ĐẠI HỌC QUỐC GIA HÀ NỘI
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ

NGUYỄN VŨ CHI LOAN

NGHIÊN CỨU CÁC PHƯƠNG PHÁP TRÍCH RÚT TỪ KHOÁ TỪ TRANG WEB VÀ ỨNG DỤNG

Ngành: Công nghệ thông tin
Chuyên ngành: Kỹ thuật phần mềm
Mã số: 60480103

TÓM TẮT LUẬN VĂN THẠC SỸ
Ngành: Kỹ thuật phần mềm

HÀ NỘI - 2017

MỞ ĐẦU

Hiện nay việc trích rút từ khoá từ trang web là một việc hết sức quan trọng với một lượng thông tin khổng lồ ngày càng bùng nổ và tăng theo cấp số nhân trên Internet. Bài toán trích rút từ khoá từ trang web đã giúp giải quyết rất nhiều bài toán thực tế như: Tìm kiếm thông tin, tóm tắt văn bản…Rất nhiều người có nhu cầu tổng hợp và tóm tắt lại các thông tin để thuận lợi cho việc tổng hợp các thông tin đó.

Việc trích chọn từ khóa là ứng dụng quan trọng nhất trong các engine tìm kiếm. Vì hiện nay các engine này chủ yếu vẫn tìm kiếm dựa vào từ khóa. Đó chính là một trong những động lực để phát triển bài toán trích rút từ khoá từ trang web. Nhiệm vụ bài toán đặt ra là cần tìm được một tập các từ khoá sao cho các từ khoá này phải sát với nội dung của tài liệu văn bản.Vì thế các phương pháp tóm tắt tự động được nghiên cứu và phát triển.

Bài toán trích rút từ khoá không chỉ dừng lại ở trích rút từ khoá mà nó còn mở rộng ra trích rút câu hoặc các loại dữ liệu đa phương tiện như hình ảnh, âm thanh và video. Một ứng dụng điển hình cho việc ứng dụng của tóm tắt dữ liệu tự động là các máy tìm kiếm, trong đó nổi bật nhất là bộ máy tìm kiếm Google.

Với thực tế nêu trên, luận văn đã đề xuất một phương pháp giải quyết bài toán trích rút từ khoá từ trang web tiếng Anh qua đề tài "***Nghiên cứu các phương pháp trích rút từ khoá từ trang web và ứng dụng***". Mục tiêu của đề tài là nghiên cứu giải quyết bài toán sinh từ khoá theo phương pháp chính là: *đồ thị web*. Qua thực nghiệm cho thấy các hướng tiếp cận này là khả quan và có triển vọng với độ chính xác khá tốt, nếu kết hợp với các từ khoá của chính các chuyên gia thì tập từ khoá sinh ra là

khá đầy đủ và chính xác.

Ngoài phần **MỞ ĐẦU** và **KẾT LUẬN**, kết cấu của luận văn bao gồm các chương sau:

- ***Chương 1***: Giới thiệu về bài toán. Nêu các khái niệm cơ bản về bài toán. Các ứng dụng của bài toán. Những thách thức đặt ra cho bài toán.

- ***Chương 2***: Các phương pháp trích rút từ khoá từ trang web. Giới thiệu phương pháp TextRank áp dụng để trích rút từ khoá từ trang web.

- ***Chương 3:*** "Kết quả thực nghiệm và đánh giá". Đưa ra những kết quả đã làm, và đánh giá kết quả.

CHƯƠNG I. GIỚI THIỆU BÀI TOÁN TRÍCH RÚT TỪ KHOÁ TỪ NỘI DUNG VĂN BẢN TRÊN TRANG WEB

1.1. Đặt vấn đề

Sự phát triển nhanh chóng của Internet và đặc biệt là sự bùng nổ thông tin làm cho thông tin ngày càng khó kiểm soát, và trùng lặp nhiều. Tìm kiếm thông tin hiện nay càng là nhu cầu thiết yếu của nhiều người trên nhiều lĩnh vực khác nhau. Sự đột phá về công nghệ đã cho ra những máy tìm kiếm phần nào đã giải quyết được sự ngập lụt thông tin này. Vì nhu cầu sử dụng máy tìm kiếm hiện nay là rất lớn.Tìm kiếm và tổng hợp thông tin không thuận lợi gây ra khó khăn để có được 1 kết quả tìm kiếm đúng mục đích và ít tốn kém thời gian.

Hiện nay các máy tìm kiếm (Google, Bing, Coccoc, …) vẫn chủ yếu dựa vào từ khoá để tìm kiếm trang web. Vì vậy khi một trang web mà ta biết trước tập từ khoá sẽ giúp tìm kiếm chính xác hơn .Trích rút từ khoá tự động trong nội dung văn bản trên web là một bài toán được đặt ra trước nhu cầu thực tế. Ứng dụng quan trọng nhất của trích chọn từ khoá sử dụng phương pháp TextRank chính là tìm kiếm.

Các từ khóa là các từ, cụm từ nhằm miêu tả nội dung của trang web, văn bản một cách ngắn gọn nhất, chính xác nhất.

Nhận thấy đây là 1 đề tài mới, có tính khoa học là nền tảng của nhiều ứng dụng thực tế, nên tác giả đã quyết định chọn đề tài " Nghiên cứu các phương pháp trích rút từ khoá từ trang web và ứng dụng". Đề tài này nghiên cứu các phương pháp trích rút từ khoá và tập trung chủ yếu vào phương pháp TextRank để trích rút từ khoá tự động từ nội dung văn bản trên web.

1.2 Khái niệm và các đặc trưng của từ khóa

Từ khóa là một từ hay một cụm từ dùng để mô tả một cách chính xác, ngắn gọn nhất nội dung chính của một tài liệu (văn bản, hay các trang web). Trong tiếng Anh, từ khóa được thể hiện dưới nhiều thuật ngữ khác nhau như: keywords, term, query term, hay tags; nhưng ý nghĩa của chúng là giống nhau. Tập các từ khóa có thể coi như là một **bản tóm tắt đơn giản nhất** của văn bản. Tập các từ khóa sẽ nói lên rõ hơn ý nghĩa của văn bản hay trang web đó.
Một số đặc điểm, tiêu chí ảnh hưởng đến quá trình rút trích từ khóa:
Từ dừng, loại từ, từ có lien quan đến tiêu đề, số lượng…

1.3 Đánh giá các từ khoá

Dựa vào

a. Tính phổ biến

b. Tính đặc trưng

c. Hướng ngư ời sử dụng

1.4. Thách thức của bài toán sinh từ khóa cho trang web

1.4.1. Đối với các trang có nội dung tập trung

Các kĩ thuật trích xuất từ khóa đối với văn bản sẽ được áp dụng như tần số từ, vị trí từ trong các đoạn văn, độ tương đồng từ..Nói chung, việc lọc nhiễu cho các trang có nội dung tập trung là một điều quan trọng giúp tăng chất lượng của việc trích xuất từ khóa. Với những bài viết quá dài thì thời gian chạy cũng khá lâu.

1.4.2. Đối với các trang có nội dung tổng hợp

Các trang web luôn muốn những thông tin cập nhật sẽ được hiển thị trên trang đầu khi mà người dùng tới trang của họ. Những trang đầu này còn gọi là các trang chủ. Ngoài thỏa mãn là một công cụ tìm kiếm, web portal cung cấp các thông tin dịch vụ khác như báo tin tức, chứng khoán, giải trí. Ví dụ về các web portal như: AOL, MSN, yahoo, iGoogle. Nếu áp dụng việc trích xuất từ khóa áp dụng đối với nội dung trong các trang web này sẽ dẫn đến kết quả không chính xác. Cần có những phương pháp khác để có thể sinh từ khóa cho loại trang này, và trong luận văn này tôi áp dụng phương pháp dùng đồ thị Web và log hỗ trợ.

1.5. Ứng dụng của từ khóa trong các lĩnh vực

Phạm vi ứng dụng:

- Các kho dữ liệu văn bản lớn như các thư viện số phát triển rất nhanh dẫn đến gia tăng giá trị thông tin tóm tắt.

- Hỗ trợ người dùng nhận biết về nội dung của tài liệu và kho tài liệu.
- Ứng dụng trong truy vấn thông tin cho phép mô tả những tài liệu trả về từ kết quả truy vấn. Đính hướng tìm kiếm cho người dùng.
- Nền tảng cho chỉ mục tìm kiếm.
- Là đặc trưng dùng trong kỹ thuật phân loại, gom cụm tài liệu.

1.6. Tổng kết chương

Chương này tôi đã trình bày những khái niệm của từ khóa, và bài toán trích xuất từ khóa cho trang web, thách thức của nó trong các tài liệu web. Và qua đây, chúng ta cũng thấy được tầm quan trọng của việc sinh từ khóa trên các lĩnh vực khác nhau.

CHƯƠNG 2. CÁC PHƯƠNG PHÁP TRÍCH RÚT TỪ KHOÁ TỪ TRANG WEB

2.1. Phương pháp tần số từ

- Cách tiếp cận của TF x IDF sẽ ước lượng được độ quan trọng của 1 từ đối với 1 văn bản trong danh sách tập tài liệu văn bản cho trước. Nguyên lý cơ bản của TF x IDF là: " Độ quan trọng của 1 từ sẽ tăng lên cùng với số lần xuất hiện của nó trong văn bản và sẽ giảm xuống nếu từ đó xuất hiện trong nhiều văn bản khác

- Lý do đơn giản là vì nếu 1 từ xuất hiện trong nhiều văn bản khác nhau thì có nghĩa là nó là từ rất thông dụng , vì thế khả năng nó là từ khoá sẽ giảm xuống(Ví dụ như các từ " Vì thế", " Tuy nhiên", " Nhưng", " và"

- Do đó độ đo sự quan trọng của 1 từ trong tài liệu f sẽ được tính = tf x idf

Với tf: độ phổ biến của từ t trong tài liệu f

idf : nghịch đảo độ phổ biến của từ t trong các tài liệu còn lại

Công thức tính tổng quát:

$Weight_{wi}$ = tf * idf

Với tf = Ns(t)/ $\sum w$

Idf = log ($\sum d$(d: t $\in$ d)

Ns(t) : Số lần xuất hiện của từ t trong tài liệu f

$\sum w$: Tổng số các từ trong tài liệu f

$\sum d$: Tổng số văn bản

d: t ∈ d: số tài liệu có chứa t

Ví dụ: 1 văn bản có 100 từ, trong đó từ " máy tính" xuất hiện 10 lần thì độ phổ biến: tf(" máy tính") = 10/100 = 0.1

Giả sử có 1000 tài liệu, trong đó có 200 tài liệu chứa từ " máy tính"

→ Idf = log(1000/200) = 0.699

Như vậy ta tính được độ đo tf x idf = 0.1x 0.699 = 0.0699

→ Nếu tf x idf vượt một ngưỡng xác định, các cụm từ khoá được tìm thấy và được gán trọng số. Những từ nào có trọng số cao thì được chọn.

2.2 Phương pháp TextRank để trích rút từ khoá cho trang web

Phương pháp TextRank đề xuất một phương pháp xử lý ít nhất một văn bản ngôn ngữ tự nhiên sử dụng một đồ thị.

2.2.1 Mô hình TextRank

Như trên ta thấy thuật toán xếp hạng dựa trên đồ thị là cách đưa ra cách chọn đỉnh quan trọng trong đồ thị dựa trên các thông tin toàn cục của các đỉnh trong đồ thị. Ý tưởng của thuật toán này dựa trên hai yếu tố: bỏ phiếu và đề cử. ". Khi đỉnh đầu tiên liên kết với đỉnh thứ hai, ví dụ như thông qua mối quan hệ kết nối hoặc cạnh biểu đồ. Mỗi một liên kết đến đỉnh đang xét thì nó được 1 phiếu bầu. Như vậy, càng nhiều phiếu bầu thì đỉnh đó càng quan trọng. Từ cách xác định trên thì trọng số của một đỉnh chính là số phiếu bầu cho đỉnh đó.

Ta có đồ thị G = (V, E) là đồ thị có hướng. Trong đó:

V: là tập các đỉnh

E: là tập các cạnh của đồ thị, E là tập con của V x V(E $\subseteq$ V xV). Với mỗi đỉnh V_i thì ta có:

- In (V_i) là tập các đỉnh trỏ đến V_i

- Out(V_i) là tập các đỉnh mà V_i trỏ đến.

Trọng số của đỉnh V_i được xác định như sau:(Brin and Page, 1998):

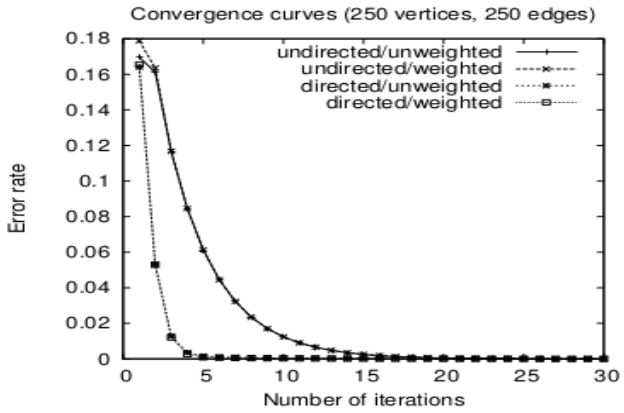
$$S(V_j) = (1 - d) + d* \sum_{j \in \ln(V_j} \frac{1}{|Out(V_j)|} S(V_j) \quad (1)$$

Trong đó d là nhân tố giảm, có giá trị từ 0 đến 1. Nó là xác suất mà một đỉnh có liên kết đến một đỉnh bất kỳ trong đồ thị. Đối với các trang web thì d là xác suất người dùng nhấn vào một liên kết bất kỳ và xác suất để người dùng vào một trang web hoàn toàn mới là 1 – d. Theo PageRank thì d = 0.85. Đây cũng là xác suất sẽ được sử dụng trong TextRank.

2.2.2. Đồ thị vô hướng

Việc áp dụng thuật toán TextRank vào đồ thị vô hướng cũng giống như với đồ thị có hướng. Có một điểm cần lưu ý, đó là trong đồ thị vô hướng thì số đỉnh vào bằng số đỉnh ra.

Ta có các hình vẽ sau:



Hình 2.2: Đường cong hội tụ của phương pháp xếp hạng dựa trên đồ thị với đồ thị có hướng – vô hướng, có trọng số - không có trọng số, 250 đỉnh và 250 cạnh

Trong hình 10 thì đường cong hội tụ cho đồ thị được sinh ngẫu nhiên với 250 đỉnh và 250 cạnh, với ngưỡng dừng là 10^{-5}(ngưỡng này được xác định đủ nhỏ để thuật toán dừng tính toán) cho thấy số lần lặp của quá trình tính toán không cao mặc dù số lượng đỉnh và cạnh lớn.

2.*2.3 Đồ thị có trọng số*

Vì thuật toán Pagerank ban đầu chỉ sử dụng đồ thị không trọng số do gần như không có tình huống một trang web có nhiều liên kết đến một trang nào đó trong môi trường web. Tuy nhiên đối với các văn bản trong ngôn ngữ tự nhiên thì việc một văn bản nào đó có nhiều thành phần tham chiếu đến một văn bản khác là hoàn toàn xảy ra. Do đó, để cải tiến Pagerank cho phù hợp với ngôn ngữ tự nhiên, thuật toán Textrank sử dụng đồ thị có trọng số. Trọng số ở đây được định nghĩa là độ dài kết nối giữa hai đỉnh V_i và V_j kí hiệu w_{ij}. Từ đó suy ra công thức (1) phải được thay đổi để phù hợp với đồ thị có trọng số trong thuật toán Textrank. Ta được công thức mới như sau:

$$S(V_j) = (1 - d) + d* \sum_{j \in \ln(V_j} \frac{w_{ij}}{\sum_{v_k \in Out(V_j)} w_{jk}} WS(V_j) \quad (2)$$

Như vậy, theo hình (1) ỏ trên thì số lần lặp lại tính toán để có độ tụ đạt ngưỡng 10^{-5} của đồ thị có trọng số và đồ thị không có trọng số là tương đương nhau.

2.2.4 Đồ thị hoá văn bản

Tuỳ vào các loại và đặc trưng để đưa vào đồ thị mà có các cách thức làm việc. nhưng cách thức hoạt động của thuật toán xếp hạng dựa trên đồ thị áp dụng cho ngôn ngữ tự nhiên có các bước như sau:

- Xác định đơn vị văn bản dùng tốt nhất cho từng công việc, thêm vào là đỉnh của đồ thị.
- Xác định quan hệ kết nối giữa các đơn vị văn bản đã xác định ở

trên để vẽ các cạnh giữa các đỉnh trong đồ thị. Các cạnh này có thể là vô hướng hoặc có hướng, có trọng số hoặc không có trọng số

- Lặp lại thuật toán xếp hạng cho đến khi độ tụ thoả mãn ngưỡng.
- Sắp xếp các đỉnh dựa trên các trọng số đã được tính toán trong bước trên.

Như vậy, thuật toán này giúp cho chúng ta làm được hai việc: Trích rút từ khoá và trích rút câu trong văn bản ngôn ngữ tự nhiên. Vấn đề được đề cập ngay sau đây.

2.2.5 Sử dụng TextRank để trích rút từ khoá

Thuật toán trích rút từ khoá TextRank là thuật toán hoàn toàn không giám sát. Cách thức hoạt động như sau:

- Tách từ và gán nhãn, có các bộ lọc ngữ nghĩa. Để tránh gia tăng kích thước đồ thị thì áp dụng các đơn vị từ vựng phái có độ dài nhất định(n- gram).
- Đưa tất cả các đơn vị từ vựng có ở bước trên vào đồ thị. Các cạnh được đưa vào để liên kết các đơn vị từ vựng đồng xuất hiện với khoảng cách N từ. Sau khi dựng xong đồ thị(vô hướng, không trọng số) thì khởi tạo trọng số cho các đỉnh giá trị là 1. Và theo hình 10 thì số lần lặp lại từ 20-30 của thuật toán sẽ cho kết quả đạt ngưỡng 10^{-5}.

  Sau khi có kết quả cho mỗi đỉnh thì thực hiện quá trình sắp xếp ngược trọng số. T đỉnh đầu tiên sẽ được đưa vào quá trình tiếp theo, $5 \leq T \leq 20$. Ở đây thì T được lấy theo kích thước văn bản đầu vào.
- Sau bước trên ta được một tập các đơn vị từ vựng. Các đơn vị liền kế nhau thì được ghép lại với nhau để tạo thành từ khoá dài.

❖ **Thuật toán TextRank gồm 5 giai đoạn như sau:**

Bước 1:

- Phần xử lý ngôn ngữ tự nhiên sử dụng thuật toán của Stanford (open source). Kết quả trả về là một tập các terms. Một term có thể là một danh

từ, hoặc một tính từ

Ví dụ**:** trong câu**: "**the cars are loaded onto a train car with the help of Wrench**"** thì các term là**: cars| train| car| help|Wrench.**

Bước 2:

- Tiếp theo sử dụng thuật toán TextRank để đánh trọng số cho các term trong bước 1. Ý tưởng là như sau: (Theo bài báo của Rada Mihalcea and Paul Tarau, 2004)

a. Tất cả các term sẽ được biểu diễn như các đỉnh của graph, 2 term được nối với nhau nếu chúng cùng thuộc một sentence và cách nhau từ 2 terms.-10 terms

Ví dụ: Từ các term ở trên thì **cars** sẽ được liên kết với **train, car**. Term **train** sẽ được liên kết với các term **cars, car, help**.

Như vậy một graph đã được xây dựng. Để đánh trọng số cho các đỉnh của graph, chúng ta sử dụng thuật toán được phát triển từ thuật toán PageRank trong bài báo mới nhất

b. Giả sử đối với mỗi đỉnh v_i , gọi $S(v_i)$ là trọng số của nó. Vậy thì phương trình quan hệ giữa đỉnh và các đỉnh kề của nó sẽ là:

$$S(v_i) = (1-d) + d \times \sum_{v_j \in C(v_i)} \frac{attr(v_i, v_j)}{\sum_{v_k \in C(v_j)} attr(v_j, v_k)} \times S(v_j)$$

Trong đó d = 0.85 là hằng số của thuật toán, $attr(v_i, v_j) = \frac{freq(v_i) x freq(v_j)}{freq(v_i) + freq(v_j)}$

ở đó $freq(v_i)$ là tần số xuất hiện của từ v_i trong văn bản

$freq(v_j)$ là tần xuất hiện của từ v_j trong văn bản

- Giải hệ thống phương trình hàm này bằng cách đưa vào các giá trị trong khởi tạo bất kỳ và số vòng lặp, chúng ta đạt được các trọng số cho mỗi đỉnh
- Sau bước b) chúng ta lấy ra 5% các đỉnh có giá trị trọng số cao nhất. Một đỉnh có trọng số càng cao nếu như đỉnh đó xuất hiện nhiều lần trong văn bản hoặc có nhiều liên kết đến các đỉnh khác hoặc có liên kết đến các đỉnh có trọng số cao khác.
- Chúng ta coi các đỉnh này sẽ là các **topic** chính của phim.

Bước 4:

Sử dụng thuật toán n-gram để tìm các keword phrase từ các term tìm được trong bước 1. Trọng số của phrase sẽ bằng tổng các trong số của các term mà nó chứa được tính trong bước 3.

Ví dụ: trong câu**: "the cars are loaded onto a train car with the help of Wrench"** thì các term là**: cars| train| car| help|Wrench.** Các term phrases sẽ là**: cars| train car|help|Wrench.**

2.3 Tổng kết chương

Chương này đã giới thiệu những phương pháp cơ bản để giải quyết bài toán trích rút từ khóa trong nội dung văn bản trên các trang Web. Các phương pháp này hiệu quả đối với một số miền, và có thể áp dụng trong nhiều bài toán khác nữa. Trong chương tiếp, tôi xin trình bày về thực nghiệm và đánh giá.

CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ

Trong chương này tôi chỉ tập trung vào thực nghiệm và đánh giá cho phương pháp TextRank, lí do vì tác giả nhận thấy đây là một phương pháp mới, hơn nữa nó có tính ứng dụng cao trong thực tế. Tại sao nói phương pháp này có tính phổ biến cao vì trong luận văn này hướng nghiên cứu của tác giả dựa vào bài báo của tác giả Rada Mihalcea and Paul Tarau năm 2004 có đến 1640 lượt được trích dẫn và được các chuyên gia xây dựng riêng một Package thực nghiệm bằng các ngôn ngữ khác nhau trong đó có Java và python . Các phương pháp còn lại đã có các cá nhân, tổ chức hay công ty nghiên cứu và áp dụng.

Để đánh giá độ tốt của giải pháp đề xuất, tôi đã thực hiện đánh giá theo 2 cách:

- Thu thập dữ liệu là các văn bản thô thuộc nhiều chủ đề khác nhau đã được các chuyên gia đánh giá và trích rút từ khoá, so sánh kết quả trích rút từ khoá của các chuyên gia với của hệ thống trích rút bởi TextRank.
- Thu thập dữ liệu là các văn bản thô thuộc chủ đề về phim ảnh và có từ khoá đã được trích rút sẵn trên trang web cho từng văn bản. So sánh kết quả trích rút từ khoá trên web do các chuyên gia đánh giá với hệ thống trích rút từ khoá thực hiện bởi Textrank.

3.1 Yêu cầu thử nghiệm và tập dữ liệu thử nghiệm

Tập dữ liệu thực nghiệm

1. Dữ liệu thực nghiệm tác giả sử dụng trong luận văn được lấy từ tập dữ liệu tải về trên trang web: https://github.com/zelandiya/keyword-extraction-datasets do các chuyên gia tổng hợp và đánh giá thuộc các chủ đề khác nhau và có độ dài khác nhau. Chi tiết như sau:

Bảng 3.1 : Danh sách chủ đề và số lượng văn bản tương ứng

STT	Chủ đề	Dung lượng
1	Hệ thống phân tán	300KB

2	Khoa học	300KB

2. Cùng với tập dữ liệu được tác giả sưu tầm về chủ đề phim ảnh và diễn viên. Chi tiết như sau:

Bảng 3.2: Danh sách chủ đề và số lượng văn bản tương ứng

STT	Chủ đề	Số văn bản
1	Phim	50
2	Phim hoạt hình	50

3.2. Cài đặt thử nghiệm ứng dụng

3.2.1. Yêu cầu phần cứng và phần mềm

Cấu hình phần cứng máy tính sử dụng để cài đặt chương trình:

Bảng 3.3: Cấu hình phần cứng máy tính sử dụng để cài đặt chương trình

Thành phần	Chỉ số
CPU	Intel® Core™ i5 CPU
RAM	2.00 GB
OS	Windows 7 Ultimate
Bộ nhớ ngoài	300GB

Danh mục phần mềm sử dụng trong thực nghiệm:

Chương trình thực nghiệm được viết bằng ngôn ngữ python phiên bản 2.7 và các thư viện Numpy và Scipy. Trong luận văn có sử dụng công cụ phần mềm hỗ trợ trong quá trình thực hiện thực nghiệm:

Bảng 3.4: Danh mục phần mềm sử dụng trong thực nghiệm

STT	Tên phần mềm	Tác giả	Nguồn
1	Package index Owner: summanlp	Federico Barries, Federico	http://pypi.python.org/pypi/summa/0.0.7

		lopez	

3.2.2. Giới thiệu cấu trúc chương trình

Các bước của chương trình bao gồm:

- Thu thập các file text cần trích rút từ khoá là đầu vào của bài toán trích rút
- Trích rút từ khoá của các file dựa vào thuật toán TextRank đã trình bày ở chương 2
- Đánh giá chung về kết quả thu được

3.3 Phương pháp đánh giá

Công thức tính độ chính xác (precision) và độ nhớ lại (recall) của mỗi phương pháp áp dụng trên văn bản thứ i như sau:

$$\text{Precision(i)} = \frac{A \cap B}{B}$$

$$\text{Recall(i)} = \frac{A \cap B}{A}$$

Một hệ thống IR (Information Retrieval – Trích xuất thông tin) cần phải cân đối giữa recall và precision, bởi vậy một độ đo khác cũng thường được sử dụng đó là

F – score được xây dựng dựa trên recall và precision.

$$\text{Fscore} = \frac{\text{Re}\,call x \,\text{Pr}\,ecision}{(recall + precision)/2}$$

Precision, recall và F- score là các độ đo cơ bản của 1 tập các tài liệu được trích rút. Trên thực tế, đôi khi ta không thể sử dụng trực tiếp các độ đo này để so sánh hai danh sách có sắp xếp các tài liệu trả về, bởi chúng không hề quan tâm đến thứ tự nội tại các tài liệu[7].

Ví dụ: chúng ta hãy so sánh một tập hợp 15 cụm từ khóa hàng đầu được tạo ra bởi một trong những phương pháp sử dụng bộ đệm Porter:

*grid comput, grid, **grid servic discoveri**, web servic, servic*

*discoveri, grid servic, **uddi**, distribut hash tabl, discoveri of grid, **uddi registri**, rout, proxi **registri**, **web servic discoveri**, qos, **discoveri***

Với bộ tiêu chuẩn vàng tương đương với 19 cụm từ chính (một tập hợp được chỉ định bởi cả tác giả và độc giả):

grid servic discoveri**, **uddi**, distribut web-servic discoveri architectur, dht base uddi registri hierarchi, deploy issu, bamboo dht code, case-insensit search, queri, longest avail prefix, qo-base **servic** discoveri, autonom control, **uddi registri**, scalabl issu, soft state, dht, **web servic**, grid comput, md, **discoveri

Hệ thống đã xác định chính xác 6 cụm từ chính, dẫn đến độ chính xác 40% (6/15) và độ hồi tưởng lại 31,6% (6/19). Với kết quả cho từng tài liệu riêng lẻ, tôi tính toán độ chính xác, hồi tưởng trung bình và điểm F có thể đạt được qua cụm từ khóa kết hợp là khoảng 75%, bởi vì không phải tất cả các cụm từ khóa thực sự xuất hiện trong tài liệu.

3.4. Một số kết quả thu được

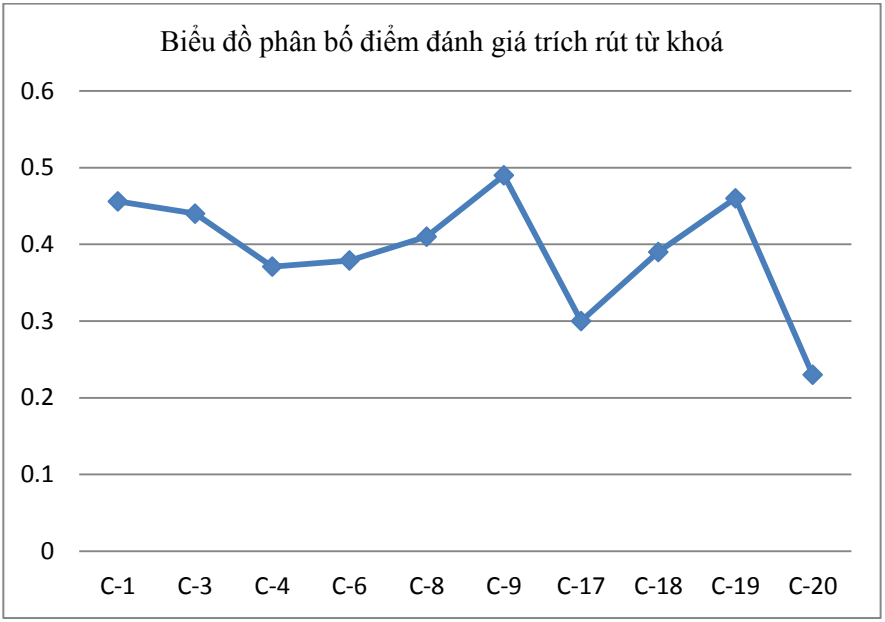
Kết quả đánh giá với chủ đề " Hệ thống phân tán"

Bảng 3.5: So sánh kết quả đánh giá hệ thống tóm tắt tự động sử dụng Textrank và các chuyên gia

STT	Tên file	Từ khoá của chuyên gia	Từ khoá trích rút của TextRank	Từ khoá chung	Recall	Precision	F-score
1	C-1	42	50	21	0.5	0.42	0.456
2	C-3	40	50	20	0.5	0.4	0.44
3	C-4	47	50	18	0.383	0.36	0.371
4	C-6	29	50	15	0.517	0.3	0.379
5	C-8	38	50	18	0.474	0.36	0.41
6	C-9	23	50	18	0.783	0.36	0.49

7	C-17	37	50	13	0.351	0.26	0.3
8	C-18	27	50	15	0.56	0.3	0.39
9	C-19	19	50	16	0.84	0.32	0.46
10	C-20	20	50	8	0.4	0.16	0.23
TB					0.53	0.324	0.393

Từ dữ liệu bảng 3.5, ta có biểu đồ như hình 7. Biểu đồ thể hiện điểm đánh giá độ đo F-score của các tập dữ liệu.

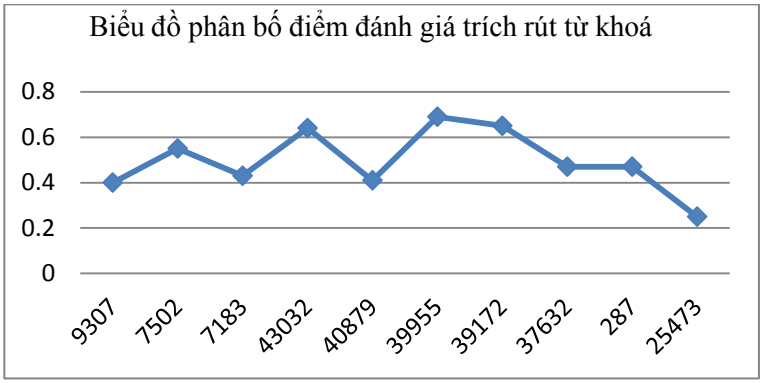


Hình 3.1: Biểu đồ phân bố điểm đánh giá trích rút từ khoá từ tập dữ liệu mẫu

STT	Tên file	Từ khoá của chuyên gia	Từ khoá của TextRank	Từ khoá chung	Recall	Precision	F-score
1	9307	10	20	6	0.6	0.3	0.4
2	7502	9	20	8	0.89	0.4	0.55
3	7183	8	20	6	0.75	0.3	0.43
4	43032	11	20	10	0.9	0.5	0.64
5	40879	14	20	7	0.5	0.35	0.41
6	39955	12	20	11	0.92	0.55	0.69
7	39172	14	20	11	0.79	0.55	0.65
8	37632	10	20	7	0.7	0.35	0.47
9	287	10	20	7	0.7	0.35	0.47
10	25473	12	20	4	0.33	0.2	0.25
TB					0.71	0.39	0.5

kết quả đánh giá với chủ đề " Khoa học"

Từ dữ liệu bảng 3.6, ta có biểu đồ như hình 8. Biểu đồ thể hiện điểm đánh giá độ đo F- score của các tập dữ liệu



Hình 3.2: Biểu đồ phân bố điểm đánh giá trích rút từ khoá từ tập dữ liệu mẫu

Kết quả đánh giá với dữ liệu chủ đề “ phim và phim hoạt hình”

Bảng 3.7: So sánh kết quả từ khoá của TextRank và từ khoá trên trang web về phim và phim hoạt hình

STT	Tên file	Từ khoá trên web	Từ khoá trích rút từ TextRank	Từ khoá chung	Recall	Precision	F-score
1	A1	5	6	2	0.4	0.33	0.36
2	A2	5	6	1	0.2	0.17	0.18
3	A3	5	12	3	0.6	0.25	0.35
4	A4	5	4	2	0.4	0.5	0.45
5	A5	5	2	1	0.2	0.5	0.29
6	A6	5	6	2	0.4	0.33	0.36
7	A7	5	6	2	0.4	0.33	0.36
8	A8	5	4	1	0.2	0.25	0.22
9	A9	5	13	3	0.6	0.23	0.33
10	A10	5	5	2	0.4	0.4	0.4
11	A11	5	4	1	0.4	0.33	0.36
12	A12	5	5	2	0.4	0.4	0.4
13	A13	5	5	2	0.4	0.4	0.4
14	A14	5	5	1	0.2	0.2	0.2
15	A15	5	9	3	0.6	0.33	0.43
16	A16	5	9	3	0.6	0.33	0.43
17	A17	5	6	2	0.4	0.33	0.36
18	A18	5	11	1	0.2	0.1	0.13
19	A19	5	6	2	0.4	0.33	0.36
20	A20	5	4	1	0.2	0.25	0.22
21	A21	5	3	1	0.2	0.33	0.25
22	A22	5	4	1	0.2	0.25	0.22
23	A23	5	4	1	0.2	0.25	0.22
24	A24	5	9	3	0.6	0.33	0.43
25	A25	5	8	3	0.6	0.38	0.47
26	A26	5	7	2	0.4	0.29	0.34
27	A27	5	6	2	0.4	0.33	0.36

28	A28	5	6	2	0.4	0.33	0.36
29	A29	5	7	2	0.4	0.29	0.34
30	A30	5	6	2	0.4	0.33	0.36
31	A31	5	1	1	0.2	1	0.33
32	A32	5	2	2	0.4	1	0.57
33	A33	5	5	1	0.2	0.2	0.2
34	A34	5	5	1	0.2	0.2	0.2
35	A35	5	5	1	0.2	0.2	0.2
36	A36	5	6	1	0.2	0.17	0.18
37	A37	5	11	2	0.2	0.18	0.19
38	A38	5	4	1	0.2	0.25	0.22
39	A39	5	4	1	0.2	0.25	0.22
40	A40	5	9	2	0.4	0.22	0.28
41	A41	5	6	2	0.4	0.33	0.36
42	A42	5	5	2	0.4	0.4	0.4
43	A43	5	4	1	0.2	0.25	0.22
44	A44	5	1	1	0.2	0.2	0.2
45	A45	5	4	1	0.2	0.25	0.22
46	A46	5	2	1	0.2	0.5	0.29
47	A47	5	3	1	0.2	0.33	0.25
48	A48	5	2	1	0.2	0.5	0.29
49	A49	5	6	2	0.4	0.33	0.36
50	A50	5	5	2	0.4	0.4	0.4
TB					0.33	0.33	0.31

Từ dữ liệu bảng 3.7, ta có:

Nhận xét:

- Độ đo F1 cho kết quả khá tốt, các điểm đánh giá trên toàn tập dữ liệu đều trên 0.31. Tập dữ liệu cho kết quả tốt nhất là tâp file 39955 với điểm số đạt 0.92. Tuy nhiên có vài tập dữ liệu cho kết quả thấp so với các tập còn lại như C-20, C-17, C-4, C-6, 25473.

- Các biểu đồ thể hiện sự khác biệt rõ giữa điểm đánh giá của các tập dữ

liệu. Đó cũng thể hiện mức độ chính xác, chất lượng của phương pháp TextRank đối với các tập dữ liệu với các đặc điểm khác nhau.

- Từ bảng phân tích dữ liệu thực nghiệm →tốc độ trích rút từ khoá phụ thuộc vào độ dài văn bản.

- Thời gian trích rút cho một văn bản chỉ khoảng vài giây tuỳ thuộc độ dài ngắn của văn bản. Đây là con số ấn tượng nói lên tiềm năng áp dụng phương pháp TextRank vào thực tế

- Vì có một số văn bản có điểm đánh giá thấp. Vì vậy tác giả đã loại bỏ đi những văn bản khó trích rút hoặc trích rút có điểm đánh giá thấp, kết quả là điểm đánh giá trên toàn tập dữ liệu tăng lên đáng kể → Phương pháp TextRank sẽ cho kết quả tốt nhất ở những văn bản có độ nhiễu ít, độ dài văn bản ngắn, chứa ít các từ nối, từ quan hệ.

3.5. Đánh giá kết quả thực nghiệm

Đánh giá chính xác kết quả của một danh sách các từ khoá là một việc làm rất khó khăn vì thực ra phương pháp mà tác giả ứng dụng trong luận văn là hoàn toàn không giám sát. Từ khoá được sinh ra tự động, hơn nữa cách đánh giá từ khoá của các chuyên gia cũng có thể rất khác nhau cho cùng một tài liệu văn bản. Chủ yếu việc đánh giá vẫn dựa vào ý kiến đánh giá của các chuyên gia con người. Những từ khoá phải mang ý nghĩa cao, nói lên nội dung của tài liệu văn bản.

Với lượng từ khoá được trích rút khá nhiều bởi phương pháp TextRank tất nhiên có thể khống chế lượng từ khoá sinh ra khi dùng thuật toán, nhưng từ khoá vẫn bị lặp lại nhiều, một số từ khoá không có ý nghĩa quan trọng, không nêu được đặc trưng của văn bản đó cũng là nhược điểm của phương pháp. Tuy nhiên thì ưu điểm của phương pháp là thời gian trích rút từ khoá nhanh, không cần những kiến thức chuyên sâu về ngôn ngữ học vì thế bài toán này có tính ứng dụng thực tế cao

KẾT LUẬN

Những vấn đề đã giải quyết được trong luận văn

- Luận văn đã nghiên cứu các phương pháp trích rút từ khoá từ nội dung văn bản trên các trang web và ứng dụng. Đặc biệt là đi sâu nghiên cứu phương pháp mới là trích rút từ khoá bằng phương pháp TextRank.

- Đồng thời, luận văn cũng đã đề xuất sử dụng một công cụ được xây dựng sẵn để trích rút từ khoá của văn bản tiếng Anh. Thực nghiệm trên dữ liệu tiếng anh của bộ dữ liệu đã được xây dựng bởi các chuyên gia.

- Tác giả cũng đã sưu tầm dữ liệu trên Internet cho tập dữ liệu với chủ đề về phim ảnh và so sánh kết quả trích rút của phương pháp TextRank với kết quả từ khoá trên trang web được xây dựng bởi các chuyên gia.

- Khảo sát phương pháp trích rút từ khoá sử dụng Textrank cho kết quả khả quan có thể ứng dụng trong các bài toán thực tế về tìm kiếm thông tin, hay tóm tắt văn bản. Và trên đây tôi cũng đã trình bày những ưu điểm, nhược điểm còn tồn tại của phương pháp.

Hướng phát triển tiếp theo

Mặc dù kết quả thu được của luận văn là đáng khích lệ và khá tốt nhưng do thời gian có hạn và việc ước lượng các trọng số cho phương pháp có thể chưa được tối ưu. Trong thời gian tới, tôi sẽ tiến hành thu thập thêm các dữ liệu và hoàn thiện những gì còn thiếu sót của phương pháp mà tôi đề xuất.

Cũng trên cơ sở đã đạt được của luận văn, tôi dự định sẽ cải tiến chương trình để có thể thực hiện được trên tập dữ liệu các văn bản Tiếng Việt.

TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Nguyễn Hoàng Tú Anh, Nguyễn Trần Kim Chi, Nguyễn Hồng Phi(2008), "Mô hình biểu diễn văn bản thành đồ thị", *tạp chí phát triển KH&CN tập 12 số 07 năm 2009*

[2] Nguyễn Quang Châu, Lê Trọng Ngọc, Tôn long Phước, Nguyễn Văn Tân(2011), "Một hướng tiếp cận xây dựng Ontology Tiếng Việt", *tạp chí Đại học Công Nghiệp T25 năm 2011*

[3] Trương Quốc Định(2015), "Phân loại văn bản dựa trên rút trích tự động tóm tắt của văn bản", *kỷ yếu Hội nghị Quốc gia về nghiên cứu cơ bản và ứng dụng công nghệ thông tin năm 2015*.

[4] Trương Quốc Định, Nguyễn Quang Dũng(2012), "Một giải pháp tóm tắt văn bản Tiếng Việt tự động", *hội thảo Quốc gia lần thứ XV về một số vấn đề chọn lọc của công nghệ thông tin và truyền thông năm 2012*.

[5] Chu Anh Minh(2009), *Bài toán trích xuất từ khoá cho trang web áp dụng phương pháp phân tích thẻ HTML và đồ thị web,* Luận văn thạc sĩ, Trường đại học Công nghệ, Đại học Quốc gia Hà Nội.

[6] Nguyễn Văn Nghiệp(2015), *Tóm tắt văn bản Tiếng Việt sử dụng phương pháp TextRank,* Luận văn thạc sĩ, Trường đại học Công nghệ, Đại học Quốc gia Hà Nội.

[7] Lê Hoàng Thanh(2012). *Text mining – Kỹ thuật trích xuất thông tin từ văn bản*

[8] Trần Ngọc Phúc(2012), *Phân loại nội dung tài liệu web,* Luận văn thạc sĩ, Trường đại học Lạc Hồng, Đồng Nai.

[9] Nguyễn Trọng Phúc, Lê Thanh Hương(2008), "Tóm tắt văn bản Tiếng Việt sử dụng cấu trúc diễn ngôn"

[10] Website: http://vietseo.net

Tiếng Anh

[11] J. Han and M. Kamber, Data mining concepts and techniques. San Francisco: Morgan Kawfmann Publishers, 2006

[12] SuNamKim,OlenaMedelyan,Min-Yen Kan&TimothyBaldwin.Automatic

keyphrase extraction from scientific articles;2010

[13] Rada Mihalcea and Paul Tarau. TextRank: Bringing Order into Texts; 2004.

[14] Kazi Saidul Hasan and Vincent Ng. Automatic Keyphrase Extraction: A Survey of the State of the Art; 2014

[15] Simone Teufel, Marc Moens. Sentence extraction as a classification task; 2002

[16] Brian Loff. Survey of Keyword Extraction Techniques; 2012.

[17] Gonenc Ercan, Ilyas Cicekli. Using Lexical Chains for Keyword Extraction. Inf; 2007

Process. Manage., Vol. 43, No. 6. (November 2007), pp. 1705-1714.

[18] H.Edmundson(1969). New methods in automatic abstracting, Journal of ACM; 1969.

[19] HPLuhn(1958). The automatic creation of literature abstracts. IBM journal of research development.

[20] J. Kleinberg. Authoritative sources in a hyperlinked environment. J. of the ACM , 1999, to appear. Also appears as IBM Research Report RJ 10076 91892 May 1997.

[21] P. D. Turney, Learning Algorithms for Keyphrase Extraction, Information Retrieval; 1999.

[22] Qiang Yang, Advertising keyword suggestion based on concept hierarchy presented by Qiang Yang, HongKong Univ of Science and Technology.

[23] S. Brin and L. Page. The anatomy of a large-scale hypertextual Web search engine.Proc. 7th WWW Conf; 1998.

[24] Y. MATSUO,M. Ishizuka.Keyword Extraction from a Single Document using Word Co-occurrence Statistical Information.International Journal on Artificial Intelligence Tools; 2003.

[25] Yasin Uzun. Keyword Extraction Using Naive Bayes. Bilkent University, Department of Computer Science, Turkey; 2015.

[26] Zhu Mengxiao ,Cai Zhi ,Cai Qingsheng.Automatic Keywords Extraction Of Chinese Document Using Small World Structure. Department of Computer Science, University of Science and Technology of China; 2014.

[27] Soumen Chakrabarti, Data mining for hypertext: A tutorial survey. Volume 1 ACM – 2000

[28] Yi-fang Brook Wu, Quanzhi Li, Razvan Stefan Bot, Xin Chen, Domanin – specific keyphrase extraction, Proceedings of the 14th ACM international conference on information and knowledge management, October 31- November 05, 2005, Bremen, Germany.

[29] Vibhanshu Abhishek, Kartik Hosanagar, Keyword generation for search engine advertising using semantic similarity between terms, Proceeding of the ninth international conference on Electronic commerce, August 19-22, 2007, Mineapolis, MN, USA.

[30] M. Sahami and T. Heilman. A web-based kernel function for matching short text snippets. In International Conference on Machine Learning, 2005.

[31] Python http://pypi.python.org/pypi/summa/0.07

[32] Tf,IDF http://en.wikipedia.org/wiki/Tf-idf

[33] Website: http://searchengineguide.com

Công cụ và dữ liệu sử dụng

[34] Website : http://pypi.python.org/pypi/summa/0.07

[35] Website: http://www.imdb.com

[36] Website: http://google.com