Computational Intelligence and Learning Doctoral School

Kernels on Graph Nodes and Their Application to Link Analysis

Masashi Shimbo

Computational Linguistics Lab
Graduate School of Information Science
Nara Institute of Science and Technology, Japan

5 November 2007 @ Université Catholique de Louvain

Background

- Several kernels on graph nodes have been proposed in machine learning community.
- Kernels are a generalization of inner products, and inner products are the basis of similarity computation between entities.
- Hence these *graph kernels* should be providing measures of similarity between graph nodes.

Background

- Measuring relatedness between graph nodes is a major task of link analysis.

- Is graph kernels useful for link analysis as well?

Motivation

Graph kernels define inner products of nodes in a graph

inner product	≈ similarity	in machine learning
similarity	≈ relatedness	synonym

- inner product ≈ relatedness — for all graph kernels???
- inner product ≠ importance — for any graph kernels???

Objective

This talk provides an overview of graph kernels, with emphasis on their applicability to link analysis.

- What kind of relatedness measures do these graph kernels provide? (provided that they do)
- Do graph kernels have no use in another major task of link analysis, i.e., measuring the importance of individual nodes?

Contents

1. Background
2. Link analysis preliminaries
3. Graph kernels and link analysis
4. Dealing with multiple communities in a graph

Link analysis

Given a network-structured data

WWW, citation networks, social networks, …

find useful information from the underlying graph structure

- Which node is most important?
- Which node is most related to a given node?

Popular link analysis measures

Relatedness between two nodes

- Co-citation, bibliographic (co-reference) coupling

Importance of individual nodes

- PageRank, HITS

Review of existing link analysis measures

Relatedness measures

- Co-citation / bibliographic coupling

Importance measure

- HITS

Co-citation/bibliographic coupling "relatedness"

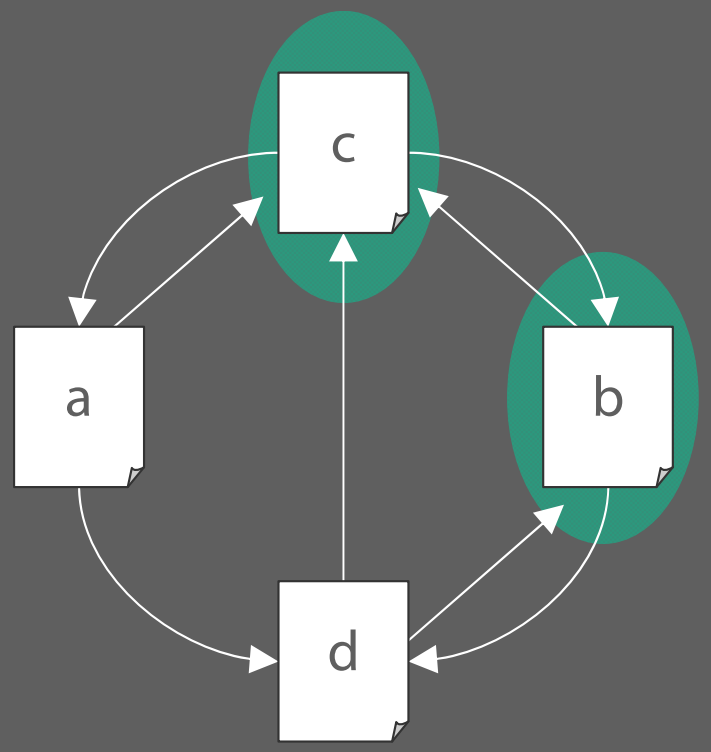
Index of the degree to which two scientific papers are related to each other

Co-citation coupling [Small & al., 1973]

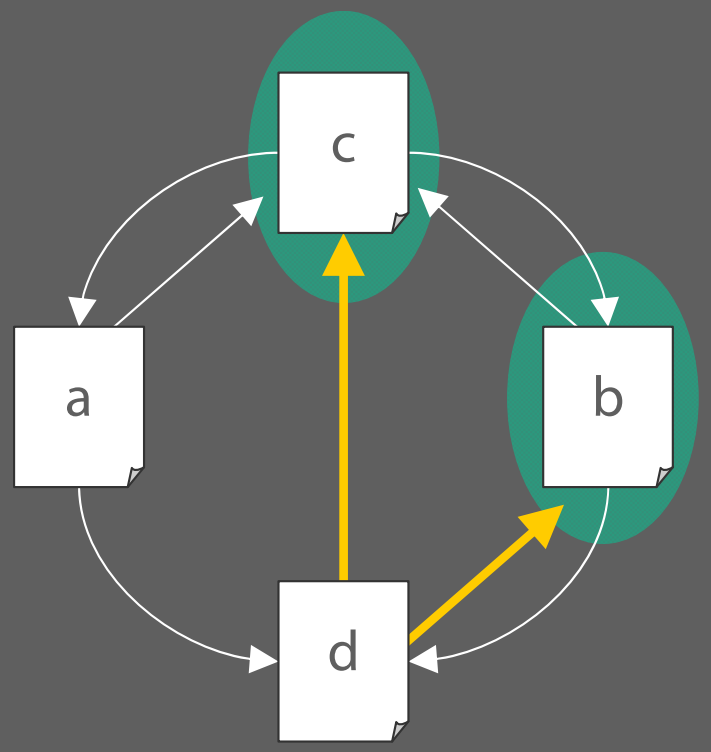
Number of papers jointly citing the two

Bibliographic (co-reference) coupling [Kessler,1963]

Number of papers jointly cited by the two



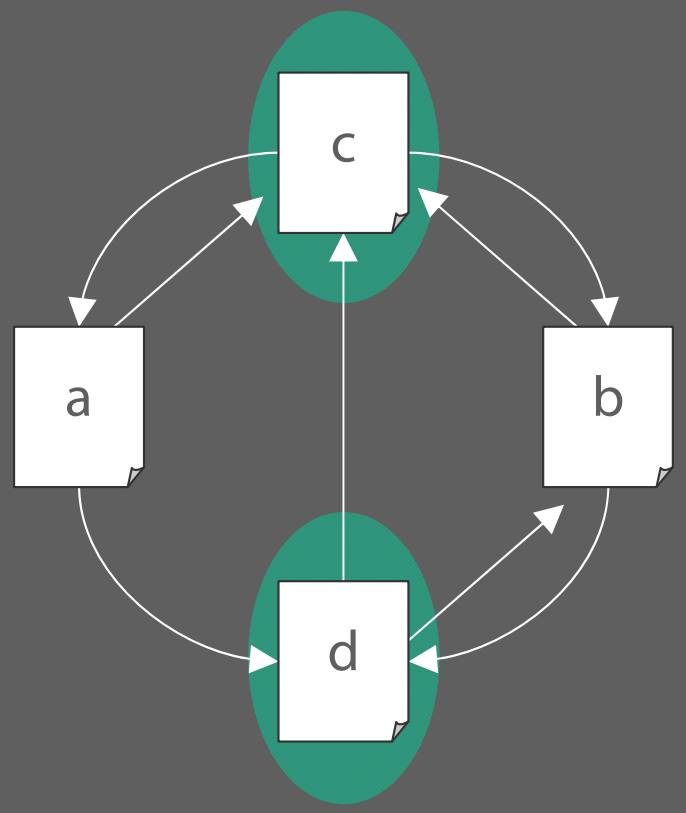
Consider nodes b and c



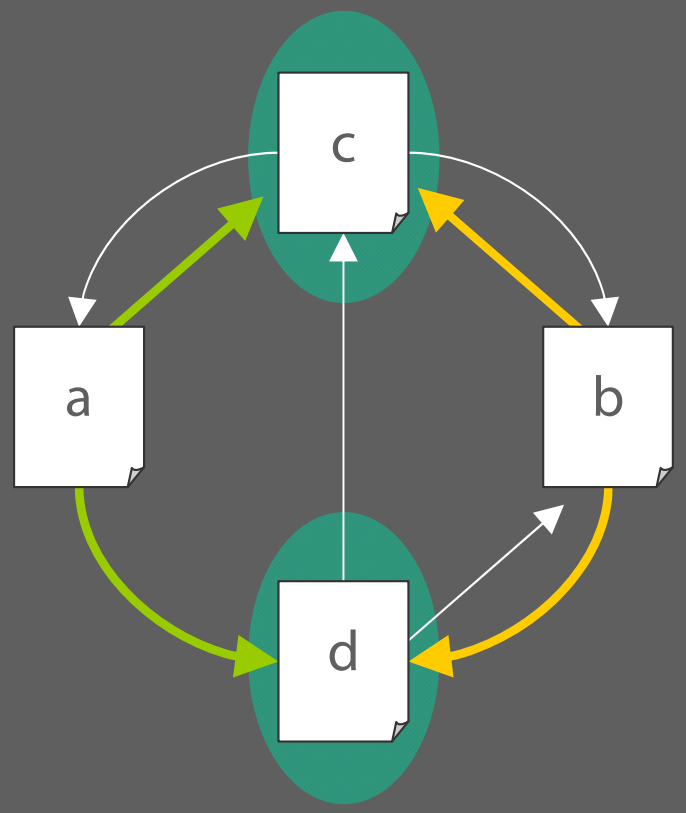
Consider node b and c

d is the only node
that jointly cites b and c

Hence, co-citation(b,c) = 1



Next consider nodes c and d



Next consider nodes c and d

Two nodes a and b jointly cite c and d

Hence, co-citation(c,d) = 2

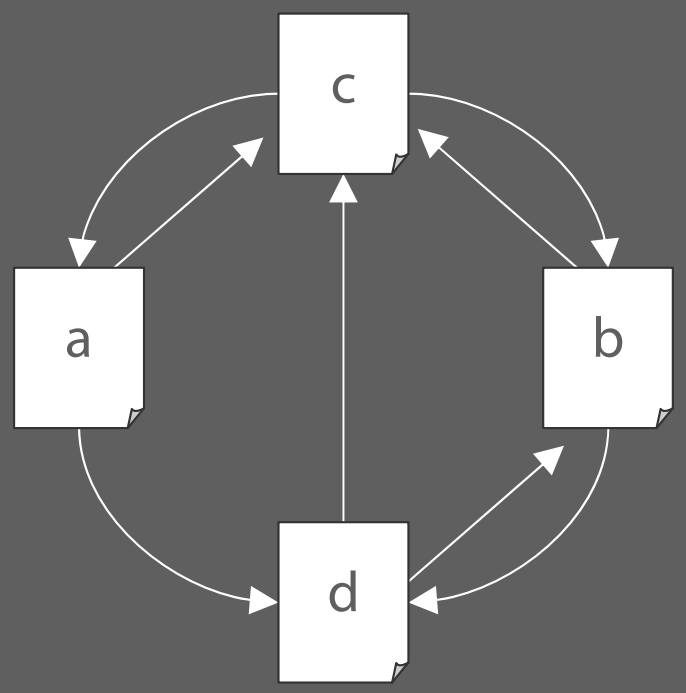
How do we count numbers of co-citations / co-references?

A = adjacency matrix of a citation graph

A^T = matrix transpose of A

$A^TA[i, j]$ ➔ co-citation relatedness of (i, j)

$AA^T[i, j]$ ➔ bibliographic coupling relatedness of (i, j)



Matrix-based computation of co-citation coupling

First compute adjacency matrix A

Then compute $A^T A$

adjacency matrix $A =$

$$\begin{array}{c|cccc} \text{from} \backslash \text{to} & a & b & c & d \\ \hline a & 0 & 0 & 1 & 1 \\ b & 0 & 0 & 1 & 1 \\ c & 1 & 1 & 0 & 0 \\ d & 0 & 1 & 1 & 0 \end{array}$$

A^TA = co-citation matrix

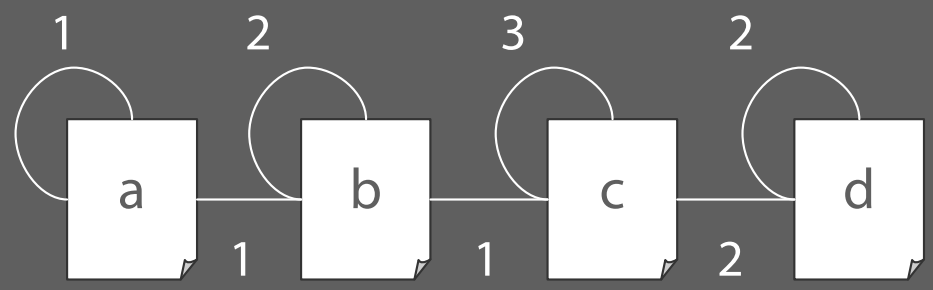
$$A^TA = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 3 & 2 \\ 0 & 0 & 2 & 2 \end{pmatrix} \end{array}$$

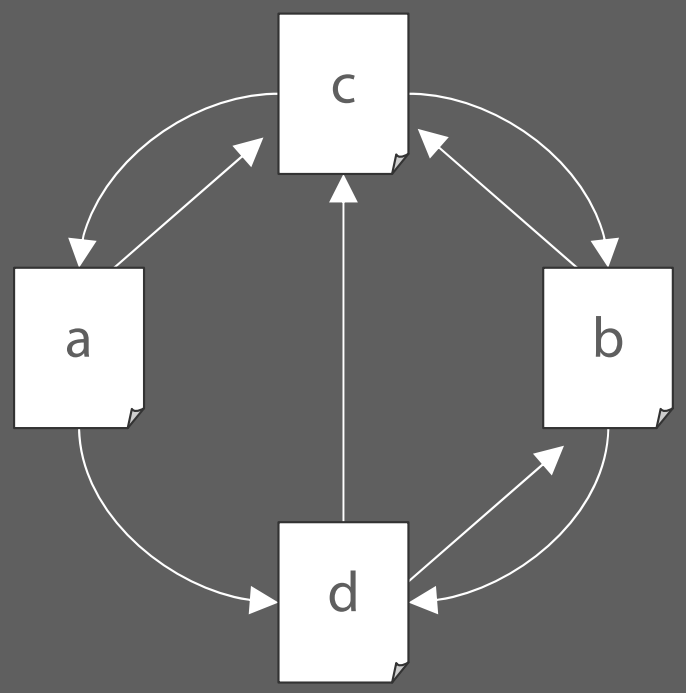
$$A = \begin{array}{c|c} \text{from} \backslash \text{to} & \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \end{array} & \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

A^TA = co-citation matrix

$$A^TA = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 3 & 2 \\ 0 & 0 & 2 & 2 \end{pmatrix} \end{array}$$

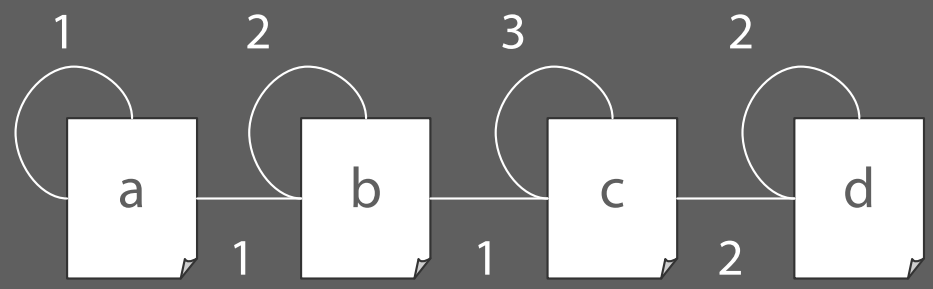
Induced co-citation graph

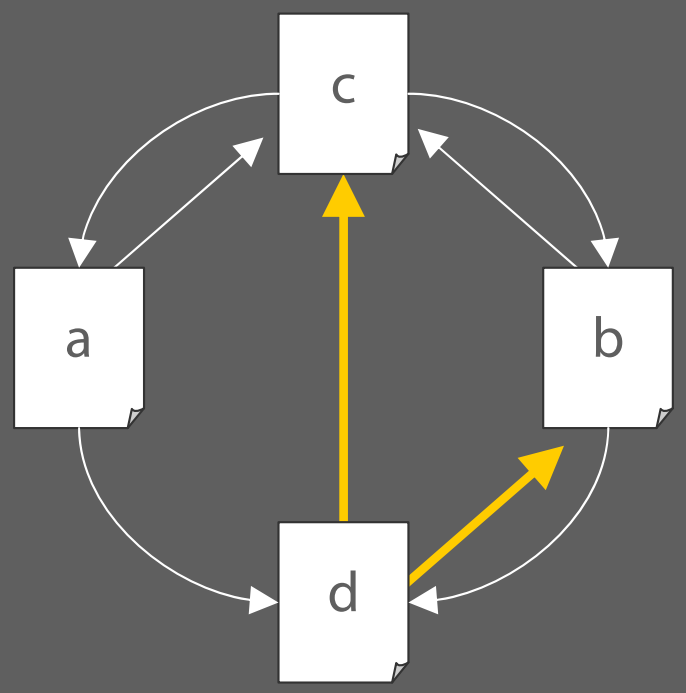




Original
citation graph

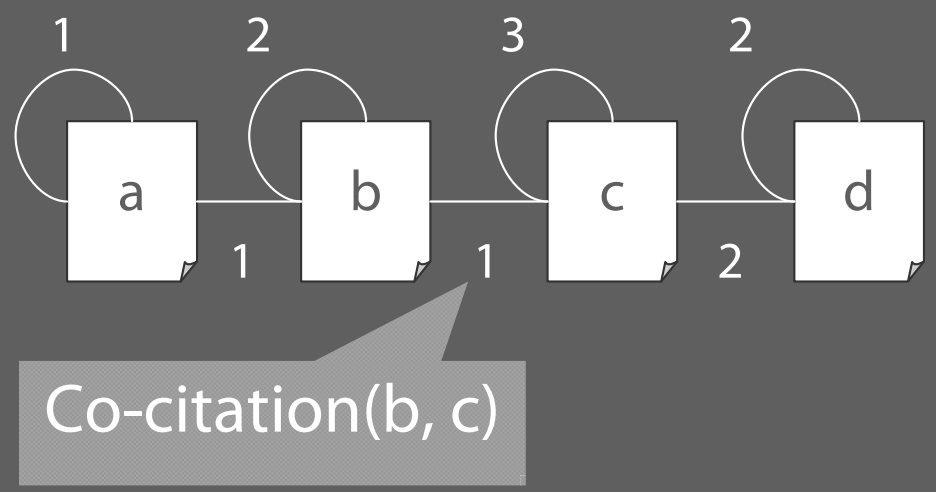
Induced
co-citation graph

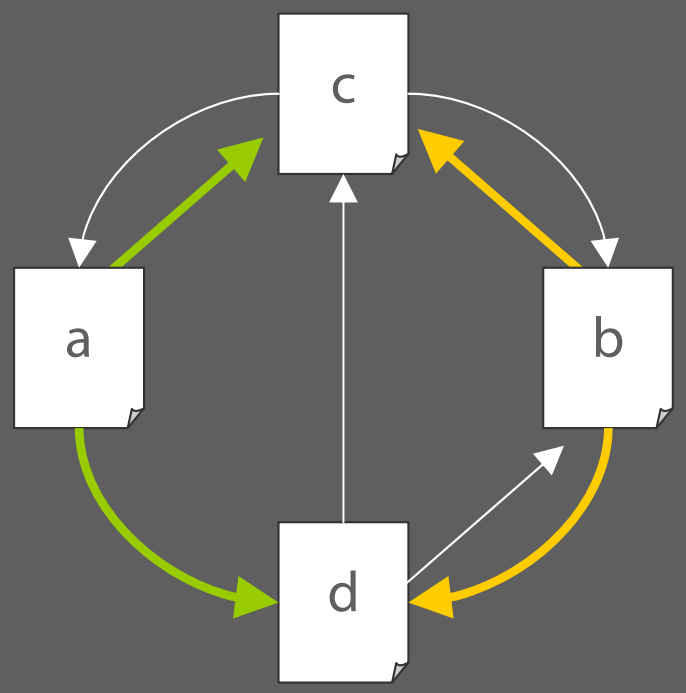




Original
citation graph

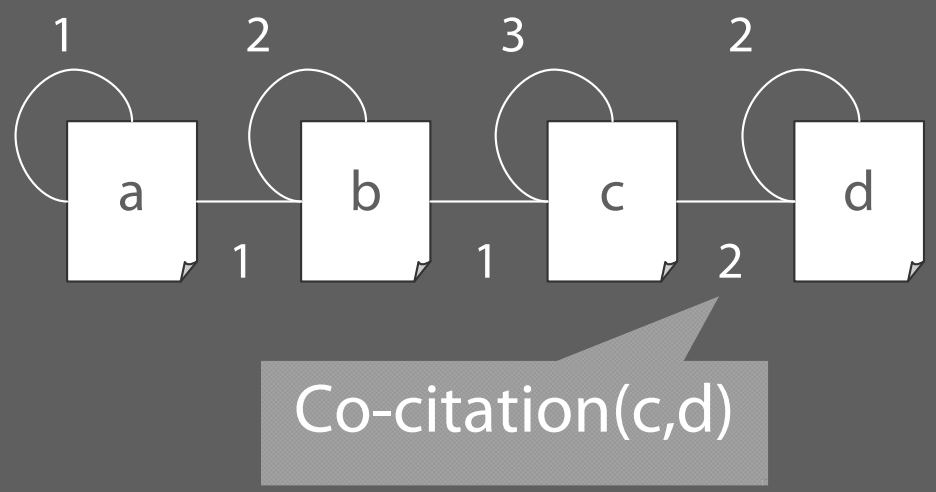
Induced
co-citation graph

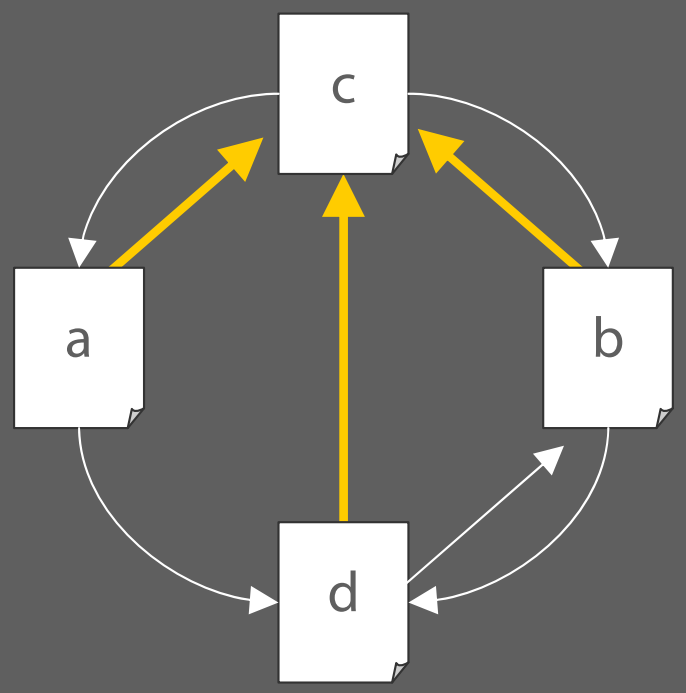




Original
citation graph

Induced
co-citation graph

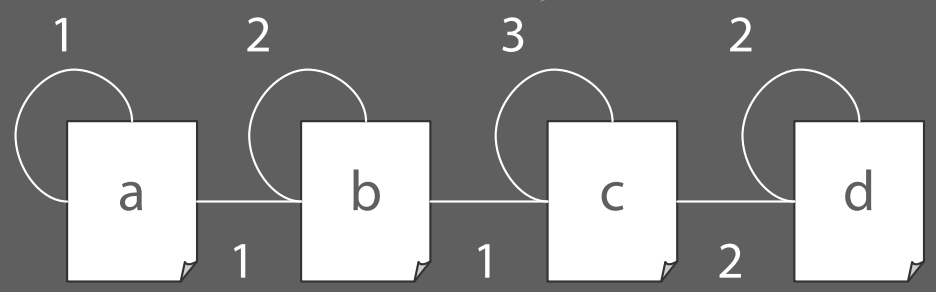




Original
citation graph

in-degree of
c in original
graph

Induced
co-citation graph



Similarly,

Bibliographic coupling (number of shared citations from two nodes) relatedness is given by the elements of AA^{T}

Summary: co-citation/bibliographic coupling

A = adjacency matrix of citation graph

$A^T A$ = co-citation matrix

AA^T = bibliographic coupling matrix

Review of existing link analysis measures

Relatedness

- Co-citation / bibliographic coupling

Importance

- HITS

HITS

Hyperlink-Induced Topic Search

[Kleinberg, 1999]

HITS "procedure"

1. **Collect the base set**
    1. Obtain set S of pages containing query keywords
    2. Expand the set to the "base set"

    E.g., by augmenting S with pages linking to and linked from pages in S

2. **Compute HITS rankings**

    Assign two scores to each page in the graph induced by the base set

Two types of important pages

Kleinberg distinguished two types of pages useful for topic search

Authority pages

- Authoritative source of the query topic

Hub pages

- The page itself may not be an authority but contains many pointers to authoritative pages – still useful

HITS "importance measures"

Each page n is associated with *two* importance scores

Authority score $a[n]$

Hub score $h[n]$

Define (for later convenience)

Authority *vector* $\boldsymbol{a} = (a[n_1], \ldots, a[n_N])^T$

Hub *vector* $\boldsymbol{h} = (h[n_1], \ldots h[n_N])^T$

(superscripted T denotes a matrix transpose)

How to determine authorities/hubs?

A good hub page is one that points to many good authority pages

A good authority page is likely to be cited by many good hub pages

Circular dependence → make it a recursive computation!

HITS recursion

Compute scores a_t and h_t for each t=1,2, …

$$a_t[n] \leftarrow \gamma \sum_{m \to n} h_{t-1}[m]$$

$$h_t[n] \leftarrow \eta \sum_{n \to m} a_t[m]$$

until convergence (start with a uniform vector $\boldsymbol{h}_0$)

γ, η: normalization constant (to keep $|\boldsymbol{a}_t|=|\boldsymbol{h}_t|=1$)

HITS recursion

Equivalently, in matrix notation

$$\boldsymbol{a}_t \leftarrow \gamma \mathrm{A}^{\mathrm{T}} \boldsymbol{h}_{t-1}$$
$$\boldsymbol{h}_t \leftarrow \eta \mathrm{A} \boldsymbol{a}_t$$

(A : adjacency matrix of the base graph)

HITS scores: definition

HITS scores are defined as the limit of the recursive computation

$\boldsymbol{a} = \lim_{t \to \infty} \boldsymbol{a}_t$ … HITS authority vector

$\boldsymbol{h} = \lim_{t \to \infty} \boldsymbol{h}_t$ … HITS hub vector

HITS recursion

$$\boldsymbol{a}_t \leftarrow \gamma \mathrm{A}^{\mathrm{T}} \boldsymbol{h}_{t-1}$$
$$\boldsymbol{h}_t \leftarrow \eta \mathrm{A} \boldsymbol{a}_t$$

Hence

$$\boldsymbol{a}_t \leftarrow \gamma\eta \mathrm{A}^{\mathrm{T}} \mathrm{A} \boldsymbol{a}_{t-1}$$
$$\boldsymbol{h}_t \leftarrow \gamma\eta \mathrm{A}\mathrm{A}^{\mathrm{T}} \boldsymbol{h}_{t-1}$$

= Power method for computing dominant eigenvectors!

HITS = eigenvector computation

HITS authority vector $\boldsymbol{a}$

= dominant eigenvector of $\mathbf{A}^{\mathbf{T}}\mathbf{A}$

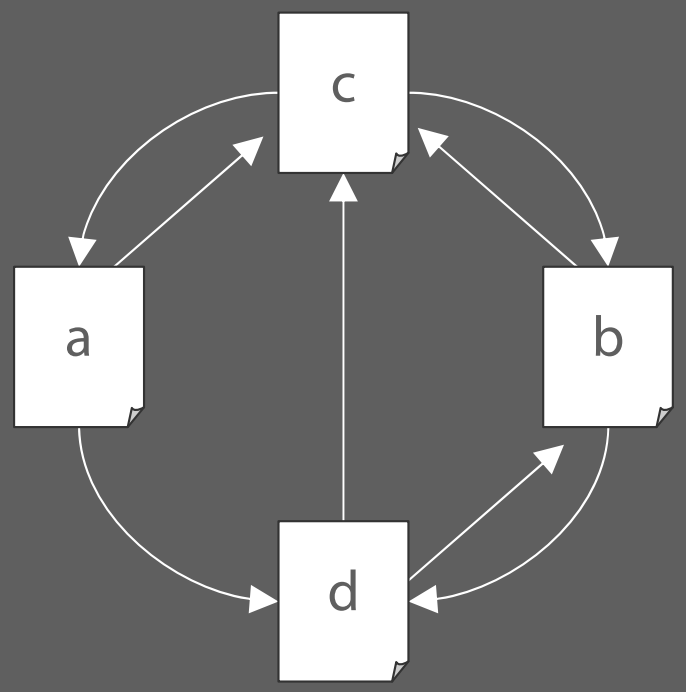
HITS hub vector $\boldsymbol{h}$

= dominant eigenvector of $\mathbf{A}\mathbf{A}^{\mathbf{T}}$

Note

$\mathbf{A}^{\mathbf{T}}\mathbf{A}$ = co-citation matrix

$\mathbf{A}\mathbf{A}^{\mathbf{T}}$ = bibliographic coupling matrix



Computation of HITS authority vector

adjacency matrix $A =$

$$
\begin{array}{c|c}
\text{from} \backslash \text{to} & \begin{array}{cccc} a & b & c & d \end{array} \\
\begin{array}{c} a \\ b \\ c \\ d \end{array} & \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix}
\end{array}
$$

A^TA = co-citation matrix

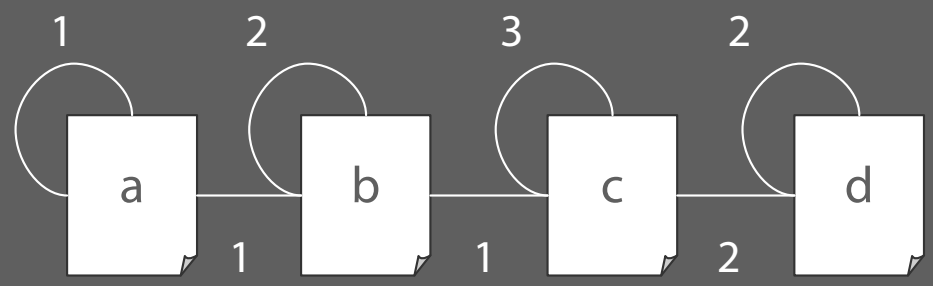
$$A^TA = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 3 & 2 \\ 0 & 0 & 2 & 2 \end{pmatrix} \end{array}$$

$$A = \begin{array}{c|c} \text{from} \backslash \text{to} & \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{array}{c} a \\ b \\ c \\ d \end{array} & \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{pmatrix} \end{array}$$

A^TA = co-citation matrix

$$A^TA = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 3 & 2 \\ 0 & 0 & 2 & 2 \end{pmatrix} \end{array}$$

co-citation graph



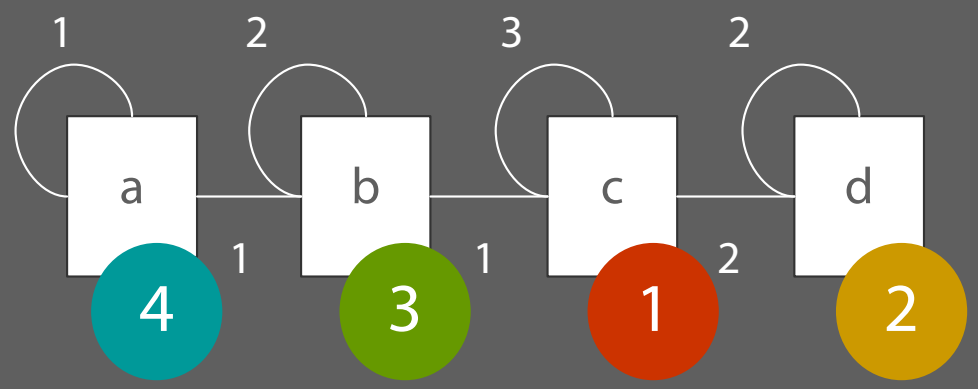
HITS authority = dominant eigenvector of A^TA

$$A^TA = \begin{array}{c} \\ a \\ b \\ c \\ d \end{array} \begin{array}{c} \begin{array}{cccc} a & b & c & d \end{array} \\ \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 0 & 1 & 3 & 2 \\ 0 & 0 & 2 & 2 \end{pmatrix} \end{array}$$

HITS authority vector

$$\begin{array}{c} a \\ b \\ c \\ d \end{array} \begin{pmatrix} 0.080 \\ 0.303 \\ 0.774 \\ 0.550 \end{pmatrix}$$

co-citation graph



Contents

1. Background
2. Link analysis preliminaries
3. Graph kernels and link analysis
4. Dealing with multiple communities in a graph

3. Graph kernels and link analysis

1. Typology of graph kernels
2. Graph kernels from link-analysis perspective
    1. von Neumann kernels
    2. regularized Laplacian
3. Graph Laplacian and its regularization
4. Application example

Kernel

Computes the inner product of examples in a (high dimensional) feature space, *without* actually computing the image of the examples in the feature space

Why kernels matter?

Because they provide a basic tool for computing various similarity metrics

Kernels

Some proposed kernels are applicable to non-vector data:

- strings
- trees
- graphs
- graph nodes
- ⋮

Graph kernels

Define inner products of nodes in a graph

➔ Induce similarity metric on nodes

But not all kernels provide a useful similarity measure!

Identity matrix, uniform matrix, …

➔ What about the graph kernels, especially in the context of link analysis?

Graph kernels

(Heat) diffusion kernels

[Chung 97, Kondor & Lafferty 02]

von Neumann kernels

[Kandola, Cristianini & Shawe-Taylor 03]

Regularized Laplacian

[Chebotarev & Shamis 97, Smola & Kondor 03]

Commute-time kernels

[Saerens, Fouss, Yen & Dupont 04]

⋮

Graph kernels

Based on adjacency matrix

von Neumann kernels

adjacency-matrix-based exponential kernels

Based on Laplacian

diffusion kernels

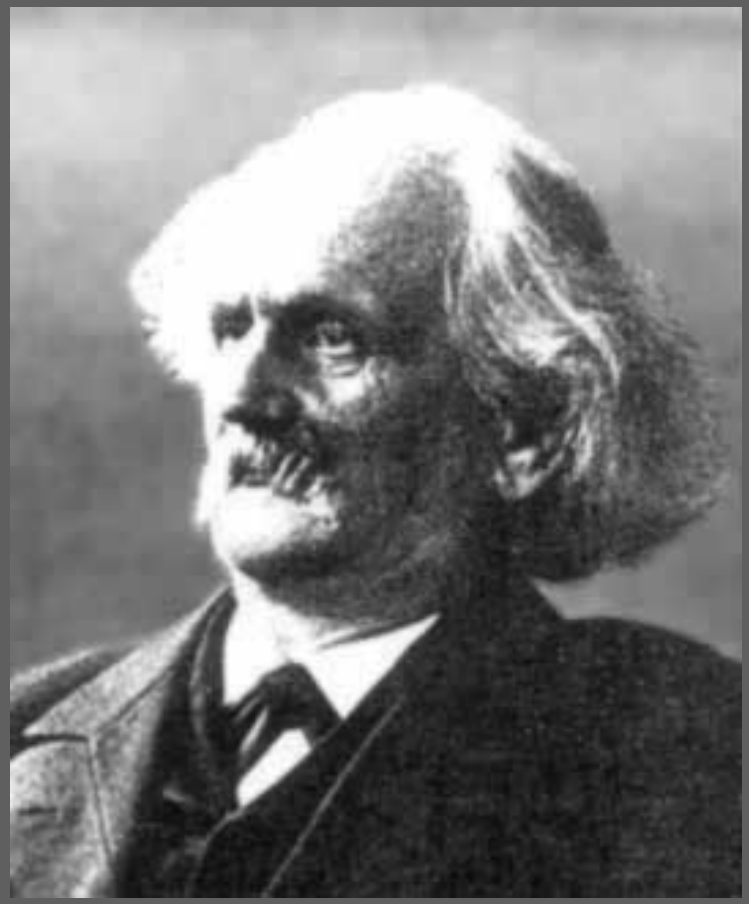
commute-time kernels

regularized Laplacian

3. Graph kernels and link analysis

1. Typology of graph kernels
2. Graph kernels from link-analysis perspective
    1. von Neumann kernels
    2. Regularized Laplacian
3. Graph Laplacian and its regularization
4. Application example

(von) Neumann kernels



von Neumann kernels

[Kandola et al., 2003]

von Neumann kernels were originally proposed as kernels for documents, computed from content words but not from citations/links

Two types of von Neumann kernels were defined

- kernels for words
- kernels for documents

von Neumann kernels

[Kandola et al., 2003]

Document similarity kernel

$$NK_\beta(X^TX) = X^TX\,(I - \beta X^TX)^{-1}$$
$$= X^TX + \beta(X^TX)^2 + \beta^2(X^TX)^3 + \ldots$$

Word similarity kernel

$$NK_\beta(XX^T) = XX^T\,(I - \beta XX^T)^{-1}$$
$$= XX^T + \beta(XX^T)^2 + \beta^2(XX^T)^3 + \ldots$$

X = Term-document matrix

β = "diffusion factor" parameter

Link analysis with von Neumann kernels

Apply von Neumann kernels to citation graph directly

Term-document matrix X ➔ Adjacency matrix A of a citation graph

$$NK_\beta(A^TA) = A^TA\,(I - \beta A^TA)^{-1}$$
$$= A^TA + \beta(A^TA)^2 + \beta^2(A^TA)^3 + \ldots$$
$$NK_\beta(AA^T) = AA^T\,(I - \beta AA^T)^{-1}$$
$$= AA^T + \beta(AA^T)^2 + \beta^2(AA^T)^3 + \ldots$$

Link analysis with von Neumann kernels

$$NK_\beta(A^TA) = A^TA + \beta(A^TA)^2 + \beta^2(A^TA)^3 + \ldots$$

$$NK_\beta(AA^T) = AA^T + \beta(AA^T)^2 + \beta^2(AA^T)^3 + \ldots$$

Link analysis with von Neumann kernels

$$NK_\beta(A^TA) = A^TA + \beta(A^TA)^2 + \beta^2(A^TA)^3 + \ldots$$

$$NK_\beta(AA^T) = AA^T + \beta(AA^T)^2 + \beta^2(AA^T)^3 + \ldots$$

What is the meaning of terms $(A^TA)^n$ and $(AA^T)^n$?

$(A^T A)^n$

(i, j)-element of $(A^T A)^n$

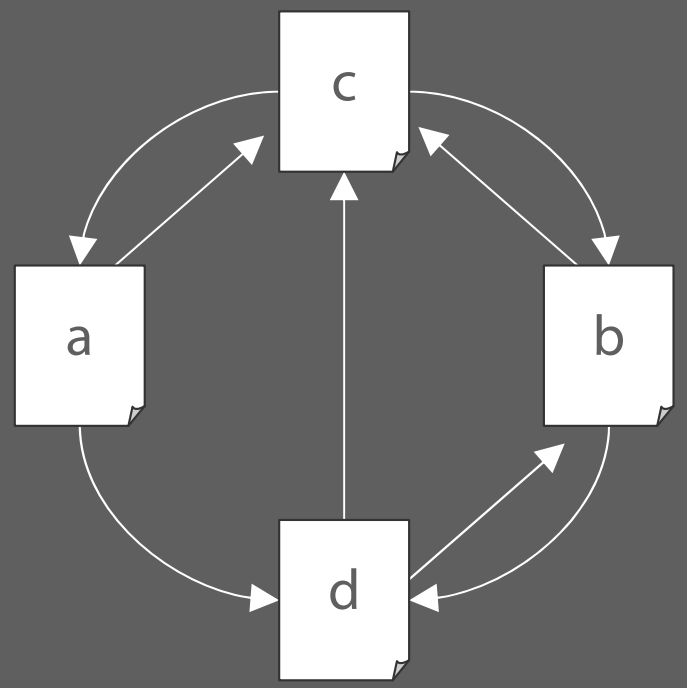
Number of paths of length n from node i to node j in co-citation graph

Interpretation in terms of link analysis

As n grows from 1 towards ∞

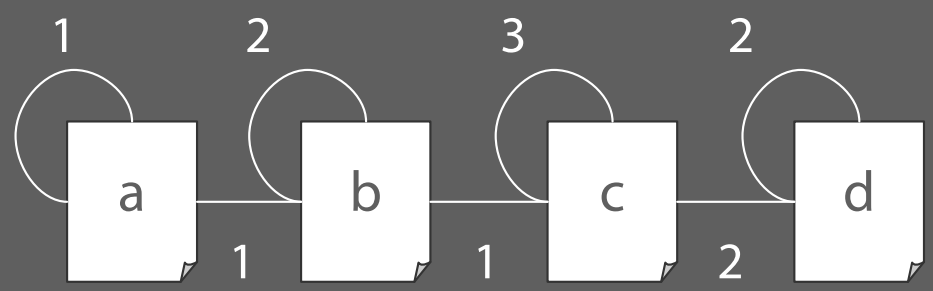
$(A^T A)^n$ changes its property from (local) relatedness towards (global) importance

Powers of A^TA



Original
citation graph

Induced
co-citation graph



$(A^TA)^1$ = co-citation matrix

$(A^TA)^1 =$

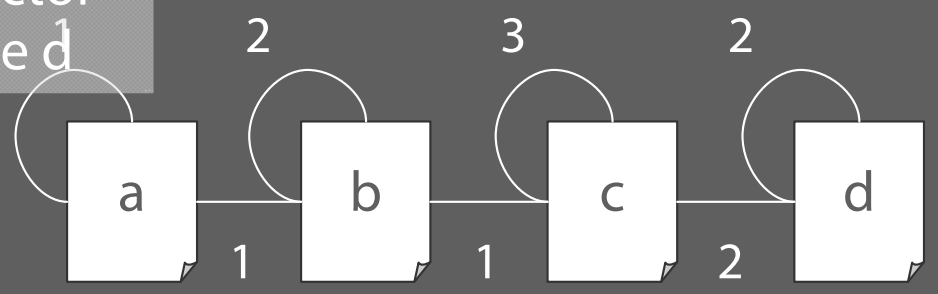
	a	b	c	d
a	1	1	0	0
b	1	2	1	0
c	0	1	3	2
d	0	0	2	2

score vector for node a

score vector for node b

score vector for node c

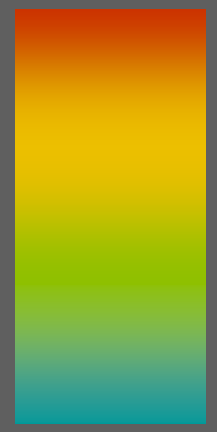
score vector for node d

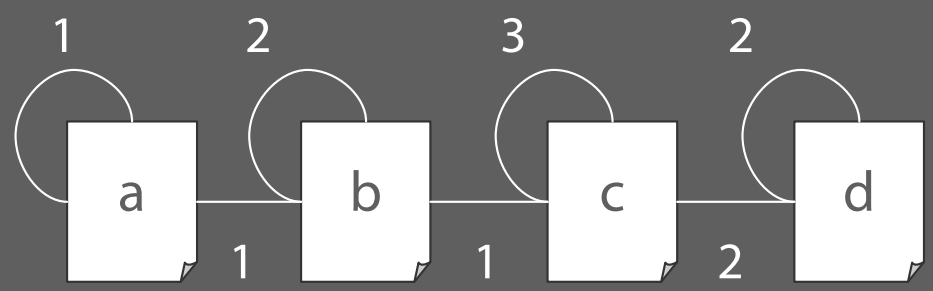


$(A^TA)^1$ = co-citation matrix

$(A^TA)^1 =$

	a	b	c	d
a	1	1	0	0
b	1	2	1	0
c	0	1	3	2
d	0	0	2	2

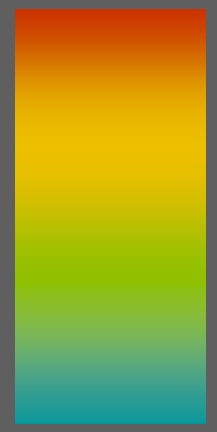


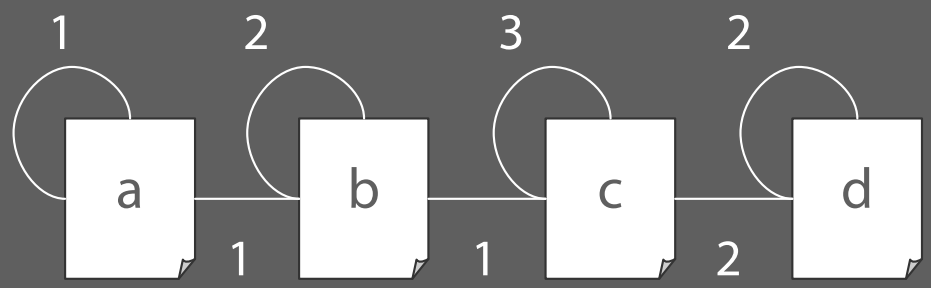


$(A^TA)^2$ = numbers of paths of length 2

$(A^TA)^2 =$

	a	b	c	d
a	2	3	1	0
b	3	6	5	2
c	1	5	14	10
d	0	2	10	8

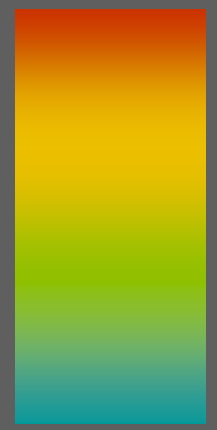


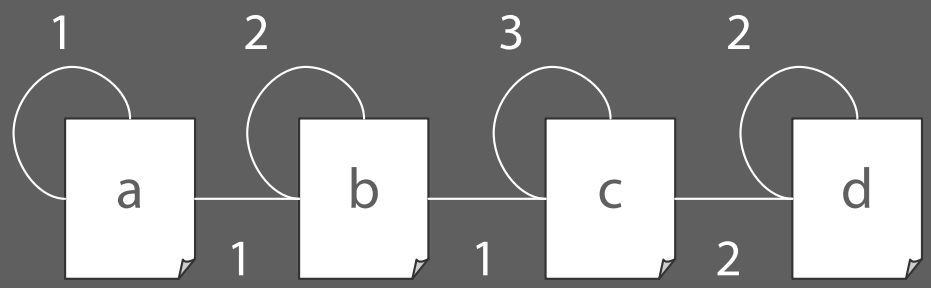


$(A^TA)^3$ = numbers of paths of length 3

$(A^TA)^3 =$

	a	b	c	d
a	5	9	6	2
b	9	20	25	14
c	6	25	67	48
d	2	14	48	36

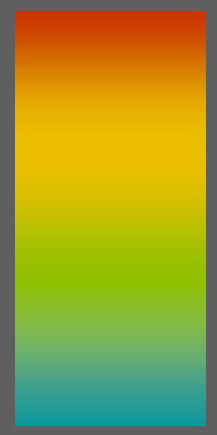


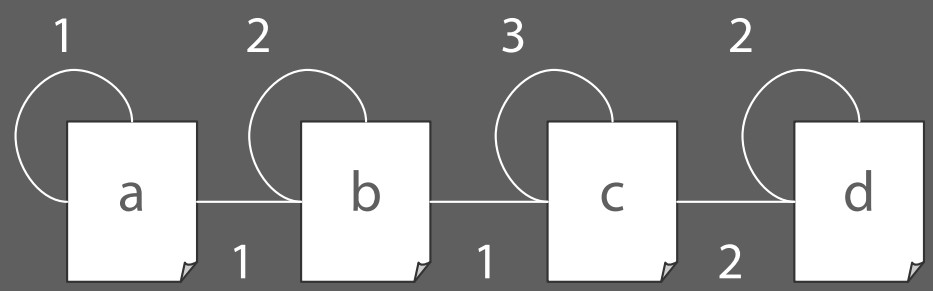


$(A^TA)^4$ = numbers of paths of length 4

$(A^TA)^4 =$

	a	b	c	d
a	14	29	31	16
b	29	74	123	78
c	31	123	322	230
d	16	78	230	168

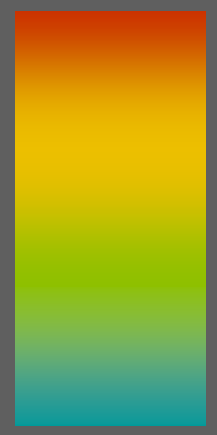


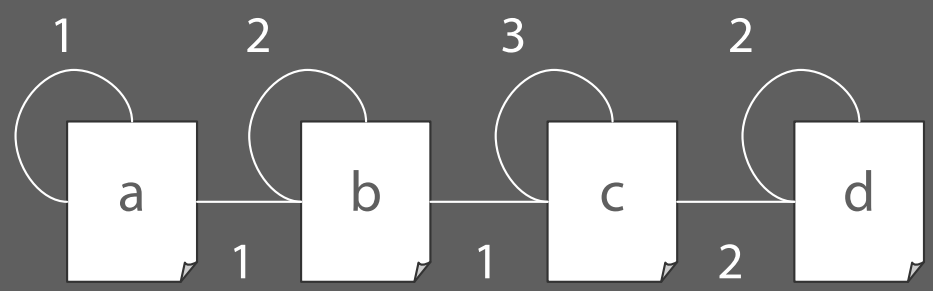


$(A^TA)^5$ = numbers of paths of length 5

$(A^TA)^5 =$

	a	b	c	d
a	146	403	753	496
b	403	1302	2901	2003
c	753	2901	7454	5306
d	496	2003	5306	3800



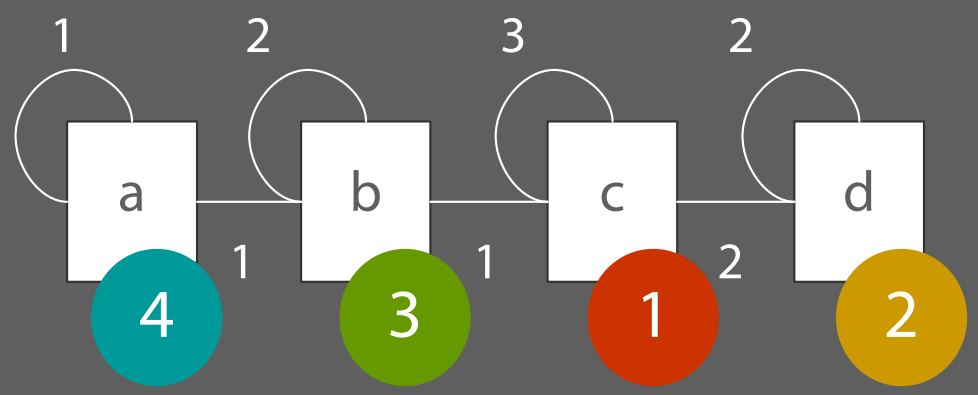


$(A^TA)^5$ = numbers of paths of length 5

$(A^TA)^5 =$

	a	b	c	d
a	146	403	753	496
b	403	1302	2901	2003
c	753	2901	7454	5306
d	496	2003	5306	3800

For all nodes, the ranking is identical to HITS



$\lim_{n\to\infty}(A^TA)^n$ = HITS authority ranking

Theorem

$$\left(\frac{A^TA}{\lambda}\right)^n \to xx^T \quad \text{as } n \to \infty$$

λ = dominant eigenvalue of A^TA

x = dominant eigenvector of A^TA

(= HITS authoritativeness vector)

Every row/column of xx^T gives a ranking identical to HITS

$$\left(A^T A\right)^n = \sum_k \lambda_k^n x_k x_k^T$$

$$= \lambda_1^n \left(x_1 x_1^T + \sum_{k \neq 1} \left(\frac{\lambda_k}{\lambda_1} \right)^n x_k x_k^T \right)$$

vanish!

λ_i = i-th dominant eigenvalue of $A^T A$

x_i = eigenvector corresponding to λ_i

von Neumann kernels

Weighted sum of all path weights between nodes

A mixture of relatedness and importance

$$NK_\beta(A^TA) = A^TA + \beta(A^TA)^2 + \beta^2(A^TA)^3 + \ldots$$

Relatedness ⟷ Importance

von Neumann kernels

A mixture of relatedness and importance

$$NK_\beta(A^TA) = A^TA + \beta(A^TA)^2 + \beta^2(A^TA)^3 + \ldots$$

β small ➔ measure biased towards relatedness

In particular, $\beta = 0$ ➔ $NK_\beta(A^TA)$ = co-citation matrix

β large ➔ measure biased towards importance

Indeed, at the limit of $\beta \rightarrow 1/\lambda$,

$NK_\beta(A^TA)$ in the limit = HITS authoritativeness

Theorem

$$\left(\frac{1}{\lambda}-\beta\right)NK_\beta(A^TA) \to xx^T \quad \text{as } \beta \to \frac{1}{\lambda}-0$$

λ = Dominant eigenvalue of A^TA

x = Eigenvector corresponding to λ

(= HITS authority vector)

Similar relation holds for

- bibliographic coupling matrix AA^T
- HITS hub vector
- the von Neumann kernel $NK(AA^T)$

von Neumann kernels: summary

Existing link analysis measures

Co-citation/bibliographic coupling "relatedness"

HITS authority/hub "importance"

von Neumann kernels = unified framework

$0 < \beta < 1/\lambda$ ➔ intermediate between relatedness and importance

Relative importance
[White & Smyth, 2003]

"Importance of nodes relative to a particular node"

von Neumann kernels

Intermediate between relatedness and importance

➔ vNK $\in$ Relative importance measures

Adjacency-based exponential kernels

$$K_\beta(A^T A) = \exp(\beta A^T A)$$
$$= I + \beta A^T A + \frac{1}{2!}(\beta A^T A)^2 + \frac{1}{3!}(\beta A^T A)^3 + \cdots$$

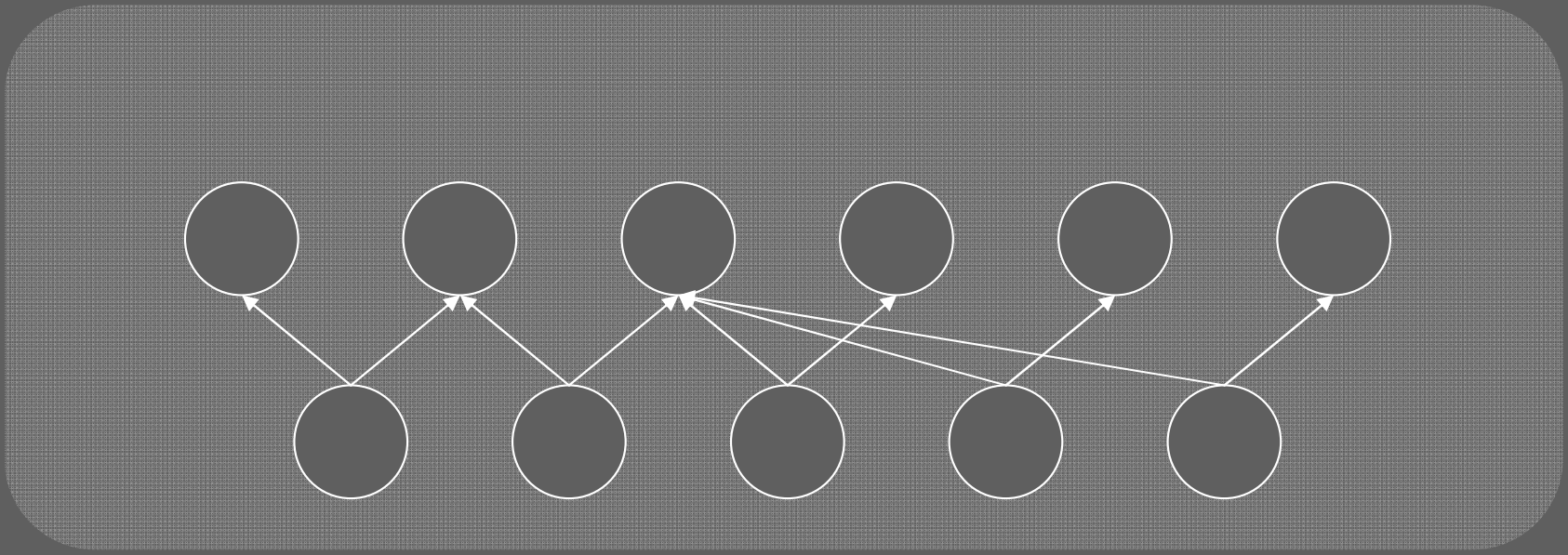
Also converges to xx^T as $\beta \to \infty$ with appropriate scaling

3. Graph kernels and link analysis

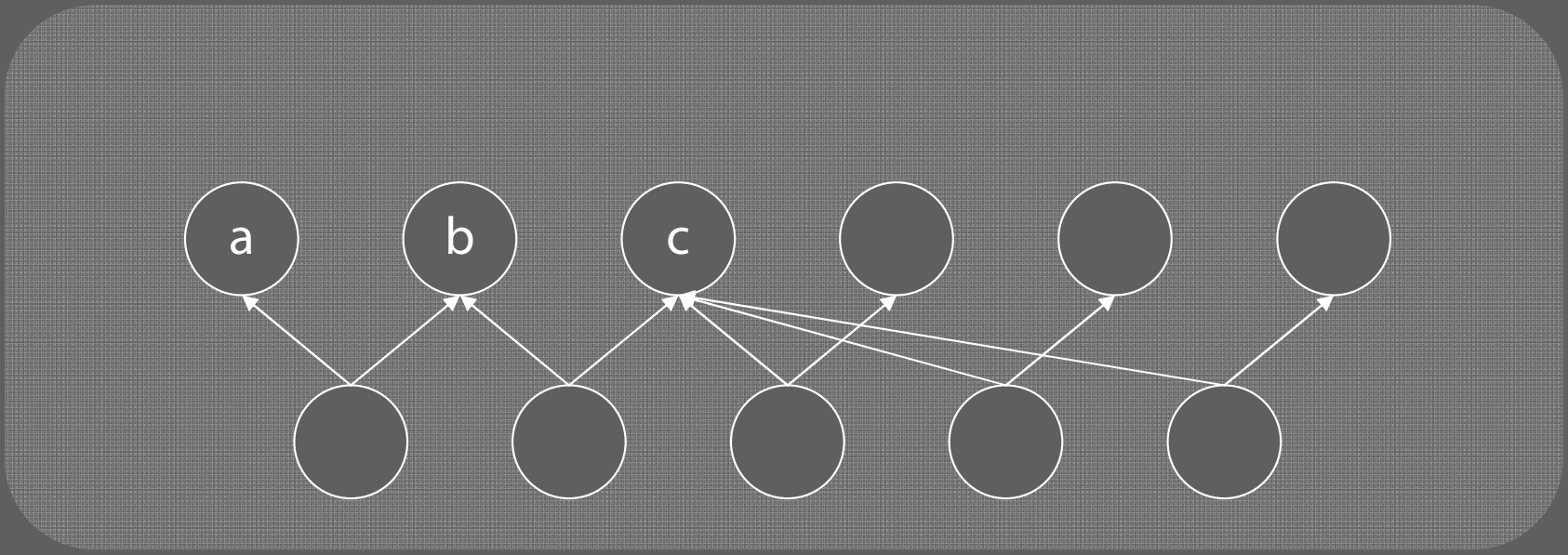
1. Typology of graph kernels
2. Graph kernels from link-analysis perspective
    1. von Neumann kernels
    2. Regularized Laplacian
3. Graph Laplacian and its regularization
4. Application example

Regularized Laplacian

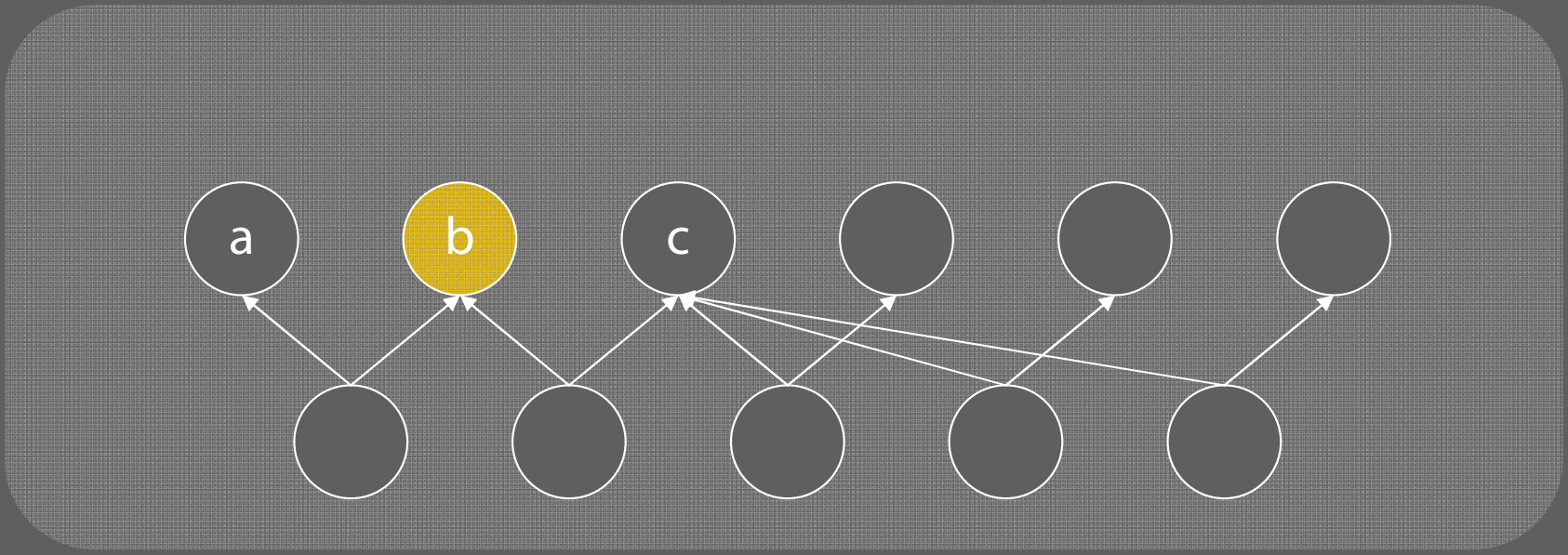
Quiz



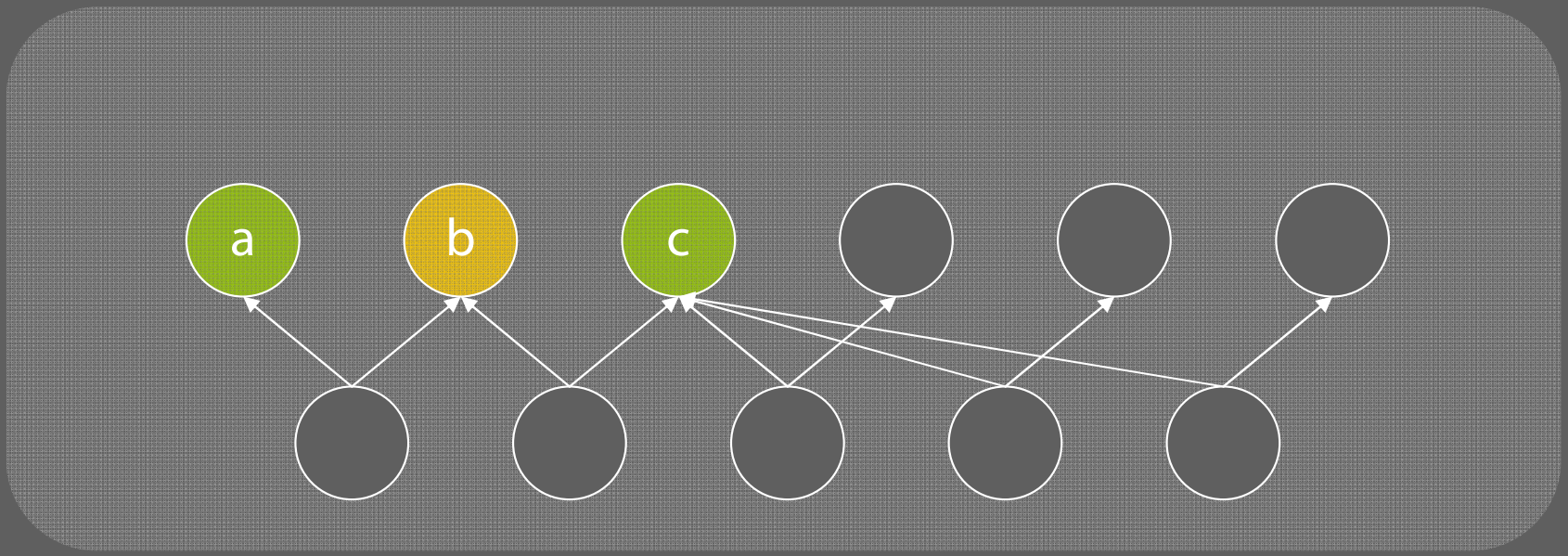
Quiz



Quiz



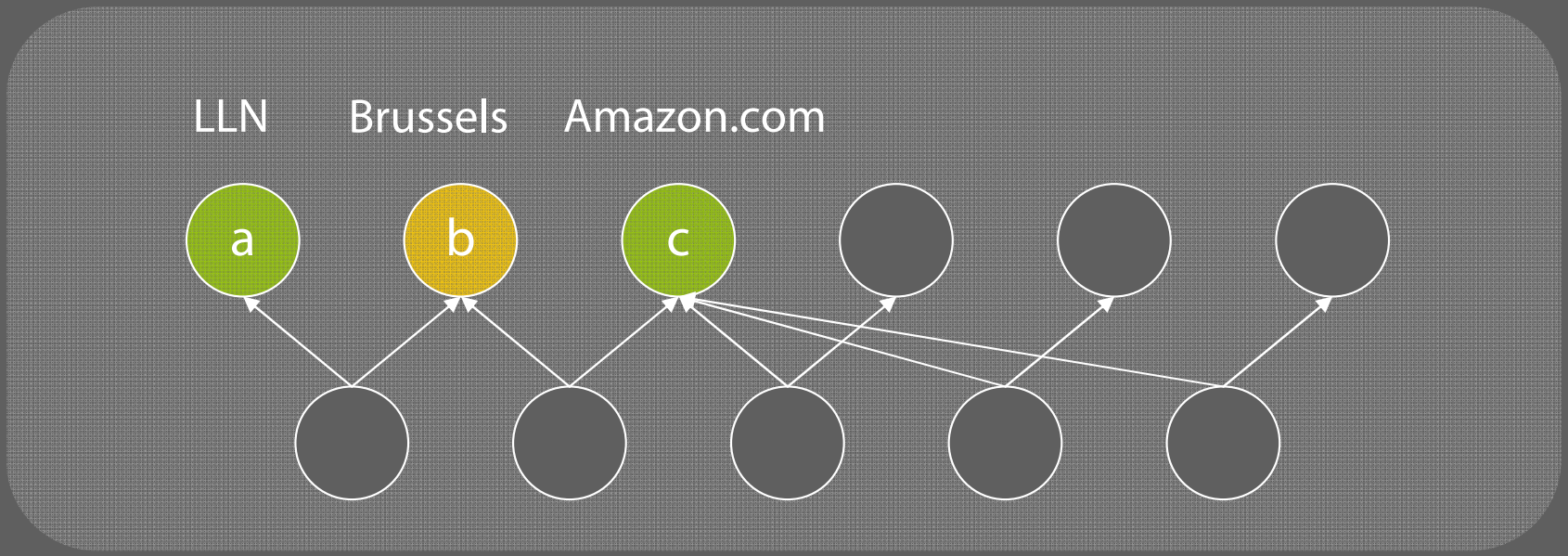
Quiz



Which node is more related to b?

a or c?

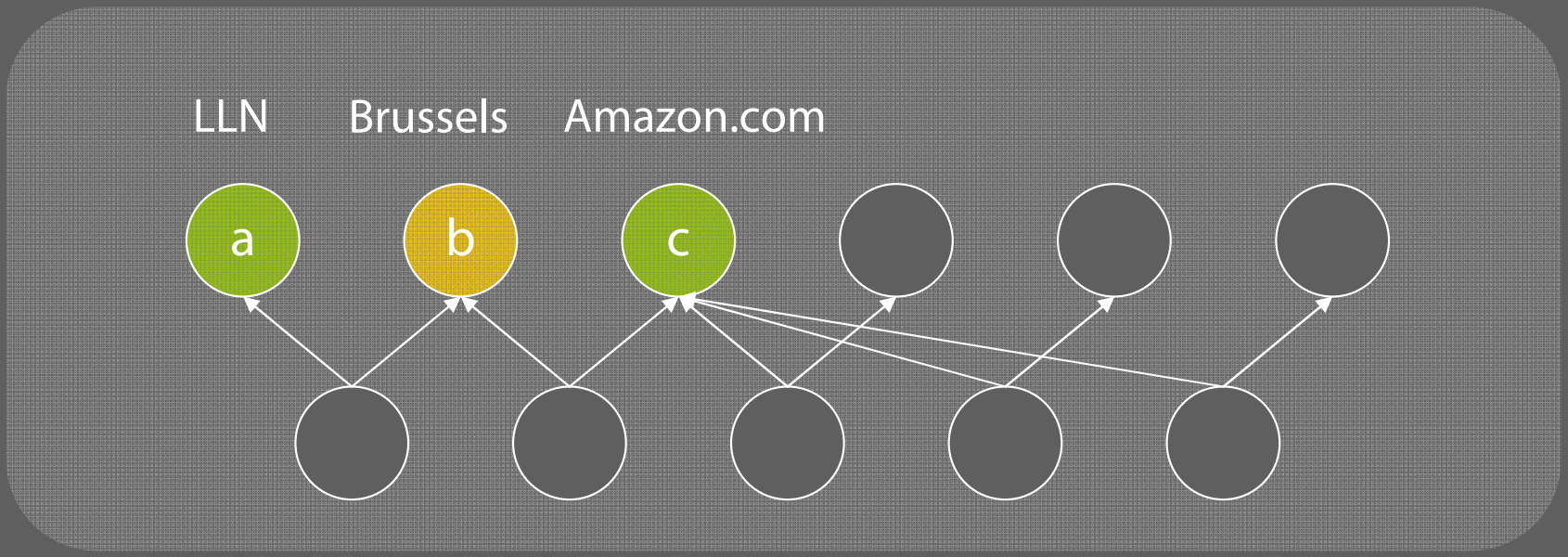
Suppose these are web pages



Which node is more related to b?

a or c?

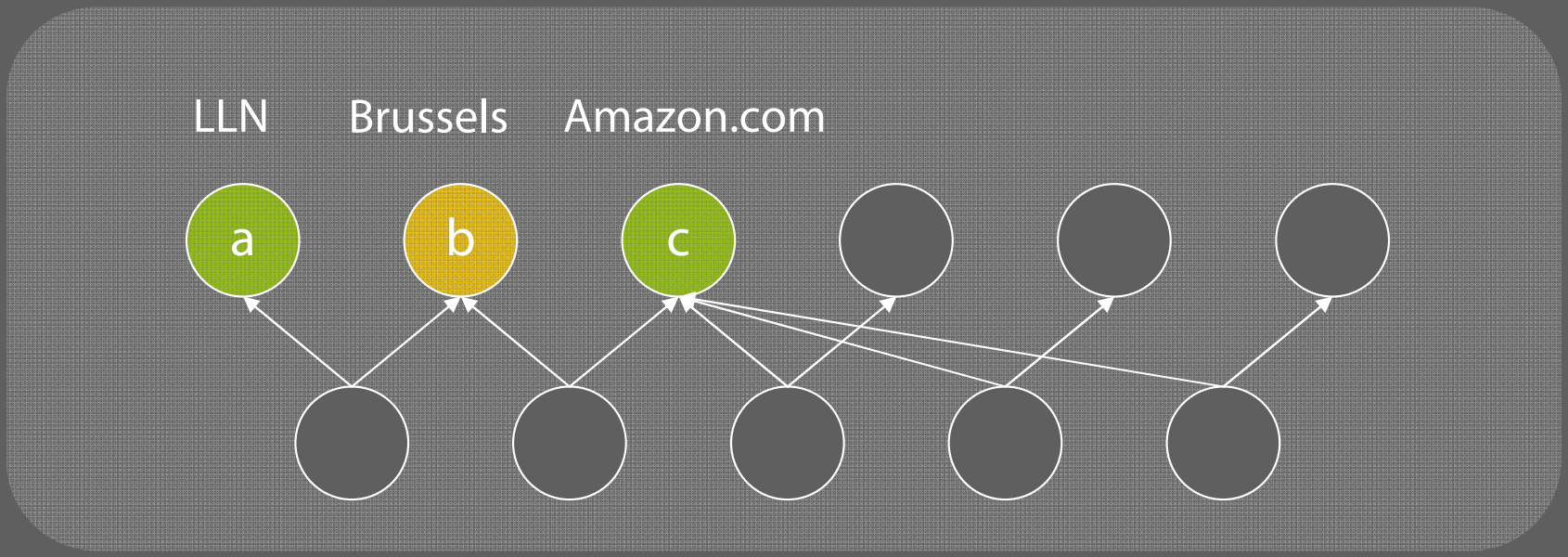
Intuitively,



relatedness(a, b) > relatedness(c, b)

- c (Amazon) is co-cited with many other pages - which probably have nothing to do with Belgian cities

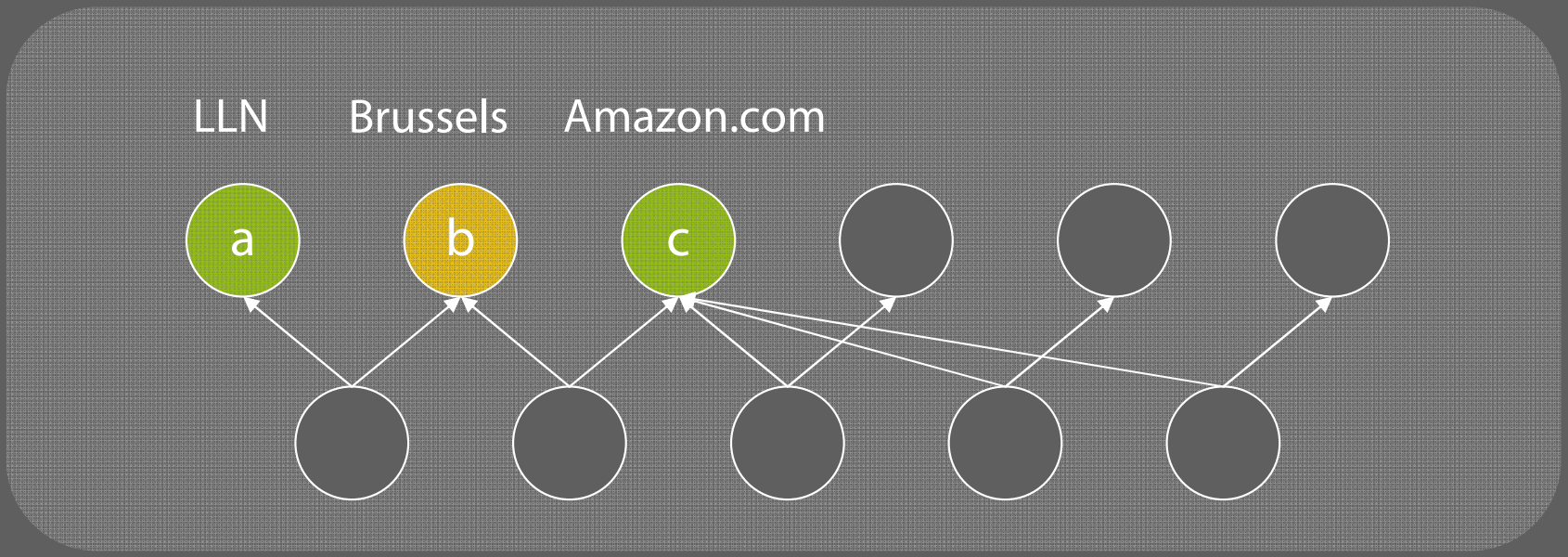
Output of co-citation coupling



Co-citation(a, b) = Co-citation(c, b)

- Both a (LLN) and c (Amazon) have one co-citation with b (Brussels)
- Ignores the number of other citations to c (Amazon)

von Neumann kernels



$vNK_\beta[c, b] > vNK_\beta[a, b]$ whenever $\beta > 0$

- c (Amazon) is judged to be more relevant than a (LLN) !
- Because k has higher global importance (HITS ranking)

Regularized Laplacian

[Chebotarev and Shamis 97, Smola and Kondor 03]

$$RL_\beta(A^TA) = (I + \beta L(A^TA))^{-1}$$

$L(A^TA)$ = Laplacian of a co-citation graph

$= D(A^TA) - A^TA$

$D(A^TA)$ = degree matrix of A^TA

diagonal matrix with (i, i)-element = degree of node i in co-citation graph (sum of ith row of A^TA)

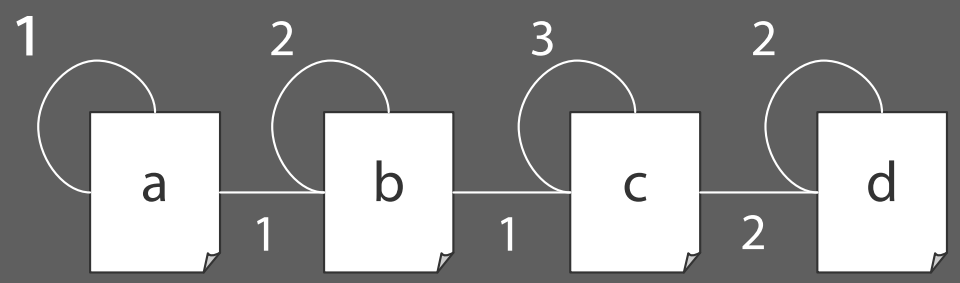
Graph Laplacian: an interpretation

Change the weight of self-loops

Sum weights of all non-loop edges incident to the node, and negate the sign

Let the adjacency matrix of this matrix be M – this is the negated Laplacian.

And the Laplacian is $\mathrm{L} = -\mathrm{M}$.



Original (co-citation) graph

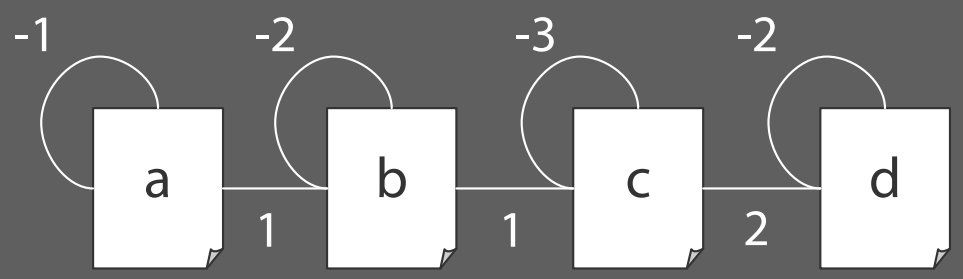
Graph Laplacian: an interpretation

Change the weight of self-loops

Sum weights of all non-loop edges incident to the node, and negate the sign

Let the adjacency matrix of this matrix be M – this is the negated (combinatorial) Laplacian.

And the (combinatorial) Laplacian is $\mathrm{L} = -\mathrm{M}$.



Graph representation of a negative Laplacian

Regularized Laplacian

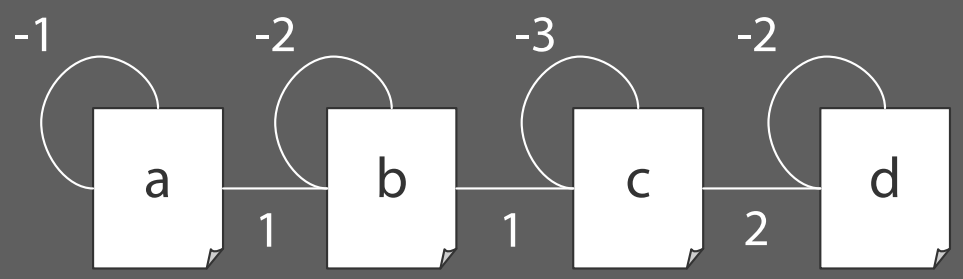
Scoring scheme is identical to von Neumann kernels

= Weighted sum (Neumann series) of path weights

Nodes with many edges (important nodes)

➡ Large discount

Does not tend to importance even if β is increased



Graph representation of a negative Laplacian

Link analysis with regularized Laplacian

$$\begin{aligned} RL_\beta(A^TA) &= (I + \beta L(A^TA))^{-1} \\ &= I - \beta L(A^TA) + \beta^2 L(A^TA)^2 - \beta^3 L(A^TA)^3 + \ldots \end{aligned}$$

Although $-L(A^TA)$ contains negative elements, it can be shown that $RL_\beta(A^TA)$ is non-negative

Hence each row/column of the regularized Laplacian can be treated as a score vector (just like co-citation/vNK)

There must be more simple methods!

E.g., normalize edge weights with number of out-links before applying von Neumann kernel

β small ➔ OK, it works!

β large ➔ Unintuitive ranking

Important nodes get too much discount

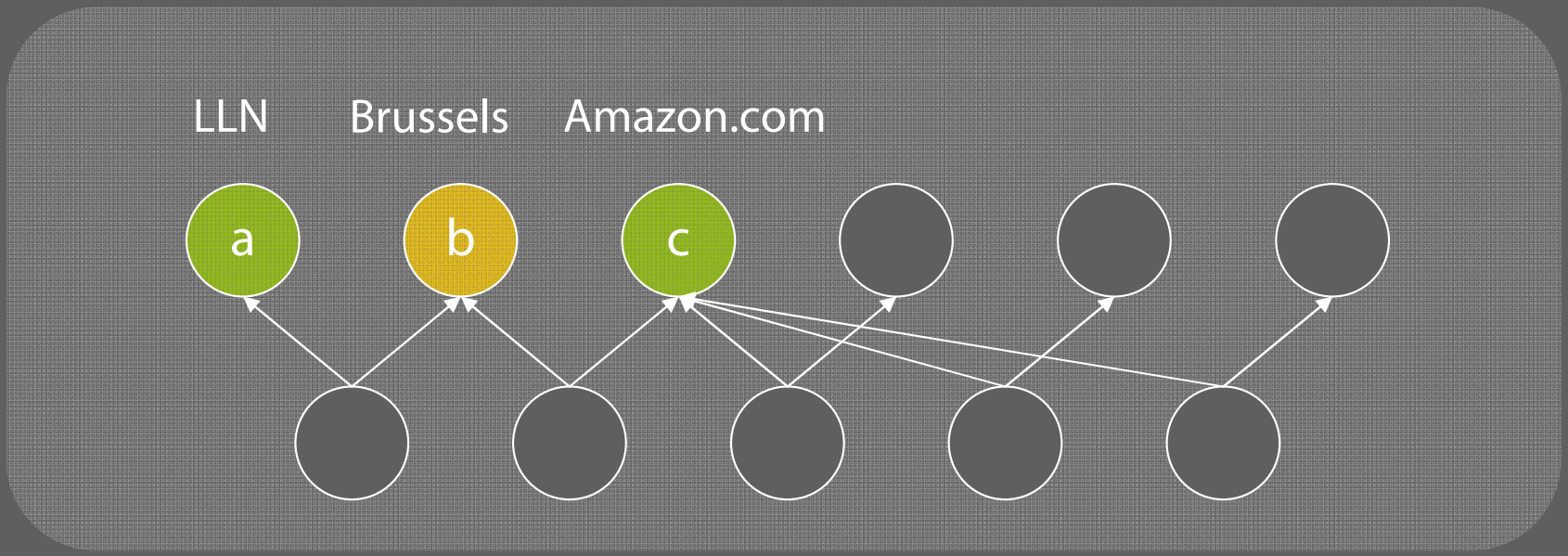
Regularized Laplacian

In the limit $\beta \rightarrow \infty$ uniform matrix

= trivial, but still a relatedness matrix

(which regards all nodes as equally related)

Output of Regularized Laplacian



$RL_\beta(A^TA)[a, b] > RL_\beta(A^TA)[c, b]$

- Matches our intuition

Other Laplacian-based kernels

All Laplacian-based kernels enjoy favorable property as a relatedness measure

- E.g., parametric kernels like regularized Laplacian and diffusion kernels all tend to a uniform matrix as β tends to ∞

β controls how much the kernel should discount distant node

Summary: regularized Laplacian

Gives a relatedness measure regardless of parameter β

β controls how much the kernel should discount distant nodes

$\beta=0$	only consider the node itself to be related
$\beta\rightarrow\infty$	all nodes are equally related

3. Graph kernels and link analysis

1. Typology of graph kernels
2. Graph kernels from link-analysis perspective
    1. von Neumann kernels
    2. regularized Laplacian
3. Graph Laplacian and its regularization
4. Application example

Graph Laplacian and its regularization

Connection to “ordinary” Laplacian and ideas behind Laplacian-based graph kernels

Laplacian ∇^2 in classical physics

x, y, z : coordinates

$$\nabla^2 = \frac{d^2}{dx^2} + \frac{d^2}{dy^2} + \frac{d^2}{dz^2}$$

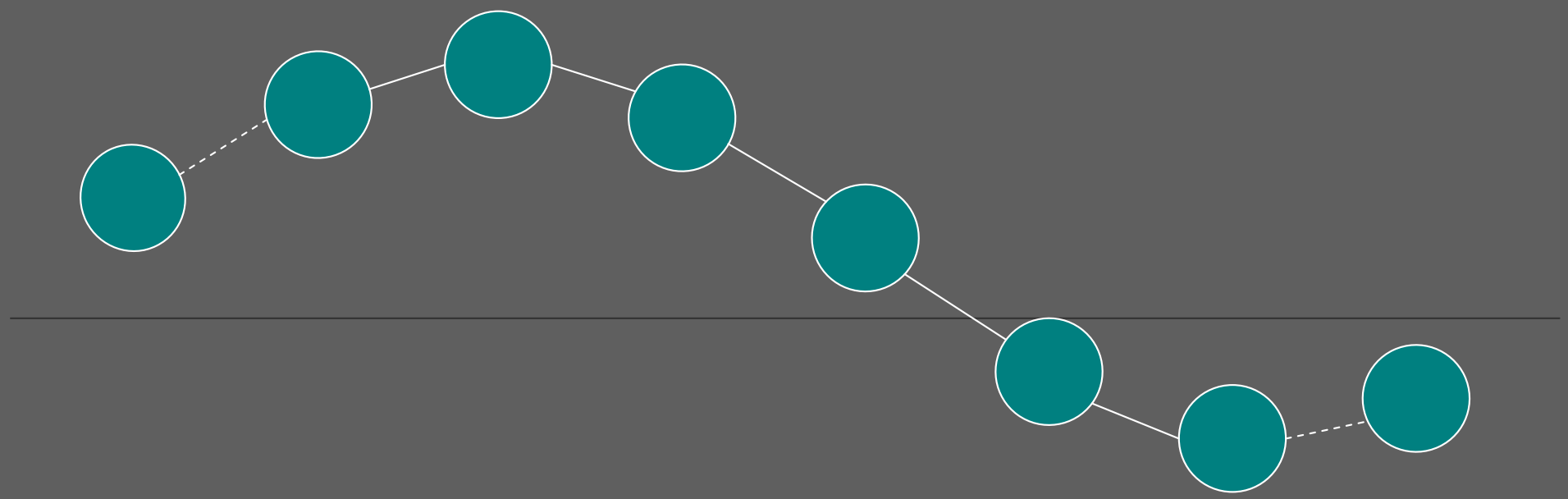
Why is matrix L called "Laplacian"?

$$L = D - A$$

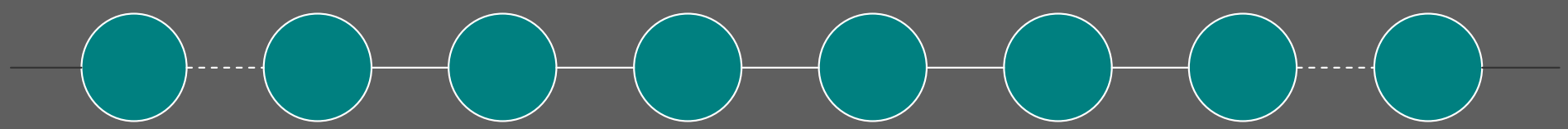
$$\Updownarrow$$

$$\nabla^2 = \frac{d^2}{dx^2} + \frac{d^2}{dy^2} + \frac{d^2}{dz^2}$$

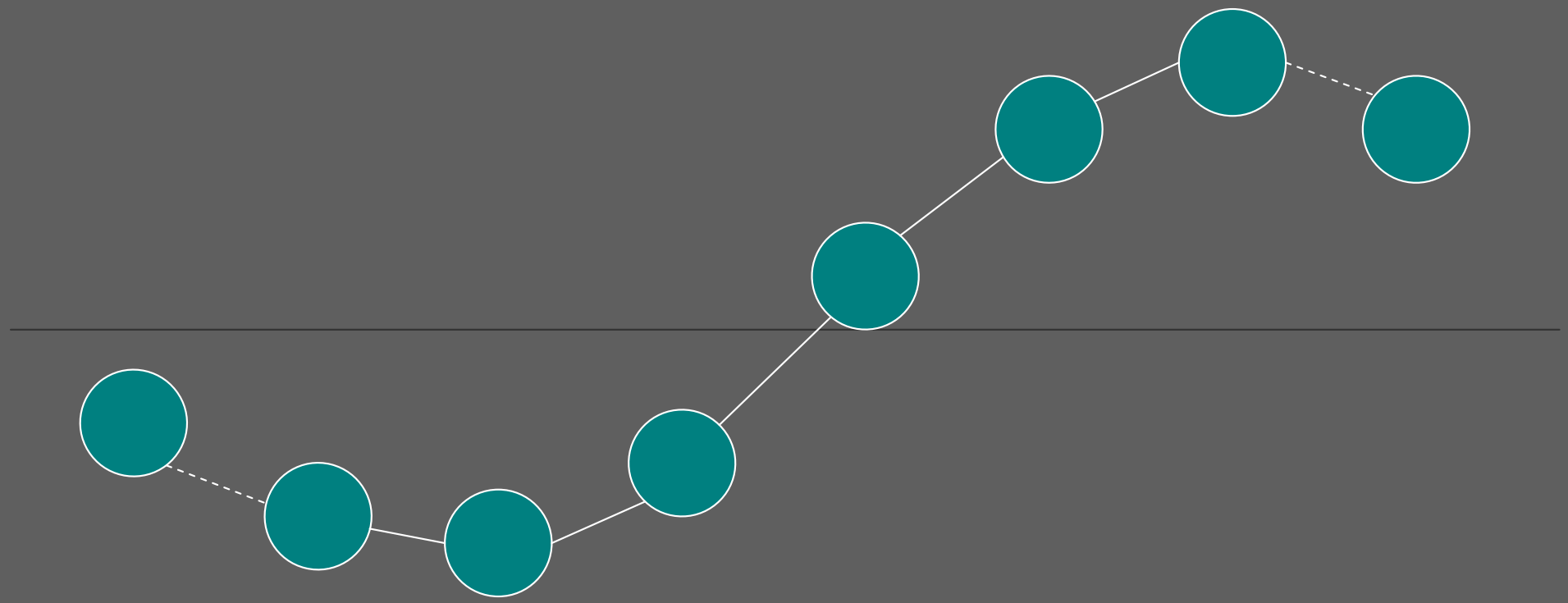
Vibrating beads threaded with a string



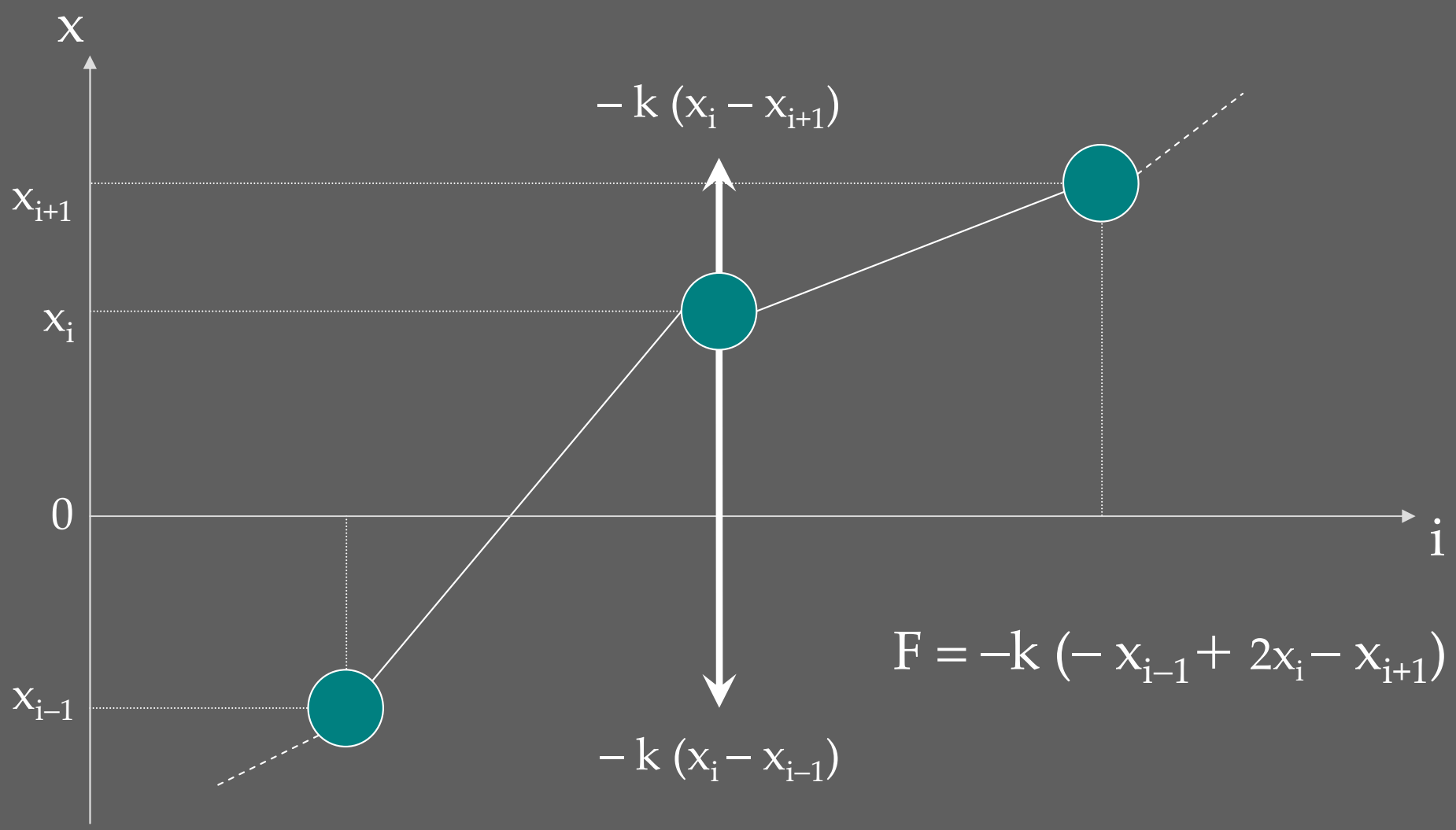
Vibrating beads threaded with a string



Vibrating beads threaded with a string



Force on i-th bead



$$F = -k(-x_{i-1} + 2x_i - x_{i+1})$$

Force on i-th bead

$$F = m\,(d^2x_i/dt^2) = -k\,(-x_{i-1} + 2x_i - x_{i+1})$$

$$(d^2x_i/dt^2) = -(k/m)\,(-x_{i-1} + 2x_i - x_{i+1})$$

For simplicity let $\sigma = k/m$

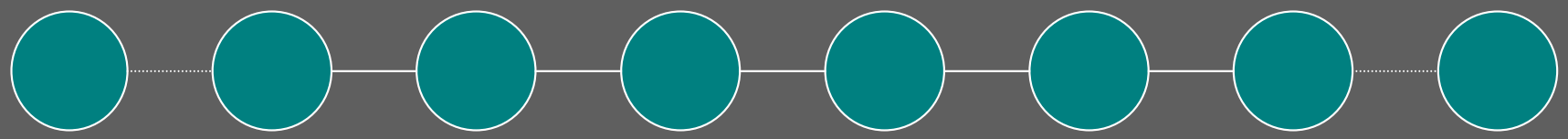
$$(d^2x_i/dt^2) = -\sigma(-x_{i-1} + 2x_i - x_{i+1})$$

Matrix notation

$$(d^2x_i/dt^2) = -\sigma\,(-\,x_{i-1} + 2x_i - x_{i+1})$$

$$\frac{d^2}{dt^2}\begin{pmatrix} \vdots \\ x_{i-1} \\ x_i \\ x_{i+1} \\ \vdots \end{pmatrix} = -\sigma \begin{pmatrix} \ddots & \ddots & \ddots & & & & 0 \\ & -1 & 2 & -1 & & & \\ & & -1 & 2 & -1 & & \\ & & & -1 & 2 & -1 & \\ 0 & & & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} \vdots \\ x_{i-1} \\ x_i \\ x_{i+1} \\ \vdots \end{pmatrix}$$

Threaded beads = graph (chain)



Adjacency matrix A

$$\begin{pmatrix} 0 & 1 & & & & & & & \\ 1 & 0 & 1 & & & & & & \\ & 1 & 0 & 1 & & & & & \\ & & 1 & 0 & 1 & & & & \\ & & & 1 & 0 & 1 & & & \\ & & & & 1 & 0 & 1 & & \\ & & & & & 1 & 0 & 1 & \\ & & & & & & 1 & 0 & 1 \\ & & & & & & & 1 & 0 \end{pmatrix}$$

Degree diagonal matrix D

$$\begin{pmatrix} 1 & & & & & & & & & 0 \\ & 2 & & & & & & & & \\ & & 2 & & & & & & & \\ & & & 2 & & & & & & \\ & & & & 2 & & & & & \\ & & & & & 2 & & & & \\ & & & & & & 2 & & & \\ & & & & & & & 2 & & \\ 0 & & & & & & & & 2 & \\ & & & & & & & & & 1 \end{pmatrix}$$

Laplacian matrix L=D–A

$$
\begin{pmatrix}
1 & -1 & & & & & & & & \\
-1 & 2 & -1 & & & & & & 0 & \\
 & -1 & 2 & -1 & & & & & & \\
 & & -1 & 2 & -1 & & & & & \\
 & & & -1 & 2 & -1 & & & & \\
 & & & & -1 & 2 & -1 & & & \\
 & & & & & -1 & 2 & -1 & & \\
 & & & & & & -1 & 2 & -1 & \\
 & 0 & & & & & & -1 & 2 & -1 \\
 & & & & & & & & -1 & 1
\end{pmatrix}
$$

Matrix notation

$$(d^2x_i/dt^2) = -\sigma\,(-\,x_{i-1} + 2x_i - x_{i+1})$$

$$\frac{d^2}{dt^2}\begin{pmatrix} \vdots \\ x_{i-1} \\ x_i \\ x_{i+1} \\ \vdots \end{pmatrix} = -\sigma \begin{pmatrix} \ddots & \ddots & \ddots & & & & 0 \\ & -1 & 2 & -1 & & & \\ & & -1 & 2 & -1 & & \\ & & & -1 & 2 & -1 & \\ 0 & & & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} \vdots \\ x_{i-1} \\ x_i \\ x_{i+1} \\ \vdots \end{pmatrix}$$

Matrix notation

$$(d^2x_i/dt^2) = -\sigma\,(-x_{i-1} + 2x_i - x_{i+1})$$

$$\frac{d^2}{dt^2}\begin{pmatrix} \vdots \\ x_{i-1} \\ x_i \\ x_{i+1} \\ \vdots \end{pmatrix} = -\sigma \begin{pmatrix} \ddots & \ddots & \ddots & & & & 0 \\ & -1 & 2 & -1 & & & \\ & & -1 & 2 & -1 & & \\ & & & -1 & 2 & -1 & \\ 0 & & & & \ddots & \ddots & \ddots \end{pmatrix} \begin{pmatrix} \vdots \\ x_{i-1} \\ x_i \\ x_{i+1} \\ \vdots \end{pmatrix}$$

$$\ddot{\boldsymbol{x}} = -\sigma\,\mathbf{L}\,\boldsymbol{x}$$

Matrix notation

$$\frac{\partial^2}{\partial t^2}\boldsymbol{x} = -\sigma L \boldsymbol{x}$$

Wave equation

$$\frac{\partial^2}{\partial t^2} x = -\sigma L x$$

$$\frac{\partial^2}{\partial t^2} x = \sigma \nabla^2 x$$

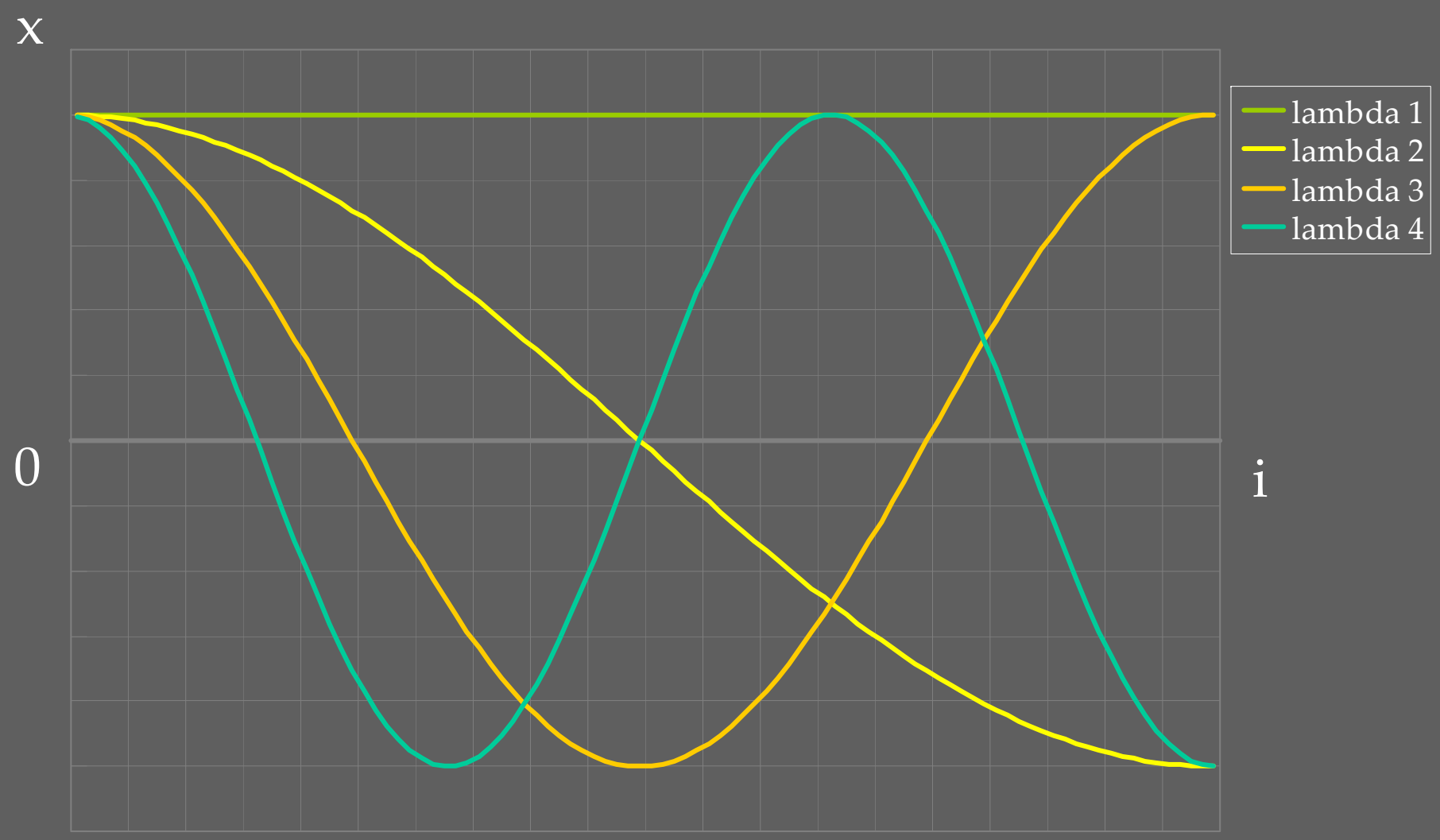
The sign is reversed – it's how the graph Laplacian is defined
(To make it positive semidefinite)

Modes of vibration

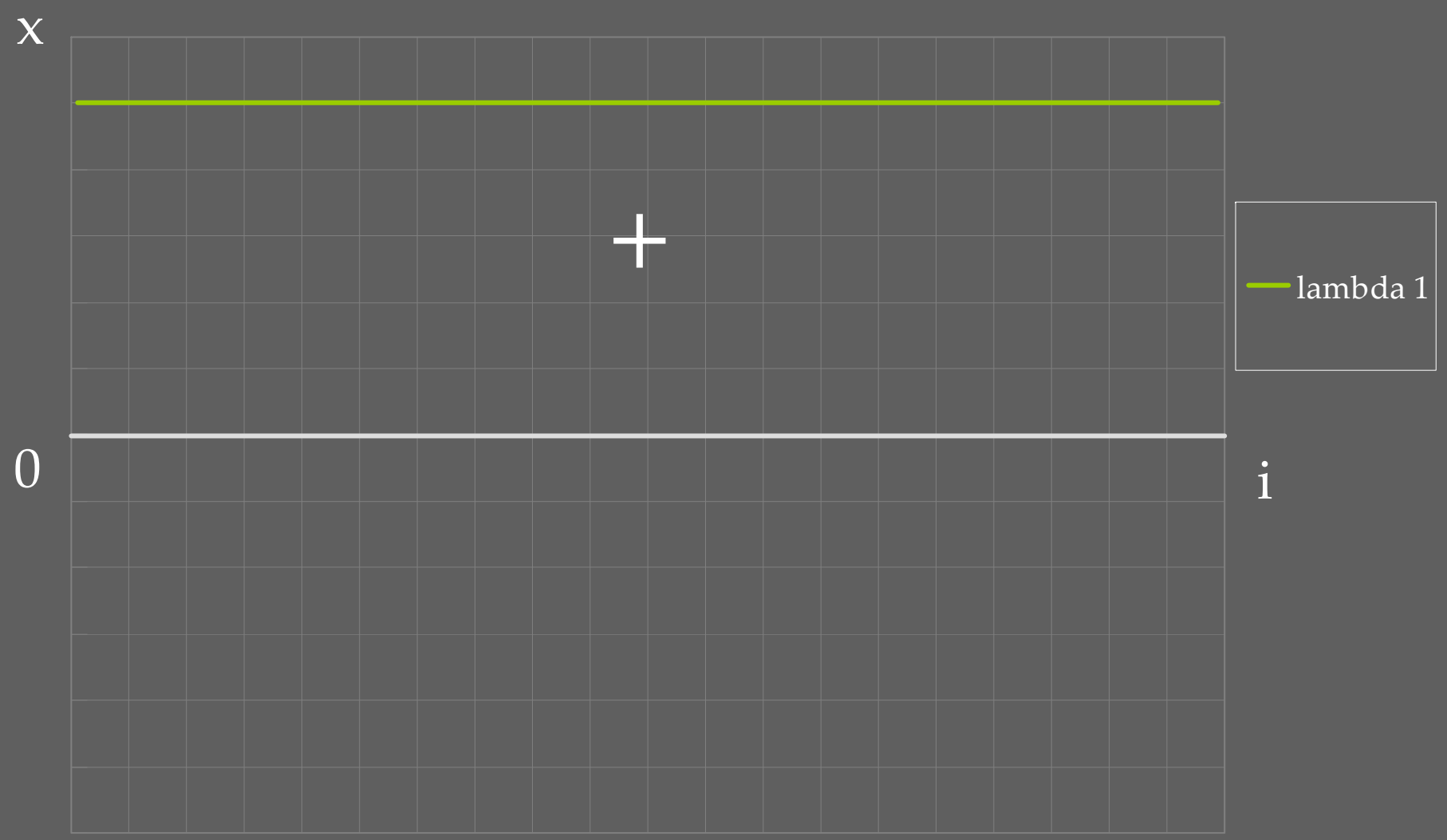
Eigensystem λ of Laplacian give the modes of vibration

$$\mathrm{L}\boldsymbol{x} = \lambda \boldsymbol{x}$$

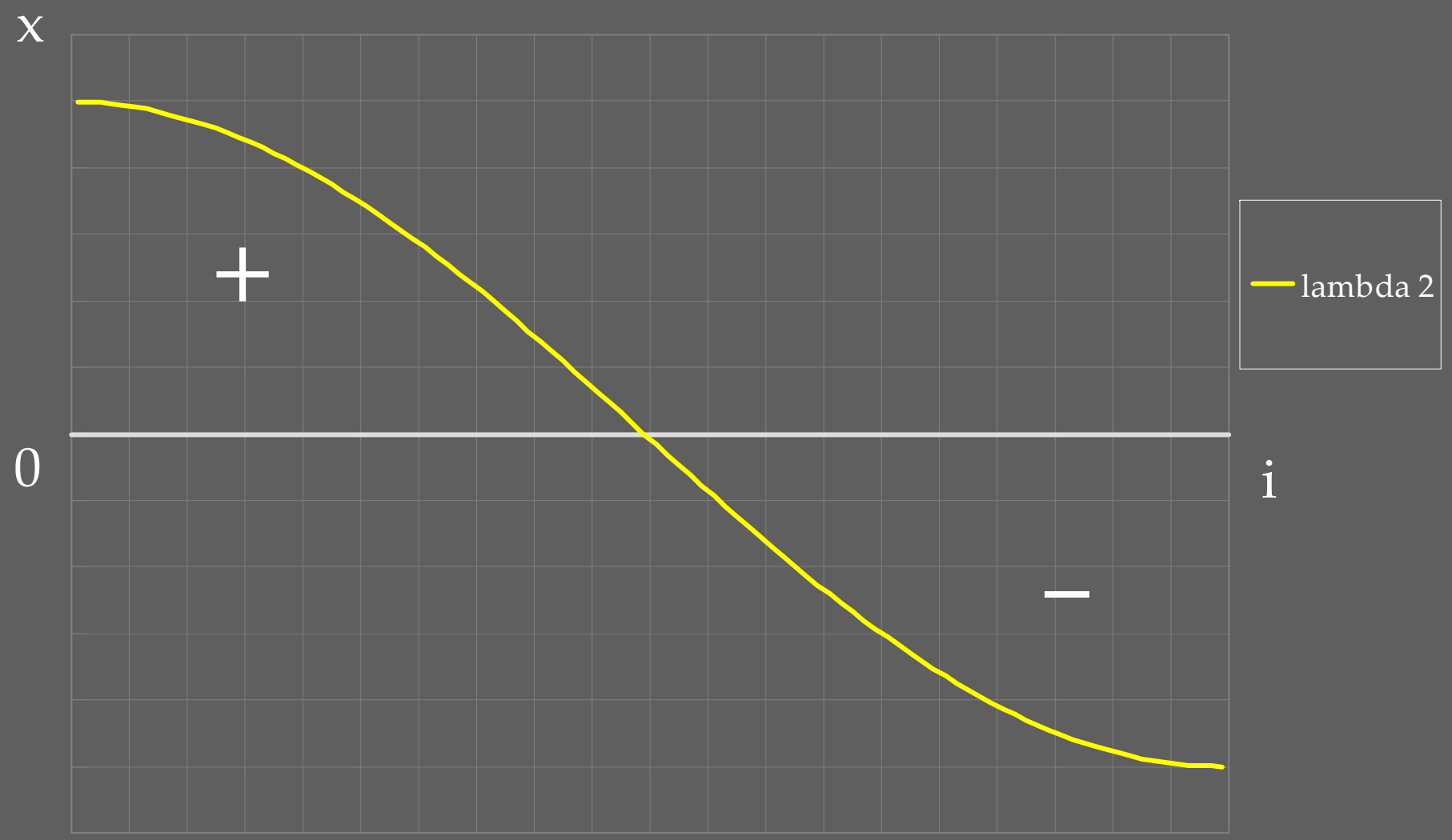
Modes of vibration corresponding to smallest eigenvalues



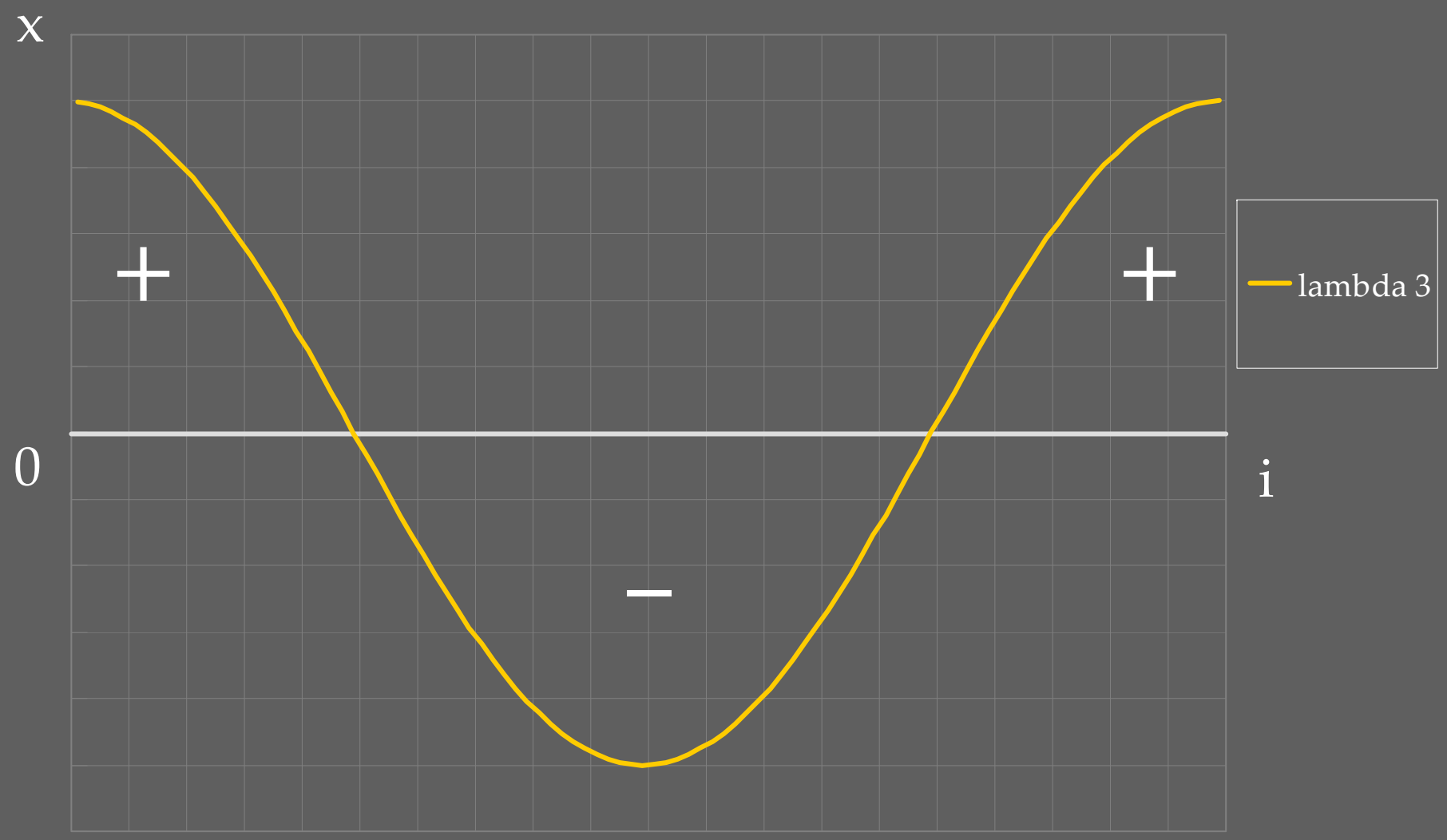
Mode of vibration (smallest)



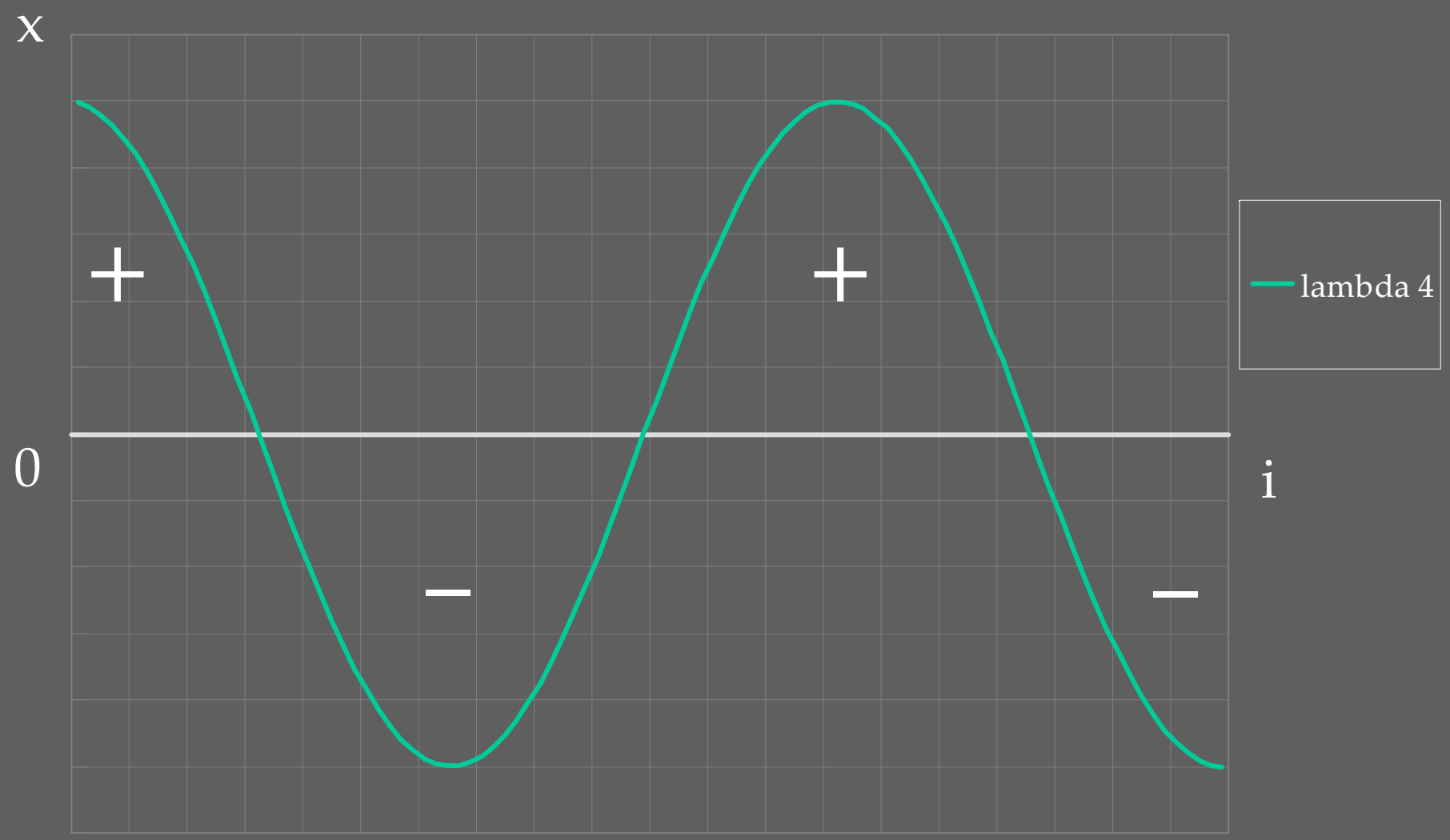
Mode of vibrations (2nd smallest)



Mode of vibration (3rd smallest)



Mode of vibration (4th smallest)



A bit off-topic: spectral clustering

Interested in global structure

- ➔ ignore high frequency modes

  (= large eigenvalues of Laplacian)

  Large eigenvectors allow abrupt change of class between nearby nodes

- ➔ Use only the smallest eigenvectors to form a kernel matrix, and use it to compute distance between nodes – and cluster them

Relatedness/similarity between nodes

We want a smooth kernel matrix –

e.g., relatedness/similarity scores do not drop to 0 all of a sudden as the distance between nodes exceeds a certain value

How do we impose smoothness?

By encouraging smoother eigencomponents corresponding to smaller modes (eigenvalues) of Laplacian, and discouraging less smooth ones.

Regularization of Laplacian

Compute spectral decomposition of L

$$\mathrm{L} = \mathrm{U}\Lambda\mathrm{U}^{\mathrm{T}}$$ (Λ: diagonal eigenvalue matrix)

and apply *regularizer* function $f(\lambda)$ to diagonals of Λ

Resulting kernel:

$$\mathrm{K} = \mathrm{U}\, f(\Lambda)\, \mathrm{U}^{\mathrm{T}}$$

Requirement on regularizer $f(\lambda)$

- Decreasing function of λ

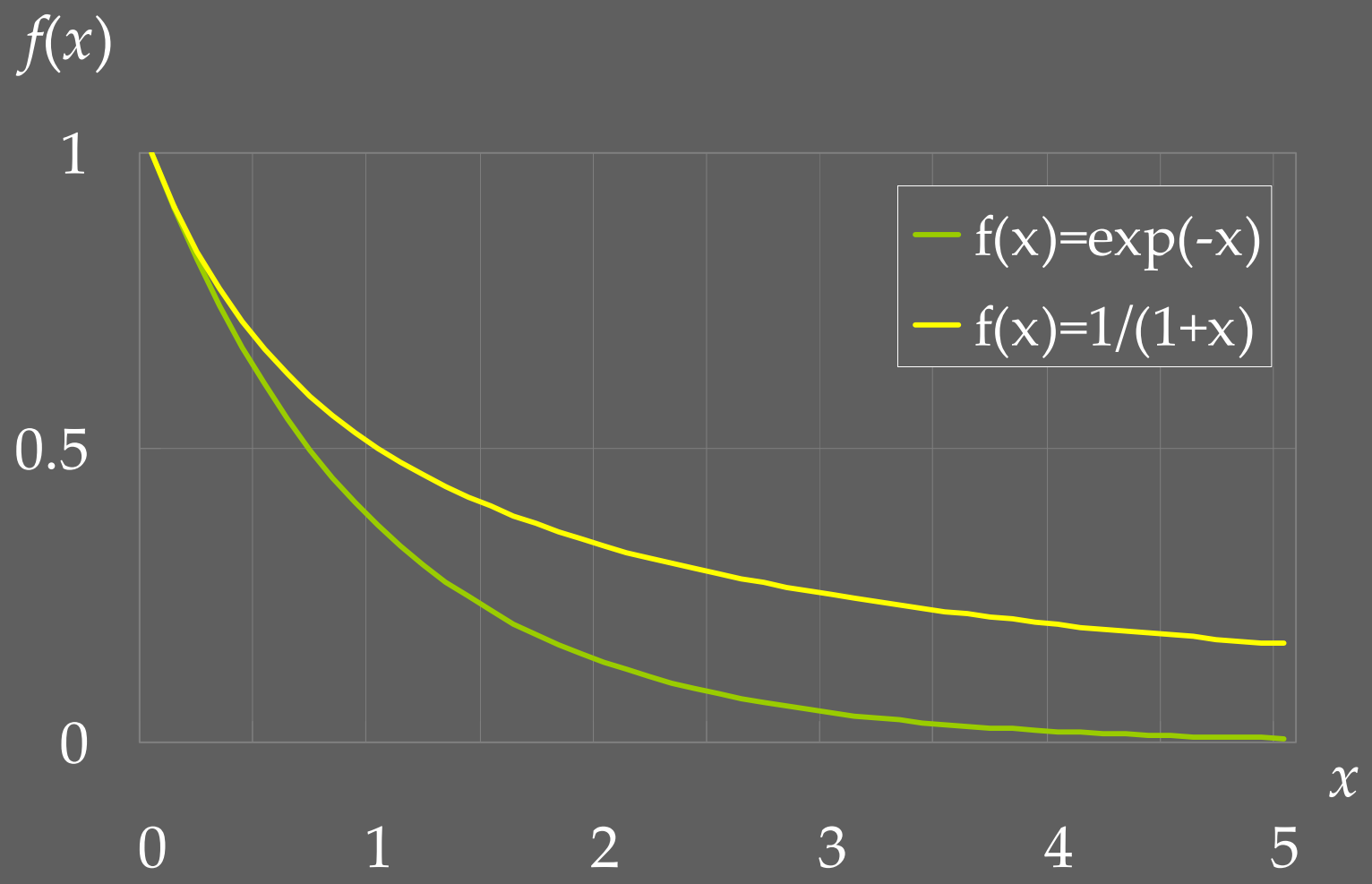
  To encourage smaller λ's and discourage larger ones

- Non-negative function

  To assure positive semidefiniteness

  ($f(\lambda)$ will be the eigenvalue of the induced kernel)

Some regularizers



Different regularizers lead to different graph kernels

$1/(1 + \beta\lambda)$

- ➔ Regularized Laplacian

$\exp(-\beta\lambda)$

- ➔ Diffusion kernels

$1/\lambda$ (or 0 if $\lambda=0$)

- ➔ Commute time kernels

Heat equation

$$\frac{\partial u(t,x,y,z)}{\partial t} = \sigma \nabla^2 u(t,x,y,z)$$

By substituting $\nabla^2 = -L$ and $u(t, \cdot, \cdot, \cdot) = K_t$

$$\frac{\partial K_t}{\partial t} = -\sigma L K_t$$

… discrete approximation

Solution is given by

$$K_t = \exp(-t\sigma L)$$

Substituting $\beta = t\sigma$ gives

$$K_\beta = \exp(-\beta L)$$

… diffusion kernels

Interpretation of kernels as link analysis measures

von Neumann kernels

Give a unified framework covering both importance and relatedness, and something in-between

Laplacian-based kernels

Have a favorable property as a relatedness measure

3. Graph kernels and link analysis

1. Typology of graph kernels
2. Graph kernels from link-analysis perspective
    1. von Neumann kernels
    2. regularized Laplacian
3. Graph Laplacian and its regularization
4. Application example

Research paper recommendation

Input

a small number of query papers

Output

list of papers ordered by their relevance to the query papers

Outline: experiment

Compared algorithms

- von Neumann kernels
- Regularized Laplacian
- HITS
- Co-citation

Dataset

Citation graph on natural language processing (NLP)

☛ about 2 thousand nodes

The NLP citation data

The most dominant topics (communities):

parsing

word-sense disambiguation

. . .

Paper with top HITS authority ranking

= Paper on standard corpus data for parsing

"Penn Treebank"

Procedure

1. Extract the row vector for the following query paper from kernel matrices

   M. A. Walker and J. D. Moore
   Empirical studies in discourse
   Computational Linguistics 23(1), 1997

2. Rank papers according to the magnitude of components in the extracted vector

Query paper

M. A. Walker and J. D. Moore. Empirical studies in discourse. *Computational Linguistics* 23(1), 1997.

A paper on discourse processing

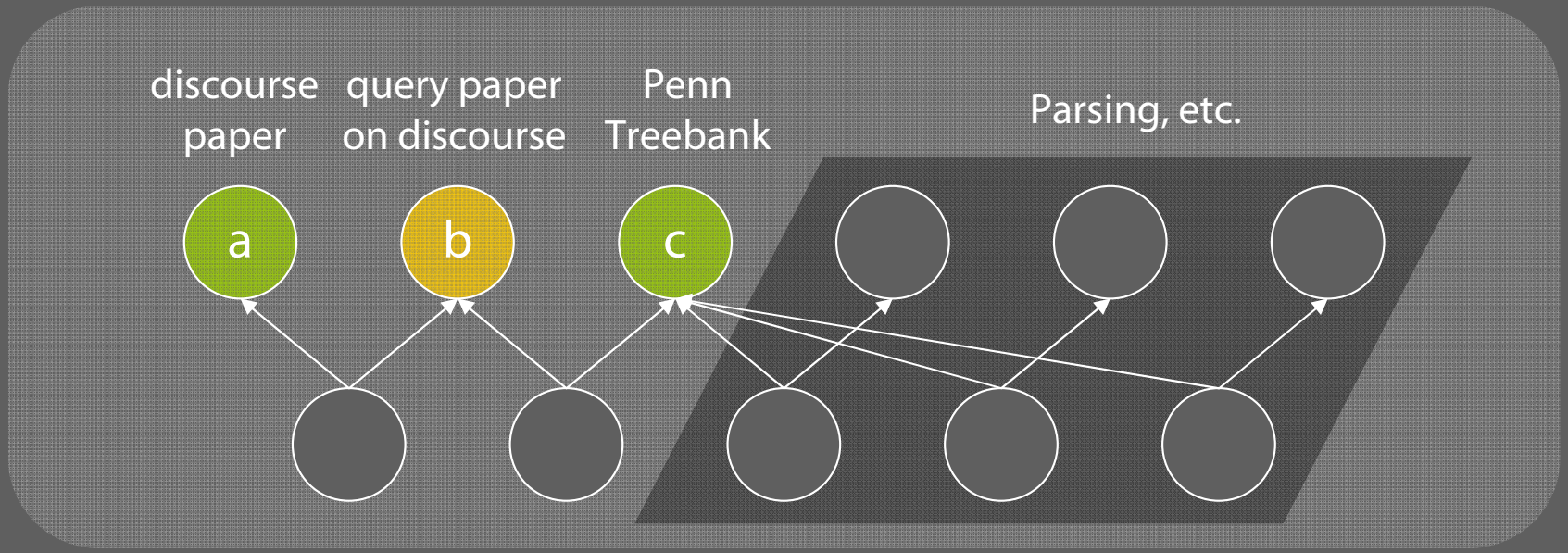
Not the most major topic

HITS ranking = 771st

Query paper

M. A. Walker and J. D. Moore. Empirical studies in discourse. *Computational Linguistics* 23(1), 1997.

Co-cited with the Penn Treebank paper (ranked no. 1 according to HITS)



von Neumann kernel (β=0.999)

NK	HITS	Paper title
1	1	Building a large annotate
2	2	A stochastic parts progra
3	3	Statistical decision-tree n
4	4	A new statistical parser b
5	5	Unsupervised word sens
6	6	Word-sense disambiguat
7	7	The mathematics of stati
8	8	Three generative, lexicali
9	9	Transformation-based erro
10	10	Integrating multiple knowledge sources to disambiguate word sense …

Identical to HITS

Topics of the top-ranked papers

- Parsing
- Word sense disambiguation
- Machine translation

Topic of the query paper

- Discourse processing

von Neumann kernel (with β=0.1)

NK	CC	HITS	Paper title
1	1	771	Empirical studies in discourse
2	2	1	Building a large annotated corpus of English: the Penn Treebank
3	2	50	Attention, intention, and the structure of discourse
4	2	76	Assessing agreement on classification tasks: the Kappa statistic
5	2	201	The reliability of dialogue structure coding schemes
6	2	604	Meaning and the structure of discourse ...
7	2	1061	Effects of ...
8		3	Statistical decision trees for parsing
9		4	A new statistical parser based on bigram lexical dependencies
10		96	Centering: a framework for modeling the local coherence of discourse

Top 7 papers are those with nonzero co-citation score

- Relatedness
- But papers in ranks 2-7 are in the order of HITS ranking

Papers ranked at 8 and 9 are not discourse papers, but are papers with high HITS ranking

- Ranking shows bias towards importance even if β is small

Regularized Laplacian

RL	CC	HITS	Paper title
1	1	771	Empirical studies in discourse
2	2	1061	Effect of … computer spoken natural language dialogue
3	2	604	Message Understanding Conf
4	2	201	The reliability of a dialogue st
5	2	76	Assessing agreement on class ification tasks: …
6	2	50	Attention, intentions, and the structure of discourse
7	2	1	Building a large annotated corpus of English: the Penn Treebank
8		198	A prosodic analysis of discourse segments in …
9		96	Centering: a framework for modeling the … discourse
10		115	Combining multiple knowledge sources for discourse …

Top 7 papers are identical to co-citation

Papers ranked 8 – 10 are discourse papers

Comparison of vNK and HITS

Correlation between rankings

Top-10 K-min distance [Fagin et al. 03]

a modified version of Kendall Tau restricted to top-k partial ranking

Similar rankings ⇒ Small K-min distance

Dissimilar rankings ⇒ Large K-min distance

Averaged over all query papers

MAX: 100 MIN: 0

$0 \leq \beta/\lambda^{-1} < 1$	0.01	0.1	0.5	0.9	0.99	0.999	0.9999
K-min	87.4	87.3	86.3	72.0	26.4	5.5	1.1

Some additional notes

Parameter tuning in von Neumann kernels

How do we choose an appropriate β ?

At some place near the upper bound of β, it changes abruptly from relatedness to importance

Clue?

Partial derivative wrt β may help estimating the amount of possible change at β

$$\frac{\partial}{\partial \beta} \mathrm{NK}_{\beta}(\mathrm{M}) = \left(\mathrm{NK}_{\beta}(\mathrm{M})\right)^2$$

Computation costs

Requires matrix inversion / eigenvalue computation

➔ expensive computation cost

Even if a citation graph is sparse, its kernel matrix is often dense

➔ memory requirement is tight as well

If a ranking relative to a single node is all that is needed, a power-method like approach can be used.

No need to keep an entire kernel matrix - only a relevant vector needs to be kept in memory

Contents

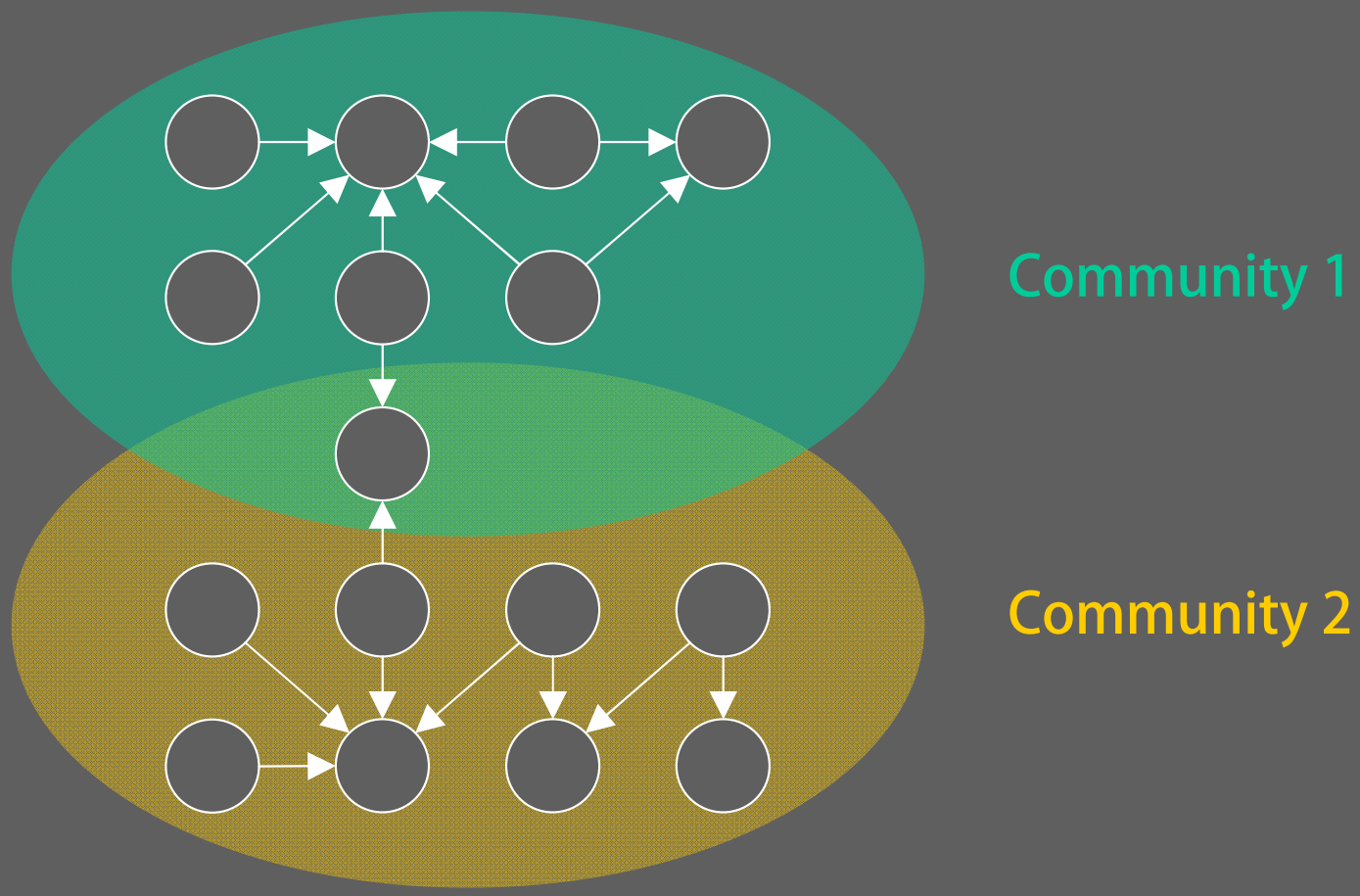
1. Background
2. Link analysis preliminaries
3. Graph kernels and link analysis
4. Dealing with multiple communities in a graph

   (more on von Neumann kernels)

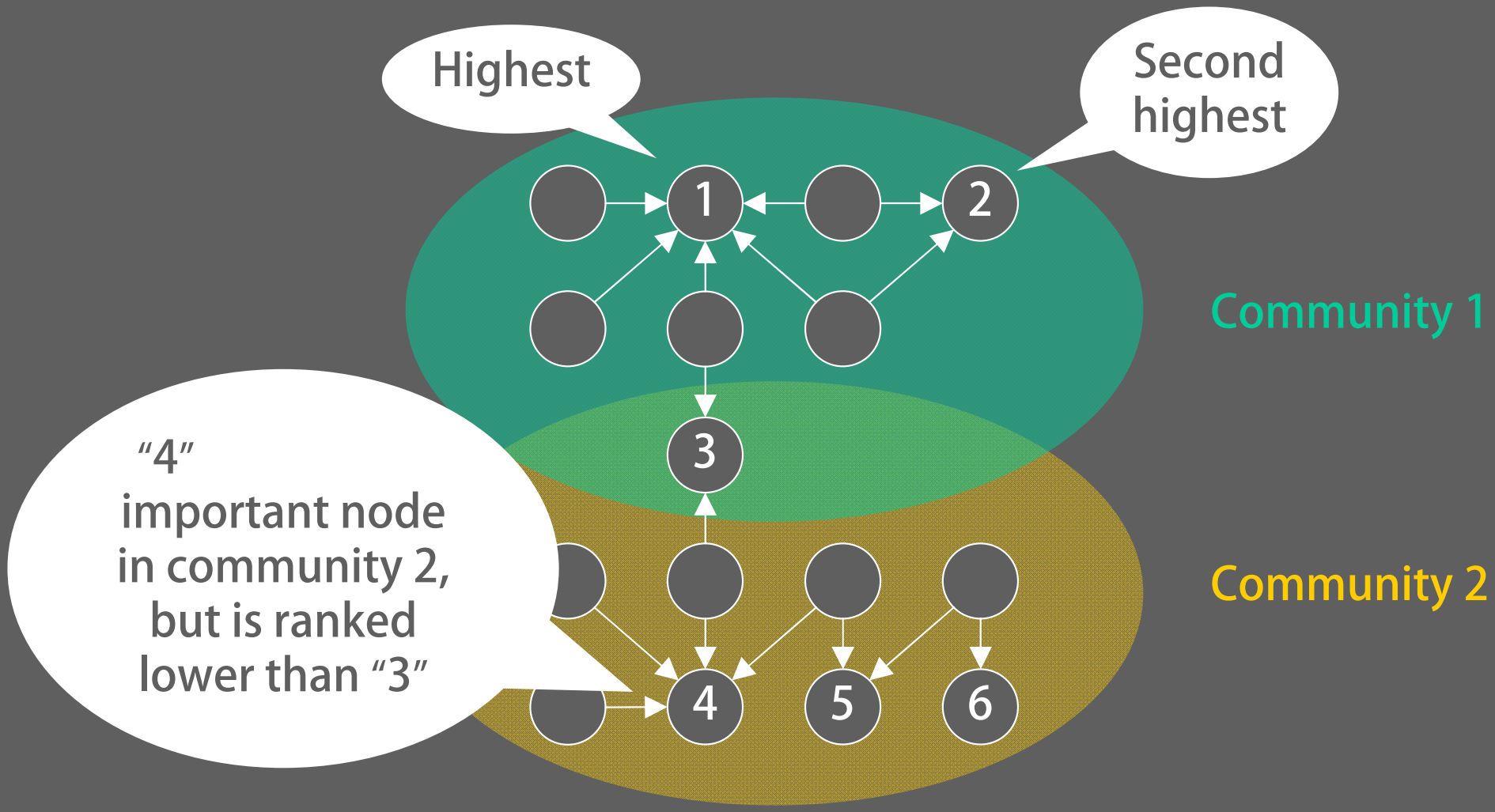
What is wrong with von Neumann kernel link analysis measures?

Let us see the behavior of von Neumann kernels on graphs with multiple communities

A multiple-community graph



HITS authority ranking



HITS authority ranking

von Neumann kernels also show a similar tendency if β is increased

i.e., they tend to rank nodes in community 1 higher irrespective of the "query" node

➔ Even if the query node is in community 2, nodes in community 1 can be output as most relevant

Topic drift

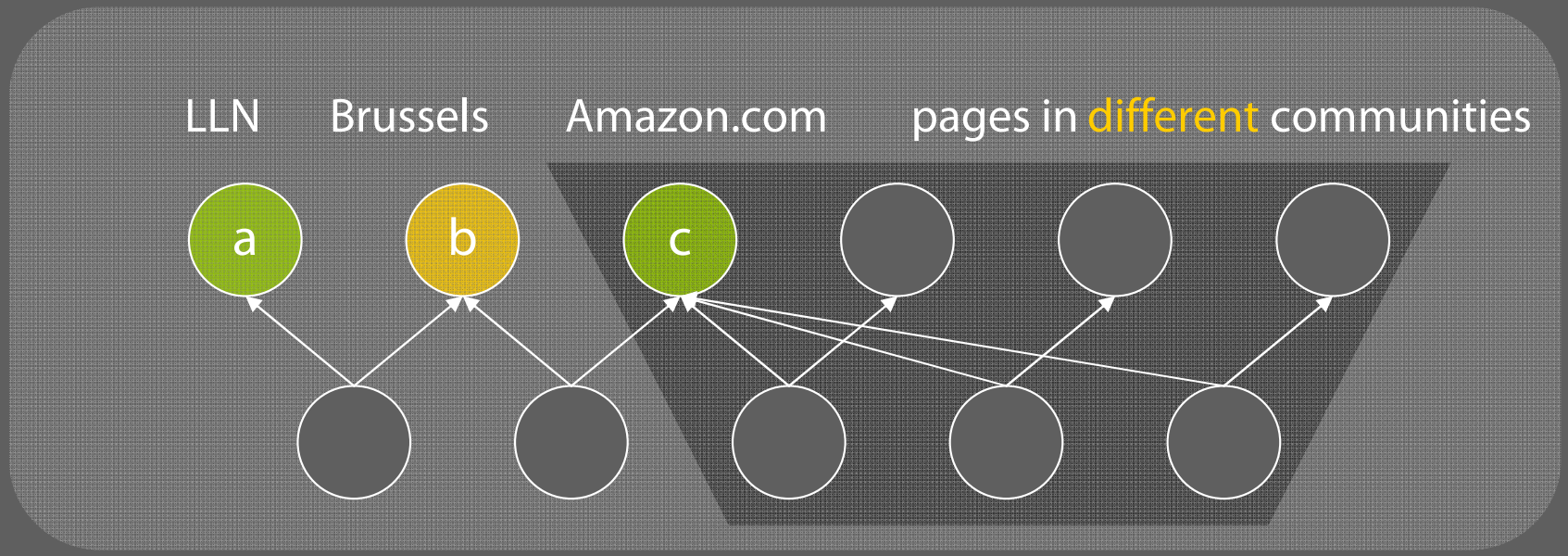
For global ranking algorithms, including HITS, this is almost inevitable as they return a single ranking for all nodes

- Normally this is avoided with the help of content information

But this is more of a problem for kernels

- It computes a ranking for each individual node, but all rankings uniformly display bias in favor of a single community

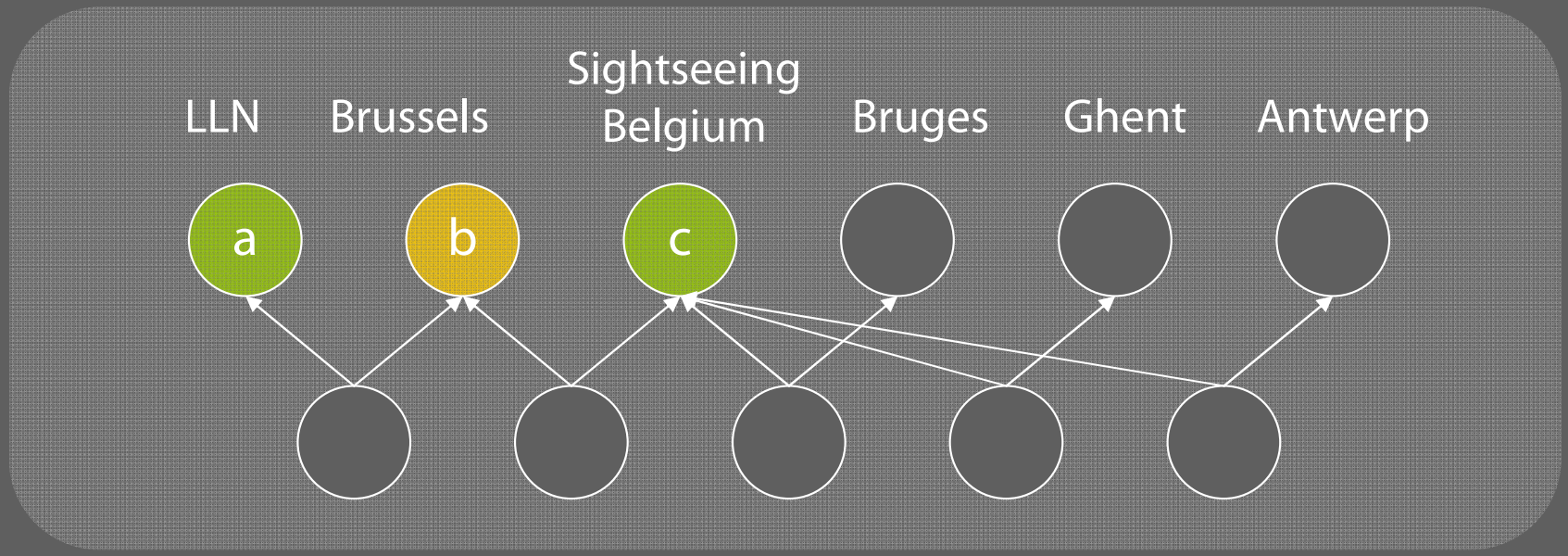
von Neumann kernels



$NK_\beta(A^TA)[c, b] > NK_\beta(A^TA)[a, b]$ whenever $\beta > 0$

Unintuitive *because* we assumed many pages citing c belong to different communities

What if graph contains a single "topic"?



In this case, recommending c over a as more relevant to b may be acceptable – given that c is more important in this *community* and probably more useful to most users

So…

von Neumann kernels might be useful *provided that* the whole graph is concerned about a single topic (or, community).

For a graph with multiple communities, decomposition of a graph into these communities is necessary - to apply von Neumann kernels independently to them.

Decompose a graph into communities – how?

Topic = community

There are papers that are interdisciplinary — i.e., containing multiple topics

- machine learning *and* link analysis
- information retrieval *and* natural language processing

➔ We would like to avoid "hard" clustering

PLSA-based preprocessing for von Neumann kernels

von Neumann kernel + PLSA preprocessing

Step 1 Apply PLSA to citation graph to obtain multiple "community graphs"

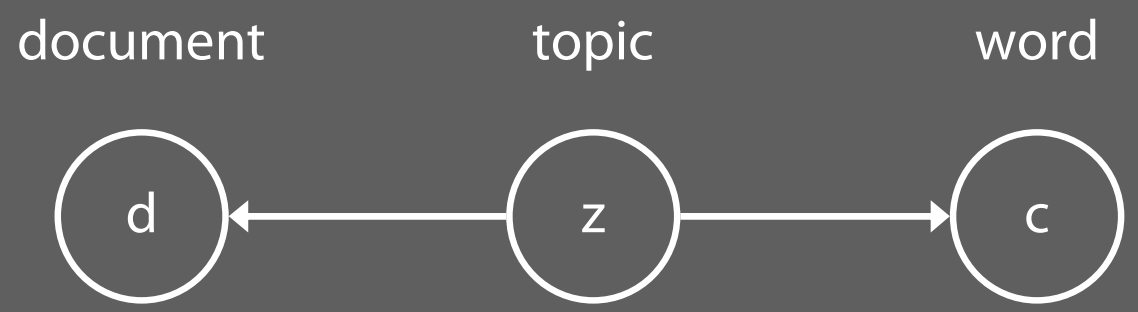
Step 2 Apply von Neumann kernels independently to the obtained community graphs

Step 3 Sum all kernel matrices to yield a final kernel matrix

Probabilistic LSA [Hofmann 1999]

A generative model of words and documents

Based on the assumption that document d and word c are conditionally independent given a hidden "topic" z



Note

We use PLSA, but not content (word) information

Instead

We use a variation of PLSA-based generative model for citations [Cohn 2001]

PLSA

$$p(c_i, d_j) = \sum_k p(c_i \mid z_k)\, p(d_j \mid z_k)\, p(z_k)$$

c_i = i th term
d_j = j th document
z_k = k th hidden topic

An EM algorithm is used to estimate $p(c|z)$, $p(d|z)$ and $p(z)$.

PLSA for citations

$$p(c_i, d_j) = \sum_k p(c_i \mid z_k)\, p(d_j \mid z_k)\, p(z_k)$$

c_i = citation from document *i*

d_j = citation to document *j*

z_k = *k* th community (hidden topic)

An EM algorithm is used to estimate $p(c|z)$, $p(d|z)$ and $p(z)$.

How to use PLSA to decompose a graph into communities?

Weight of an edge from node i to node j in community k can be computed as

$$
\begin{aligned}
& p(z_k \mid c_i, d_j) \\
& \quad = \frac{p(c_i, d_j \mid z_k) p(z_k)}{\sum_k p(c_i, d_j \mid z_k) p(z_k)} \\
& \quad = \frac{p(c_i \mid z_k) p(d_j \mid z_k) p(z_k)}{\sum_k p(c_i \mid z_k) p(d_j \mid z_k) p(z_k)}
\end{aligned}
$$

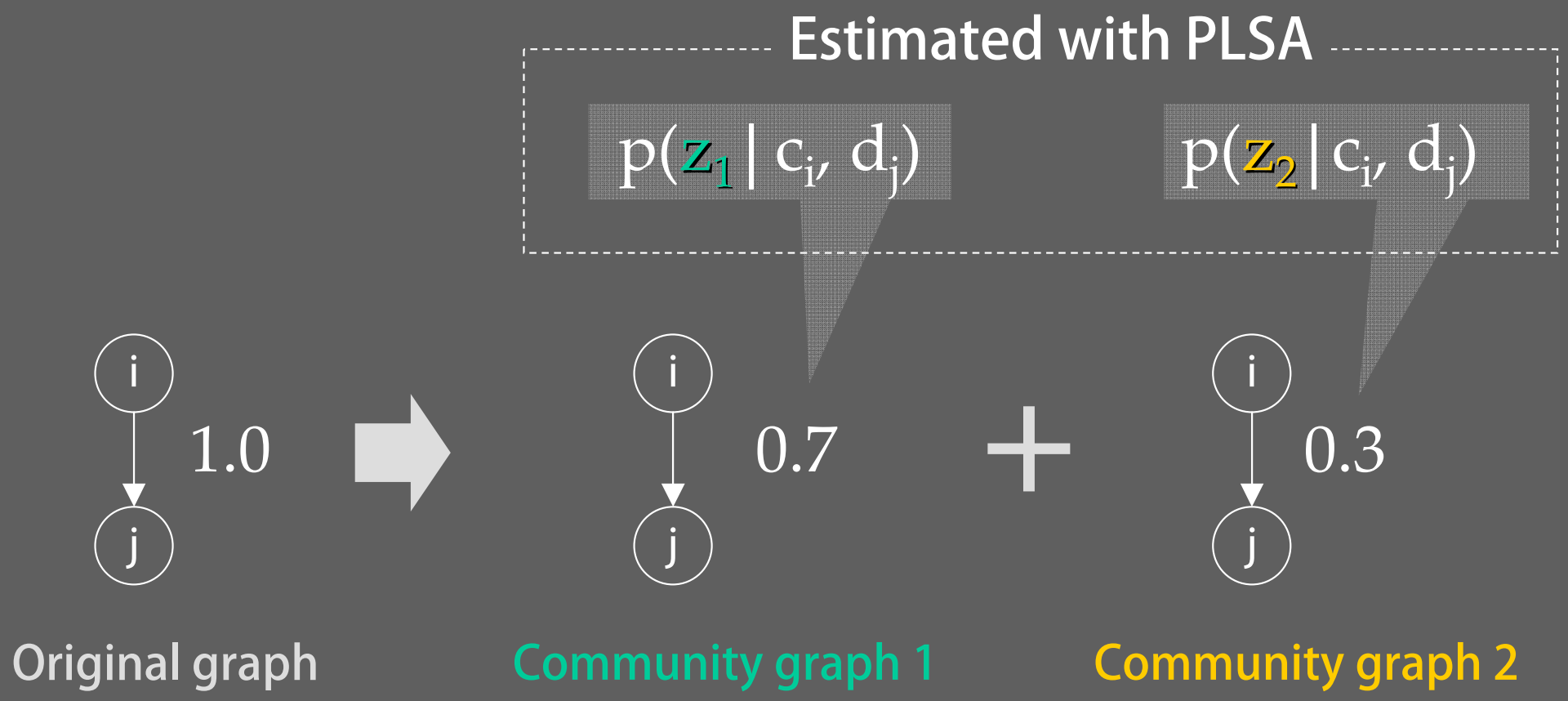
How to use PLSA to decompose a graph into communities?

So we are *not* clustering nodes!

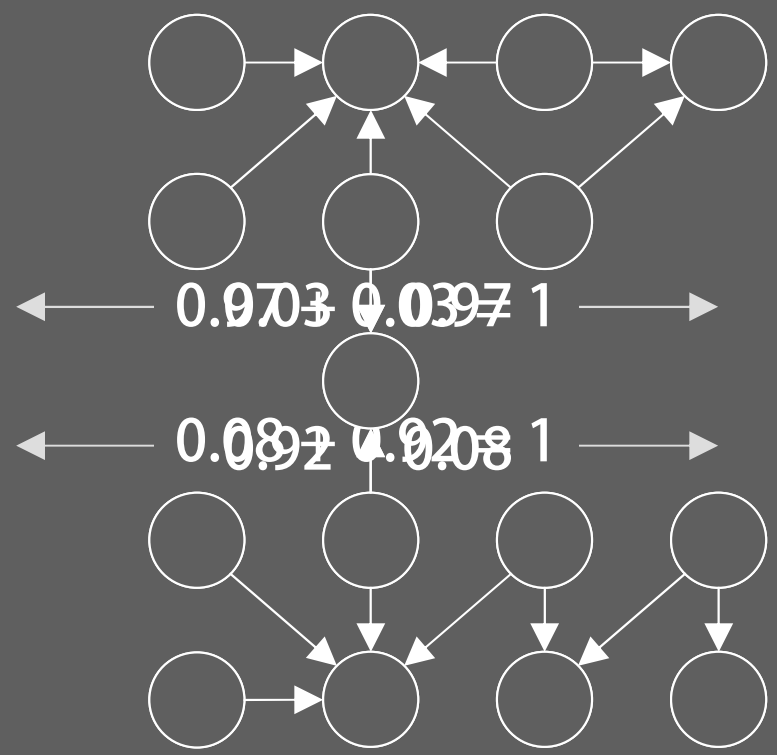
Instead,

Edge weights are distrubuted over k communities

Distribute edge weights



Graph decomposition with PLSA



Community graph 1

Community graph 2

Relation to PLSA-based Fisher kernel

Similarity and difference between

- von Neumann kernels with PLSA processing
- PLSI-based Fisher kernel [Hofmann 2001]

Fisher kernels [Jaakkola & Haussler 2000]

A generic method for obtaining kernels from generative models

- Uses derivatives of data log likelihood wrt to model parameters

Fisher kernels [Jaakkola & Haussler 2000]

$$K(x, y) = \nabla \ell_\theta(x)^T F(\theta)^{-1} \nabla \ell_\theta(y)$$

θ Model parameter

$\ell_\theta(x)$ Log likelihood of x under parameter θ

$F(\theta)$ Fisher information matrix $E_{p(x|\theta)}\left[\nabla_\theta \ell(x) \nabla_\theta \ell(x)^T\right]$

not easy to compute - often approximated by an identity matrix

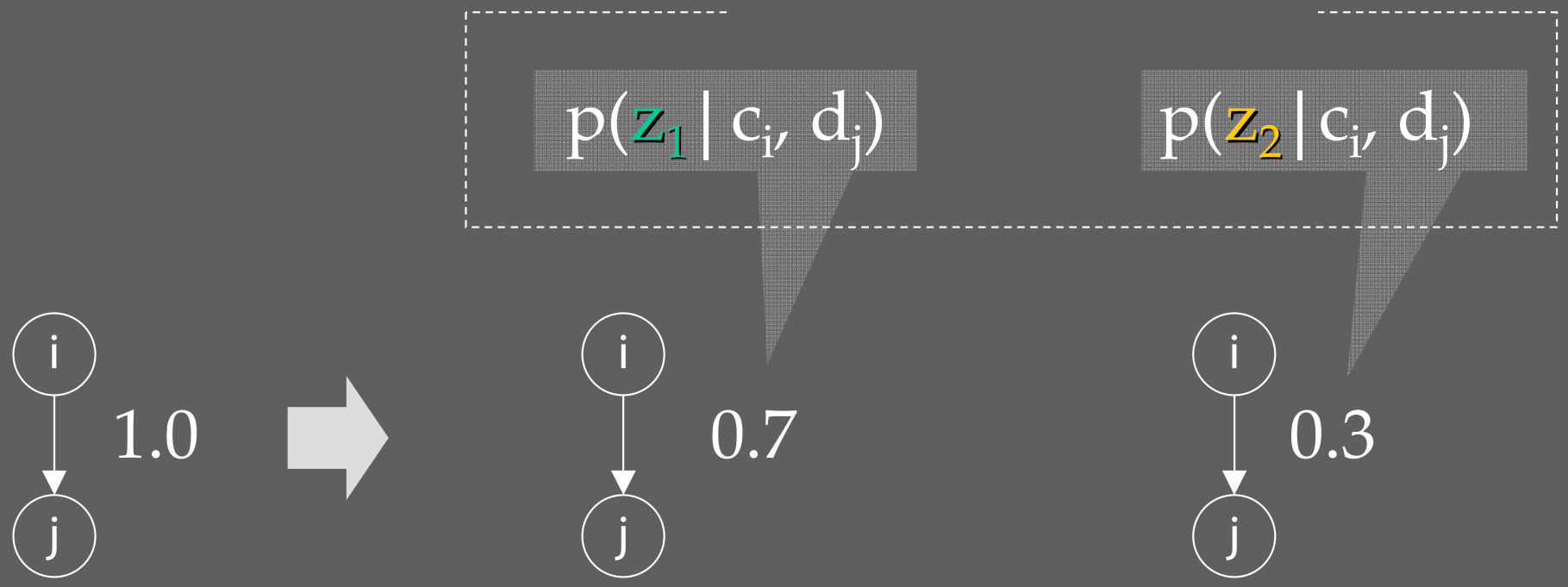
PLSA-based Fisher kernels

[Hofmann 2001]

Kernel based on the derivative wrt $p(c_i \mid z_k)$

$$K(d_1, d_2) = \sum_i p(c_i \mid d_1) p(c_i \mid d_2) \sum_k \frac{p(z_k \mid c_i, d_1) p(z_k \mid c_i, d_2)}{p(c_i \mid z_k)}$$

Recall how we distributed edge weights



PLSA-based Fisher kernels

[Hofmann 2001]

Kernel based on the derivative wrt $p(c_i \mid z_k)$

$$K(d_1, d_2) = \sum_i p(c_i \mid d_1) p(c_i \mid d_2) \sum_k \frac{p(z_k \mid c_i, d_1) p(z_k \mid c_i, d_2)}{p(c_i \mid z_k)}$$

A weighted sum of *co-citations* over all communities

➔ Long-distance relations are not captured

PLSA-based Fisher kernels

[Hofmann 2001]

The other PLSI-based Fisher kernel - this time based on the derivative wrt $p(z_k)$

$$K(d_1, d_2) = \sum_k \frac{p(z_k \mid d_1) p(z_k \mid d_2)}{p(z_k)}$$

von Neumann kernel + PLSA preprocessing

Step 1 Apply PLSA to citation graph to obtain multiple "community graphs"

Step 2 Apply von Neumann kernels independently to the obtained community graphs

Step 3 Sum all kernel matrices to yield a final kernel matrix

Experimental setting

Compare the performance of the von Neumann kernels

- with PLSA preprocessing
- without PLSA preprocessing – original vNK

under 3 parameter settings

- Large $\beta/\lambda^{-1} = 0.9999$
- Medium $\beta/\lambda^{-1} = 0.95$
- Small $\beta/\lambda^{-1} = 0.01$

Without PLSA preprocessing ($\beta/\lambda^{-1} = 0.9999$)

NK	HITS	Title
1	1	Building a large anno
2	2	A stochastic parts pro
3	3	Statistical decision-tr
4	4	A new statistical pars
5	5	Unsupervised word s
6	6	Word-sense disambig
7	7	The mathematics of s
8	8	Three generative, lexicalis
9	9	Transformation-based error-driven learning and naturally language …
10	10	Integrating multiple knowledge sources to disambiguate word sense …

Topics

- Parsing
- Word-sense disambig.
- Machine translation

Topic of the query paper

- Discourse processing

With PLSA preprocessing ($\beta/\lambda^{-1} = 0.9999$)

PNK	HITS	Title
1	50	Attention, intentions, and the structur
2	61	Multiple-paragraph segmentation of e
3	77	Lexical cohesion computed by thesau
4	111	Combining multiple knowledge sourc
5	194	A prosodic analysis of discourse segment
6	76	Assessing agreement on classification tasks: the Kappa statistic
7	150	An automatic methods of finding community boundaries
8	162	Text segmentation based on similarity between words
9	317	Intention-based segmentation: human reliability and correlation with …
10	340	Replicability of transaction and action coding in the map task corps

Topic

- discourse

Without PLSA preprocessing ($\beta/\lambda^{-1} = 0.95$)

NK	HITS	Title
1	1	Building a large annotated corpus of English: the Penn Treebank
2	3	Statistical decision-tree models for parsing
3	2	A stochastic parts program and noun phrase parser for …
4	4	A new statistical parser based on bigram lexical dependencies
5	50	Attention, intentions, and the structure of discourse
6	771	Empirical studies in discourse
7	8	Three generative, lexicalised models for statistical parsing
8	76	Assessing agreement on classification tasks: the Kappa statistic
9	6	Word-sense disambiguation using statistical models of Roget's
10	5	Unsupervised word sense disambiguation rivaling supervised methods

With PLSA preprocessing ($\beta/\lambda^{-1} = 0.95$)

PNK	HITS	Title
1	771	Empirical studies in discourse
2	201	The reliability of a dialogue structure coding scheme
3	76	Assessing agreement on classification tasks: the Kappa statistic
4	50	Attention, intentions, and the structure of discourse
5	1061	Effects of variable initiative … spoken natural language dialogue
6	1	Building a large annotated corpus of English: the Penn Treebank
7	96	Centering: a framework for modeling the local coherence of discourse
8	61	Multiple-paragraph segmentation of expository text
9	77	Lexical cohesion computed by thesaural relations as …
10	115	Combining multiple knowledge sources for discourse segmentation

Without PLSA preprocessing ($\beta/\lambda^{-1} = 0.01$)

NK	HITS	Title
1	771	Empirical studies in discourse
2	1	Building a large annotated corpus of English: the Penn Treebank
3	50	Attention, intentions, and the structure of discourse
4	76	Assessing agreement on classification tasks: the Kappa statistic
5	201	The reliability of a dialogue structure coding scheme
6	604	Message Understanding Conference (MUC) Tests of discourse …
7	1061	Effects of variable initiative … spoken natural language dialogue
8	3	Statistical decision-tree models for parsing
9	4	A new statistical parser based on bigram lexical dependencies
10	96	Centering: a framework for modeling the local coherence of discourse

With PLSA preprocessing ($\beta/\lambda^{-1} = 0.01$)

PNK	HITS	Title
1	771	Empirical studies in discourse
2	201	The reliability of a dialogue structure coding scheme
3	76	Assessing agreement on classification tasks: the Kappa statistic
4	1061	Effects of variable initiative … spoken natural language dialogue
5	50	Attention, intentions, and the structure of discourse
6	1	Building a large annotated corpus of English: the Penn Treebank
7	604	Message Understanding Conference (MUC) Tests of discourse …
8	96	Centering: a framework for modeling the local coherence of discourse
9	374	A trainable document summarizer
10	60	Evaluation a focus-based approach to anaphora resolution

Another preliminary result for paper recommendation simulation

Scenario

A student is given a homework. He has to make a survey of a research field he knows nothing of.

His sole clue to the field is a paper his professor has handed him.

He wants to find the most relevant papers in the field, by using a recommendation system (to finish his homework quickly).

But *which* recommendation system?

Survey papers on NLP

10 survey papers

Topics vary:

parsing, generation, discourse, summarization, …

Procedure

Step 1 From a survey paper containing many references, pick up one reference at random.

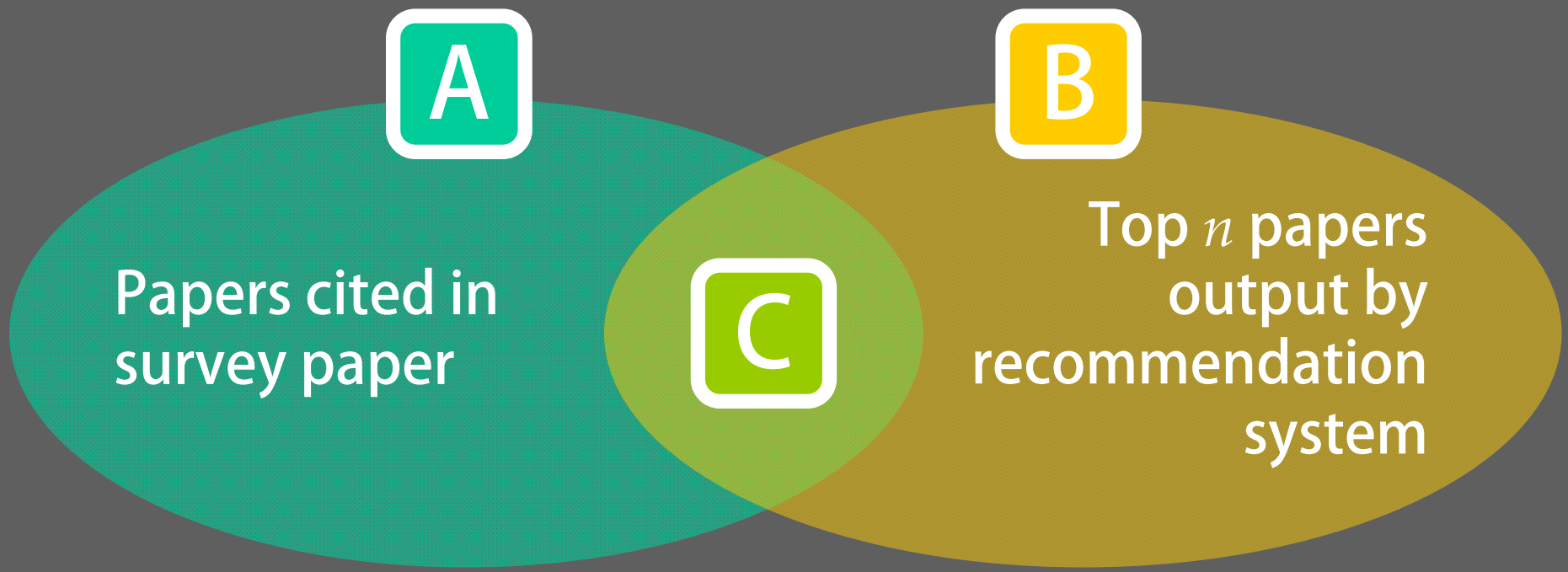
Step 2 Input the referenced paper to a recommendation system as a query.

Step 3 Check how many remaining references in the survey paper are included in the top-n output of the system.

Evaluation metric

Recall of the top-n ranking list output by the recommendation system

"Of the top-n papers output by the recommendation system, how many are actually referenced in the survey paper?"



$$\text{Recall} = \frac{C}{A}$$

Result

Ranking methods	Top-30	Top-50
von Neumann kernel	0.42	0.50
von Neumann kernel + PLSA / 5 communities	0.43	0.52
von Neumann kernel + PLSA / 8 communities	**0.45**	**0.55**
von Neumann kernel + PLSA / 10 communities	0.44	0.54
Regularized Laplacian	0.40	0.51
Co-citation coupling	0.40	0.46
HITS authority ranking	0.17	0.25

Result

Ranking methods	Top-30	Top-50
von Neumann kernel	0.42	0.50
von Neumann kernel + PLSA / 5 communities	0.43	0.52
von Neumann kernel + PLSA / 8 communities	**0.45**	**0.55**
von Neumann kernel + PLSA / 10 communities	0.44	0.54
Regularized Laplacian	0.40	0.51
Co-citation coupling	0.40	0.46
HITS authority ranking	0.17	0.25

Summary

Summary

von Neumann kernels

Unified framework for relatedness and importance, and their intermediate

Preprocessing to avoid “topic drift” desirable

PLSA has a nice property suitable for vNK

Summary (continued)

Laplacian-based kernels

Based on the regularization of graph Laplacian

Remain a relatedness measure over entire parameter range

References

- F. R. K. Chung. *Spectral Graph Theory*. AMS, 1997
- R. Fagin, R. Kumar, and D. Sivakumar. Comparing top *k* lists. *SODA*, 2003.
- T. Hofmann. Probabilistic latent semantic indexing. *SIGIR*, 1999.
- T. Hofmann. Learning the similarity of documents: an information theoretic approach to document retrieval and categorization. *NIPS* 12. MIT Press, 2000.
- T. Ito, M. Shimbo, T. Kudo, and Y. Matsumoto. Application of kernels to link analysis. *KDD*, 2005.
- T. Ito, M. Shimbo, D. Mochihashi, and Y. Matsumoto. Exploring multiple communities with kernel-based link analysis. *PKDD*, 2006.
- M. M. Kessler. Bibliographic coupling between scientific papers. *Am. Documentation* 14, 1963

References (continued)

- J. Kandola, J. Shawe-Taylor, and N. Cristianini. Learning semantic similarity. *NIPS* 15. MIT Press, 2003.
- R. Kondor and J. Lafferty. Diffusion kernels on graphs and other discrete input spaces. *ICML*, 2002.
- M. Shimbo, T. Ito, and Y. Matsumoto. Evaluation of kernel-based link analysis measures on research paper recommendation. *JCDL*, 2007.
- H. Small. Co-citation in the scientific literature: a new measure of the relationship between two documents. *J. Am. Soc. Information Science* 24, 1973.
- A. J. Smola and R. Kondor. Kernels and regularization of graphs. *COLT*, 2003.
- S. White and P. Smyth. Algorithms for estimating relative importance in networks. *KDD*, 2003.

Thank you

http://cl.naist.jp/~shimbo

shimbo@is.naist.jp