Phân tích liên kết
(Link Analysis)

Nguyễn Mạnh Hiển
hiennm@tlu.edu.vn

Trang (web) uy tín

- **Trang uy tín** (authority) là trang cung cấp những thông tin quan trọng, tin cậy và hữu ích về một chủ đề nào đó.
- **Bậc vào** (in-degree) là số liên kết trỏ đến một trang, là một độ đo đơn giản cho uy tín của một trang.
- Tuy nhiên, bậc vào xem các liên kết có vai trò như nhau:
  - Có nên cho liên kết từ một trang uy tín có trọng số lớn hơn hay không?

Trang (web) trung tâm

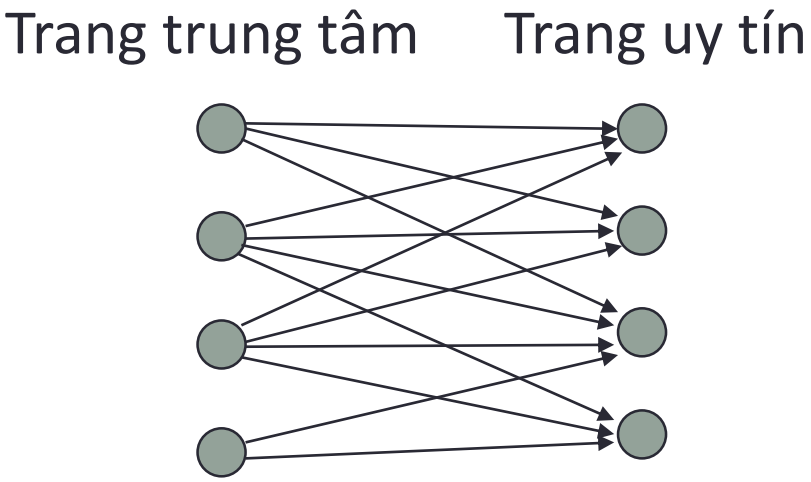
- **Trang trung tâm** (hub) là trang chỉ dẫn (index), cung cấp nhiều liên kết hữu ích tới các trang nội dung phù hợp (tức là các trang uy tín).

Thuật toán HITS

- Viết tắt của “Hyperlink-Induced Topic Search”.
- Được Kleinberg đề xuất vào năm 1998.
- Tìm các trang trung tâm và uy tín về một chủ đề đã cho thông qua phân tích **đồ thị web**:
  - Một trang trung tâm trỏ tới nhiều trang uy tín.
  - Một trang uy tín được nhiều trang trung tâm trỏ tới.

Trang trung tâm và trang uy tín

- Chúng lập thành một đồ thị hai bên (bipartite graph):

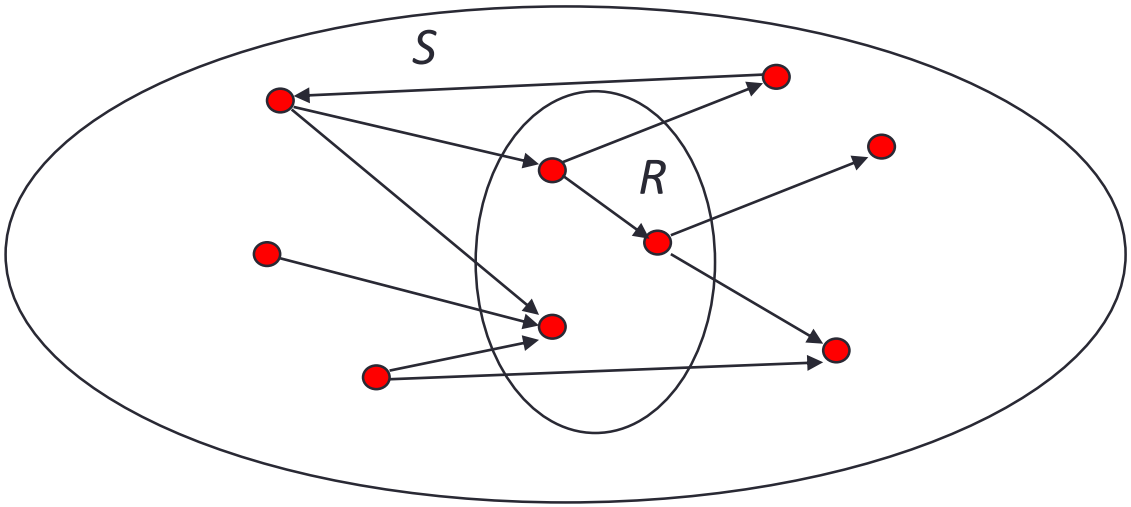


Thuật toán HITS (tiếp)

- Tính các trang trung tâm và uy tín cho một chủ đề; chủ đề đó xác định bằng một câu truy vấn.
- Trước tiên, xác định **tập cơ sở** S gồm những trang phù hợp với câu truy vấn.
- Phân tích cấu trúc liên kết của **đồ thị cơ sở** ứng với S để tìm các trang trung tâm và uy tín trên tập này.

Xây dựng đồ thị cơ sở

- Xét câu truy vấn Q, gọi R là tập các trang trả về bởi một máy tìm kiếm nào đó (như mô hình không gian véctơ).
- Khởi tạo S bằng R.
- Thêm vào S tất cả các trang được một trang trong R trỏ đến.
- Thêm vào S tất cả các trang trỏ đến một trang trong R.



Uy tín và bậc vào

- Trong tập cơ sở S, những trang có bậc vào cao nhất chưa chắc là trang uy tín (có thể đó chỉ là những trang phổ biến như Amazon).
- Trang uy tín thực sự là trang được nhiều trang trung tâm trỏ đến (biết rằng trang trung tâm là trang trỏ đến các trang uy tín).

Thuật toán lặp

- Lặp lại việc tính điểm số trung tâm và uy tín cho các trang cho đến khi hội tụ, tức là không còn thay đổi nữa.
- Mỗi trang $p \in S$ có:
  - Điểm số uy tín a_p; điểm số cho tất cả các trang lập thành vector **a**.
  - Điểm số trung tâm h_p; điểm số cho tất cả các trang lập thành vector **h**.
- Khởi tạo $a_p = h_p = 1$.
- Chuẩn hóa điểm số sao cho chiều dài các vector **a** và **h** bằng 1:

$$\sum_{p \in S} (a_p)^2 = 1 \qquad \sum_{p \in S} (h_p)^2 = 1$$

Cập nhật điểm số

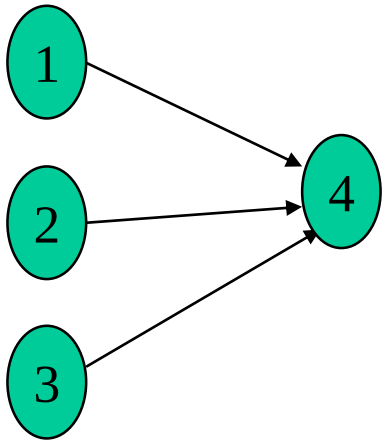
- Trang uy tín được nhiều trang trung tâm trỏ đến:

$$a_p = \sum_{q:q \to p} h_q$$

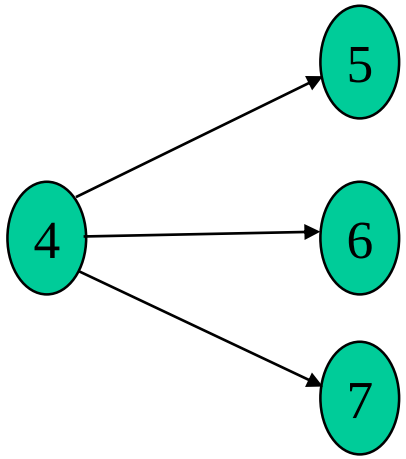
- Trang trung tâm trỏ đến nhiều trang uy tín:

$$h_p = \sum_{q:p \to q} a_q$$

Minh họa cập nhật điểm số



$a_4 = h_1 + h_2 + h_3$



$h_4 = a_5 + a_6 + a_7$

Thuật toán lặp HITS

Khởi tạo cho mọi p $\in$ S: $a_p = h_p = 1$

For i = 1 to k:

Với mọi p $\in$ S: $a_p = \sum_{q:q\to p} h_q$

Với mọi p $\in$ S: $h_p = \sum_{q:p\to q} a_q$

Với mọi p $\in$ S: $a_p = a_p/c$, trong đó c: $\sum_{p\in S} \left(a_p / c\right)^2 = 1$

Với mọi p $\in$ S: $h_p = h_p/c$, trong đó c: $\sum_{p\in S} \left(h_p / c\right)^2 = 1$

Tìm trang tương tự dùng cấu trúc liên kết

- Xét trang P, gọi R là tập các trang trỏ đến P.
- Xây dựng tập cơ sở S từ R.
- Chạy thuật toán HITS trên S.
- Trả về những trang uy tín nhất trong S làm các trang tương tự nhất với P.

Kết quả tìm trang tương tự

- Cho “honda.com”, tìm được:
  - toyota.com
  - ford.com
  - bmwusa.com
  - saturncars.com
  - nissanmotors.com
  - audi.com
  - volvocars.com

Thuật toán PageRank

- Một phương pháp phân tích liên kết khác, được dùng bởi Google (Brin & Page, 1998).
- Không cố bắt lấy sự khác biệt giữa các trang trung tâm và uy tín.
- Phân hạng các trang chỉ theo uy tín.
- Áp dụng vào toàn bộ web thay vì vùng lân cận quanh các trang trả về cho một câu truy vấn.

Ý tưởng PageRank khởi đầu

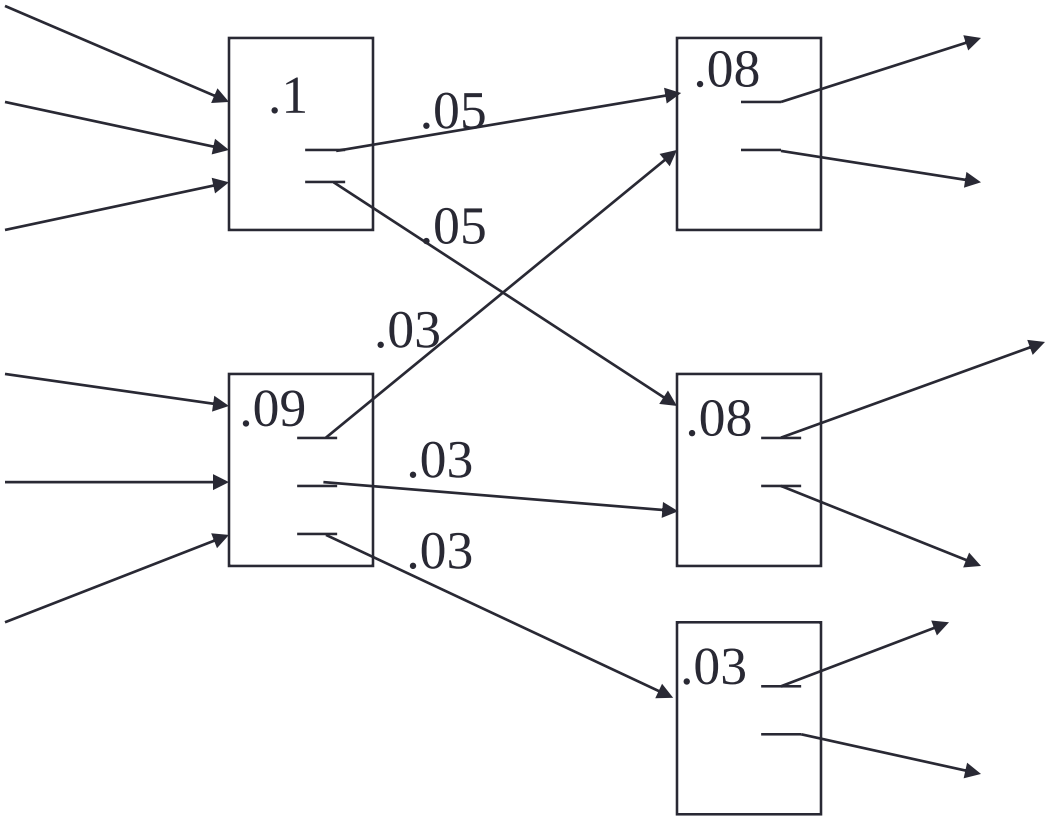
Hạng khởi đầu của một trang p:

$$R(p) = c \sum_{q:q \to p} \frac{R(q)}{N_q}$$

- N_q là tổng số liên kết đi ra từ trang q.
- Trang q chia uy tín đều nhau cho tất cả các trang nó trỏ đến.
- c là hằng số chuẩn hóa có giá trị sao cho hạng của tất cả các trang có tổng bằng 1.

Ý tưởng PageRank khởi đầu

- Có thể xem PageRank như một tiến trình "chảy" từ trang này sang trang khác theo các liên kết.



Thuật toán khởi đầu

- Lặp lại tiến trình chảy cho đến khi hội tụ:

  Gọi S là tập các trang web

  Khởi tạo R(p) = 1/|S| $\forall p \in S$

  Cho đến khi hạng không thay đổi (nhiều):

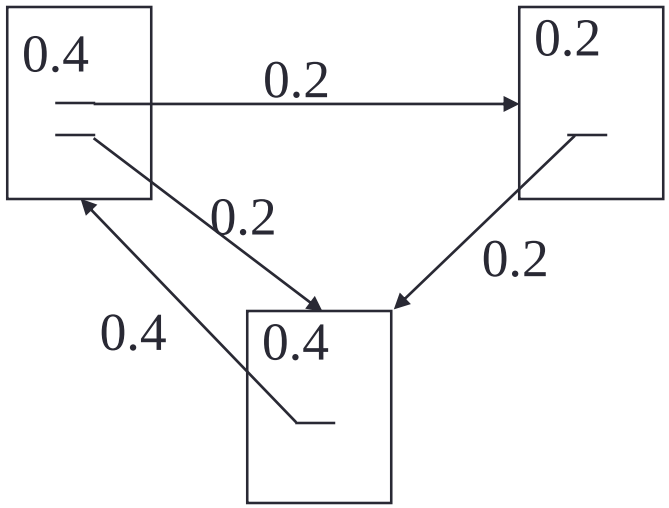
  Với mỗi $p \in S$:

$$R'(p) = \sum_{q:q \to p} \frac{R(q)}{N_q}$$

$$c = 1/\sum_{p \in S} R'(p)$$

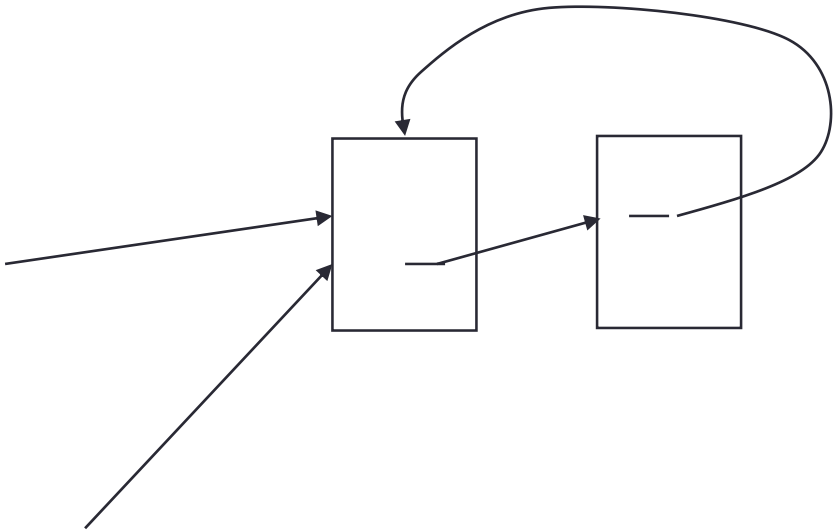
  Với mỗi $p \in S$: R(p) = cR'(p)

Trạng thái ổn định



Vấn đề với ý tưởng khởi đầu

- Một nhóm trang chỉ trỏ tới bản thân chúng, nhưng được những trang khác trỏ đến, sẽ hoạt động giống như “đích hạng” (rank sink) và sẽ hấp thụ hết hạng của hệ thống.



Nguồn hạng (rank source)

- Ta đưa thêm vào nguồn hạng E:
  - Sẽ liên tục bổ sung vào hạng của mỗi trang p một lượng cố định E(p):

$$R(p) = c\left(\sum_{q:q\to p} \frac{R(q)}{N_q} + E(p) \right)$$

Thuật toán PageRank

Gọi S là tập các trang web

Với mỗi p ∈ S: E(p) = $\alpha/|S|$ (với $0 < \alpha < 1$, ví dụ $\alpha = 0.15$)

Với mỗi p ∈ S, khởi tạo: R(p) = $1/|S|$

Cho đến khi hạng không thay đổi (nhiều)

Với mỗi p ∈ S:

$$R'(p) = \left[(1-\alpha) \sum_{q:q \to p} \frac{R(q)}{N_q} \right] + E(p)$$

$$c = 1 / \sum_{p \in S} R'(p)$$

Với mỗi p ∈ S: R(p) = cR'(p)

Tìm kiếm tiêu đề với PageRank

- Dùng truy hồi Boole để tìm trong tiêu đề trang và phân hạng các trang trả về theo điểm số PageRank của chúng.
- Tìm kiếm university:
  - Altavista (tên gọi một máy tìm kiếm) đã trả về một tập ngẫu nhiên các trang với từ university trong tiêu đề (có vẻ thích các URL ngắn).
  - Google nguyên thủy đã trả về trang nhà của các đại học hàng đầu.

Kết luận về phân tích liên kết

- Phân tích liên kết dùng thông tin cấu trúc đồ thị web để hỗ trợ việc tìm kiếm.
- Là một trong những cách tân lớn trong tìm kiếm web.
- Là một trong những nguyên nhân chính cho sự thành công khởi đầu của Google.