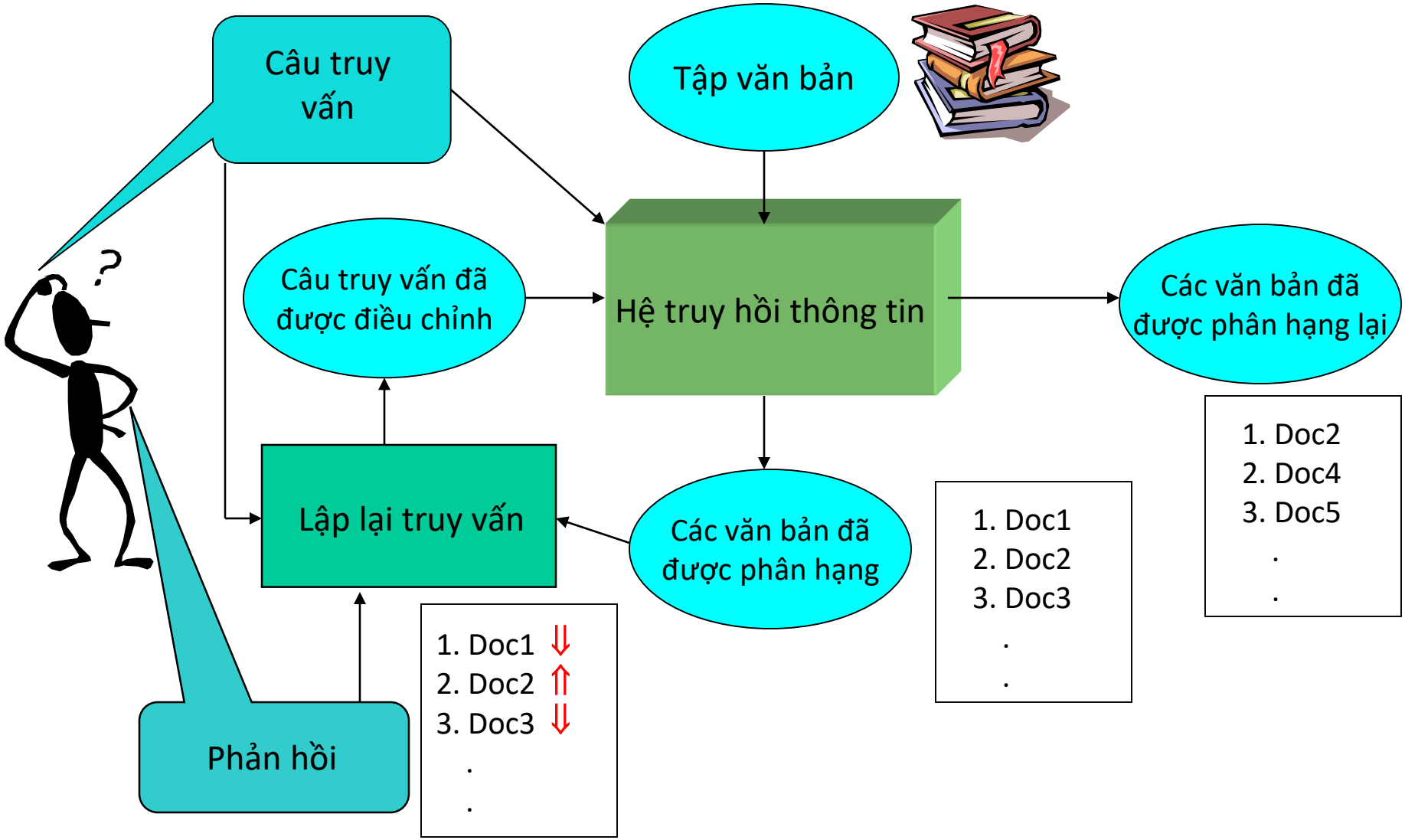
Phản hồi phù hợp và mở rộng câu truy vấn

Nguyễn Mạnh Hiển
hiennm@tlu.edu.vn

Phản hồi phù hợp (relevance feedback)

- Sau khi trả về kết quả, cho phép người dùng phản hồi về sự phù hợp của các văn bản.
- Dùng thông tin phản hồi đó để lập lại câu truy vấn.
- Tạo ra kết quả mới dựa trên câu truy vấn mới.
- Cho phép một tiến trình truy hồi thông tin có tính tương tác hơn (nhiều lần qua lại) giữa người dùng và hệ thống.

Kiến trúc phản hồi phù hợp



Lập lại câu truy vấn

Điều chỉnh câu truy vấn dựa trên thông tin phản hồi:

- **Mở rộng câu truy vấn**: Thêm vào câu truy vấn các từ mới lấy từ các văn bản phù hợp.
- **Định lại trọng số từ** (của các từ trong câu truy vấn): Tăng trọng số của các từ xuất hiện trong các văn bản phù hợp, và giảm trọng số của các từ xuất hiện trong các văn bản không phù hợp.

Lập lại câu truy vấn trong mô hình không gian vector

- Thay đổi vector truy vấn dùng đại số vector:
  - Cộng vector biểu diễn các văn bản phù hợp vào vector biểu diễn câu truy vấn.
  - Trừ vector biểu diễn các văn bản không phù hợp khỏi vector biểu diễn câu truy vấn.
- Làm như vậy vừa bổ sung các từ có trọng số dương (cộng vector) và âm (trừ vector) vào câu truy vấn, vừa định lại trọng số các từ ban đầu.

Câu truy vấn tối ưu

- Giả sử tập các văn bản phù hợp C_r đã biết.
- Câu truy vấn tối ưu như sau:

$$\vec{q}_{opt} = \frac{1}{|C_r|} \sum_{\forall \vec{d}_j \in C_r} \vec{d}_j - \frac{1}{N - |C_r|} \sum_{\forall \vec{d}_j \notin C_r} \vec{d}_j$$

trong đó N là tổng số văn bản.

Phương pháp Rocchio

Trên thực tế, vì không biết tất cả các văn bản phù hợp, ta chỉ dùng các ***tập con*** văn bản phù hợp D_r và không phù hợp D_n đã biết, cùng với câu truy vấn q ban đầu, để tạo ra câu truy vấn mới:

$$\vec{q}_m = \alpha\vec{q} + \frac{\beta}{|D_r|}\sum_{\forall\vec{d}_j\in D_r}\vec{d}_j - \frac{\gamma}{|D_n|}\sum_{\forall\vec{d}_j\in D_n}\vec{d}_j$$

α: Trọng số điều chỉnh được của câu truy vấn ban đầu.

β: Trọng số điều chỉnh được của các văn bản phù hợp.

γ: Trọng số điều chỉnh được của các văn bản không phù hợp.

(Thường chỉ cần cho các hệ số này bằng 1 sẽ đem lại cải thiện trong thực tế).

Đánh giá phản hồi phù hợp

- Do cách xây dựng, câu truy vấn mới sẽ phân hạng các văn bản phù hợp đã biết cao hơn, và phân hạng các văn bản không phù hợp đã biết thấp hơn.
- Việc đánh giá ***không nên*** dựa trên những văn bản như vậy vì sự phù hợp của chúng đã biết trước rồi.
- Trong lĩnh vực học máy, lỗi này được gọi là "kiểm thử trên dữ liệu huấn luyện".
- Việc đánh giá nên tập trung vào những văn bản ***khác***.

Đánh giá phản hồi phù hợp công bằng

- Xóa khỏi tập văn bản những văn bản đã có thông tin phản hồi kèm theo.
- Tính độ chính xác (precision) và độ thu hồi (recall) trên tập văn bản còn lại.
- So với tập văn bản đầy đủ, độ chính xác và độ thu hồi tính được có thể bị giảm đi vì một số văn bản phù hợp đã bị xóa.
- Tuy nhiên, hiệu suất ***tương đối*** trên tập văn bản còn lại vẫn cho biết (một cách công bằng) phản hồi phù hợp hiệu quả đến đâu.

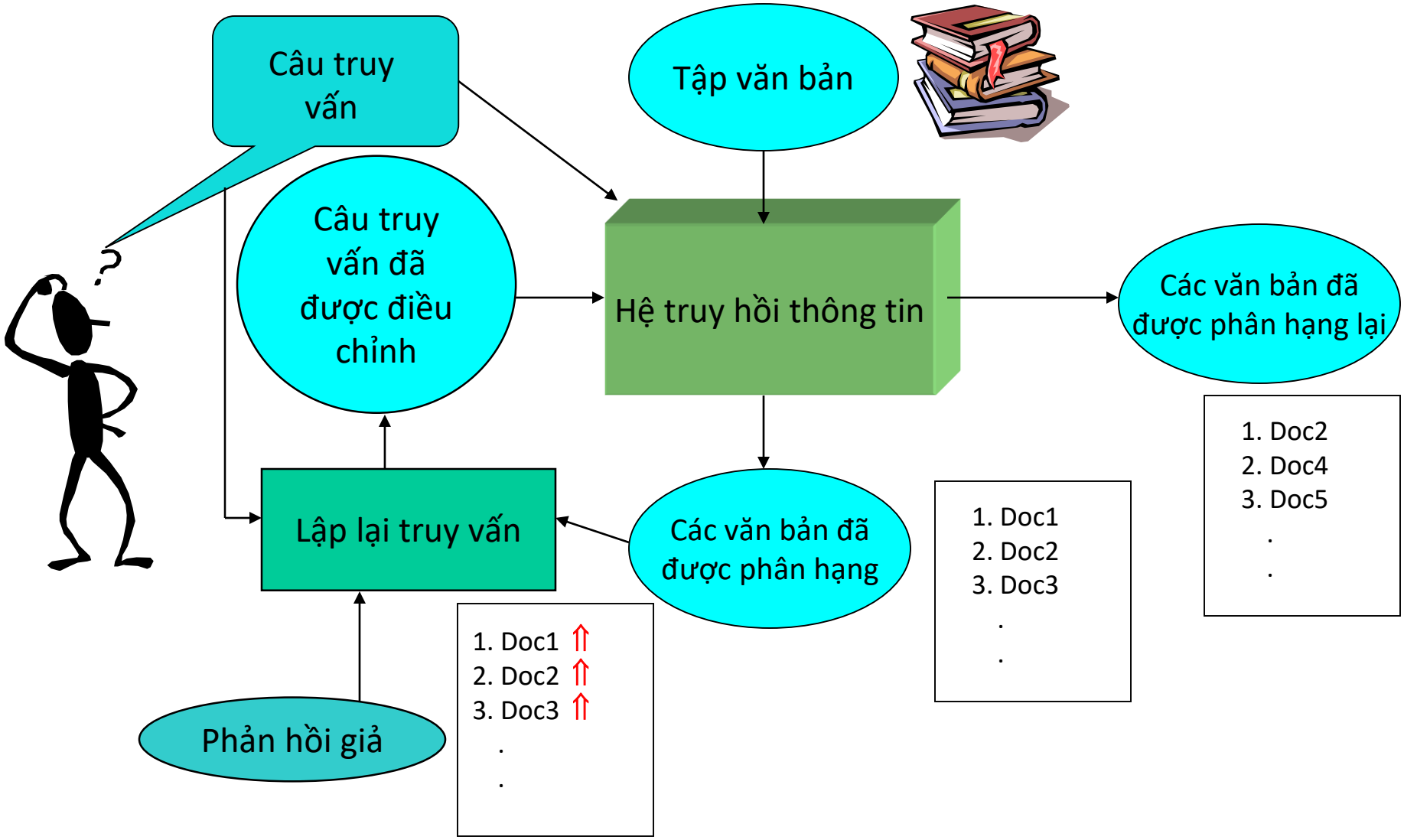
Vì sao phản hồi không được dùng rộng rãi?

- Người dùng đôi khi ngại hoặc không muốn cung cấp phản hồi tường minh.
- Dẫn đến những câu truy vấn dài, đòi hỏi tính toán nhiều hơn để trả về kết quả.
- Dẫn đến khó hiểu hơn vì sao một văn bản nào đó lại được trả về.

Phản hồi giả (pseudo feedback)

- Không dùng dữ liệu phản hồi tường minh của người dùng.
- Giả thiết m văn bản phân hạng cao nhất trong kết quả trả về là phù hợp. Ta sẽ dùng những văn bản này để lập lại câu truy vấn.

Kiến trúc phản hồi giả



Từ điển đồng nghĩa

- Cung cấp thông tin về những từ và cụm từ đồng nghĩa hoặc có liên quan về mặt ngữ nghĩa.
- Ví dụ:

  physician

  đồng nghĩa: croaker, doc, doctor, MD, medical, mediciner, medico, sawbones

  liên quan: medic, general practitioner, surgeon

Mở rộng câu truy vấn dựa trên từ điển đồng nghĩa

- Đối với mỗi từ t trong câu truy vấn, mở rộng câu truy vấn bằng cách bổ sung các từ đồng nghĩa và có liên quan của từ t (lấy trong từ điển đồng nghĩa).
- Có thể định trọng số các từ bổ sung nhỏ hơn các từ có trong câu truy vấn gốc.
- Nói chung sẽ giúp tăng độ thu hồi.
- Có thể làm giảm độ chính xác đáng kể, đặc biệt là với các từ nhập nhằng về nghĩa.

Từ điển đồng nghĩa thống kê

- Từ điển đồng nghĩa do con người xây dựng thủ công có thể không có sẵn (hoặc dễ tìm) với những ngôn ngữ nào đó.
- Từ điển đồng nghĩa của con người có thể không đủ rộng để bao trùm lên tất cả các tình huống.
- Có thể ***phân tích thống kê*** tập văn bản để phát hiện ra những từ có liên quan về mặt ngữ nghĩa.

Phân tích toàn cục tự động

- Xác định các từ tương tự nhau thông qua những tính toán thống kê trên ***toàn bộ*** tập văn bản.
- Tính ***ma trận kết hợp***: Lượng hóa tương quan giữa các từ trên cơ sở chúng xuất hiện cùng nhau trong cùng một văn bản nhiều ít ra sao.
- Mở rộng câu truy vấn bằng cách bổ sung các từ có tương quan cao với các từ trong câu truy vấn.

Ma trận kết hợp

	w_1	w_2	w_3	.....	w_n
w_1	c_{11}	c_{12}	c_{13}	.....	c_{1n}
w_2	c_{21}				
w_3	c_{31}				
.	.				
.	.				
w_n	c_{n1}				

c_{ij}: Hệ số tương quan giữa từ w_i và từ w_j:

$$c_{ij} = \sum_{d_k \in D} f_{ik} \times f_{jk}$$

f_{ik}: Tần số của từ w_i trong văn bản d_k

D: Tập văn bản

Ma trận kết hợp chuẩn hóa

- Hệ số tương quan dựa trên tần số từ (trong slide trước) thiên vị những từ phổ biến hơn.
- Chuẩn hóa điểm số về khoảng [0, 1]:

$$s_{ij} = \frac{c_{ij}}{c_{ii} + c_{jj} - c_{ij}}$$

- Điểm số đã chuẩn hóa đạt mức 1 (cao nhất) nếu hai từ có cùng tần số trong tất cả các văn bản.

Vấn đề với phân tích toàn cục

- Sự nhập nhằng nghĩa từ có thể dẫn đến những từ tương quan về mặt thống kê nhưng lại không phù hợp:
  - Ví dụ: Apple computer → Apple computer **red fruit**
- Vì các từ bổ sung có tương quan cao với các từ trong câu truy vấn, việc mở rộng truy vấn có thể không đem lại nhiều văn bản mới.

Phân tích cục bộ tự động

- Ở thời điểm xử lý câu truy vấn, xác định các từ tương tự (với các từ truy vấn) bằng cách phân tích các văn bản trả về với thứ hạng cao nhất.
- Phân tích chỉ diễn ra trên một tập ***cục bộ*** gồm những văn bản trả về cho một câu truy vấn cụ thể.
- Tránh được nhập nhằng nghĩa vì tìm các từ tương tự chỉ trong các văn bản phù hợp:
  - Ví dụ: Apple computer → Apple computer **Macbook laptop**

So sánh phân tích toàn cục và cục bộ

- Phân tích toàn cục đòi hỏi tính toán cường độ cao, nhưng chỉ một lần ở thời gian phát triển hệ thống.
- Phân tích cục bộ đòi hỏi tính toán cường độ cao cho mỗi câu truy vấn ở thời gian vận hành hệ thống (mặc lượng tính toán mỗi lần ít hơn phân tích toàn cục).
- Phân tích cục bộ đem lại kết quả tốt hơn.

Tinh chỉnh phân tích toàn cục

- Chỉ mở rộng câu truy vấn bằng những từ k_i tương tự với tất cả các từ k_j trong câu truy vấn Q:

$$sim(k_i, Q) = \sum_{k_j \in Q} c_{ij}$$

  - VD1: fruit không được thêm vào Apple computer vì nó gần với Apple nhưng lại cách xa computer.
  - VD2: fruit được thêm vào apple pie vì fruit gần với cả hai từ apple và pie.
- Có thể dùng cách định trọng số từ tinh vi hơn (thay vì tần số từ đơn giản) khi tính hệ số tương quan giữa các từ.

Kết luận về mở rộng truy vấn

- Mở rộng câu truy vấn bằng cách bổ sung các từ liên quan có thể giúp cải thiện hiệu suất, đặc biệt là độ thu hồi (recall).
- Tuy nhiên, phải chọn các từ liên quan thật cẩn thận để tránh mất mát độ chính xác (precision).

Bài tập

Giả sử câu truy vấn ban đầu của người dùng là cheap CDs cheap DVDs extremely cheap CDs. Người dùng kiểm tra hai văn bản d_1 và d_2, đánh giá d_1 với nội dung CDs cheap software cheap CDs là phù hợp và d_2 với nội dung cheap thrills DVDs là không phù hợp. Giả sử ta đang dùng tần số từ để định trọng số từ và không chuẩn hóa các vector về chiều dài đơn vị.

Tính vector câu truy vấn sau khi điều chỉnh theo phương pháp phản hồi phù hợp Rocchio; giả sử $\alpha = 1$, $\beta = 0.75$ và $\gamma = 0.25$.