# 데이터 과학

## L15: PageRank

Kookmin University

# 목차

- **PageRank 개요 및 응용**
- PageRank 심플버전
- PageRank 오리지널

# PageRank

- Google 검색엔진의 기반 알고리즘
- 하이퍼링크를 이용한 웹 페이지 중요도 측정

## The PageRank Citation Ranking: Bringing Order to the Web

January 29, 1998

**Abstract**

The importance of a Web page is an inherently subjective matter, which depends on the readers interests, knowledge and attitudes. But there is still much that can be said objectively about the relative importance of Web pages. This paper describes PageRank, a method for rating Web pages objectively and mechanically, effectively measuring the human interest and attention devoted to them.

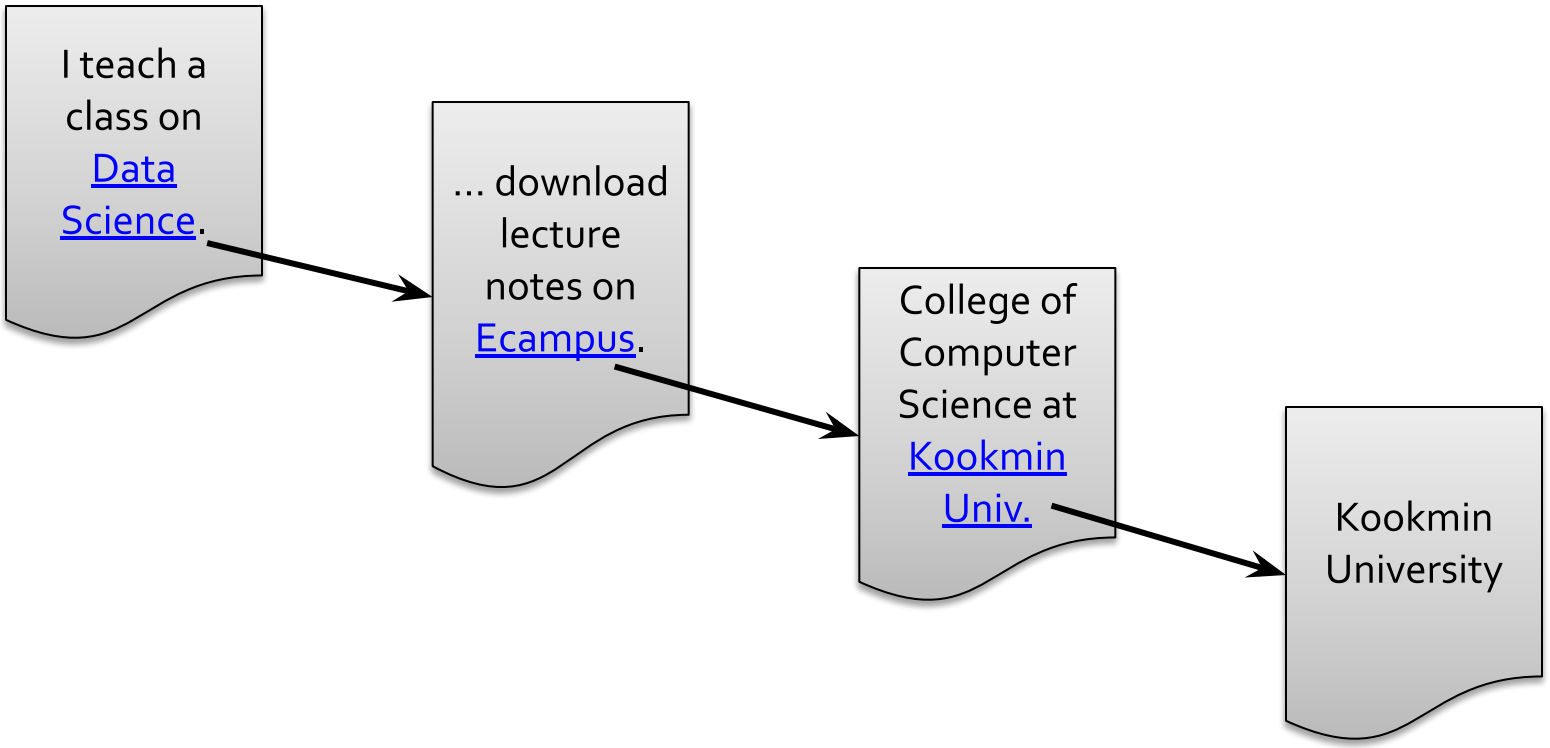
We compare PageRank to an idealized random Web surfer. We show how to efficiently compute PageRank for large numbers of pages. And, we show how to apply PageRank to search and to user navigation.

### 1 Introduction and Motivation

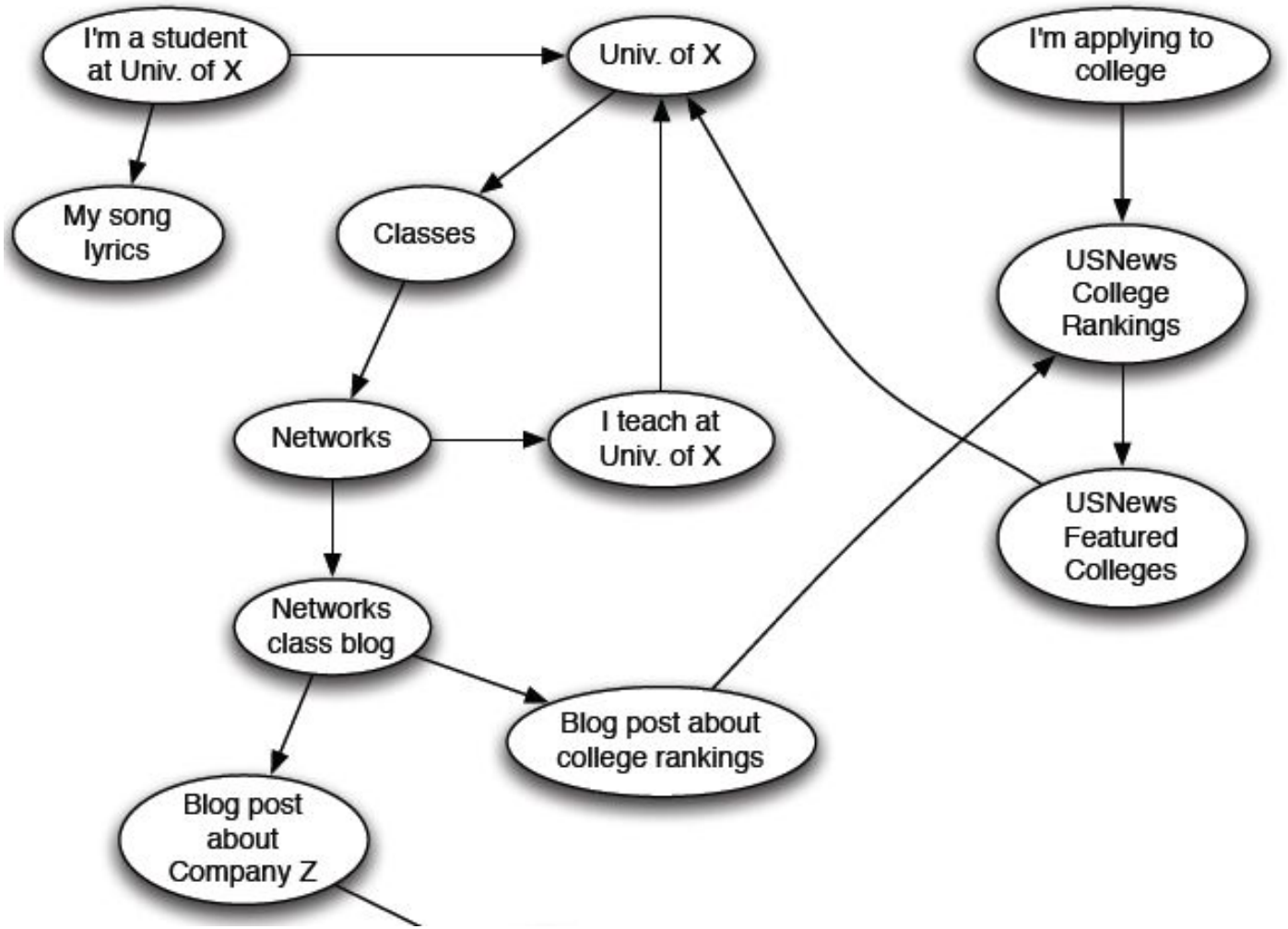
The World Wide Web creates many new challenges for information retrieval. It is very large and heterogeneous. Current estimates are that there are over 150 million web pages with a doubling life of less than one year. More importantly, the web pages are extremely diverse, ranging from "What is Joe having for lunch today?" to journals about information retrieval. In addition to these major challenges, search engines on the Web must also contend with inexperienced users and pages engineered to manipulate search engine ranking functions.

# 하이퍼링크 네트워크

- Web을 directed graph로 보기
  - Node: 웹페이지
  - Edge: 하이퍼링크



# 하이퍼링크 네트워크

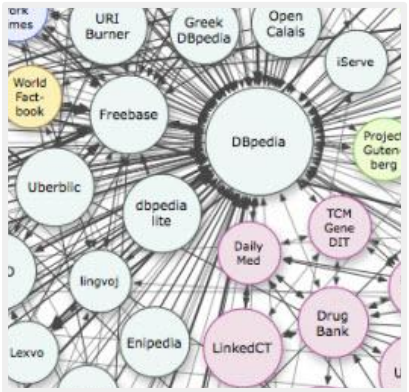

I'm a student at Univ. of X
Univ. of X
I'm applying to college
My song lyrics
Classes
USNews College Rankings
Networks
I teach at Univ. of X
USNews Featured Colleges
Networks class blog
Blog post about college rankings
Blog post about Company Z

# PageRank 의 응용

- 모든 그래프에서, 노드의 중요도 측정에 사용
  - 예시)
  - 웹 페이지의 중요도 측정
  - 친구관계 네트워크에서 핵심 인물 탐색
  - 생물학 그래프에서 중요한 단백질 조사
  - 컴퓨터 네트워크 통신 로그에서 DDoS 공격 탐지
  - …

Friendship Network

Phonecall Network

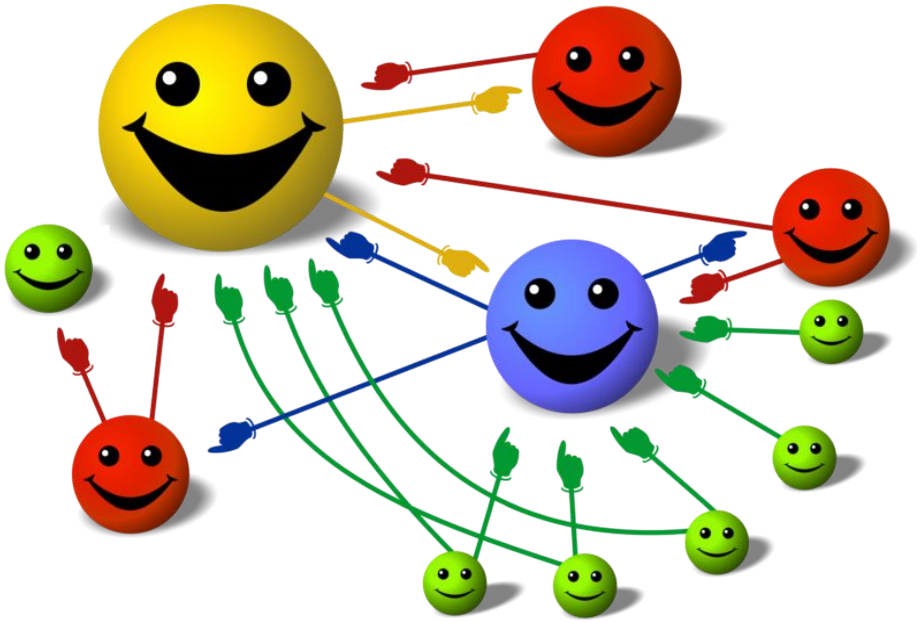



Knowledge Base

Internet, WWW

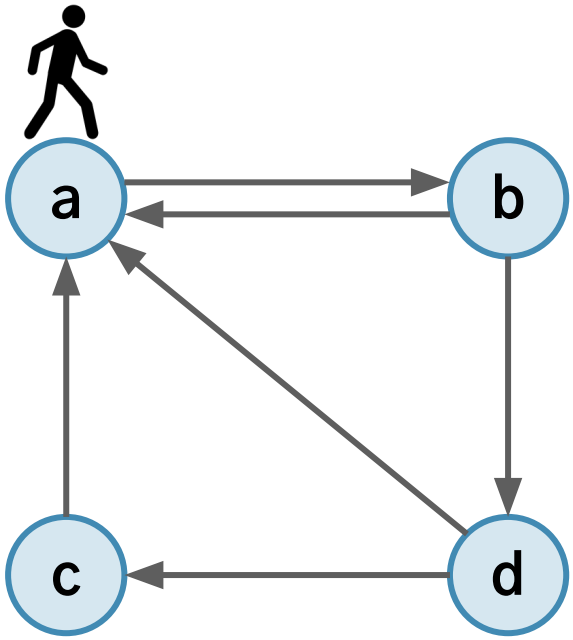
# PageRank

- 핵심 아이디어: 링크 정보를 이용하여 페이지에 순위(점수)를 매기자!
  - 많은 링크를 받는 페이지 → 높은 점수
  - 높은 점수의 페이지로부터 링크를 받으면 → 높은 점수



# Random Walk Interpretation

- PageRank = 간선을 따라 그래프를 떠돌아다니는 행인이 각 정점에 머무를 확률
- 웹을 서핑하는 과정과 유사

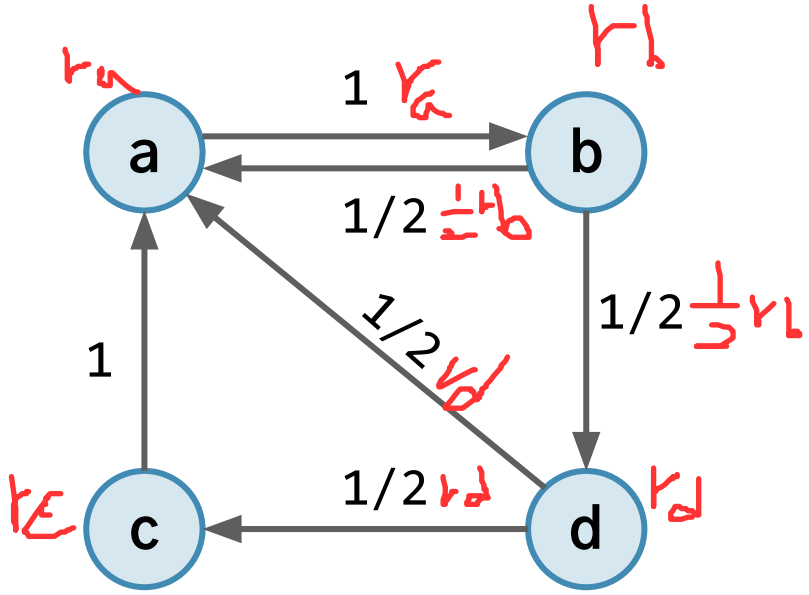


# 목차

- PageRank 개요 및 응용
- **PageRank 심플버전**
- PageRank 오리지널

# PageRank Score (심플버전)

- 내 점수를 골고루 이웃에게 나눠주기
- PageRank score = 받은 점수의 합

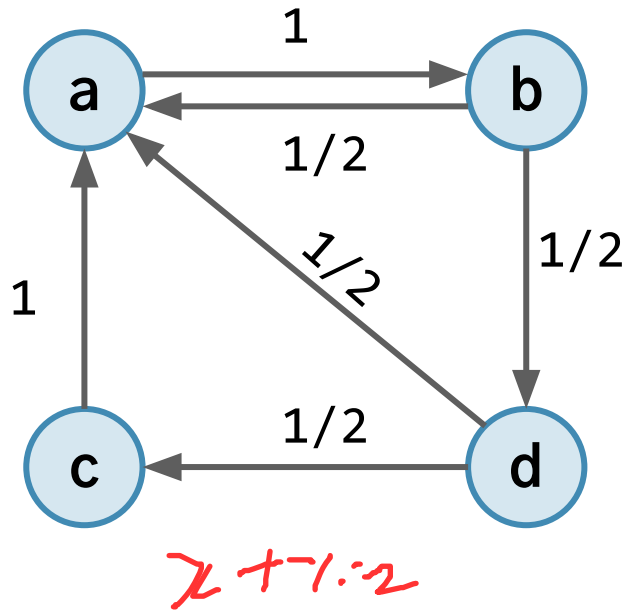



$$r_j = \sum_{i \to j} \frac{r_i}{\mathrm{d_i}}$$

# PageRank Score (심플버전)

- 내 점수를 골고루 이웃에게 나눠주기
- PageRank score = 받은 점수의 합

$$r_j = \sum_{i \to j} \frac{r_i}{\mathrm{d_i}}$$




| | a | b | c | d |
|---|---|---|---|---|
| a | 0 | 1/2 | 1 | 1/2 |
| b | 1 | 0 | 0 | 0 |
| c | 0 | 0 | 0 | 1/2 |
| d | 0 | 1/2 | 0 | 0 |

$\times$ $\begin{bmatrix} r_a \\ r_b \\ r_c \\ r_d \end{bmatrix}$ $=$ $\begin{bmatrix} r_a \\ r_b \\ r_c \\ r_d \end{bmatrix}$

# Power Iteration

- 모든 정점의 점수를 1/n 으로 초기화
- 정점마다 새로운 점수를 계산
  - 받은 점수의 합
- 수렴할 때까지 반복

| | a | b | c | d | | | |
|---|---|---|---|---|---|---|---|
| a | 0 | 1/2 | 1 | 1/2 | × | $r_a$ | = $r_a$ |
| b | 1 | 0 | 0 | 0 | | $r_b$ | $r_b$ |
| c | 0 | 0 | 0 | 1/2 | | $r_c$ | $r_c$ |
| d | 0 | 1/2 | 0 | 0 | | $r_d$ | $r_d$ |

| | | | | | | |
|---|---|---|---|---|---|---|
| $r_a$ | 0.25 | 0.50 | 0.31 | 0.31 | | 0.36 |
| $r_b$ | 0.25 | 0.25 | 0.50 | 0.50 | ... | 0.36 |
| $r_c$ | 0.25 | 0.13 | 0.06 | 0.06 | | 0.09 |
| $r_d$ | 0.25 | 0.13 | 0.13 | 0.13 | | 0.18 |

# Power Iteration

- Power Iteration: 행렬의 **주고유벡터** 계산 방법
- PageRank = M의 주고유벡터
  - M: Stochastic Adjacency Matrix
  - Largest eigenvalue of M = 1

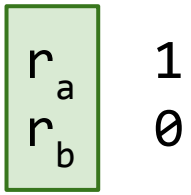
$$r(k+1) = \frac{M \cdot r(k)}{|M \cdot r(k)|}$$

# 목차

- PageRank 개요 및 응용
- PageRank 심플버전
- **PageRank 오리지널**

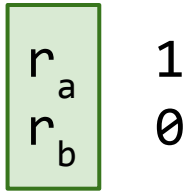
# 심플버전의 문제점

- 수렴하는가?

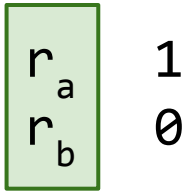




- 결과가 그럴듯한가?

(dangling node)





# 심플버전의 문제점

- 수렴하는가?


$r_a$ $r_b$

1 0 1 0 ...
0 1 0 1

- 결과가 그럴듯한가?

dead end
(dangling node)


$r_a$ $r_b$

1 0 0 0 ...
0 1 0 0

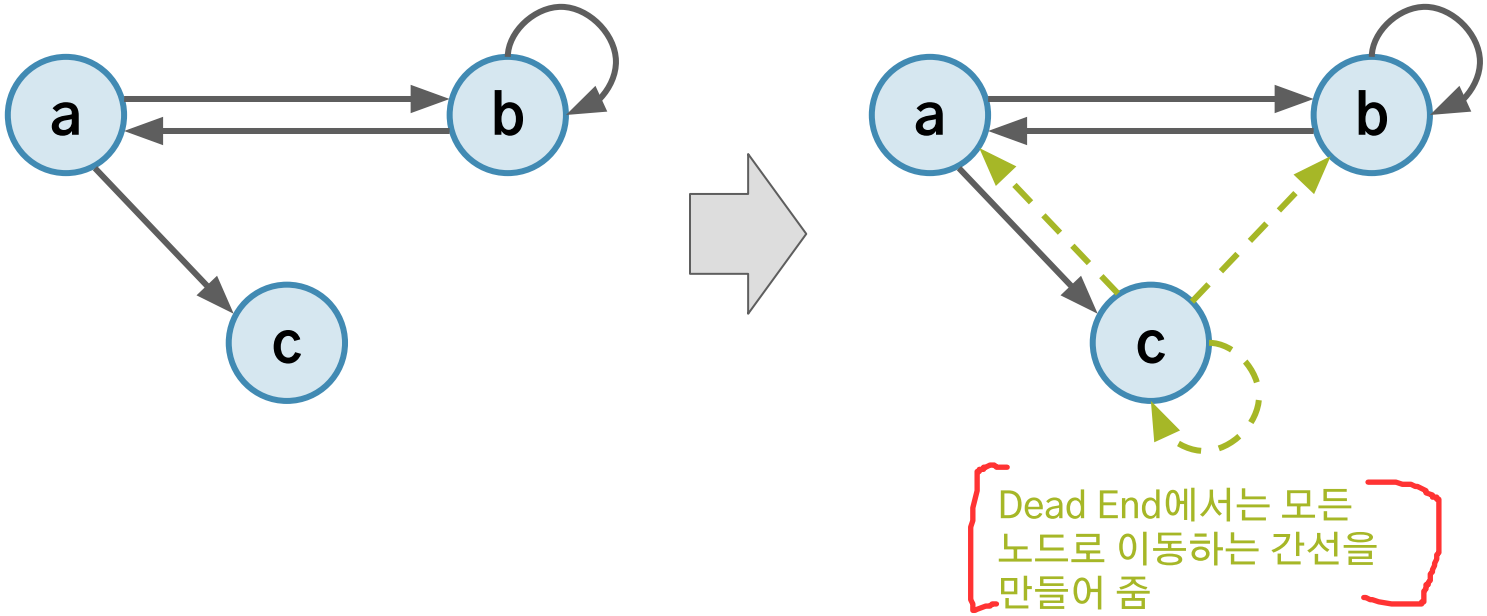
spider trap


$r_a$ $r_b$

1 0 0 0 ...
0 1 1 1

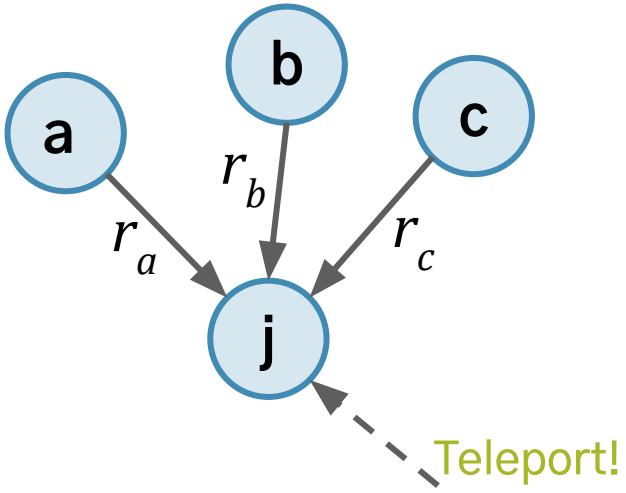
# Random Teleport!

- 매 스텝마다, 특정 확률(β)로 아무 노드로 순간이동
  - 수렴 문제 해결? 해결
  - Spider Trap 해결? 언젠가 빠져나옴
  - Dead End 해결? 무조건 순간이동



# PageRank (오리지널)

- 노드 점수?
  - β*이웃에게 받은 점수 + β*랜덤으로 받는 점수




$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1-\beta)\frac{1}{N}$$

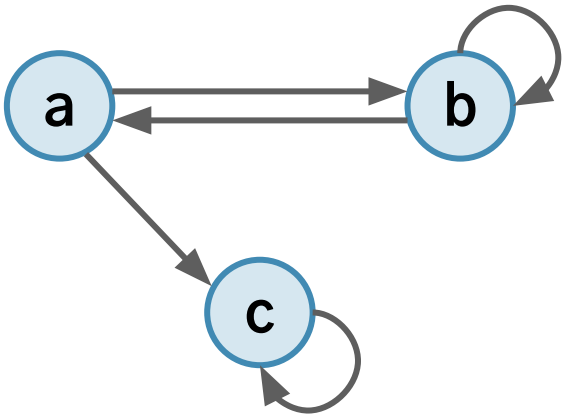
# PageRank (오리지널)

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1-\beta)\frac{1}{N}$$

- The Google Matrix

β값은 보통 0.8~0.9 정도를 사용
(평균적으로 5번 이동하면 점프)

$$A = \beta\, M + (1-\beta)\left[\frac{1}{N}\right]_{N\times N}$$




$$0.8\times \begin{bmatrix} 0 & 1/2 & 0 \\ 1/2 & 1/2 & 0 \\ 1/2 & 0 & 1 \end{bmatrix} + 0.2\times \begin{bmatrix} 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \\ 1/3 & 1/3 & 1/3 \end{bmatrix}$$

$M$ $\qquad$ $[1/N]_{N\times N}$

$$= \begin{bmatrix} 7/15 & 7/15 & 1/15 \\ 7/15 & 1/15 & 1/15 \\ 1/15 & 7/15 & 13/15 \end{bmatrix}$$

$A$

# PageRank (오리지널)

$$r_j = \sum_{i \to j} \beta \frac{r_i}{d_i} + (1-\beta)\frac{1}{N}$$

β값은 보통 0.8~0.9 정도를 사용
(평균적으로 5번 이동하면 점프)

- The Google Matrix

$$A = \beta\, M + (1-\beta)\left[\frac{1}{N}\right]_{N\times N}$$

M

- 여전히 Power Iteration을 사용할 수 있다!

$$\boldsymbol{r} = \boldsymbol{A} \cdot \boldsymbol{r}$$

# Questions?