cho từng giải thuật.

Đầu ra: Tạo văn bản tóm tắt.

Đây là chức năng thực hiện tạo kết quả tóm tắt cuối cùng để cung cấp cho người sử dụng. Đối với hệ thống, đây là chức năng rất đơn giản và tác giả cũng không đề cập chi tiết. Tuy rất đơn giản, nhưng tác giả cũng thiết kế nó như một module riêng để có thể phát triển hệ thống lên cao nữa sau này (ví dụ tạo tóm tắt abstract).

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.7 Cài đặt hệ thống.

4.7.1 Môi trường và công cụ cài đặt.

*** Môi trường cài đặt:**

Hệ thống được cài đặt trong môi trường hệ điều hành Windows XP (có thể hoạt động trong các hệ điều hành Windows 9x trở lên có hỗ trợ Unicode).

Bộ nhớ trong : 25 MB hoặc hơn.

Bộ nhớ ngoài: 25 MB hoặc hơn.

(Qua thử nghiệm khi chương trình đang hoạt động, bộ nhớ hệ điều hành cung cấp cho chương trình lúc lớn nhất là khoảng 17MB)

*** Công cụ cài đặt:**

Hệ thống được cài đặt bằng ngôn ngữ Visual C++ 6.0. Đặc điểm của hệ thống là hoạt động dựa trên các giải thuật tính toán phức tạp và sử dụng nhiều bộ nhớ. Giao diện của hệ thống không quá phức tạp. Do vậy VC++ có khả năng hỗ trợ rất tốt tốc độ tính toán xử lý dữ liệu. Đồng thời thư viện MFC của VC++ có rất nhiều lớp phục vụ tính toán trên các xâu (đối tượng xử lý chính đối với mỗi bài toán trong Khai thác văn bản)

4.7.2 Mô tả chương trình.

Chương trình được đặt tên là VietSum.

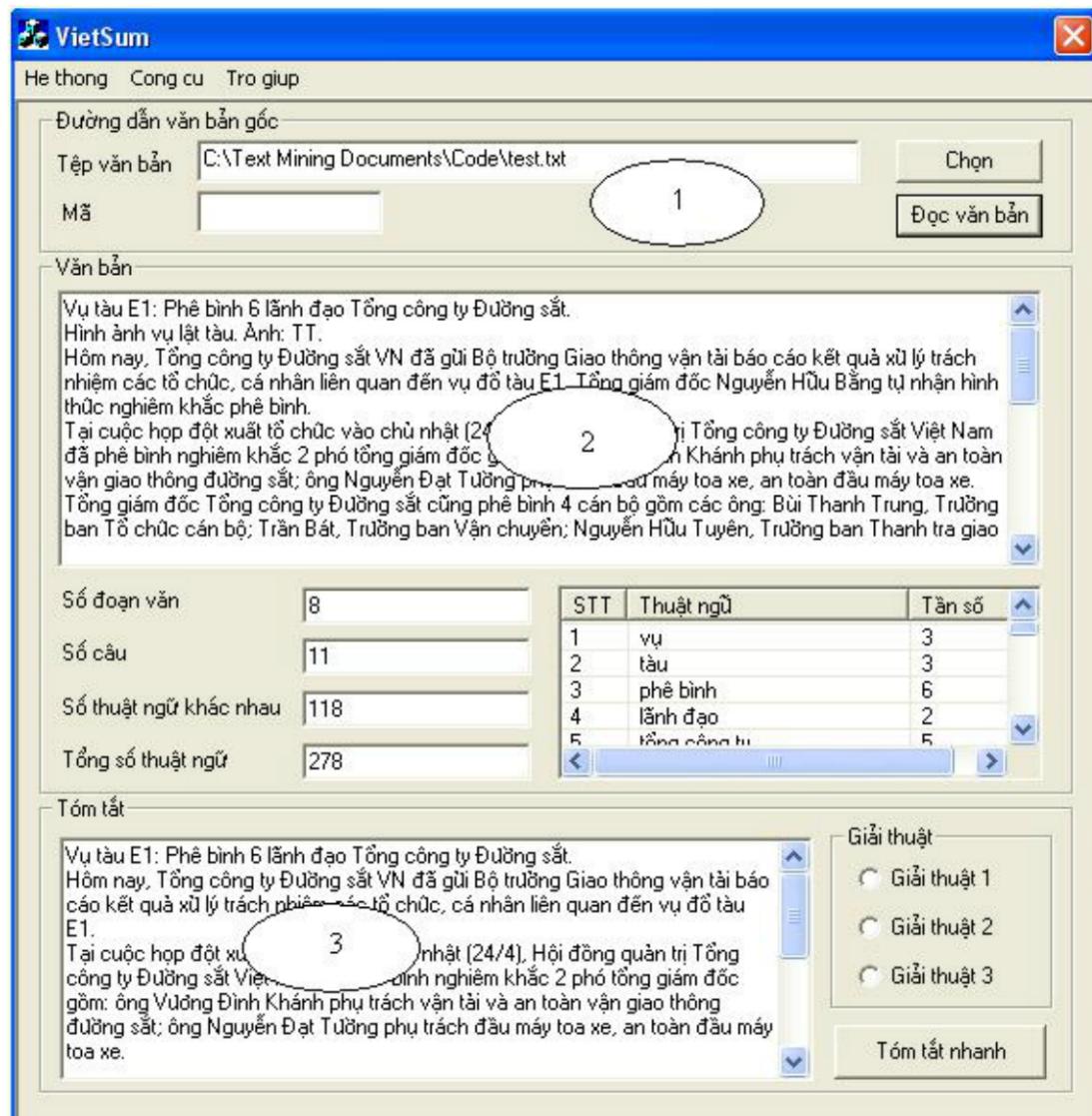
4.7.2.1 Các lớp chính được thiết cho chương trình:

- *Các lớp quản lý dữ liệu.*
 - + lớp CVNDocument lưu trữ dữ liệu có cấu trúc của một văn bản.
 - + lớp CVNParagraph lưu trữ dữ liệu có cấu trúc của một đoạn văn trong văn bản. Nó là lớp con của lớp CVNDocument.
 - + lớp CVNSentence lưu trữ dữ liệu của một câu trong văn bản. Nó là lớp con của lớp CVNParagraph.
- *Các lớp quản lý giao diện.*
 - + lớp CVietSumDlg quản lý giao diện chính.
 - + lớp CAalgo1Dlg, CAalgo2Dlg, CDlg3Dlg quản lý giao diện các chức năng tóm tắt theo từng giải thuật.
 - + lớp CVietSumApp quản lý ứng dụng, tức là phần khung của chương trình.
- *Các lớp thực hiện giải thuật*

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

- + lớp CAlgorithm1, CAlgorithm2, CAlgorithm3 chứa các hàm thực hiện giải thuật.
- + Lớp CTextFile quản lý việc tương tác với các file văn bản: đọc, ghi, nhận dạng mã ký tự.
- + lớp CTextPreprocessing thực hiện các bước tiền xử lý văn bản.

4.7.2.2 Giao diện chính chương trình



Hình 29: Giao diện chính của chương trình.

Trong đó:

Vùng 1 thực hiện chọn một văn bản để tóm tắt. Hệ thống chỉ được nghiên cứu thiết kế để tóm tắt các văn bản đơn lẻ (Single Document Summarization - SDS) chứ không phải các tập văn bản (Multi Documents Summarization - MDS). Tóm

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

nhiên có thể tóm tắt một tập văn bản bằng cách tóm tắt từng văn bản trong chúng. Tuy nhiên về tính chất đây cũng chỉ là tóm tắt SDS bởi hệ MDS cần phải thực hiện tóm tắt dựa trên cả sự liên kết về nội dung, tính chất của các văn bản trong cùng một tập dữ liệu kết hợp với các giải thuật khác.

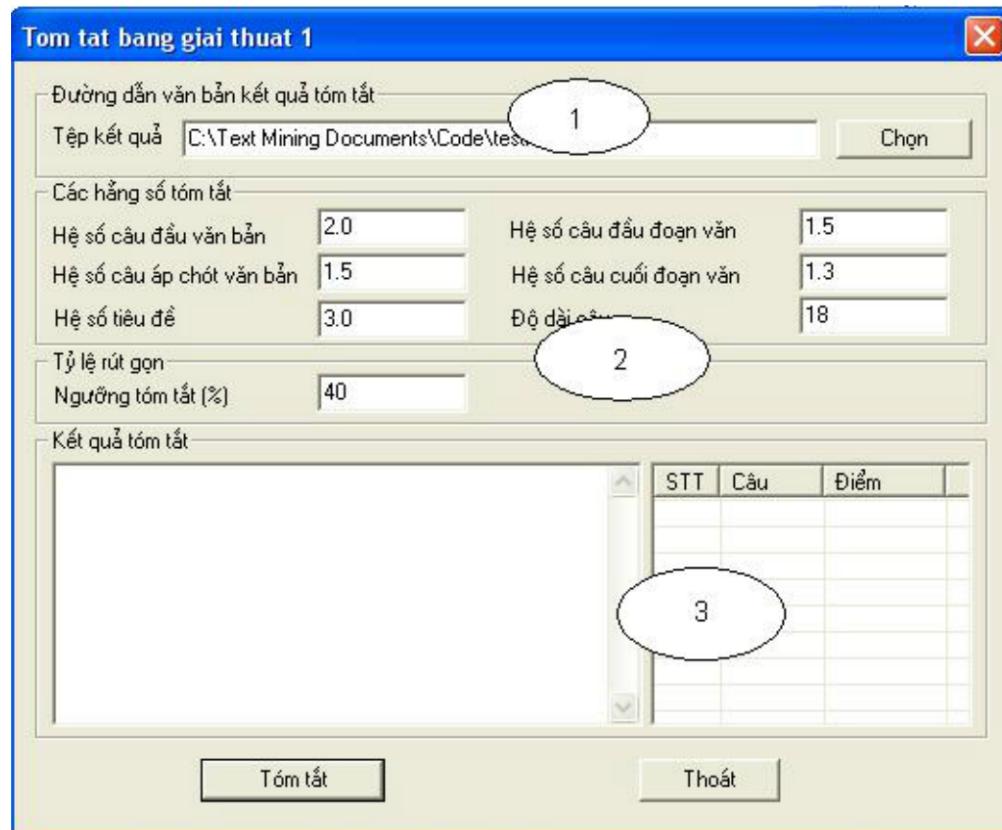
Vùng 2 cung cấp nội dung của văn bản cần tóm tắt. Trong đó, cửa sổ phía trên chứa văn bản gốc và phía dưới là các con số thống kê nội dung của văn bản cùng danh sách thuật ngữ xuất hiện trong văn bản.

Vùng 3 chứa kết quả của hệ thống. Một văn bản cần tóm tắt có thể được tóm tắt nhanh sử dụng một trong ba giải thuật với các hệ số và tùy chọn mặc định. Kết quả tóm tắt thể hiện trong cửa sổ bên trái của vùng.

Cũng có thể thực hiện tóm tắt cho một văn bản bằng cách áp dụng cụ thể từng giải thuật với các hệ số do người dùng đưa ra.

4.7.2.3 Giao diện giải thuật 1

Chức năng này được kích hoạt bằng cách chọn **Công cụ/Giải thuật 1** trên giao diện chính.



Hình 30: Giao diện giải thuật 1.

Trong đó, vùng 1 để tạo kết quả tóm tắt ghi ra tệp văn bản.

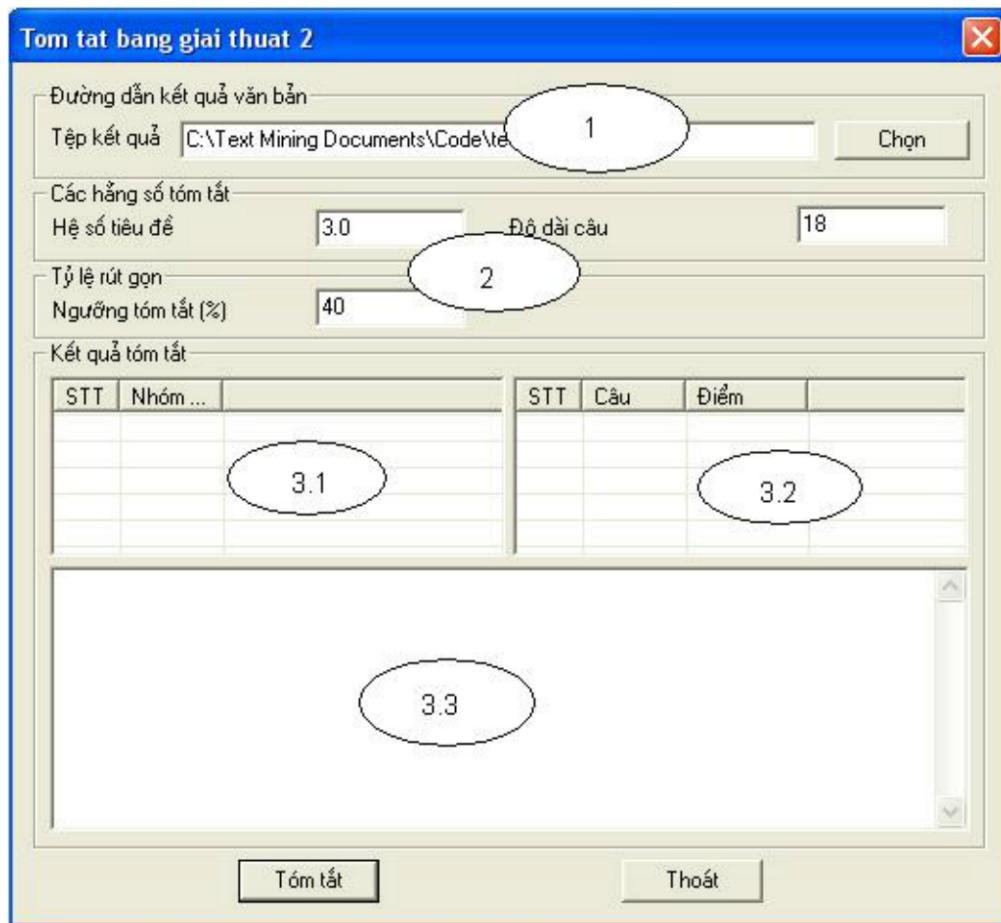
Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Vùng 2 để người dùng có thể nhập các hằng số tóm tắt để có thể tối ưu hoá tóm tắt.

Vùng 3 đưa ra kết quả tóm tắt, danh sách các câu trong văn bản gốc cùng với điểm của chúng để minh họa cụ thể cho kết quả.

4.7.2.4 Giao diện giải thuật 2

Giải thuật 2 được kích hoạt bằng cách chọn **Công cụ/Giải thuật 2** từ giao diện chính.



Hình 31: Giao diện giải thuật 2

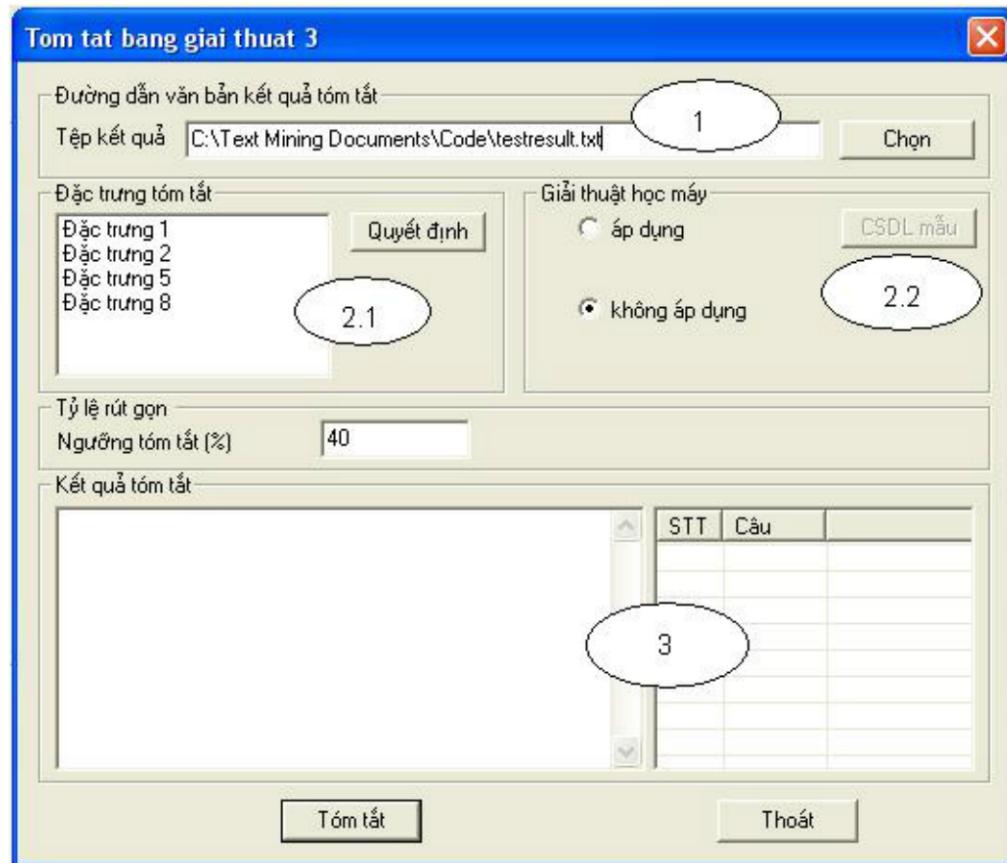
Trong đó, vùng 1 và vùng 2 cũng có chức năng tương tự như giao diện giải thuật 1.

Ở vùng 3.1, minh họa cho kết quả tóm tắt là danh sách các nhóm đoạn văn/câu văn đã được phân nhóm. 3.2 là kết quả ghi điểm cho mỗi câu trong văn bản và 3.3 thể hiện kết quả TTVB.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

4.7.2.5 Giao diện giải thuật 3

Giải thuật 3 được kích hoạt bằng cách chọn **Công cụ/Giải thuật 3** từ giao diện chính.



Hình 32: Giao diện giải thuật 3

Trong đó vùng 2.1 để lựa chọn các đặc trưng tóm tắt sẽ dùng trong giải thuật. Bởi vì không phải nhiều đặc trưng cùng kết hợp sẽ làm cho hiệu quả của giải thuật tốt hơn.

Vùng 2.2 cho người dùng hai lựa chọn có/không sử dụng giải thuật học máy. Giải thuật học máy được dùng để tìm ra luật kết hợp tốt nhất các đặc trưng tóm tắt. Nếu chọn áp dụng giải thuật học máy, người dùng phải cung cấp đường dẫn đến tập Tóm tắt mẫu cho chương trình. Mỗi một cặp văn bản - tóm tắt trong tập mẫu được lưu dưới dạng văn bản gốc - danh sách các câu tóm tắt của văn bản.

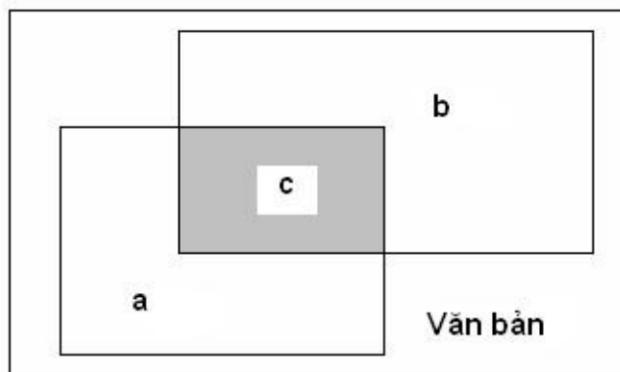
Vùng 3 thể hiện kết quả tóm tắt và danh sách các câu thoả mãn đặc trưng.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.8 Minh họa một số thực nghiệm và đánh giá

4.8.1 Đại lượng đánh giá độ chính xác.

Để đánh giá sự chính xác của quá trình thực hiện TTVB, hai giá trị sau được sử dụng: độ chính xác (precision) và độ bao (recall).



Hình 33: Precision và Recall

Giả sử một văn bản cần tóm tắt trong đó có a câu đúng (dựa theo tập tóm tắt mẫu), b câu mà hệ thống tìm kiếm được và c là giao của a và b.

* Độ chính xác (Precision).

Độ chính xác hay giá trị Precision được tính bằng:

$$precision = \frac{c}{b}$$

* Độ bao (Recall)

Độ bao hay giá trị Recall được tính bằng:

$$recall = \frac{c}{a}$$

Ví dụ: Một văn bản có 40 câu. Tóm tắt được cho là chính xác tuyệt đối do tác giả tạo ra bao gồm 15 câu. Văn bản này được đưa vào hệ thống tóm tắt tự động và cho ra kết quả sau (tương ứng với kết quả tìm được là 6 / 10 / 20 câu):

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

Kết quả tìm được	Kết quả đúng tìm được	<i>Precision</i>	<i>Recall</i>
6	4	0.27 (4/15)	0.67 (4/6)
10	6	0.40 (6/15)	0.60 (6/10)
20	9	0.60 (9/15)	0.45 (9/20)

Bảng 5: Minh họa các giá trị Precision và Recall

Có thể thấy nếu giá trị Precision càng cao thì giá trị Recall càng thấp và ngược lại Recall càng cao thì Precision càng thấp. Để đánh giá chính xác kết quả của một hệ thống không thể chỉ dựa vào một trong hai giá trị này mà phải kết hợp cả 2. Giá trị *precision = recall* khi kích thước tập kết quả tìm được bằng với kích thước tập kết quả mong muốn.

4.8.2 Cơ sở dữ liệu thực nghiệm

Các văn bản mẫu là các bài báo được lấy từ địa chỉ trang web của báo điện tử VnExpress: <http://www.vnexpress.net>.

Các thông số của tập dữ liệu văn bản:

- Số văn bản: 594 văn bản.
- Tổng dung lượng: 2.6 MB.
- Kích thước văn bản lớn nhất: 15 KB.
- Kích thước văn bản nhỏ nhất: 2 KB.
- Kích thước trung bình một văn bản: 4.5 KB.

Tập văn bản - tóm tắt mẫu cũng được lấy trong CSDL này, có 20 văn bản cùng với tóm tắt mẫu:

STT	Tiêu đề	Kích thước	Số câu tóm tắt
1	Bệnh nhân SARS dễ mắc lao phổi	4KB	8
2	Bibica thay đổi nhân sự cao cấp	2KB	6
3	Chỉ số giá tiêu dùng tháng 7 sẽ tiếp tục tăng	3KB	9
4	Chuẩn bị ban hành khung giá đất mới	2KB	6

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

5	Cỗ phiếu Bảo Minh đắt giá	4KB	10
6	Đua khuyến mại điện thoại VoIP	8KB	18
7	Sắp có thêm hạn ngạch dệt may đi EU	4KB	8
8	IncomBank tung ra thẻ chip vô danh đầu tiên	3KB	8
9	Vụ kiện tôm đe doạ tới xuất khẩu của Mỹ	5KB	12
10	Khiêm trách phó chủ nhiệm đoàn luật sư Hà Nội	4KB	9
11	Vàng Trung Quốc giá rẻ xâm lấn thị trường	2KB	6
12	Vinafood2 trúng thầu xuất khẩu 150.000 tấn gạo	2KB	8
13	Yukos vỡ nợ, có nguy cơ phá sản	4KB	10
14	Công ty Việt đầu tư xây nhà ở Mỹ	3KB	8
15	Cỗ phần hoá cần dứt khoát hơn	4KB	10
16	Chiến dịch săn lùng Rooney sôi sục khắp châu Âu	6KB	15
17	Lỗi nghiêm trọng trong game Unreal	3KB	8
18	Bán dân chau Á-Thái Bình Dương sẽ tăng trưởng 27,4%	3KB	7
19	Giới nữ trong thời đại công nghệ	4KB	9
20	Bảo vệ rùa bằng cá mập giả	5KB	12

Bảng 6: Tập tóm tắt mẫu

Tất cả dữ liệu được thử nghiệm trên máy Pentium III 866 Mhz với 256 MB bộ nhớ.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê

4.8.3 Thực nghiệm trên modul Tiền xử lý văn bản.

Để thử nghiệm hiệu quả của module Tiền xử lý văn bản, cần đánh giá tốc độ và độ chính xác của thuật ngữ được tách. Tốc độ của quá trình tách thuật ngữ và chuyên chung về dạng dữ liệu chuẩn được kiểm tra có số liệu sau:

Kích thước tập văn bản (KB)	Chiều dài thuật ngữ lớn nhất (ký tự)	Thời gian tách (s)	Tốc độ theo dung lượng (KB/s)	Tốc độ theo văn bản (văn bản/s)
607	30	6	101	19
607	15	4	152	28
1253	30	13	96	18
1253	15	8	157	30

Bảng 7: Kết quả tách thuật ngữ.

Đánh giá: Có thể thấy tốc độ tách thuật ngữ không phụ thuộc vào dung lượng của văn bản. Nhưng khi chiều dài của thuật ngữ lớn nhất thay đổi ảnh hưởng đáng kể đến tốc độ phân tách.

Nhận xét rằng trong từ điển thuật ngữ tiếng Việt, phần lớn các thuật ngữ đều có độ dài là 2 từ và rất ít thuật ngữ có độ dài 4 từ trở lên. Do vậy nếu cần tăng tốc độ tách thuật ngữ, có thể giảm chiều dài của thuật ngữ lớn nhất phải xét bằng chiều dài lớn nhất của một thuật ngữ có 2 từ trong từ điển. Hệ thống khi đó vẫn cho kết quả tốt trong khi tốc độ tách thuật ngữ giảm đáng kể.

4.8.4 Thực nghiệm trên các module Tóm tắt.

Việc đánh giá độ chính xác của các giải thuật tóm tắt tiếng Việt gặp nhiều khó khăn do hạn chế về nguồn dữ liệu mẫu chuẩn. Chưa có một đơn vị nào xây dựng các tóm tắt mẫu với số lượng lớn và công bố chúng rộng rãi.

Điều này gây ra nhiều trở ngại đối với tác giả trong quá trình xây dựng hệ thống, không chỉ bởi việc không đánh giá được kết quả chương trình mà còn bởi giải thuật 3 được xây dựng trong hệ thống phụ thuộc rất nhiều vào tập dữ liệu mẫu này.

Để giải quyết trước mắt vấn đề này, tác giả đề xuất phương án tự xây dựng tập tóm tắt mẫu bằng cách tận dụng kinh nghiệm đọc - hiểu - lượng giá thông tin của một số chuyên gia - con người tiếp xúc nhiều với dữ liệu văn bản (nhà báo, sinh viên, học sinh,...). Mỗi chuyên gia sẽ đọc một số văn bản sau đó tự đưa ra tóm tắt dựa trên kinh nghiệm của mình. Kết quả tuy chưa tạo nên các tóm tắt chính xác

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thông kê tuyệt đối xong đối với hệ thống tóm tắt tự động, đây cũng là những tập mẫu mong muốn.

Tuy vậy do thời gian có hạn, số lượng các tóm tắt mẫu này không lớn (20 - như trên đã liệt kê). Vì vậy tác giả hy vọng có thể tiếp tục mở rộng thêm tập dữ liệu mẫu này trong thời gian tới để có thể đánh giá cũng như nâng cao chất lượng của hệ thống.

Dưới đây là số liệu thông kê kết quả của ba giải thuật tóm tắt được sử dụng trong hệ thống, độ rút gọn thông tin là 50%:

	Giải thuật 1	Giải thuật 2	Giải thuật 3
Kết quả (Precision, Recall)	60.07%	72.45%	70.42%

Bảng 8. Đánh giá độ chính xác các giải thuật

Đánh giá: Hệ thống cho kết quả thấp đi khi hệ số rút gọn thông tin giảm. Bởi vì việc lựa chọn một câu làm tóm tắt sẽ khó hơn nếu như tỷ lệ câu đó nằm trong tóm tắt nhỏ hơn.

Tác giả đã thực hiện đánh giá về ngữ nghĩa qua các tóm tắt được tạo bởi hệ thống. Với 20 tóm tắt, đa phần đã mang đủ hết nội dung quan trọng của văn bản gốc. Sai só về sự chính xác được cảm nhận là không đáng kể. Bởi vậy tính thực tế của hệ thống lớn.

Việc thu thập tập dữ liệu mẫu mất khá nhiều thời gian nên kích thước của tập mẫu vẫn còn nhỏ. Chính vì vậy hệ thống chưa có nhiều điều kiện để thử nghiệm với dữ liệu lớn. Tác giả vẫn đang thu thập thêm các mẫu tóm tắt để có thể đưa đánh giá đúng hơn về tính chính xác của hệ thống bằng thực nghiệm.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

TỔNG KẾT

Có thể thấy bài toán TTVB là bài toán có giá trị ứng dụng rất lớn. Với sự phát triển của các kho dữ liệu khổng lồ và các kỹ thuật nâng cao khả năng tính toán của máy móc, các ứng dụng của TTVB sẽ được thực hiện ngày càng nhiều hơn theo nhu cầu của con người. Các kỹ thuật TTVB nói chung và TTVB tiếng Việt nói riêng sẽ còn được nghiên cứu và phát triển thêm trong khoảng thời gian tới.

Qua việc nghiên cứu và thực hiện đề tài này, tác giả đưa ra một số tổng kết sau:

(*) Các vấn đề đã giải quyết:

Trong phạm vi đồ án, tác giả đã thực hiện giải quyết được những vấn đề:

- Nghiên cứu lý thuyết tổng quan về bài toán TTVB, các phương pháp và xu hướng giải quyết bài toán.
- Phân tích các phương pháp có thể áp dụng cho bài toán TTVB tiếng Việt. Cụ thể là các phương pháp sử dụng kỹ thuật lượng giá, thống kê.
- Xây dựng một hệ thống TTVB cho tiếng Việt dựa trên các kỹ thuật đã phân tích.

(*) Hướng phát triển:

Trong thời gian tới tác giả hy vọng sẽ phát triển đề tài theo các hướng:

- Phát triển các kỹ thuật lượng giá để tăng thêm tính hiệu quả cho hệ thống.
- Tìm kiếm một số đặc trưng Tóm tắt cho kết quả cao đối với tiếng Việt.
- Xây dựng từ điển đồng nghĩa phục vụ cho hệ thống, từ điển WordNet tiếng Việt để mở rộng hệ thống với các kỹ thuật dựa trên độ liên kết ngữ nghĩa trong văn bản. Đặc biệt kỹ thuật áp dụng các chuỗi từ vựng (Lexical Chains) rất có tính khả thi.
- Nghiên cứu các phương pháp làm “mượt” (smoothing) kết quả để có thể từ tóm tắt Extract tạo nên tóm tắt Abstract.
- Phát triển hệ thống kết hợp với các hệ thống tìm kiếm bằng tiếng Việt trên Internet.

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

TÀI LIỆU THAM KHẢO

Tiếng Việt:

- [1] H. Kiếm, Đ. Phúc, “**Rút trích ý chính từ văn bản tiếng Việt hỗ trợ tạo nội dung**”, Trường Đại học Khoa học Tự Nhiên Tp. HCM, Việt nam.
- [2] P. Liêm, “**Ứng dụng mô hình tập thô dung sai trong xử lý văn bản**”, Trường Đại học Bách Khoa Hà Nội, (2004).
- [3] C. Trang, “**Bài toán phân nhóm văn bản tiếng Việt**”, Trường Đại học Bách Khoa Hà Nội, (2004).

Tiếng Anh:

- [4] J Larocca Neto, AD Santos, CAA Kaestner, and AA Freitas, “**Document Clustering and Text Summarization**”. In N Mackin, editor, Proc 4th International Conference Practical Applications of Knowledge Discovery and Data Mining (PADD-2000), (2000).
- [5] M. Mitra, A. Singhal, and C. Buckley. “**Automatic text summarization by paragraph extraction**”. In ACL’97/EACL’97 Workshop on Intelligent Scalable Text Summarization, (1997).
- [6] H. P. Luhn, “**The Automatic Creation of Literature Abstracts**”, IBM Journal of Research Development, (1959).
- [7] R. Barzilay and M. Elhadad. “**Using lexical chains for text summarization**”, (1997).
- [8] Chinatsu Aone, Mary Ellen Okurowski, James Gorlinsky, and Bjornar Larsen. “**A Scalable Summarization System Using Robust NLP**”, (1997).
- [9] Jaime Carbonell and Jade Goldstein. “**The use of MMR, diversity-based reranking for reordering documents and producing summaries**”. In Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, (1998).
- [10] D. Radev, H. Jing, and M. Budzikowska. “**Centroid-based summarization of multiple documents: sentence extraction, utility-based evaluation and user studies**”, (2000).
- [11] Karen Sparck-Jones and Tetsuya Sakai. “**Generic summaries for indexing in IR**”, New Orleans, LA, (2001).
- [12] K. Zechner. “**Fast generation of abstracts from general domain text corpora by extracting relevant sentences**”, (1996).

Xây dựng hệ thống Tóm tắt văn bản tiếng Việt sử dụng các kỹ thuật lượng giá, thống kê

- [13] J. Kupiec, J. Pedersen, F. Chen, “**A Trainable Document Summarizer**”, Xerox Research Center, (1995).
- [14] AI Berger and Mittal, “**A system for summarization web pages**”, In Proc ACM SIGIR, (2000).
- [15] Darin Brezeale, “**The Organization of Internet Web pages Using Wordnet and Self-Organizing maps**”, MSC Thesis, The University of Texas at Arlington, USA, (1999).
- [16] Daniel Mallett, “**Text summarization-an annotated bibliography**”, (2003).
- [17] Smaranda Muresean, “**Combining Linguistic and machine learning techniques for eamil summarization**”, Columbia University, (2001).