**MỤC LỤC**

[MỞ ĐẦU 1](#_Toc388271105)

[1. Ôn tập đại số tuyến tính 2](#_Toc388271108)

[2. Sự phân tách ma trận 4](#_Toc388271109)

[3. Ma trận thuật ngữ-văn bản và phân tách giá trị đơn (SVD) 5](#_Toc388271110)

[4. Xấp xỉ hạng thấp 7](#_Toc388271111)

[5. Chỉ mục ngữ nghĩa tiềm ẩn cho truy xuất thông tin dựa trên không gian véc tơ 9](#_Toc388271112)

[KẾT LUẬN 12](#_Toc388271113)

[TÀI LIỆU THAM KHẢO 13](#_Toc388271114)

# MỞ ĐẦU

Cùng với sự phát triển nhanh chóng của công nghệ thông tin thì khối lượng dữ liệu đa phương tiện (Multimedia) được thu thập và lưu trữ dưới dạng số ngày càng nhiều dẫn tới việc tìm kiếm dữ liệu đa phương tiện trở nên khó khăn. Vì vậy cần có các hệ thống tìm kiếm thông tin (Information Retrieval) hỗ trợ người dùng tìm kiếm một cách chính xác và nhanh chóng các thông tin mà họ cần trên kho tư liệu khổng lồ này.

Hiện nay có một số hệ thống tìm kiếm như *GoogleDesktop, DTSearch, Lucene,* tuy nhiên các hệ thống này sử dụng các kỹ thuật tìm kiếm đơn giản nên hiệu quả còn chưa cao. Vì vậy mục tiêu của báo cáo này nhằm tìm hiểu một kỹ thuật nâng cao tìm kiếm thông tin, cụ thể ở đây là tìm kiếm nội dung văn bản trong không gian vecto bằng kỹ thuật chỉ mục ngữ nghĩa tiềm ẩn (LSI).

1. **Ôn tập đại số tuyến tính**

Cho ma trận C cỡ m x n với các phần tử là các số thực; Cho một ma trận term-document, tất cả các giá trị đầu vào trong thực tế không âm. Hạng (rank) của ma trận là số hàng (hoặc cột) độc lập tuyến tính trong nó. Do đó, rank(C) ≤ {m, n}. Một ma trận vuông r x r  trong đó tất cả các phần tử không nằm trên đường chéo đều bằng 0 được gọi là ma trận chéo; Hạng của nó được tính bằng số các thành phần khác 0 trên đường chéo. Nếu tất cả r phần tử trên đường chéo của ma trận chéo bằng 1, nó được gọi là ma trận đồng nhất (hay ma trận đơn vị) cỡ r và được biểu diễn bởi Ir.

Cho ma trận vuông C cỡ m x m và một véc tơ n-chiều khác 0,các giá trị λ thỏa mãn

C = λ (1)

được gọi là giá trị đặc trưng của ma trận C. Véc tơ n-chiều thỏa mãn phương trình (1) với một giá trị đặc trưng λ là một véc tơ đặc trưng bên phải tương ứng. Véc tơ đặc trưng tương ứng với giá trị đặc trưng lớn nhất được gọi là véc tơ đặc trưng chính. Một cách biểu diễn tương tự, các véc tơ đặc trưng bên trái của C là véc tơ m chiều như sau:

T C = λT (2)

Số lượng các giá trị đặc trưng khác 0 của C là hạng lớn nhất của C.

Giá trị đặc trưng của một ma trận được tìm ra bằng cách giải phương trình đặc trưng thu được bằng cách viết lại phương trình (1) dưới dạng (C – λIm) = 0. Các giá trị đặc trưng của C là các đáp án của |C – λIm| = 0, trong đó |S| biểu thị yếu tố quyết định của một ma trận vuông S. Phương trình |C – λIm| = 0 là một phương trình đa thức bậc m trong λ và có thể có nhiều nhất m nghiệm, chính là giá trị đặc trưng của C. Những giá trị đặc trưng này có thể nói chung là phức tạp, ngay cả khi tất cả phần tử của C là số thực.

Bây giờ chúng ta xem xét một số đặc tính nữa của các giá trị đặc trưng và véc tơ đặc trưng, để thiết lập tưởng trung tâm của sự phân tách giá trị đơn trong mục 2 sau đây. Đầu tiên, chúng ta nhìn vào mối quan hệ giữa ma trận véc tơ nhân và các giá trị đặc trưng.

Ví dụ 1: Xét ma trận S =

Xóa ma trận có rank 3, và có 3 giá trị đặc trưng khác 0 là λ1 = 30, λ2 = 20, λ3 = 1, với 3 véc tơ đặc trưng tương ứng:

1 = , 2 = , 3 =

Với mỗi véc tơ đặc trưng, nhân bởi luật S bằng cách nếu chúng ta nhân véc tơ đặc trưng bằng bội số của ma trận đơn vị; bọi số là khác nhau cho mỗi véc tơ đặc trưng. Bây giờ, xét một véc tơ tùy ý, chẳng hạn như = . Chúng ta luôn có thể biểu diễn như một sự kết hợp tuyến tính của của 3 véc tơ đặc trưng của S; ở ví dụ hiện tại ta có:

= = 21 + 42 + 63.

Giả sử ta nhân với S:

S = S(21 + 42 + 63)

= 21 + 42 + 63

= 2λ11 + 4λ22 + 6λ33

= 601 + 802 + 63. (3)

Ví dụ 1 cho thấy rằng mặc dù là một véc tơ tùy ý, ảnh hưởng của phép nhân với S được xác định bởi các giá trị đặc trưng và véc tơ đặc trưng của S. Hơn nữa, điều đó nhìn thấy hiển nhiên từ phương trình (3) mà tích S là tương đối không bị ảnh hưởng bởi các điều kiện phát sinh từ các giá trị đặc trưng nhỏ của S.

Ví dụ 2: Xét ma trận đối xứng

S = (4)

Từ đặc trưng của phương trình |S - λI| = 0, chúng ta có phương trình bậc hai (2 - λ)2 – 1 = 0, đáp án cho giá trị đặc trưng 3 và 1. Các véc tơ đặc trưng tương ứng và là vuông góc với nhau.

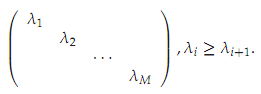
1. **Sự phân tách ma trận**

Chúng ta bắt đầu bằng cách đưa ra hai định lý về sự phân tách của ma trận vuông đến kết quả (tích) của 3 ma trận của một hình thức đặc biệt.

**Định lý 1 (**Định lý về sự chéo hóa ma trận**)**: *Cho S là một ma trận giá trị thực vuông M x M với M là véc tơ đặc trưng độc lập tuyến tính. Tồn tại một sự phân tách đặc trưng*

*S = U ˄ U-1,*

*nơi các cột của U là các véc tơ đặc trưng của S và ˄ là một ma trận đường chéo mà các phần tử trên đường chéo là các giá trị đặc trưng của S theo thứ tự giảm*

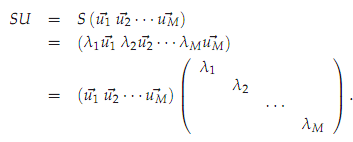
*(6)*

*Nếu các giá trị đặc trưng là hiển nhiên thì phép phân tách này là duy nhất.*

Để hiểu định lý 1 làm việc như thế nào, ta lưu ý rằng U có các véc tơ đặc trưng của S như các cột

U = (1 2 … M). (7)

Sau đó ta có:



Do đó, chúng ta có SU = U*˄*, hay S = *U ˄ U-1*.

**Định lý 2 (**Định lý về sự chéo hóa đối xứng**)**: *Cho S là một ma trận giá trị thực vuông M x M đối xứng với M là véc tơ đặc trưng độc lập tuyến tính. Tồn tại một sự phân tách đối xứng*

*(8)*

*Nơi mà các cột của Q là vuông góc với nhau và đã chuẩn hóa (đơn vị chiều dài, số thực) các véc tơ đặc trưng của S, và ᴧ là ma trận chéo mà các phần tử là các giá trị đặc trưng của S. Hơn nữa, tất cả các phần tử của Q là số thực và ta có Q-1 = QT.*

Ta sẽ xây dựng trên sự phân tách đối xứng chéo này để xây dựng các xấp xỉ hạng thấp đến các ma trận term-document.

1. **Ma trận thuật ngữ-văn bản và phân tách giá trị đơn (SVD)**

Các phép phân tách chúng ta đã nghiên cứu cho đến nay mới chỉ áp dụng cho ma trận vuông. Tuy nhiên, ma trận chúng ta quan tâm là ma trận term-document C cỡ M x N với M ≠ N (trừ một số trường hợp trùng hiếm gặp); Hơn nữa, C rất ít có khả năng đối xứng. Để giải quyết điều này, đầu tiên ta mô tả một phần mở rộng của sự phân tách đối xứng chéo được gọi là sự phân tách giá trị suy biến. Cho C, U là ma trận cỡ M x M có các cột là các véc tơ đặc trưng vuông góc của CCT, và V là ma trận cỡ N x N có các cột là các véc tơ đặc trưng vuông góc của CTC. Ký hiệu CT là ma trận chuyển vị của C.

**Định lý 3** : *Cho r là rank của ma trận C cỡ M x N. Khi đó một sự phân tách giá trị đơn (SVD) của C với công thức :*

*(9)*

*Trong đó,*

1. *Các giá trị đặc trưng λ1, …, λr của CCT là giống các giá trị đặc trưng của CTC ;*
2. *Với mỗi 1 ≤ i ≤ r, cho i = , với λi ≥ λi+1. Sau đó ma trận M x N được thành lập bởi = i với mỗi 1 ≤ i ≤ r, bằng 0 nếu ngược lại.*

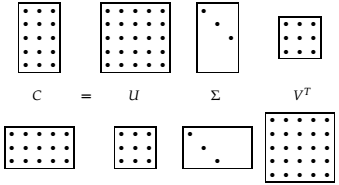
Các giá trị *i*  được tham chiếu đến như các giá trị suy biến của C. Nó là bài học để kiểm tra các mối quan hệ cảu Định lý 3 với Định lý 2; Ta làm điều này hơn là việc chứng minh Định lý 3.

Bằng cách nhân phương trình (9) với hoán vị của nó, ta có:

(10)

Lưu ý với phương trình (10), vế bên trái là ma trận vuông đối xứng giá trị thực, và phía bên phải thể hiện sự phân tách đối xứng chéo của nó như trong Định lý 2. Phía bên trái của *CCT* biểu diễn cái gì? Nó là một ma trận vuông với một dòng và một cột tương ứng với mỗi term M. Phần tử (i,j) trong ma trận là một biểu diễn giao của term thứ i và thứ j, dựa trên sự xuất hiện đồng thời của chúng trong các tài liệu. Ý nghĩa toán học chính xác phụ thuộc vào cách thức mà C được xây dựng dựa trên trọng số của thuật ngữ.

Khi viết ra giá trị các số SVD, nó là quy ước để biểu diễn hư một ma trận r x r với giá trị suy biến trên các đường chéo, vì tất cả các phần tử nằm ngoài đều bằng 0. Do đó, thông thường bỏ qua M – r cột ngoài cùng bên phải của U tương ứng với các hàng bị bỏ qua của Tương N – r cột ngoài cùng bên phải của V bị bỏ qua bởi vì nó tương ứng với các hang tring VT mà sẽ nhân bởi N – r cột giá trị 0 trong Cách viết này của SVD đôi khi được gọi là SVD giảm hay SVD cắt ngắn.



**Hình 1**. Minh họa sự phân tách giá trị đơn. Trong hình là minh họa của phương trình (9) với hai trường hợp. Trong nửa trên của hình ta có ma trận C với M > N, nửa hình sau là trường hợp M < N.

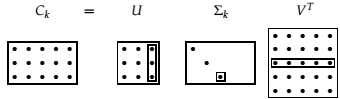
1. **Xấp xỉ hạng thấp**

Cho ma trận *C* cỡ M x N và một số nguyên dương k, ta muốn tìm một ma trận *Ck* cỡ M x N với rank lớn nhất là k, để giảm chuẩn Frobenius của ma trận khác X = C – Ck, được định nghĩa như sau:

(11)

Vì vậy, chuẩn Frobenius của ma trận X đo sự chênh lệch giữa Ck và C; mục tiêu của ta là tìm ra một ma trận Ck để sự chênh lệch này là nhỏ nhất, trong khi ràng buộc Ck có rank lớn nhất là k. Nếu r là rank của C, rõ rang Cr = C và chuẩn Frobenius của sự chênh lệch bằng 0 trong trường hợp này. Khi k nhỏ hơn rất nhiều so với r, ta xem Ck là một xấp xỉ bậc thấp.

Sự phân tách giá trị đơn có thể được sử dụng để giải quyết vấn đề xấp xỉ ma trận hạng thấp. Sau đó ta suy ra được từ nó một ứng dụng để xấp xỉ các ma trận thuật ngữ-tài liệu. Ta đưa ra 3 bước cụ thể sau đây để giải quyết:



**Hình 2**: Minh họa xấp xỉ hạng thấp sử dụng phân tách giá trị đơn.

1. Cho C, chỉ ra phép phân tách giá trị đơn của nó trong mẫu như công thức (9), do đó C = UΣVT.
2. Suy ra từ Σ ma trận Σk được tại ra bằng cách thay thế các số 0 bởi r – k giá trị suy biến nhỏ nhất trên đường chéo của Σ.
3. Tính toán và đưa ra Ck = UΣkVT bằng phép xấp xỉ rank k của C.

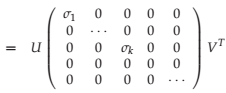
**Định lý 4**:

(12)

Gọi các giá trị suy biến theo thứ tự giảm σ1 ≥ σ2 ≥ …, ta biết từ Định lý 4 rằng Ck là xấp xỉ hạng k tốt nhất của C, phát sinh một lỗi (đo bởi chuẩn Frobenius của C - Ck) bằng σk+1. Do đó, k là lớn hơn, lỗi này nhỏ hơn (và trong trường hợp đặc biệt, với mỗi k = r, lỗi là 0 do Σr = Σ; với điều kiện r < M, N, sau đó σr+1 = 0 và do đó Cr = C).

Để có cái nhìn chi tiết hơn về lí do tại sao quá trình cắt xén các giá trị suy biến r – k nhỏ nhất trong Σ giúp tạo ra một phép xấp xỉ rank k của lỗi thấp, ta xét:

Ck = UΣkVT (13)





Với và là các cột tương ứng thứ i của U và V. Do đó, T là một ma trận rank-1, chính vì vậy ta có thể biểu diễn Ck là tổng của k ma trận rank-1 với mỗi trọng số là một giá trị suy biến. Nếu i tăng lên, sự góp phần của ma trận rank-1 T được đánh trọng số bởi một chuỗi các giá trị suy biến hẹp σi.

1. **Chỉ mục ngữ nghĩa tiềm ẩn cho truy xuất thông tin dựa trên không gian véc tơ**

Trong mô hình không gian véc tơ, mỗi tài liệu được trình diễn bởi một véc tơ trọng số thuật ngữ N-chiều, mỗi phần tử của véc tơ là một trong N thật ngữ của tài liệu đó. Nếu một bộ sưu tập có M tài liệu, sau đó bộ sưu tập được biểu diễn như một ma trận A cỡ M x N. Trong quá trình truy xuất, truy vấn cũng được biểu diễn trong một véc tơ trọng số thuật ngữ N-chiều. Sự tương đồng giữa truy vấn và mỗi tài liệu lưu trữ được tính toán như mỗi tích số hoặc hệ số côsin giữa véc tơ truy vấn và véc tơ tài liệu.

Phương pháp đơn giản trên đây có hai yếu điểm chính. Thứ nhất, một bộ sưu tập tài liệu lớn (ví như một thư viện) chứa hàng triệu tài liệu với hàng nghìn thuật ngữ (tức cả M và N đều rất lớn). Vì thế yêu cầu một con số rất lớn để lưu trữ. Ví dụ, nếu một thư viện có 1 triệu tài liệu với 10 nghìn thuật ngữ, chúng ta cần 10Gigabyte lưu trữ nếu mỗi phần tử được lưu trữ 1byte. Một vài năm trở lại đây thì điều đó là một con số khổng lồ. Thứ hai, ít nhất M phép nhân của các vecto N-chiều được yêu cầu trong suốt quá trình truy xuất nếu tích số để đo tương tự được sử dụng, và hơn thế nó được yêu cầu nếu hệ số cosin để đo tương tự được sử dụng. Khi M và N là lớn, thì yêu cầu thời gian để tính toán hoàn thiện là chấp nhận được ch truy xuất trực tuyến.

Chỉ mục ngữ nghĩa tiềm ẩn (LSI) được phát triển để giải quyết một phần các vấn đề trên. (Ta nói một phần bởi vì các kỹ thuật khác ví dụ như là phân cụm và cấu trúc dữ liệu đa khoảng cách đã được thảo luận sau đó nên được biên dịch với LSI cho việc tìm kiếm hiệu quả hơn). Ý tưởng chính của LSI là cố gắng nhóm những thuật ngữ tương tự với nhau vào một khái niệm chung hoặc chủ đề và các tài liệu được biểu diễn bởi các khái niệm này. Số lượng của các khái niệm là nhỏ hơn nhiều so với số lượng của các thuật ngữ, Yêu cầu lưu trữ và tính toán ít hơn. Thêm vào đó, bởi vì LSI có khả năng nhóm tự động đồng xảy ra và các thuật ngữ tương tự để tạo ra một kho, truy xuất hiệu quả cngx được báo cáo để cải tiến.

LSI được dựa trên khái niệm của phân tách giá trị đơn (SVD) (9). Trong phạm vi của truy xuất tài liệu văn bản, hạng r của C bằng số lượng các khái niệm. U có thể được dùng như ma trận tương tự tài liệu đến khái niệm, trong khi V là ma trận tương tự thuật ngữ đến khái niệm. Ví dụ u2,3 = 0.6 nghĩa là khái niệm 3 có trọng số 0.6 trong tài liệu 2, và v1,2 = 0.4 nghĩa là đọ tương tự giữa thuật ngữ 1 và khái niệm là 0.4.

Dựa vào SVD, ta lưu trữ các ma trận U, Σ, V thay vì lưu trữ C, giảm yêu cầu lưu trữ một cách đáng kể. Ví dụ, giả sử M = 1 000 000, N = 10 000 và r = 500, tổng số lượng không gian lưu trữ yêu cầu bằng 1 000 000 x 500 + 500 x 500 + 10 000 x 500 = 505,25 MB, ít hơn đáng kể so với yêu cầu lưu trữ 10 GB cho C.

Trong suốt quá trình lưu trữ, tài liệu truy vấn tương tự được tính toán như sau. Véc tơ truy vấn q trong không gian thuật ngữ được dịch đến qc trong không gian khái niệm với phép nhân VT như sau:

qc = VT x q

Sự tương tự giữa truy vấn với mỗi tài liệu được tính bằng tích số hoặc hệ số cosin giữa qc và mỗi dùng của U. Vì sử dụng LSI, ta thao tác với các vecto r-chiều thay vì các ecto N-chiều suốt quá trình tính toán tương tự. Do vậy r nhỏ hơn rất nhiều lần so với N, việc tính toán sử dụng LSI nhanh hơn nhiều lần với việc sử dụng các phương thức đơn giản. Tìm kiếm và truy xuất hiệu quả được cải tiến hơn bằng việc phân cụm các hàng của U dựa trên sự tương tự của chúng.

**KẾT LUẬN**

Với khối lượng thông tin khổng lồ như hiện nay thì lựa chọn các kỹ thuật tìm kiếm thông tin sao cho vừa nhanh chóng, vừa chính xác là một điều hết sức cần thiết. Báo cáo đã trình bày một kỹ thuật LSI đơn giản, dễ hiểu trong số các kỹ thuật tìm kiếm văn bản đã được nghiên cứu và phát triển. Tuy kỹ thuật này đã đem lại nhưng kết quả đáng mong đợi nhưng vẫn cần phải có những kỹ thuật tốt hơn, hiệu quả hơn nhằn đáp ứng nhu cầu truy vấn ngày càng cao của người sử dụng.

**TÀI LIỆU THAM KHẢO**

[1] Christopher D. Manning, Prabhakar Raghavan & Hinrich Schütze “***An Introduction to Information Retrieval***”, Cambridge University Press, Cambridge, England, 2009.

[2] Guojun Lu “***Multimedia Database Management Systems***”, Artech House, Boston, London, 1999.