CMU SCS

Large Graph Mining: Patterns, Tools and Case Studies

Christos Faloutsos
Hanghang Tong
CMU

CMU SCS

Outline

- Part 1: Patterns
- Part 2: Matrix and Tensor Tools
- Part 3: Proximity
- ➡ Part 4: Case Studies

CMU SCS

Outline: Part 4

- ➡ Virus/influence propagation
- Blog analysis
- Community detection and tracking
- Tensor for web mining: TOPHITS
- Scalability

CMU SCS

Reference:

D. Chakrabarti, Y. Wang, C. Wang, J. Leskovec, and C. Faloutsos, *Epidemic Thresholds in Real Networks*, ACM TISSEC, 10(4), 2008

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-4

CMU SCS

Virus propagation

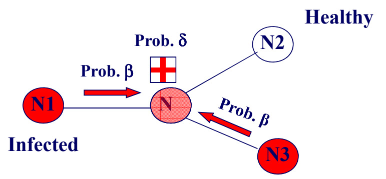
- How do viruses/rumors propagate?
- Blog influence?
- Will a flu-like virus linger, or will it become extinct soon?

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-5

CMU SCS

The model: SIS

- 'Flu' like: Susceptible-Infected-Susceptible
- Virus 'strength' s= β/δ



ICDE'09 Copyright: Faloutsos, Tong (2009) 4-6

CMU SCS

Epidemic threshold τ

of a graph: the value of τ, such that

if strength $s = \beta / \delta < \tau$

an epidemic can not happen

Thus,

- given a graph
- compute its epidemic threshold

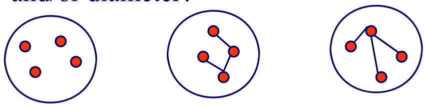
ICDE'09 Copyright: Faloutsos, Tong (2009) 4-7

CMU SCS

Epidemic threshold τ

What should τ depend on?

- avg. degree? and/or highest degree?
- and/or variance of degree?
- and/or third moment of degree?
- and/or diameter?



ICDE'09 Copyright: Faloutsos, Tong (2009) 4-8

CMU SCS

Epidemic threshold

- [Theorem] We have no epidemic, if

$$\boxed{\beta/\delta < \tau = 1/\lambda_{1,A}}$$

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-9

CMU SCS

Epidemic threshold

- [Theorem] We have no epidemic, if

recovery prob. epidemic threshold

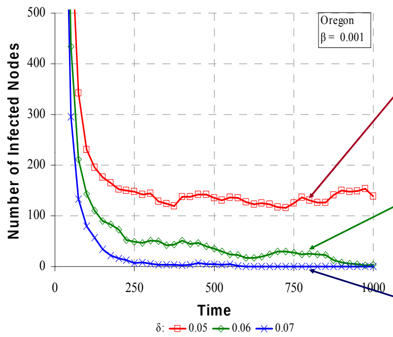
$$\boxed{\beta/\delta < \tau = 1/\lambda_{1,A}}$$

attack prob. largest eigenvalue of adj. matrix ***A***

Proof: [Wang+03]

CMU SCS

Experiments (Oregon)



$\beta/\delta > \tau$ **(above threshold)**

$\beta/\delta = \tau$ **(at the threshold)**

$\beta/\delta < \tau$ **(below threshold)**

CMU SCS

Outline: Part 4

- Virus/influence propagation
- ➡ Blog analysis
- Community detection and tracking
- Tensor for web mining: TOPHITS
- Scalability

CMU SCS

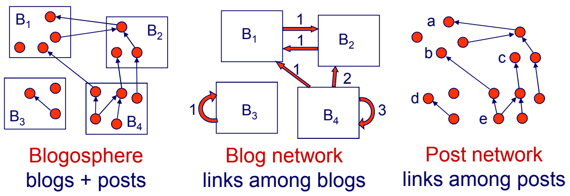
Blog analysis

- with Mary McGlohon (CMU)
- Jure Leskovec (CMU)
- Natalie Glance (now at Google)
- Mat Hurst (now at MSR)

[SDM'07]

CMU SCS

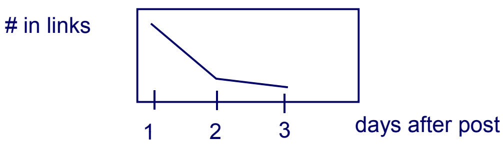
Cascades on the Blogosphere



Q1: popularity-decay of a post?
Q2: degree distributions?
Q3: cascade shapes?
Q4: cascade sizes?

CMU SCS

Q1: popularity over time

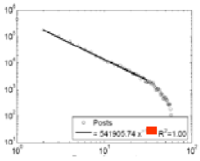


Post popularity drops-off – exponentially?

CMU SCS

Q1: popularity over time

in links (**log**)



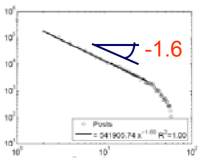
days after post (**log**)

Post popularity drops-off – ~~exponentially~~?
POWER LAW!
Exponent?

CMU SCS

Q1: popularity over time

in links (log)



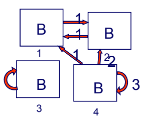
days after post (**log**)

Post popularity drops-off – ~~exponentially~~?
POWER LAW!
Exponent? -1.6 (close to -1.5: Barabasi's stack model)

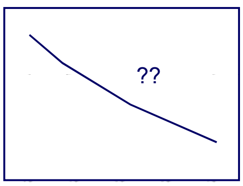
CMU SCS

Q2: degree distribution

44,356 nodes, 122,153 edges. Half of blogs belong to largest connected component.



count



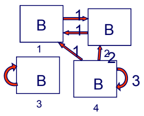
blog in-degree

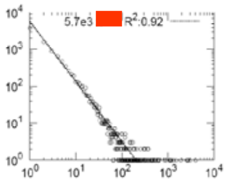
CMU SCS

Q2: degree distribution

44,356 nodes, 122,153 edges. Half of blogs belong to largest connected component.

count





blog in-degree

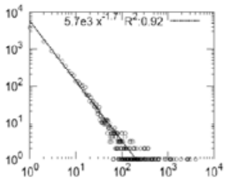
ICDE'09 Copyright: Faloutsos, Tong (2009) 4-19

CMU SCS

Q2: degree distribution

44,356 nodes, 122,153 edges. Half of blogs belong to largest connected component.

count



blog in-degree

in-degree slope: -1.7
out-degree: -3
'rich get richer'

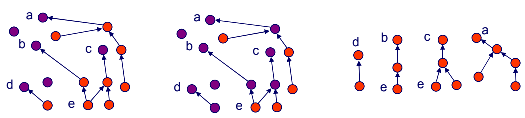
ICDE'09 Copyright: Faloutsos, Tong (2009) 4-20

CMU SCS

Topological patterns: Cascades

PProcedure for gathering cascades:

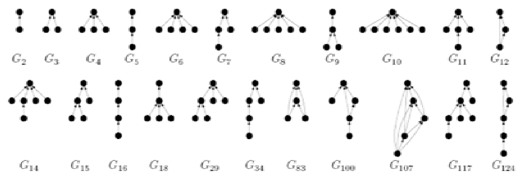
- Find all initiators (nodes with out-degree 0)
- Follow in-links
- Produces directed acyclic graph
- Count cascade shapes (use our multi-level graph isomorphism testing algorithm)



ICDE'09 Copyright: Faloutsos, Tong (2009) 4-21

CMU SCS

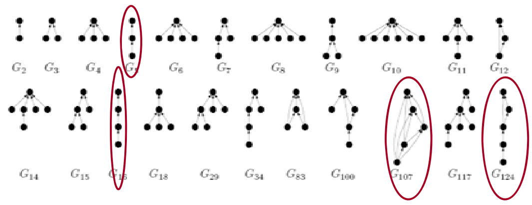
Q3: Cascade shapes?



CMU SCS

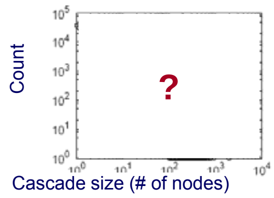
Q3: Cascade shapes?

- Mainly, stars
- Also, chains (!)



CMU SCS

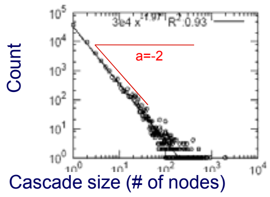
Q4: Cascade sizes?



CMU SCS

Q4: Cascade sizes?

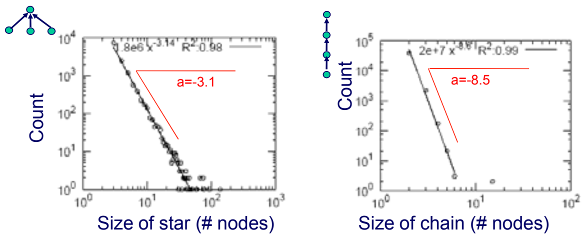
- Power law



CMU SCS

Q4: Cascade sizes?

Stars and chains also follow a power law, with different exponents (star -3.1, chain -8.5).



CMU SCS

Outline: Part 4

- Virus/influence propagation
- Blog analysis
- ➡ Community detection and tracking
- Tensor for web mining: TOPHITS
- Scalability

CMU SCS

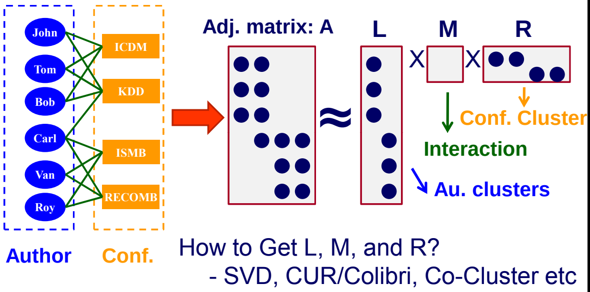
Thanks to

Deepayan Chakrabarti Jimeng Sun Spiros Papadimitriou

 Copyright: Faloutsos, Tong (2009)

CMU SCS

Matrix Factorization for Community Detection



How to Get L, M, and R?

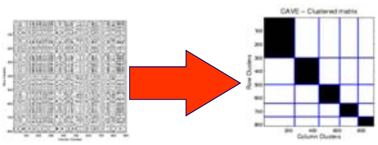
- SVD, CUR/Colibri, Co-Cluster etc

But…

 Copyright: Faloutsos, Tong (2009)

CMU SCS

Cross-association



Desiderata:

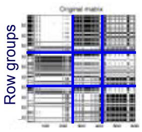
- ✓ Simultaneously discover row and column groups
- ✓ Fully Automatic: No "magic numbers"
- ✓ Scalable to large matrices

Reference:

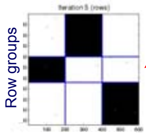
1. Chakrabarti et al. Fully Automatic Cross-Associations, KDD'04

CMU SCS

What makes a cross-association "good"?



versus

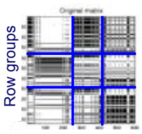


Why is this better?

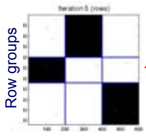
ICDE'09 Copyright: Faloutsos, Tong (2009) 4-31

CMU SCS

What makes a cross-association "good"?



versus



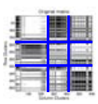
Why is this better?

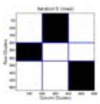
simpler; easier to describe
easier to compress!

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-32

CMU SCS

What makes a cross-association "good"?





Problem definition: given an encoding scheme
- decide on the # of col. and row groups *k* and *l*
- and reorder rows and columns,
- to achieve best compression

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-33

CMU SCS

details

Main Idea

Good Compression → Better Clustering

Total Encoding Cost = $\underbrace{\Sigma_i \, size_i * H(x_i)}_{\text{Code Cost}}$ + $\underbrace{\text{Cost of describing cross-associations}}_{\text{Description Cost}}$

Minimize the total cost (# bits)

for lossless compression

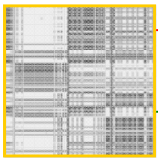
ICDE'09 Copyright: Faloutsos, Tong (2009) 4-34

CMU SCS

Cross-Association: Objective Function

(How to choose the # of communities)

one row group
one col group

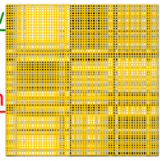


high → code cost (blocks) ← low

\+

low → description cost (blocks' model) ← high

n row groups
m col groups



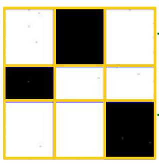
ICDE'09 Copyright: Faloutsos, Tong (2009) 4-35

CMU SCS

Cross-Association: Objective Function

(How to choose the # of communities)

k = 3 row groups
ℓ = 3 col groups



low → code cost (blocks)

\+

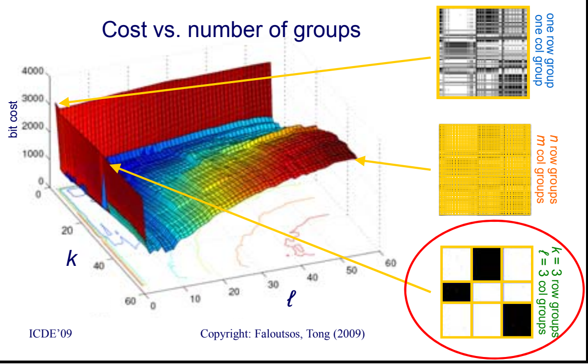
low → description cost (blocks' model)

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-36

CMU SCS

Cross-Association: Objective Function

Cost vs. number of groups

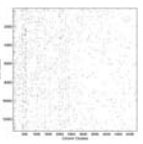


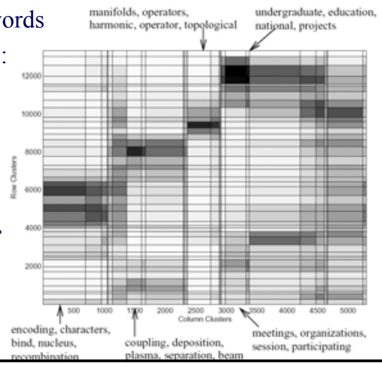
ICDE'09 Copyright: Faloutsos, Tong (2009)

CMU SCS

Cross-association Results

- NSF grants-words
- Found groups: k=41, l=28



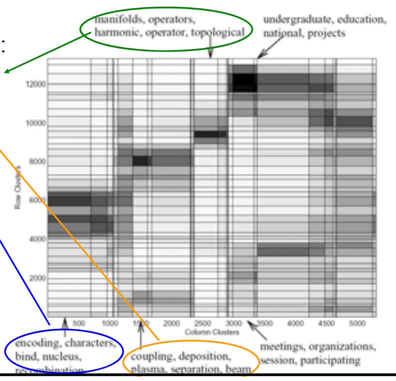


ICDE'09

CMU SCS

Cross-association Results

- Topics Found:
  - Mathematics
  - Physics
  - Genetics



ICDE'09

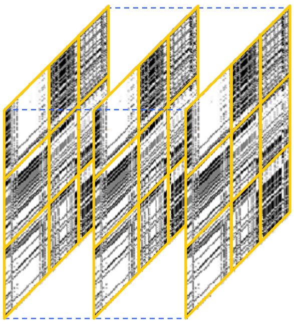
CMU SCS

GraphScope: Track Communities Over Time

Sun, J., Faloutsos, C., Papadimitriou, S., and Yu, P. S. *GraphScope: parameter-free mining of large time-evolving graphs*. KDD '07.

CMU SCS

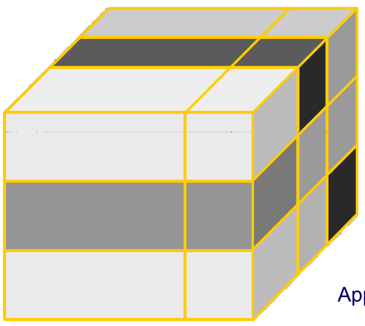
GraphScope: Track Communities Over Time



$t = 0$ $t = 1$ $t = 2$

CMU SCS

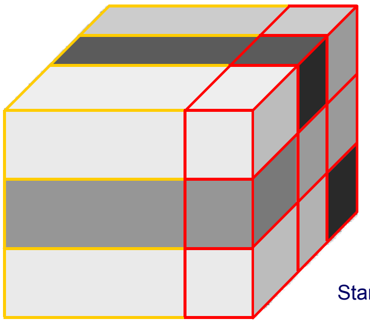
GraphScope: Cost Function



Option 1:
Append to current segment

CMU SCS

GraphScope: Cost Function

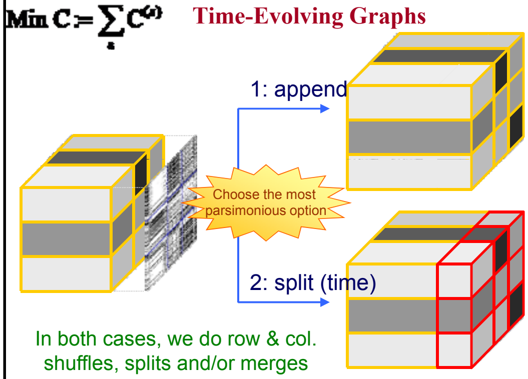


CMU SCS

Search for solution

Time-Evolving Graphs

$$\text{Min } C = \sum_s C^{(s)}$$

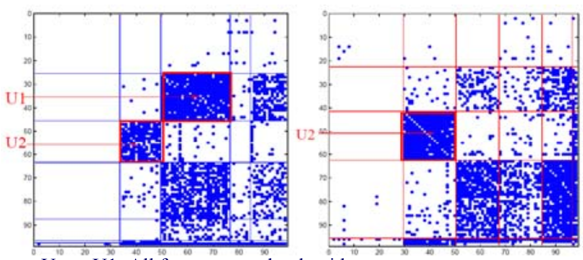


In both cases, we do row & col. shuffles, splits and/or merges

CMU SCS

Evolving communities

(BLUETOOTH)



- Users U1: All from same school, with similar schedule
- Users U2: All work in the same lab

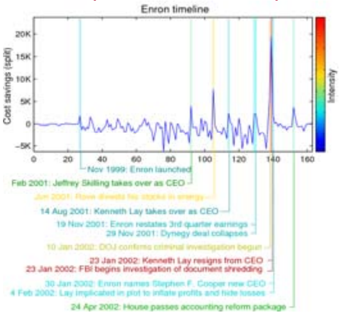
97 devices (nodes)
690 edges (on avg)
46 weeks

~ 32K edges total

Copyright: Faloutsos, Tong (2009)

CMU SCS

MDL mining on time-evolving graph (Enron emails)



ICDE'09 Copyright: Faloutsos, Tong (2009) 4-46

CMU SCS

More on this topic

- How to leverage the time smoothness
  - Jure Ferlez, Christos Faloutsos, Jure Leskovec, Dunja Mladenic, Marko Grobelnik: Monitoring Network Evolution using MDL. ICDE 2008: 1328-1330
- How to find hierarchical Communities?
  - Spiros Papadimitriou, Jimeng Sun, Christos Faloutsos, Philip S. Yu: Hierarchical, Parameter-Free Community Discovery. ECML/PKDD (2) 2008: 170-187
- Code for Cross-Association
  - www.cs.cmu.edu/~deepay/mywww/software/CrossAssociations-01-27-2005.tgz

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-47

CMU SCS

References

- Inderjit S. Dhillon, Subramanyam Mallela, Dharmendra S. Modha: Information-theoretic co-clustering, KDD2003
- Deepayan Chakrabarti, Spiros Papadimitriou, Dharmendra S. Modha, Christos Faloutsos: Fully automatic cross-associations, KDD 2004
- Jimeng Sun, Spiros Papadimitriou, Philip S. Yu, Christos Faloutsos: GraphScope: parameter-free mining of large time-evolving graphs. KDD 2007

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-48

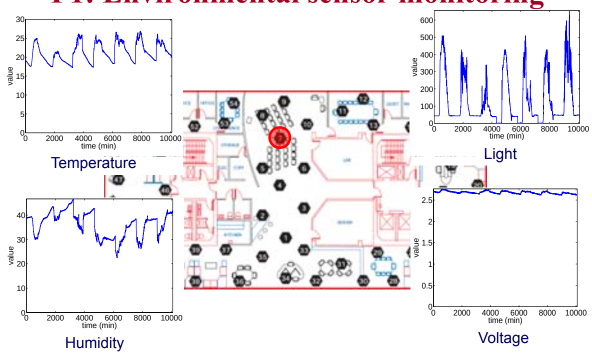
CMU SCS

Outline: Part 4

- Virus/influence propagation
- Blog analysis
- Community detection and tracking
- ➡ Tensor for web mining: TOPHITS
- Scalability

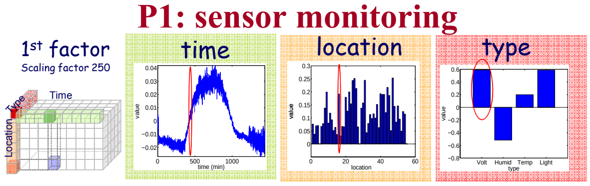
CMU SCS

P1: Environmental sensor monitoring



CMU SCS

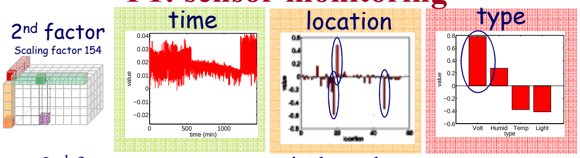
P1: sensor monitoring



- 1st factor consists of the main trends:
  - Daily periodicity on time
  - Uniform on all locations
  - Temp, Light and Volt are positively correlated while negatively correlated with Humid

CMU SCS

P1: sensor monitoring



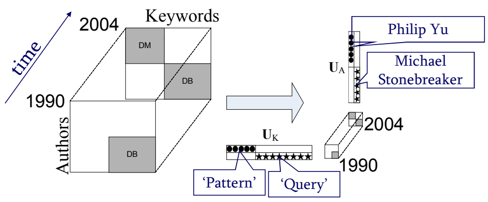
- 2nd factor captures an atypical trend:
  - Uniformly across all time
  - Concentrating on 3 locations
  - Mainly due to voltage
- Interpretation: two sensors have low battery, and the other one has high battery.

voltage light
hum.
temp.

CMU SCS

P2: Social network analysis

- Multiway latent semantic indexing (LSI)
  - Monitor the change of the community structure over time



CMU SCS

P2: Social network analysis (cont.)

Authors	Keywords	Year
michael carey, michael stonebreaker, h. jagadish, hector garcia-molina	ueri,parallel,optimization,concurr, ent	1995
surajit chaudhuri,mitch cherniack,michael stonebreaker,ugur etintemel	distribut,systems,view,storage,servic,process, cache	2004
jiawei han,jian pei,philip s. yu, jianyong wang,charu c. aggarwal	ttern,support, cluster, ner,queri	2004

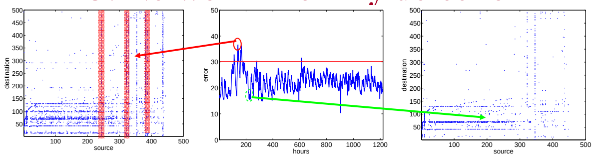
DB

DM

- Two groups are correctly identified: Databases and Data mining
- People and concepts are drifting over time

CMU SCS

P3: Network anomaly detection



- Reconstruction error gives indication of anomalies.
- Prominent difference between normal and abnormal ones is mainly due to the unusual scanning activity (confirmed by the campus admin).

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-55

CMU SCS

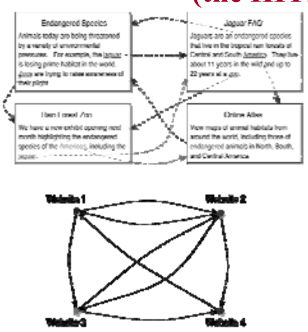
P4: Web graph mining [Kolda+ ICDM 2005]

- How to order the importance of web pages?
  - Kleinberg's algorithm HITS
  - PageRank
  - Tensor extension on HITS (TOPHITS)

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-56

CMU SCS

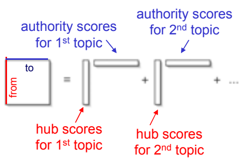
Kleinberg's Hubs and Authorities (the HITS method)



Sparse adjacency matrix and its SVD:

$$x_{ij} = \begin{cases} 1 & \text{if page } i \text{ links to page } j \\ 0 & \text{otherwise} \end{cases}$$

$$\mathbf{X} \approx \sum_r \sigma_r \, \mathbf{h}_r \circ \mathbf{a}_r$$

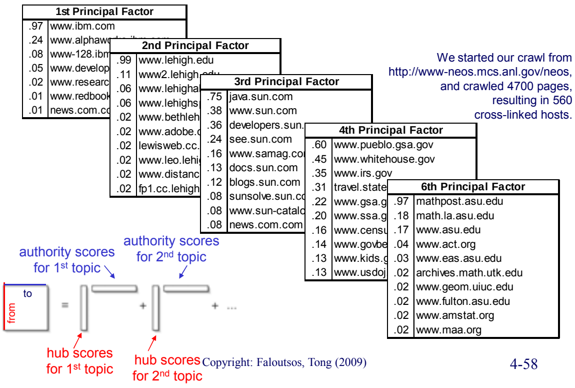


ICDE'09 Copyright: Faloutsos, Tong (2009) 4-57

Kleinberg, JACM, 1999

CMU SCS

HITS Authorities on Sample Data



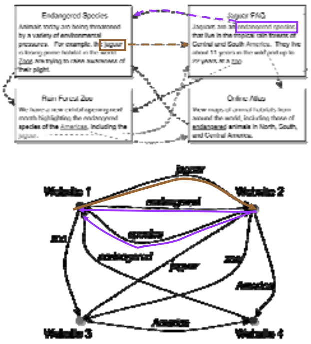
We started our crawl from http://www-neos.mcs.anl.gov/neos, and crawled 4700 pages, resulting in 560 cross-linked hosts.

Copyright: Faloutsos, Tong (2009)

4-58

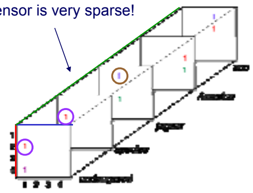
CMU SCS

Three-Dimensional View of the Web



$$x_{ijk} = \begin{cases} 1 & \text{if page } i \to \text{page } j \text{ with term } k \\ 0 & \text{otherwise} \end{cases}$$

Observe that this tensor is very sparse!



[Kolda, Bader, Kenny, ICDM05]

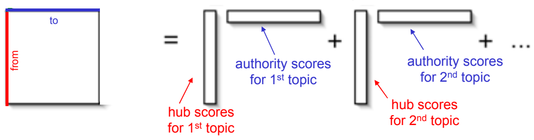
4-59

CMU SCS

Topical HITS (TOPHITS)

Main Idea: Extend the idea behind the HITS model to incorporate term (i.e., topical) information.

$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, \mathbf{h}_r \circ \mathbf{a}_r$$



ICDE'09

[Kolda, Bader, Kenny, ICDM05]

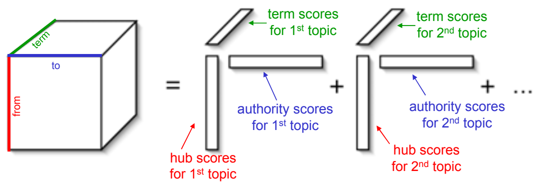
4-60

CMU SCS

Topical HITS (TOPHITS)

Main Idea: Extend the idea behind the HITS model to incorporate term (i.e., topical) information.

$$\mathcal{X} \approx \sum_{r=1}^{R} \lambda_r \, h_r \circ a_r \circ t_r$$

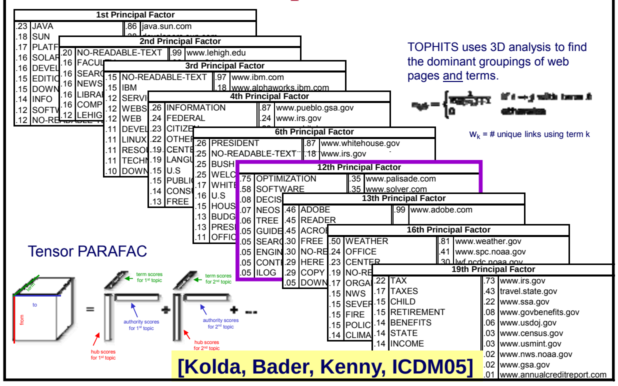


ICDE'09 4-61

[Kolda, Bader, Kenny, ICDM05]

CMU SCS

TOPHITS Terms & Authorities on Sample Data



TOPHITS uses 3D analysis to find the dominant groupings of web pages and terms.

w_k = # unique links using term k

Tensor PARAFAC

[Kolda, Bader, Kenny, ICDM05]

CMU SCS

Outline: Part 4

- Virus/influence propagation
- Blog analysis
- Community detection and tracking
- Tensor for web mining: TOPHITS
- ➡ Scalability

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-63

CMU SCS

Scalability

- How about handling huge graphs?

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-64

CMU SCS

Scalability

- Google: > 450,000 processors in clusters of ~2000 processors each [Barroso, Dean, Hölzle, *"Web Search for a Planet: The Google Cluster Architecture"* IEEE Micro 2003]
- Yahoo: 5Pb of data [Fayyad, KDD'07]
- Problem: machine failures, on a daily basis
- How to parallelize data mining tasks, then?
- A: map/reduce – hadoop (open-source clone) http://hadoop.apache.org/

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-65

CMU SCS

2' intro to hadoop

- master-slave architecture; n-way replication (default n=3)
- 'group by' of SQL (in parallel, fault-tolerant way)
- e.g, find histogram of word frequency
  - compute local histograms
  - then merge into global histogram

```
select course-id, count(*)
from ENROLLMENT
group by course-id
```

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-66

CMU SCS

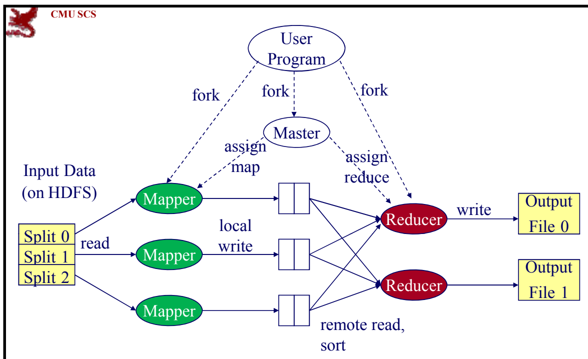
2’ intro to hadoop

- master-slave architecture; n-way replication (default n=3)
- ‘group by’ of SQL (in parallel, fault-tolerant way)
- e.g, find histogram of word frequency
  - compute local histograms
  - then merge into global histogram

select course-id, count(*) reduce
from ENROLLMENT
group by course-id map

ICDE’09 Copyright: Faloutsos, Tong (2009) 4-67

CMU SCS



By default: 3-way replication;
Late/dead machines: ignored, transparently (!)

ICDE’09 Copyright: Faloutsos, Tong (2009)

CMU SCS

D.I.S.C.

- ‘Data Intensive Scientific Computing’ [R. Bryant, CMU]
  - ‘big data’
  - http://www.cs.cmu.edu/~bryant/pubdir/cmu-cs-07-128.pdf

ICDE’09 Copyright: Faloutsos, Tong (2009) -69

CMU SCS

E.g.: self-* and DCO systems @ CMU

- >200 nodes
- target: 1 PetaByte
- Greg Ganger +:
  - www.pdl.cmu.edu/SelfStar
  - www.pdl.cmu.edu/DCO

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-70

CMU SCS

OVERALL CONCLUSIONS

Graphs pose a wealth of fascinating problems

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-71

CMU SCS

OVERALL CONCLUSIONS

- Part 1: Patterns & Generators
  - self-similarity and power laws work, when textbook methods fail!
  - New patterns (shrinking diameter, triangle laws, NLCC, WPL, etc)
  - [New generators: Kronecker, Butterfly]

ICDE, 2009 Copyright: Faloutsos, Tong (2009) 4-72

CMU SCS

OVERALL CONCLUSIONS

- Part 2: Powerful Tools
  - Matrix tools (SVD, HITS, PageRank)
  - Tensor analysis (PARAFAC, Tucker)
- Part 3: Proximity
  - Fast algorithms
  - Link prediction
  - Proximity tracking over time

 Copyright: Faloutsos, Tong (2009)

CMU SCS

OVERALL CONCLUSIONS

- Part 4: Applications
  - Virus propagation
  - Blog analysis
  - Community tracking with MDL
  - Tensor for web mining: TOPHITS
  - Scalability (map/reduce & hadoop)

 Copyright: Faloutsos, Tong (2009)

CMU SCS

Acknowledgements

Funding:

National Science Foundation — WHERE DISCOVERIES BEGIN

LAWRENCE LIVERMORE NATIONAL LABORATORY

IBM YAHOO! Sprint

PITA (PA Inf. Tech. Alliance)

 Copyright: Faloutsos, Tong (2009)

CMU SCS

Thank you!

For papers, data, code:

Christos Faloutsos
www.cs.cmu.edu/~christos

Hanghang Tong
www.cs.cmu.edu/~htong

ICDE'09 Copyright: Faloutsos, Tong (2009) 4-76