

# Mining Heterogeneous Information Networks

Jiawei Han<sup>†</sup>   Yizhou Sun<sup>†</sup>   Xifeng Yan<sup>§</sup>   Philip S. Yu<sup>‡</sup>

<sup>†</sup>University of Illinois at Urbana-Champaign

<sup>§</sup>University of California at Santa Barbara

<sup>‡</sup>University of Illinois at Chicago

Acknowledgements: NSF, ARL, NASA, AFOSR (MURI), Microsoft, IBM, Yahoo!, Google, HP Lab & Boeing

July 12, 2010

---

# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
  - **Part I:** Clustering, Ranking and Classification
    - Clustering and Ranking in Information Networks
    - Classification of Information Networks
  - **Part II:** Data Quality and Search in Information Networks
    - Data Cleaning and Data Validation by InfoNet Analysis
    - Similarity Search in Information Networks
  - **Part III:** Advanced Topics on Information Network Analysis
    - Role Discovery and OLAP in Information Networks
    - Mining Evolution and Dynamics of Information Networks
  - **Conclusions**
-

# Outline

---

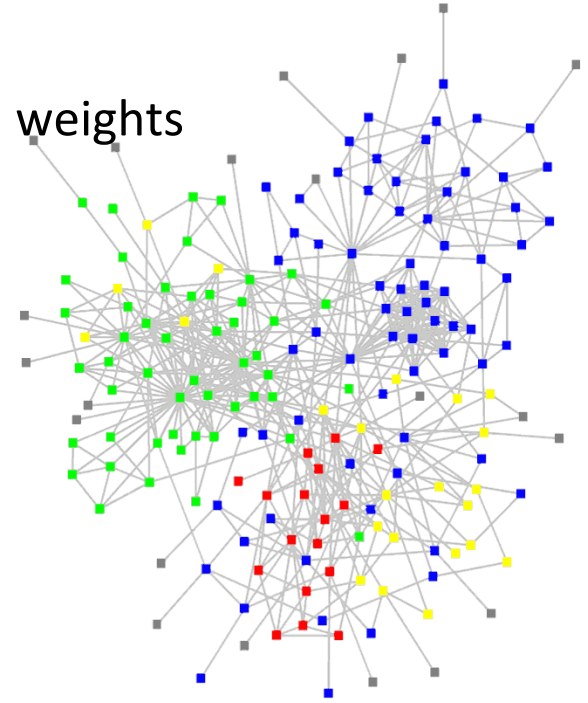


- **Motivation: Why Mining Heterogeneous Information Networks?**
  - **Part I: Clustering, Ranking and Classification**
    - Clustering and Ranking in Information Networks
    - Classification of Information Networks
  - **Part II: Data Quality and Search in Information Networks**
    - Data Cleaning and Data Validation by InfoNet Analysis
    - Similarity Search in Information Networks
  - **Part III: Advanced Topics on Information Network Analysis**
    - Role Discovery and OLAP in Information Networks
    - Mining Evolution and Dynamics of Information Networks
  - **Conclusions**
-

# What Are Information Networks?

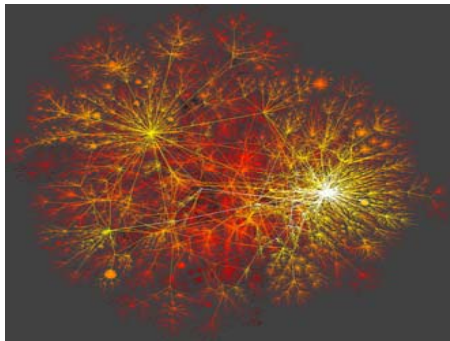
---

- Information network: A network where each node represents an entity (e.g., actor in a social network) and each link (e.g., tie) a relationship between entities
  - Each node/link may have attributes, labels, and weights
  - Link may carry rich semantic information
- Homogeneous vs. heterogeneous networks
  - Homogeneous networks
    - Single object type and single link type
    - Single model social networks (e.g., friends)
    - WWW: a collection of linked Web pages
  - Heterogeneous, multi-typed networks
    - Multiple object and link types
    - Medical network: patients, doctors, disease, contacts, treatments
    - Bibliographic network: publications, authors, venues

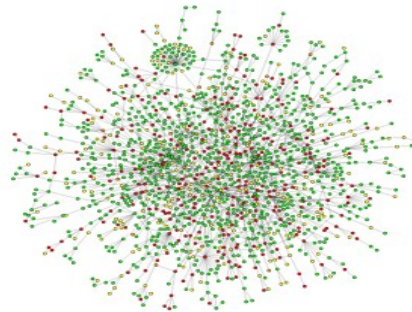


# Ubiquitous Information Networks

- Graphs and substructures
  - Chemical compounds, computer vision objects, circuits, XML
- Biological networks
- Bibliographic networks: DBLP, ArXiv, PubMed, ...
- Social networks: Facebook >100 million active users
- World Wide Web (WWW): > 3 billion nodes, > 50 billion arcs
- Cyber-physical networks



An Internet Web



Yeast protein interaction network

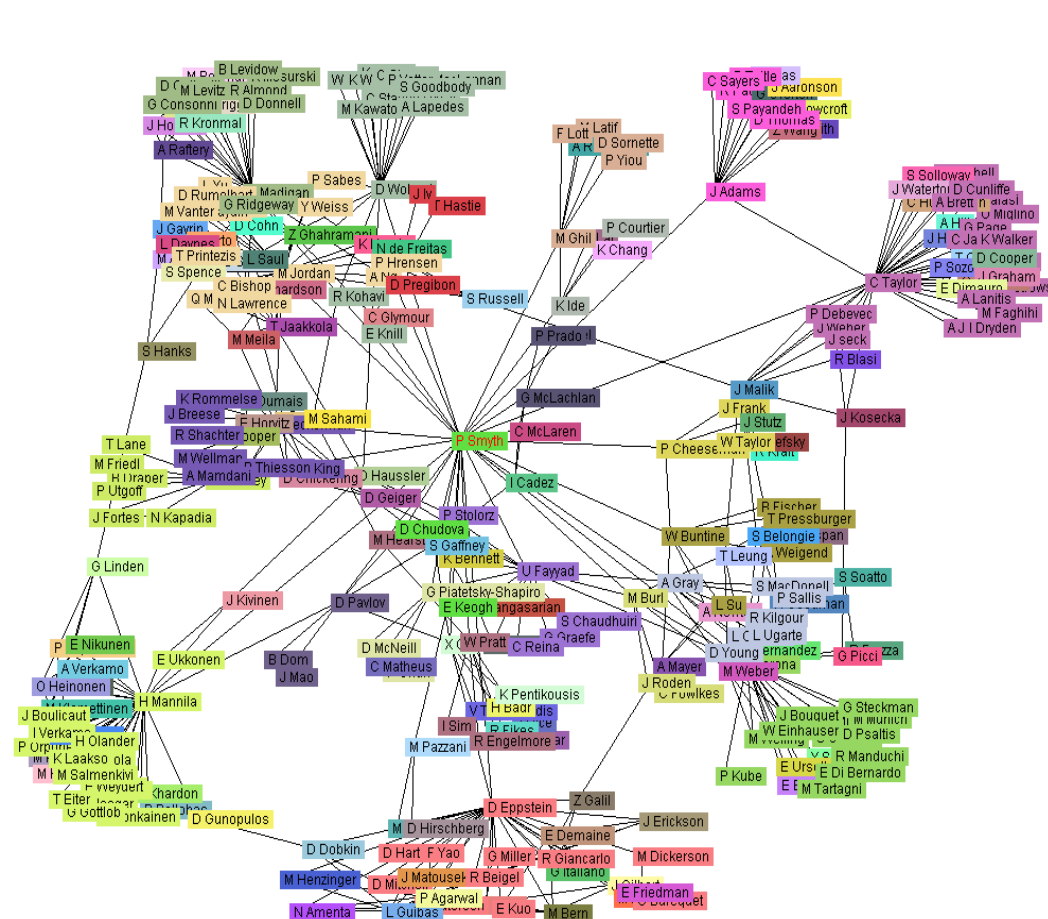


Co-author network

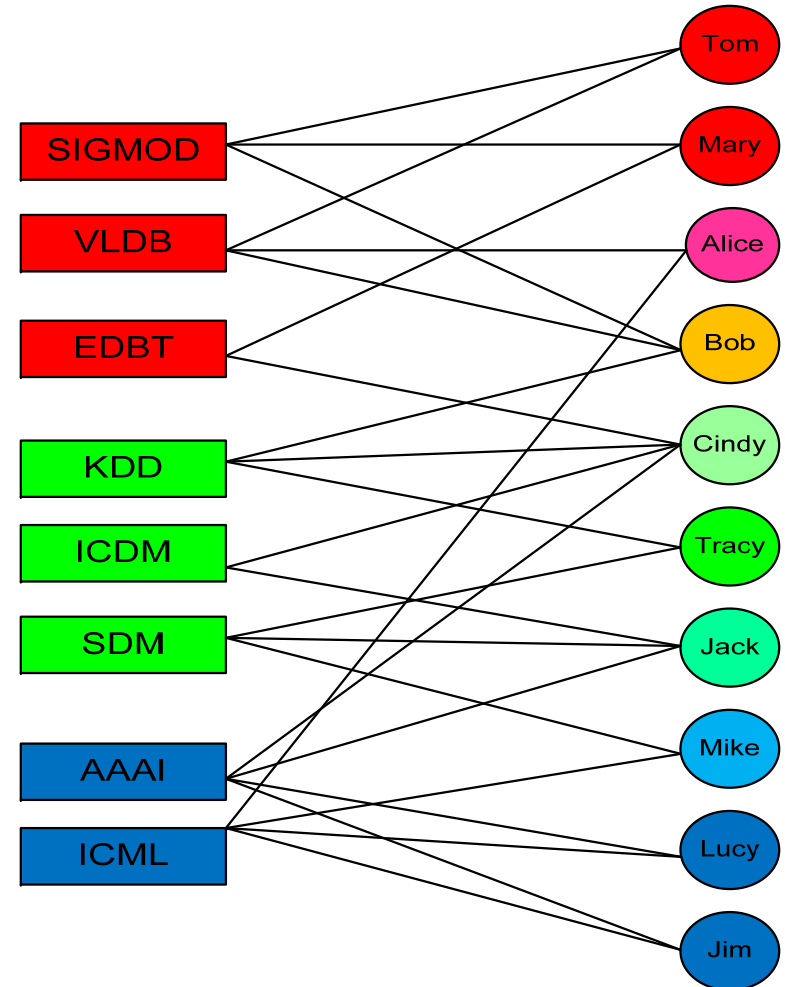


Social network sites

# Homogeneous vs. Heterogeneous Networks



Co-author Network



Conference-Author Network

# DBLP: An Interesting and Familiar Network

---

- DBLP: A computer science publication bibliographic database
  - 1.4 M records (papers), 0.7 M authors, 5 K conferences, ...
- Will this database disclose interesting knowledge about computer science research?
  - What are the popular research fields/subfields in CS?
  - Who are the leading researchers on DB or XQueries?
  - How do the authors in this subfield collaborate and evolve?
  - How many Wei Wang's in DBLP, which paper done by which?
  - Who is Sergy Brin's supervisor and when?
  - Who are very similar to Christos Faloutsos? .....
- All these kinds of questions, and potentially much more, can be nicely answered by the DBLP-InfoNet
  - How? Exploring the power of links in information networks!

# Homo. vs. Hetero.: Differences in DB-InfoNet Mining

---

- Homogeneous networks can often be derived from their original heterogeneous networks
  - Coauthor networks can be derived from author-paper-conference networks by projection on authors only
  - Paper citation networks can be derived from a complete bibliographic network with papers and citations projected
- Heterogeneous DB-InfoNet carries richer information than its corresponding projected homogeneous networks
- Typed heterogeneous InfoNet vs. non-typed hetero. InfoNet (i.e., not distinguishing different types of nodes)
  - Typed nodes and links imply a more structured InfoNet, and thus often lead to more informative discovery
- Our emphasis: Mining “structured” information networks!




# Why Mining Heterogeneous Information Networks?

---

- Most datasets can be “organized” or “transformed” into a “*structured*” heterogeneous information network!
  - Examples: DBLP, IMDB, Flickr, Google News, Wikipedia, ...
  - Structures can be progressively extracted from less organized data sets by information network analysis
  - Information-rich, inter-related, organized data sets form one or a set of gigantic, interconnected, multi-typed heterogeneous information networks
  - Surprisingly rich knowledge can be derived from such structured heterogeneous information networks
- Our goal: Uncover knowledge hidden from “organized” data
  - Exploring the power of multi-typed, heterogeneous links
  - Mining “structured” heterogeneous information networks!


# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
- **Part I: Clustering, Ranking and Classification**
  - **Clustering and Ranking in Information Networks** 
  - Classification of Information Networks
- **Part II: Data Quality and Search in Information Networks**
  - Data Cleaning and Data Validation by InfoNet Analysis
  - Similarity Search in Information Networks
- **Part III: Advanced Topics on Information Network Analysis**
  - Role Discovery and OLAP in Information Networks
  - Mining Evolution and Dynamics of Information Networks
- **Conclusions**

# Clustering and Ranking in Information Networks

---

- Integrated Clustering and Ranking of Heterogeneous Information Networks 
- Clustering of Homogeneous Information Networks
  - LinkClus: Clustering with link-based similarity measure
  - SCAN: Density-based clustering of networks
  - Others
    - Spectral clustering
    - Modularity-based clustering
    - Probabilistic model-based clustering
- User-Guided Clustering of Information Networks

# Clustering and Ranking in Heterogeneous Information Networks

---

- Ranking & clustering each provides a new view over a network
- Ranking globally without considering clusters → dumb
  - Ranking DB and Architecture confs. together?
- Clustering authors in one huge cluster without distinction?
  - Dull to view thousands of objects (this is why PageRank!)
- RankClus: Integrates clustering with ranking
  - Conditional ranking relative to clusters
  - Uses highly ranked objects to improve clusters
- Qualities of clustering and ranking are mutually enhanced
- Y. Sun, J. Han, et al., “*RankClus: Integrating Clustering with Ranking for Heterogeneous Information Network Analysis*”, EDBT'09.

# Global Ranking vs. Cluster-Based Ranking

- A Toy Example: Two areas with 10 conferences and 100 authors in each area

Table 1: A set of conferences from two research areas

DB/DM	{SIGMOD, VLDB, PODS, ICDE, ICDT, KDD, ICDM, CIKM, PAKDD, PKDD}
HW/CA	{ASPLOS, ISCA, DAC, MICRO, ICCAD, HPCA, ISLPED, CODES, DATE, VTS }

Table 2: Top-10 ranked conferences and authors in the mixed conference set

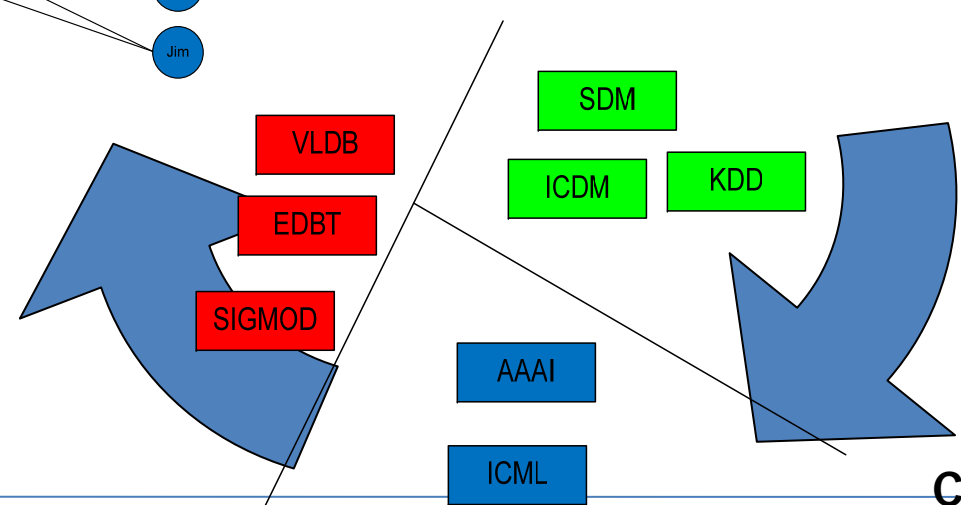
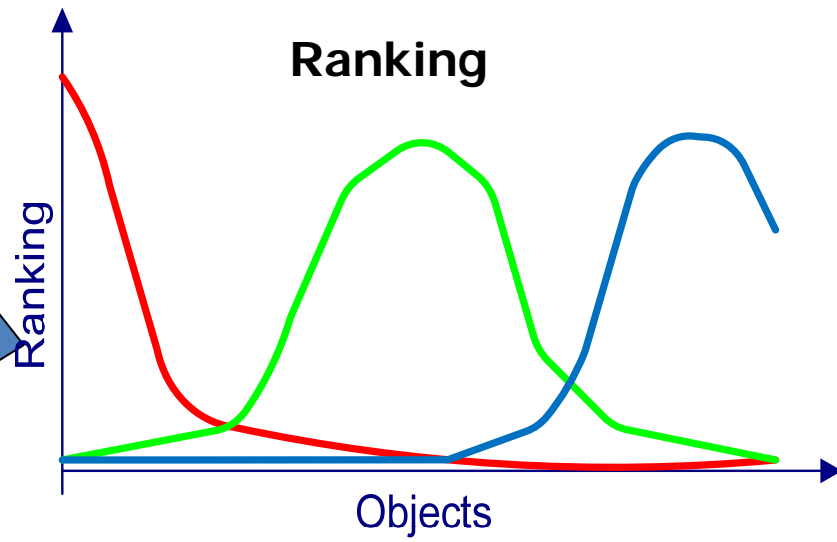
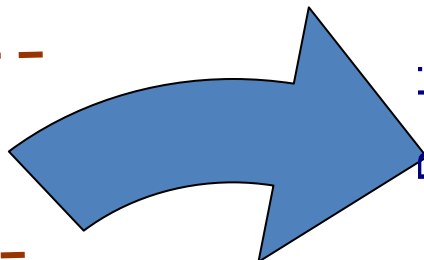
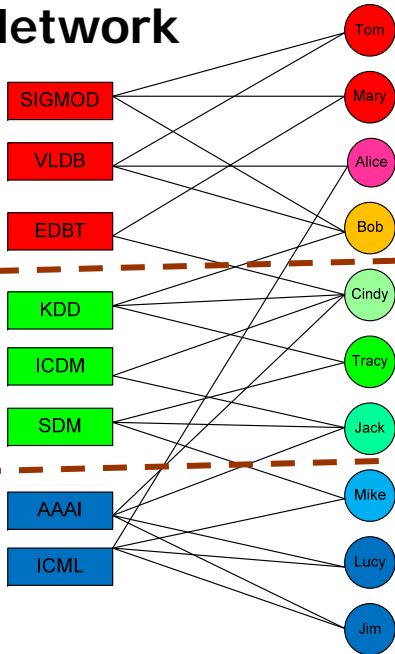
Rank	Conf.	Rank	Authors
1	DAC	1	Alberto L. Sangiovanni-Vincentelli
2	ICCAD	2	Robert K. Brayton
3	DATE	3	Massoud Pedram
4	ISLPED	4	Miodrag Potkonjak
5	VTS	5	Andrew B. Kahng
6	CODES	6	Kwang-Ting Cheng
7	ISCA	7	Lawrence T. Pileggi
8	VLDB	8	David Blaauw
9	SIGMOD	9	Jason Cong
10	ICDE	10	D. F. Wong

Table 3: Top-10 ranked conferences and authors in the DB/DM set

Rank	Conf.	Rank	Authors
1	VLDB	1	H. V. Jagadish
2	SIGMOD	2	Surajit Chaudhuri
3	ICDE	3	Divesh Srivastava
4	PODS	4	Michael Stonebraker
5	KDD	5	Hector Garcia-Molina
6	CIKM	6	Jeffrey F. Naughton
7	ICDM	7	David J. DeWitt
8	PAKDD	8	Jiawei Han
9	ICDT	9	Rakesh Agrawal
10	PKDD	10	Raghu Ramakrishnan

# RankClus: A New Framework

Sub-Network



Clustering

# The RankClus Philosophy

---

- Why integrated Ranking and Clustering?
  - Ranking and clustering can be mutually improved
  - Ranking: Once a cluster becomes more accurate, ranking will be more reasonable for such a cluster and will be the distinguished feature of the cluster
  - Clustering: Once ranking is more distinguished from each other, the clusters can be adjusted and get more accurate results
- Not every object should be treated equally in clustering!
- Objects preserve similarity under new measure space
  - E.g., VLDB vs. SIGMOD

# RankClus: Algorithm Framework

---

- Step 0. Initialization
  - Randomly partition target objects into  $K$  clusters
- Step 1. Ranking
  - Ranking for each sub-network induced from each cluster, which serves as feature for each cluster
- Step 2. Generating new measure space
  - Estimate mixture model coefficients for each target object
- Step 3. Adjusting cluster
- Step 4. Repeating Steps 1-3 until stable



# Focus on a Bi-Typed Network Case

---

- Conference-author network, links can exist between
  - Conference (X) and author (Y)
  - Author (Y) and author (Y)

DEFINITION 1. *Bi-type Information Network.* Given two types of object sets  $X$  and  $Y$ , where  $X = \{x_1, x_2, \dots, x_m\}$ , and  $Y = \{y_1, y_2, \dots, y_n\}$ , graph  $G = \langle V, E \rangle$  is called a bi-type information network on types  $X$  and  $Y$ , if  $V(G) = X \cup Y$  and  $E(G) = \{\langle o_i, o_j \rangle\}$ , where  $o_i, o_j \in X \cup Y$ .

- Use  $W$  to denote the links and their weights

$$W = \begin{pmatrix} W_{XX} & W_{XY} \\ W_{YX} & W_{YY} \end{pmatrix}$$

# Step 1: Ranking: Feature Extraction

---

- Two ranking strategies: Simple ranking vs. authority ranking
- Simple Ranking
  - Proportional to degree counting for objects, e.g., # of publications of an author
  - Considers only immediate neighborhood in the network
- Authority Ranking: Extension to HITS in weighted bi-type network
  - Rule 1: Highly ranked authors publish *many* papers in highly ranked conferences
  - Rule 2: Highly ranked conferences attract *many* papers from *many* highly ranked authors
  - Rule 3: The rank of an author is enhanced if he or she co-authors with many authors or many highly ranked authors

# Encoding Rules in Authority Ranking

---

- Rule 1: Highly ranked authors publish *many* papers in highly ranked conferences

$$\vec{r}_Y(j) = \sum_{i=1}^m W_{YX}(j, i) \vec{r}_X(i).$$

- Rule 2: Highly ranked conferences attract *many* papers from *many* highly ranked authors

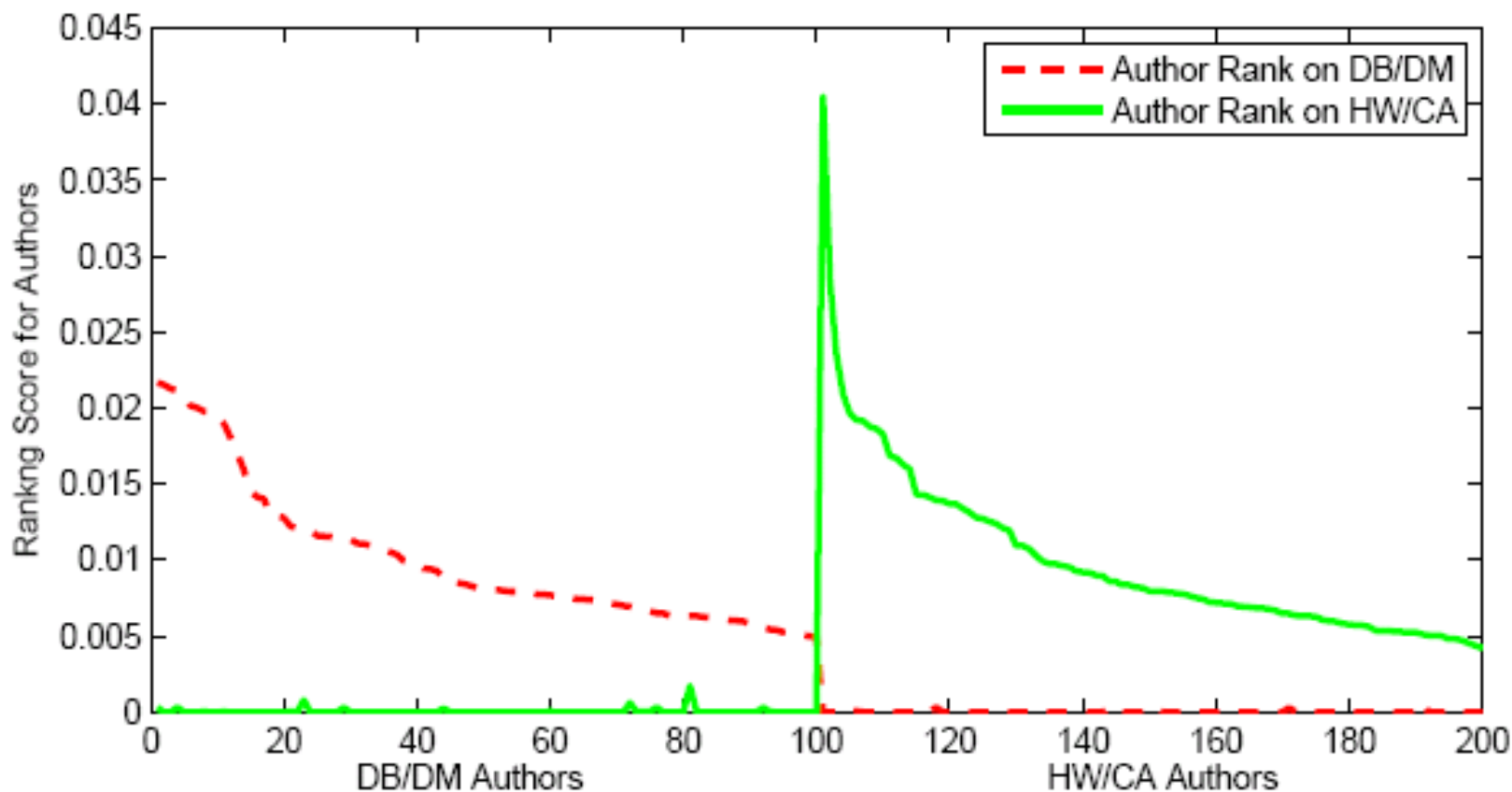
$$\vec{r}_X(i) = \sum_{j=1}^n W_{XY}(i, j) \vec{r}_Y(j)$$

- Rule 3: The rank of an author is enhanced if he or she co-authors with many authors or many highly ranked authors

$$\vec{r}_Y(i) = \alpha \sum_{j=1}^m W_{YX}(i, j) \vec{r}_X(j) + (1 - \alpha) \sum_{j=1}^n W_{YY}(i, j) \vec{r}_Y(j)$$

# Example: Authority Ranking in the 2-Area Conference-Author Network

- The rankings of authors are quite distinct from each other in the two clusters



# Step 2: Generate New Measure Space: A Mixture Model Method

---

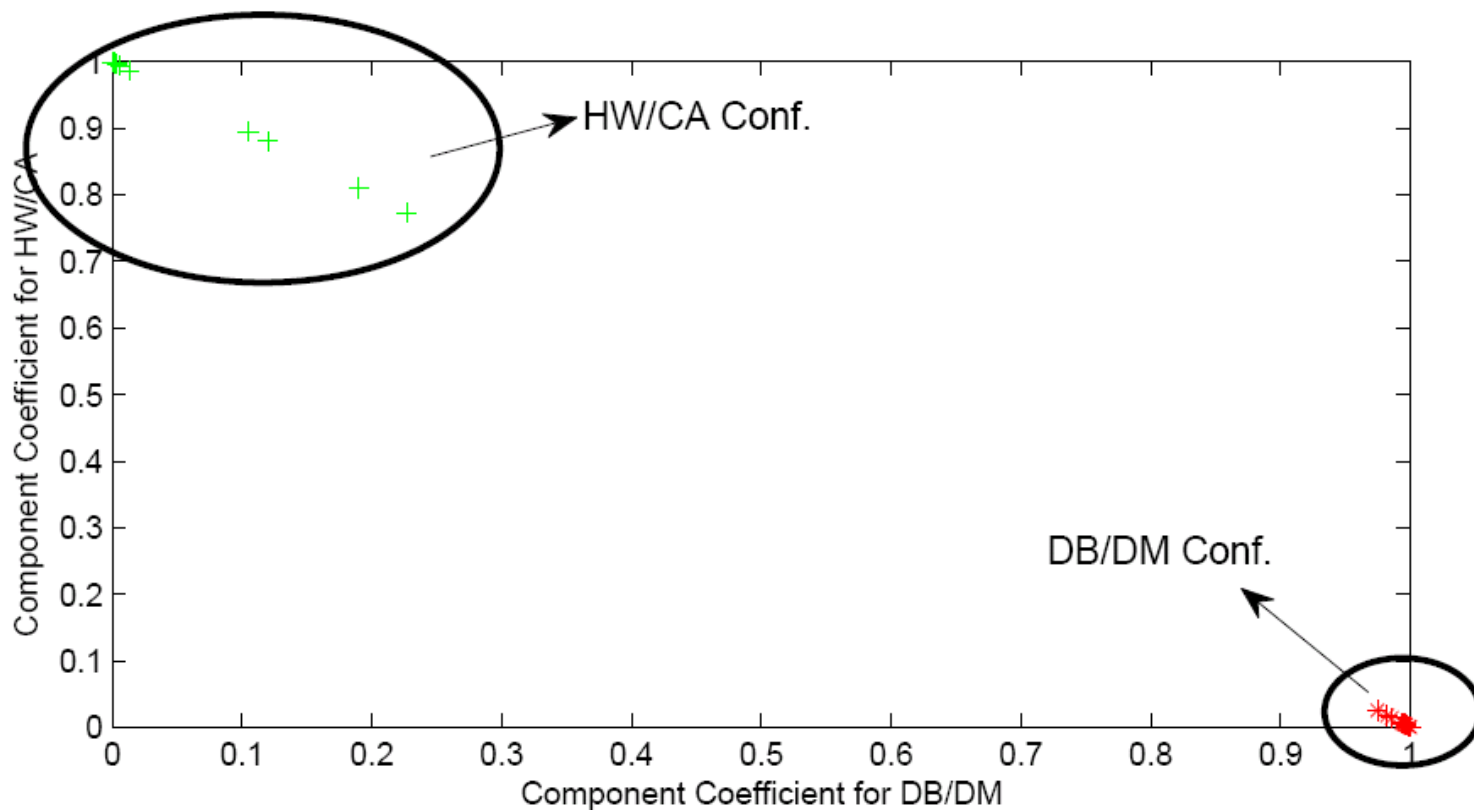
- Consider each target object's links are generated under a mixture distribution of ranking from each cluster
  - Consider ranking as a distribution:  $r(Y) \rightarrow p(Y)$

- $$p_{x_i}(Y) = \sum_{k=1}^K \pi_{i,k} p_k(Y), \text{ and } \sum_{k=1}^K \pi_{i,k} = 1$$

- Each target object  $x_i$  is mapped into a K-vector  $(\pi_{i,k})$
- Parameters are estimated using the EM algorithm
  - Maximize the log-likelihood given all the observations of links

# Example: 2-D Coefficients in the 2-Area Conference-Author Network

- The conferences are well separated in the new measure space



- Scatter plots of two conferences and component coefficients

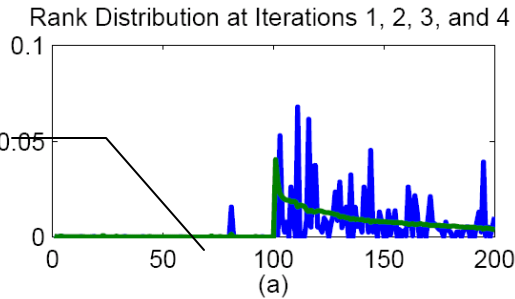
# Step 3: Cluster Adjustment in New Measure Space

---

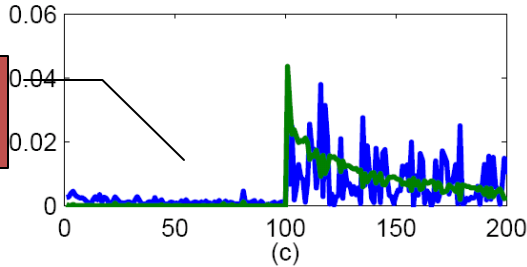
- Cluster center in new measure space
  - Vector mean of objects in the cluster (K-dimensional)
- Cluster adjustment
  - Distance measure: 1-Cosine similarity
  - Assign to the cluster with the nearest center
- **Why Better Ranking Function Derives Better Clustering?**
  - Consider the measure space generation process
    - Highly ranked objects in a cluster play a more important role to decide a target object's new measure
  - Intuitively, if we can find the highly ranked objects in a cluster, equivalently, we get the right cluster

# Step-by-Step Running Case Illustration

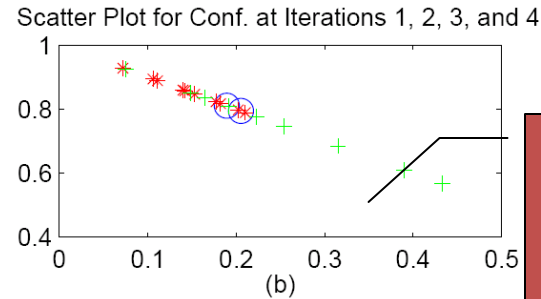
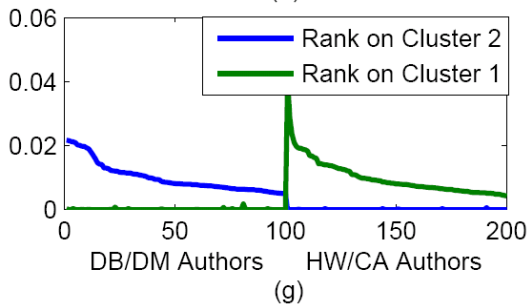
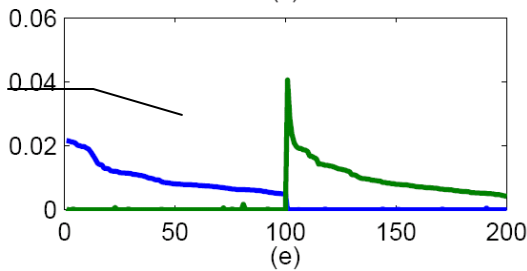
Initially, ranking distributions are mixed together



Improved a little



Improved significantly

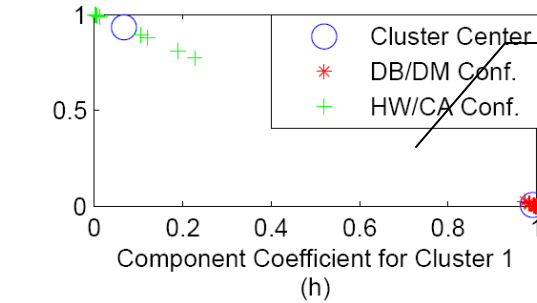
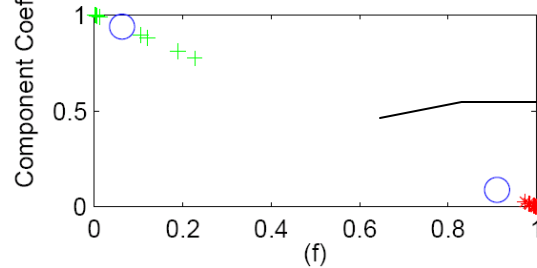
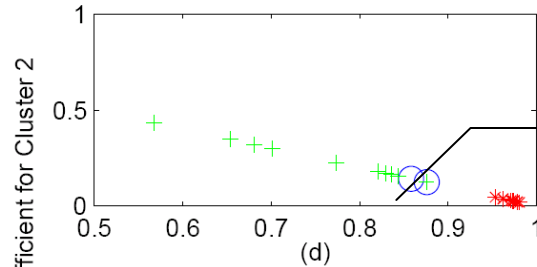


Two clusters of objects mixed together, but preserve similarity somehow

Two clusters are almost well separated

Well separated

Stable





# Time Complexity: Linear to # of Links

---

- At each iteration,  $|E|$ : edges in network,  $m$ : number of target objects,  $K$ : number of clusters
  - Ranking for sparse network
    - $\sim O(|E|)$
  - Mixture model estimation
    - $\sim O(K|E| + mK)$
  - Cluster adjustment
    - $\sim O(mK^2)$
- In all, linear to  $|E|$ 
  - $\sim O(K|E|)$
- Note: SimRank will be at least quadratic at each iteration since it evaluates distance between every pair in the network

# Case Study: Dataset: DBLP

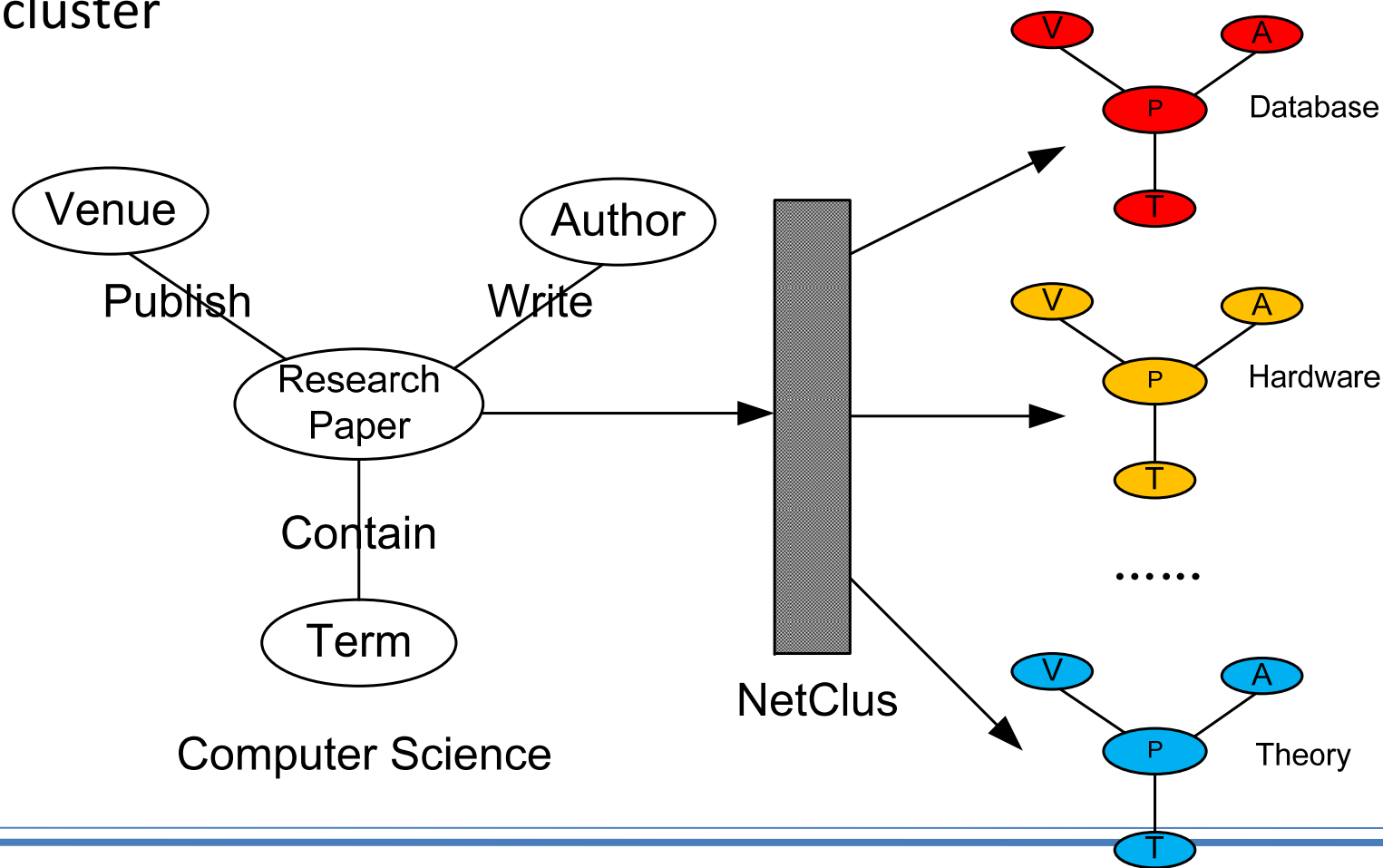
- All the 2676 conferences and 20,000 authors with most publications, from the time period of year 1998 to year 2007
- Both conference-author relationships and co-author relationships are used
- K=15 (select only 5 clusters here)

**Table 5: Top-10 Conferences in 5 Clusters Using RANKCLUS**

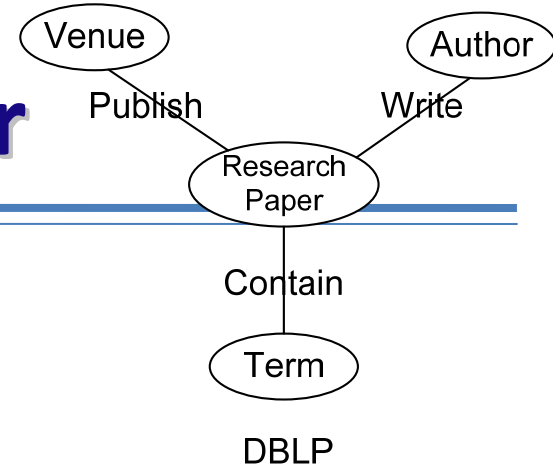
	DB	Network	AI	Theory	IR
1	VLDB	INFOCOM	AAMAS	SODA	SIGIR
2	ICDE	SIGMETRICS	IJCAI	STOC	ACM Multimedia
3	SIGMOD	ICNP	AAAI	FOCS	CIKM
4	KDD	SIGCOMM	Agents	ICALP	TREC
5	ICDM	MOBICOM	AAAI/IAAI	CCC	JCDL
6	EDBT	ICDCS	ECAI	SPAA	CLEF
7	DASFAA	NETWORKING	RoboCup	PODC	WWW
8	PODS	MobiHoc	IAT	CRYPTO	ECDL
9	SSDBM	ISCC	ICMAS	APPROX-RANDOM	ECIR
10	SDM	SenSys	CP	EUROCRYPT	CIVR

# NetClus: Ranking & Clustering with Star Network Schema

- Beyond bi-typed information network: A Star Network Schema
- Split a network into different layers, each representing by a net-cluster



# StarNet: Schema & Net-Cluster



- Star Network Schema
  - Center type: Target type
    - E.g., a paper, a movie, a tagging event
    - A center object is a **co-occurrence** of a bag of different types of objects, which stands for a **multi-relation** among different types of objects
  - Surrounding types: Attribute (property) types
- NetCluster
  - Given a information network  $G$ , a net-cluster  $C$  contains two pieces of information:
    - Node set and link set as a sub-network of  $G$
    - Membership indicator for each node  $x$ :  $P(x \text{ in } C)$
  - Given a information network  $G$ , cluster number  $K$ , a clustering for  $G$  is a set of net-clusters and for each node  $x$ , the sum of  $x$ 's probability distribution in all  $K$  net-clusters should be 1

# Jiawei Han via

List of publications from the [DBLP Bibliography Server](#) - [FAQ](#) [Facets and more with CompleteSearch](#)

[Coauthor Index](#) - [Ask others: ACM DL/Guide](#) - [CiteSeer](#) - [CSB](#) - [MetaPress](#) - [Google](#) - [MSN](#) - [Yahoo](#)

[Home Page](#)

		2009
361	EE	Jiawei Han, Xifeng Yan, Philip S. Yu: Scalable OLAP and mining of information networks. <a href="#">EDBT 2009: 1159</a>
360	EE	Yizhou Sun, Jiawei Han, Peixiang Zhao, Zhiyun Yin, Hong Cheng, Tianyi Wu: RankClus: integrating clustering with ranking for heterogeneous information network analysis. <a href="#">EDBT 2009: 565-576</a>
359	EE	Bhavani M. Thuraisingham, Latifur Khan, Murat Kantarcioglu, Sonia Chib, Jiawei Han, Sang Son: Real-Time Knowledge Discovery and Dissemination for Intelligence Analysis. <a href="#">HICSS 2009: 1-12</a>
358	EE	Bolin Ding, David Lo, Jiawei Han, Siau-Cheng Khoo: Efficient Mining of Closed Repetitive Gapped Subsequences from a Sequence Database. <a href="#">ICDE 2009: 1024-1035</a>
357	EE	Xiaolei Li, Zhenhui Li, Jiawei Han, Jae-Gil Lee: Temporal Outlier Detection in Vehicle Traffic Data. <a href="#">ICDE 2009: 1319-1322</a>
356	EE	Mohammad M. Masud, Jing Gao, Latifur Khan, Jiawei Han, Bhavani M. Thuraisingham: A Multi-partition Multi-chunk Ensemble Technique to Classify Concept-Drifting Data Streams. <a href="#">PAKDD 2009: 363-375</a>
		2008

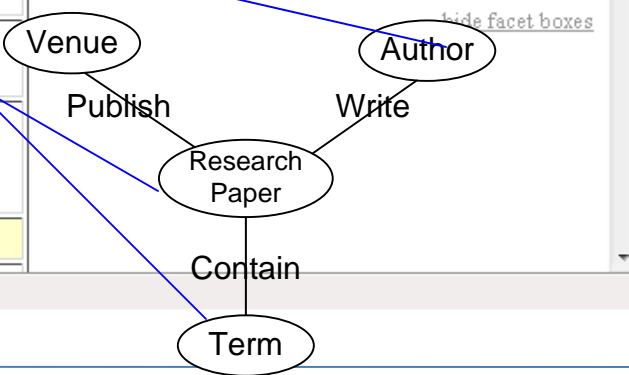
Refine by VENUE

[International Journal of Software Engineering and Knowledge Engineering \(IJSEKE\) \(1\)](#)  
[\[all 1\]](#)

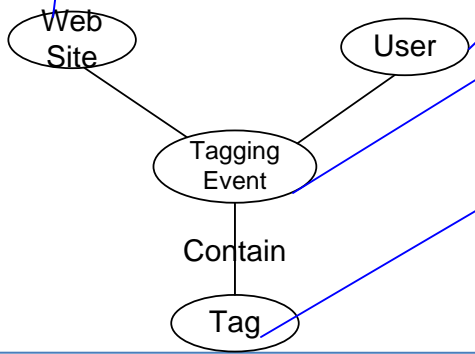
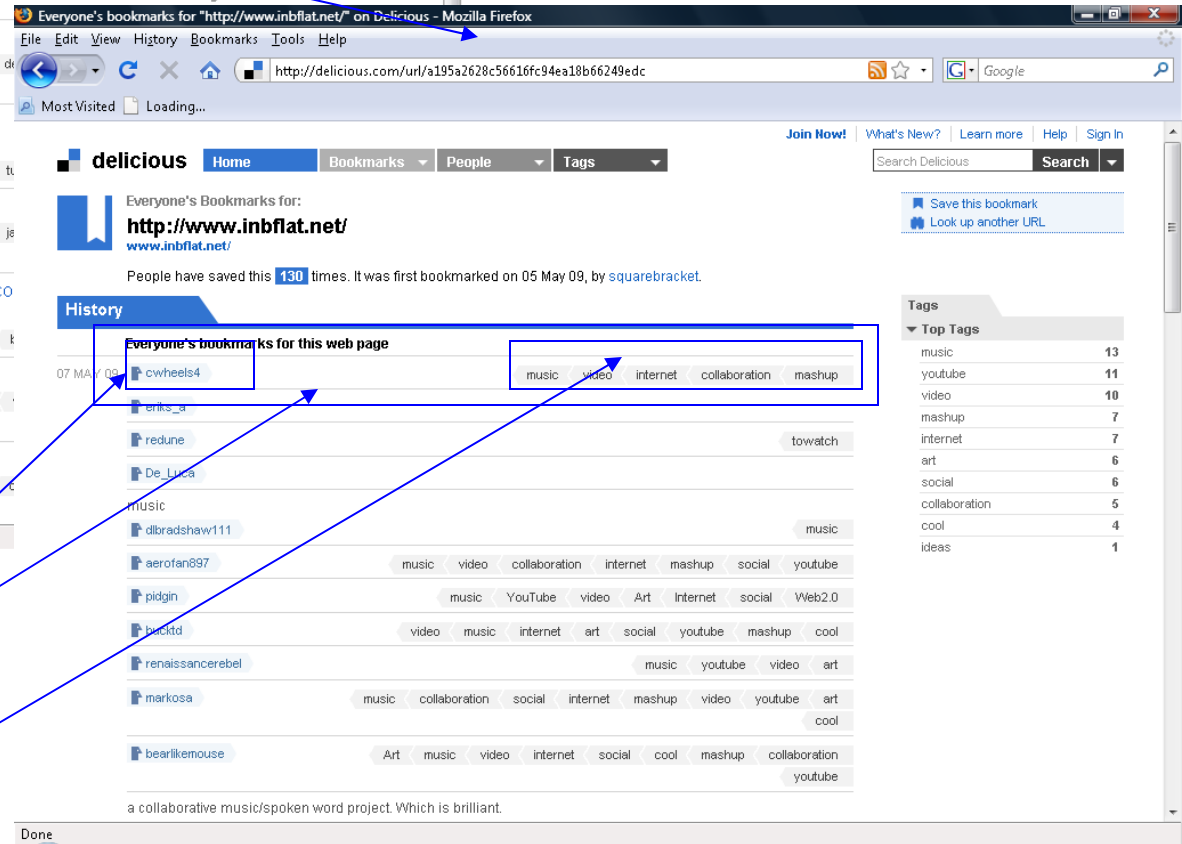
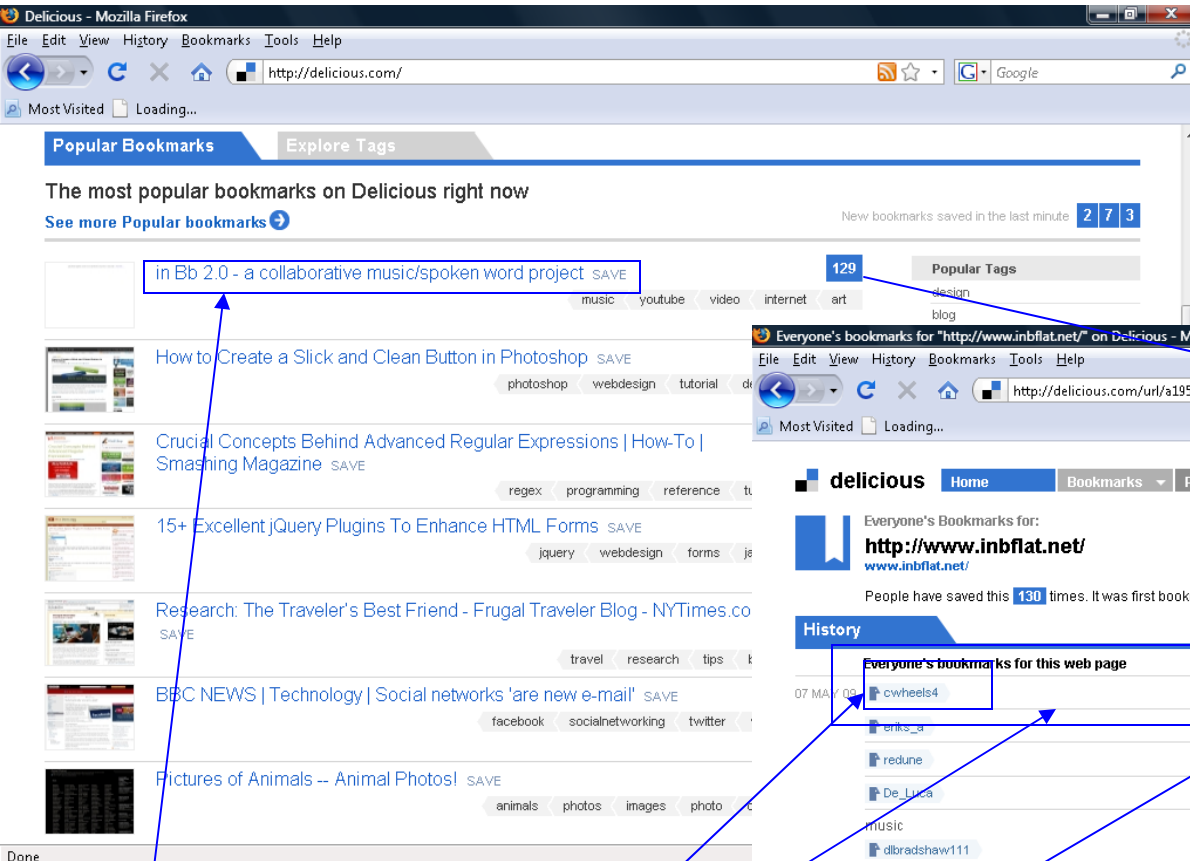
---

Refine by YEAR

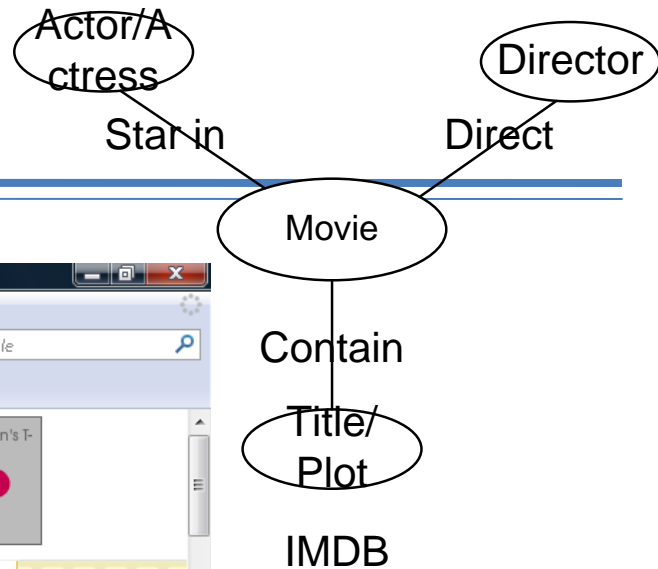
[1992 \(1\)](#)  
[\[all 1\]](#)



# StarNet of Delicious.com



# StartNet for IMDB



The Shawshank Redemption (1994) - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://www.imdb.com/title/tt0111161/#comment

Most Visited Loading...

**IMDb** The Internet Movie Database

NOW PLAYING MOVIE/TV NEWS MY MOVIES DVD & BLU-RAY IMDb TV MESSAGE BOARDS SHOWTIMES & TICKETS

Home | Top Movies | Photos | Independent Film | GameBase | Browse | Help

search All go more | tips

IMDb > The Shawshank Redemption (1994)

## The Shawshank Redemption (1994) [More at IMDbPro >>](#)

Photos [\(see all 70 | slideshow\)](#) Videos

Watch It

[Watch it at Amazon](#)

[Buy it at Amazon](#)

[Discuss in Boards](#)

[More at IMDb Pro](#)

[Add to My Movies](#)

[Update Data](#)

**Overview**

User Rating: 9.2/10 [421,872 votes](#)  
[Top 250: #1](#) [\(register to vote\)](#)

MOVIEmeter: Up 8% in popularity this week. See [why](#) on [IMDbPro](#).

Director: [Frank Darabont](#)

Writers [\(WGA\)](#): [Stephen King](#) (short story "Rita Hayworth and Shawshank Redemption")  
[Frank Darabont](#) (screenplay)

Contact: View [company](#) contact information for The Shawshank Redemption on [IMDbPro](#).

CoastalContacts.com  
World's Largest Optical Store

**Brand Name Contacts**

[Learn More](#)

Transferring data from ytaahg.vo.llnwd.net...

# Ranking Functions

---

- Ranking an object  $x$  of type  $T_x$  in a network  $G$ , denoted as  $p(x|T_x, G)$ 
  - Give a score to each object, according to its importance
- Different rules defined different ranking functions:
  - Simple Ranking
    - Ranking score is assigned according to the degree of an object
  - Authority Ranking
    - Ranking score is assigned according to the mutual enhancement by propagations of score through links
    - *“highly ranked conferences accept many good papers published by many highly ranked authors ,and highly ranked authors publish many good papers in highly ranked conferences”:*

$$P(C|T_C, G) = W_{CD} D_{DA}^{-1} W_{DA} P(A|T_A, G)$$

$$P(A|T_A, G) = W_{AD} D_{DC}^{-1} W_{DC} P(C|T_C, G)$$



# Ranking Function (Cont.)

---

- Priors can be added:
  - $P_p(X|T_x, G_k) = (1 - \lambda_p) P(X|T_x, G_k) + \lambda_p P_0(X|T_x, G_k)$ 
    - $P_0(X|T_x, G_k)$  is the prior knowledge, usually given as a distribution, denoted by only several words
    - $\lambda_p$  is the parameter that we believe in prior distribution
- Ranking distribution
  - Normalize ranking scores to 1, given them a probabilistic meaning
  - Similar to the idea of PageRank

# NetClus: Algorithm Framework

---

- Map each target object into a new low-dimensional feature space according to current net-clustering, and adjust the clustering further in the new measure space
  - Step 0: Generate initial random clusters
  - Step 1: Generate ranking-based generative model for target objects for each net-cluster
  - Step 2: Calculate posterior probabilities for target objects, which serves as the new measure, and assign target objects to the nearest cluster accordingly
  - Step 3: Repeat steps 1 and 2, until clusters do not change significantly
  - Step 4: Calculate posterior probabilities for attribute objects in each net-cluster

# Generative Model for Target Objects Given a Net-cluster

---

- Each target object stands for an co-occurrence of a bag of attribute objects
  - Define the probability of a target object  $\Leftrightarrow$  define the probability of the co-occurrence of all the associated attribute objects
- Generative probability  $P(d|G_k)$  for target object  $d$  in cluster  $C_k$  :

$$p(d|G_k) = \prod_{x \in N_{G_k}(d)} p(x|T_x, G_k)^{W_{d,x}} p(T_x|G_k)^{W_{d,x}}$$

where  $P(x | T_x, G_k)$  is ranking function,  $P(T_x | G_k)$  is type probability

- Two assumptions of independence
  - The probabilities to visit objects of different types are independent to each other
  - The probabilities to visit two objects within the same type are independent to each other

# Cluster Adjustment

---

- Using posterior probabilities of target objects as new feature space
  - Each target object => K-dimension vector
  - Each net-cluster center => K-dimension vector
    - Average on the objects in the cluster
  - Assign each target object into nearest cluster center (e.g., cosine similarity)
- A sub-network corresponding to a new net-cluster is then built
  - by extracting all the target objects in that cluster and all linked attribute objects

# Experiments: DBLP and Beyond

- Data Set: DBLP “all-area” data set
  - All conferences + “Top” 50K authors
- DBLP “four-area” data set
  - 20 conferences from DB, DM, ML, IR
  - All authors from these conferences
  - All papers published in these conferences
- Running case illustration

	Conf: KDD					Author: Michael Stonebraker					Term: Relational					Paper: SimRank[7]				
Iter	DB	DM	ML	IR	BC	DB	DM	ML	IR	BC	DB	DM	ML	IR	BC	DB	DM	ML	IR	BC
0	0.21	0.25	0.19	0.18	0.17	0.32	0.20	0.11	0.20	0.17	0.28	0.20	0.17	0.18	0.17	0.01	0.00	0.87	0.00	0.12
1	0.09	0.32	0.12	0.13	0.34	0.35	0.03	0.01	0.21	0.39	0.31	0.12	0.09	0.15	0.34	0.02	0.00	0.48	0.00	0.50
2	0.02	0.37	0.07	0.10	0.44	0.46	0.00	0.00	0.30	0.24	0.34	0.10	0.10	0.17	0.29	0.01	0.01	0.02	0.00	0.96
5	0.02	0.62	0.07	0.16	0.14	0.74	0.00	0.00	0.19	0.07	0.55	0.13	0.12	0.14	0.06	0.00	0.00	0.00	0.92	0.08
10	0.01	0.89	0.02	0.03	0.04	0.95	0.00	0.00	0.01	0.04	0.68	0.15	0.12	0.02	0.03	0.00	0.32	0.00	0.43	0.25
end	0.01	0.92	0.01	0.03	0.03	0.95	0.00	0.00	0.01	0.03	0.68	0.16	0.11	0.02	0.02	0.00	0.99	0.00	0.00	0.01

# Accuracy Study: Experiments

- Accuracy, compared with PLSA, a pure text model, no other types of objects and links are used, use the same prior as in NetClus

	NetClus (A+C+T+D)	PLSA (T+D)
Accuracy	<b>0.7705</b>	0.608

Accuracy of Paper Clustering Results

- Accuracy, compared with RankClus, a bi-typed clustering method on only one type

	RankClus $d(a) > 0$	RankClus $d(a) > 5$	RankClus $d(a) > 10$	NetClus $d(a) > 0$
NMI	0.5232	0.8390	0.7573	<b>0.9753</b>

Accuracy of Conference Clustering Results

# NetClus: Distinguishing Conferences

---

- AAI 0.0022667 0.00899168 **0.934024** 0.0300042 0.0247133
- CIKM 0.150053 0.310172 0.00723807 0.444524 0.0880127
- CVPR 0.000163812 0.00763072 **0.931496** 0.0281342 0.032575
- ECIR 3.47023e-05 0.00712695 0.00657402 **0.978391** 0.00787288
- ECML 0.00077477 0.110922 **0.814362** 0.0579426 0.015999
- EDBT **0.573362** 0.316033 0.00101442 0.0245591 0.0850319
- ICDE **0.529522** 0.376542 0.00239152 0.0151113 0.0764334
- ICDM 0.000455028 **0.778452** 0.0566457 0.113184 0.0512633
- ICML 0.000309624 0.050078 **0.878757** 0.0622335 0.00862134
- IJCAI 0.00329816 0.0046758 **0.94288** 0.0303745 0.0187718
- KDD 0.00574223 **0.797633** 0.0617351 0.067681 0.0672086
- PAKDD 0.00111246 **0.813473** 0.0403105 0.0574755 0.0876289
- PKDD 5.39434e-05 **0.760374** 0.119608 0.052926 0.0670379
- PODS **0.78935** 0.113751 0.013939 0.00277417 0.0801858
- SDM 0.000172953 **0.841087** 0.058316 0.0527081 0.0477156
- SIGIR 0.00600399 0.00280013 0.00275237 **0.977783** 0.0106604
- SIGMOD **0.689348** 0.223122 0.0017703 0.00825455 0.0775055
- VLDB **0.701899** 0.207428 0.00100012 0.0116966 0.0779764
- WSDM 0.00751654 0.269259 0.0260291 **0.683646** 0.0135497
- WWW 0.0771186 0.270635 0.029307 **0.451857** 0.171082

# NetClus: Database System Cluster

database 0.0995511  
 databases 0.0708818  
 system 0.0678563  
 data 0.0214893  
 query 0.0133316  
 systems 0.0110413  
 queries 0.0090603  
 management 0.00850744  
 object 0.00837766  
 relational 0.0081175  
 processing 0.00745875  
 based 0.00736599  
 distributed 0.0068367  
 xml 0.00664958  
 oriented 0.00589557  
 design 0.00527672  
 web 0.00509167  
 information 0.0050518  
 model 0.00499396  
 efficient 0.00465707

VLDB 0.318495  
 SIGMOD Conf. 0.313903  
 ICDE 0.188746  
 PODS 0.107943  
 EDBT 0.0436849

author	rank score
Serge Abiteboul	0.0472111
Victor Vianu	0.0348510
Jerome Simeon	0.0324529
Michael J. Carey	0.0288872
Sophie Chuet	0.0282911
Daniela Florescu	0.0241411
Sihem Amer-Yahia	0.0240869
Donald Kossmann	0.0232118
Wenfei Fan	0.0225235
Tova Milo	0.0202201
...	...

Ranking authors in XML

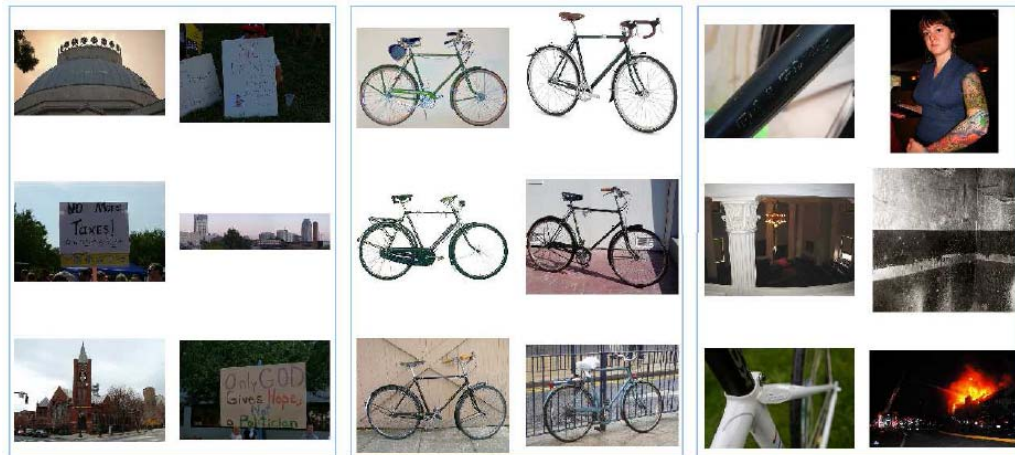
Surajit Chaudhuri 0.00678065  
 Michael Stonebraker 0.00616469  
 Michael J. Carey 0.00545769  
 C. Mohan 0.00528346  
 David J. DeWitt 0.00491615  
 Hector Garcia-Molina 0.00453497  
 H. V. Jagadish 0.00434289  
 David B. Lomet 0.00397865  
 Raghu Ramakrishnan 0.0039278  
 Philip A. Bernstein 0.00376314  
 Joseph M. Hellerstein 0.00372064  
 Jeffrey F. Naughton 0.00363698  
 Yannis E. Ioannidis 0.00359853  
 Jennifer Widom 0.00351929  
 Per-Ake Larson 0.00334911  
 Rakesh Agrawal 0.00328274  
 Dan Suciu 0.00309047  
 Michael J. Franklin 0.00304099  
 Umeshwar Dayal 0.00290143  
 Abraham Silberschatz 0.00278185



# NetClus: StarNet-Based Ranking and Clustering

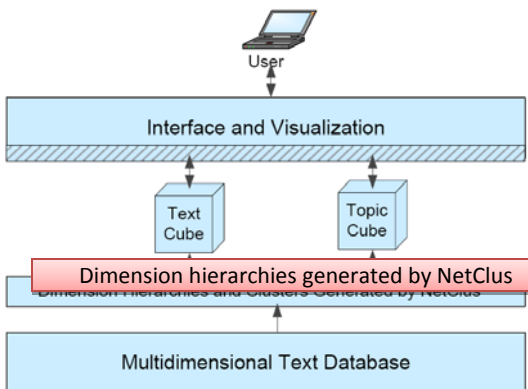
- A general framework in which ranking and clustering are successfully combined to analyze infornets
- Ranking and clustering can mutually reinforce each other in information network analysis
- NetClus, an extension to RankClus that integrates ranking and clustering and generate net-clusters in a star network with arbitrary number of types
- *Net-cluster*, heterogeneous information sub-networks comprised of multiple types of objects
- Go well beyond DBLP, and structured relational DBs

Flickr: query “Raleigh” derives multiple clusters

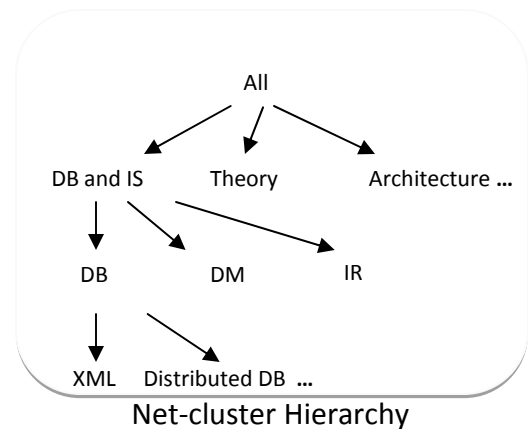


# iNextCube: Information Network-Enhanced Text Cube (VLDB'09 Demo)

Demo: [iNextCube.cs.uiuc.edu](http://iNextCube.cs.uiuc.edu)



Architecture of iNextCube



Net-cluster Hierarchy

Database and Information System

In area **Database and Information System**, the top ranked conferences/journals are:

Rank	Conf/Journal	Score
1	<a href="#">SIGMOD Conference</a>	0.075232
2	<a href="#">VLDB</a>	0.062318
3	<a href="#">ICDE</a>	0.053869
4	<a href="#">SIGIR</a>	0.048740
5	<a href="#">KDD</a>	0.028168
6	<a href="#">IEEE Trans. Knowl. Data Eng.</a>	0.024118
7	<a href="#">SIGMOD Record</a>	0.022818
8	<a href="#">IEEE Data Eng. Bull.</a>	0.020792
9	<a href="#">CIKM</a>	0.020606
10	<a href="#">ACM Trans. Database Syst.</a>	0.015887
11	<a href="#">PODS</a>	0.015577
12	<a href="#">TREC</a>	0.014045
13	<a href="#">Inf. Process. Manage.</a>	0.011904
14	<a href="#">ICDM</a>	0.011839
15	<a href="#">VLDB J.</a>	0.011544
16	<a href="#">EDBT</a>	0.011441
17	<a href="#">SIGIR Forum</a>	0.009575

Data Mining

In sub-area **Data Mining**, the top ranked authors are:

Rank	Author	Score
1	<a href="#">Philip S. Yu</a>	0.009881
2	<a href="#">Jiawei Han</a>	0.007163
3	<a href="#">Charu C. Aggarwal</a>	0.005638
4	<a href="#">Christos Faloutsos</a>	0.005140
5	<a href="#">Beng Chin Ooi</a>	0.003431
6	<a href="#">Ming-Syan Chen</a>	0.003309
7	<a href="#">Hans-Peter Kriegel</a>	0.003289
8	<a href="#">Wei Wang</a>	0.003160
9	<a href="#">Kian-Lee Tan</a>	0.003009
10	<a href="#">Nick Koudas</a>	0.002989
11	<a href="#">H. V. Jagadish</a>	0.002960

Database

In sub-area **Database**, the top ranked authors are:

Rank	Author	Score
1	<a href="#">David B. Lomet</a>	0.009049
2	<a href="#">Michael Stonebraker</a>	0.007072
3	<a href="#">Richard T. Snodgrass</a>	0.006152
4	<a href="#">David J. DeWitt</a>	0.004660
5	<a href="#">Surajit Chaudhuri</a>	0.004424
6	<a href="#">Michael J. Carey</a>	0.004195
7	<a href="#">Won Kim</a>	0.004167
8	<a href="#">Hector Garcia-Molina</a>	0.003793
9	<a href="#">Michael J. Franklin</a>	0.003773
10	<a href="#">Marianne Winslett</a>	0.003753
11	<a href="#">C. Mohan</a>	0.003580
12	<a href="#">Philip A. Bernstein</a>	0.003459
13	<a href="#">Arie Segev</a>	0.003191
14	<a href="#">Dan Suciu</a>	0.003189
15	<a href="#">Gerhard Weikum</a>	0.003043
16	<a href="#">Jennifer Widom</a>	0.003033
17	<a href="#">H. V. Jagadish</a>	0.002918
18	<a href="#">Jeffrey F. Naughton</a>	0.002911
19	<a href="#">Raghu Ramakrishnan</a>	0.002832
20	<a href="#">Joseph M. Hellerstein</a>	0.002793
21	<a href="#">Rakesh Agrawal</a>	0.002769
22	<a href="#">Umeshwar Dayal</a>	0.002763
23	<a href="#">Serge Abiteboul</a>	0.002737

Author/conference term ranking for research area. The research areas can be at different levels.


Algorithms and Theory of Computation

In area **Algorithms and Theory of Computation**, the top ranked conferences/journals are:

Rank	Conf/Journal	Rank	Author
1	<a href="#">STOC</a>	1	<a href="#">Andrew Chi-Chih Yao</a>
2	<a href="#">FOCS</a>	2	<a href="#">Christos H. Papadimitriou</a>
3	<a href="#">SIAM J. Comput.</a>	3	<a href="#">Robert Endre Tarjan</a>
4	<a href="#">SODA</a>	4	<a href="#">David Eppstein</a>
5	<a href="#">J. Comput. Syst. Sci.</a>	5	<a href="#">Micha Sharir</a>
6	<a href="#">J. ACM</a>	6	<a href="#">Avi Wigderson</a>
7	<a href="#">CoRR</a>	7	<a href="#">John H. Reif</a>
8	<a href="#">Theor. Comput. Sci.</a>	8	<a href="#">Bernard Chazelle</a>

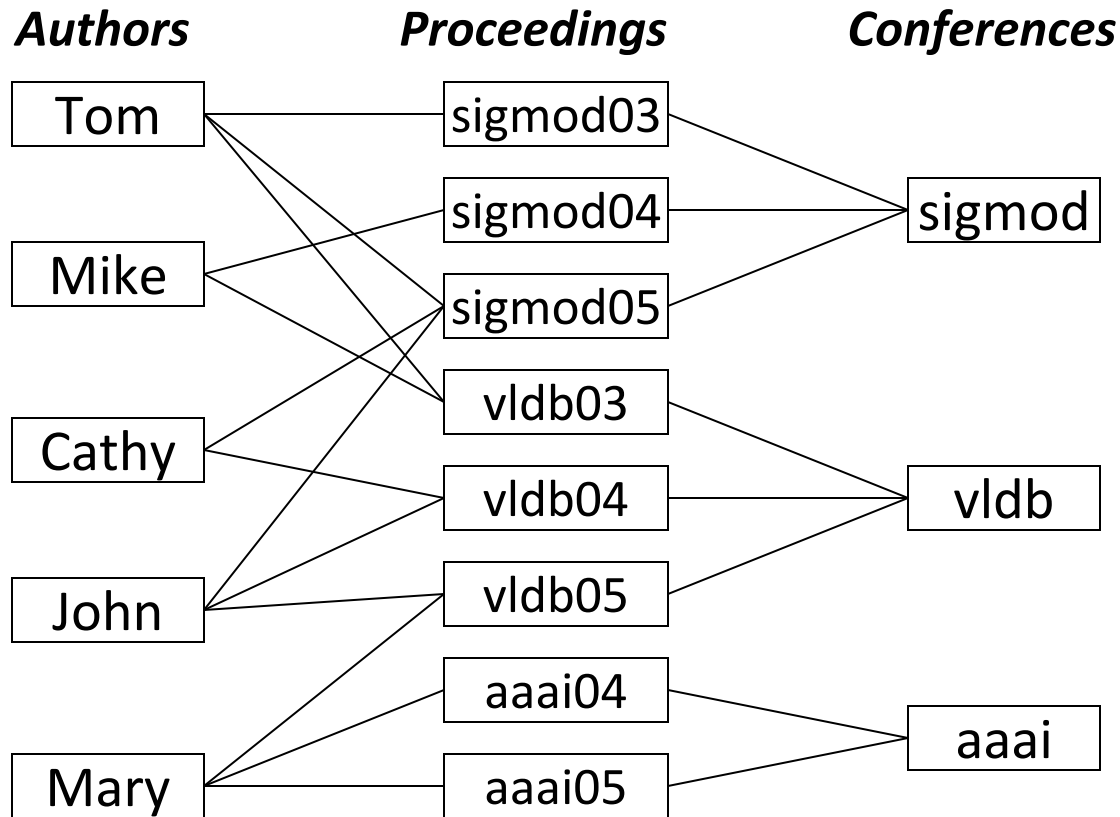
# Clustering and Ranking in Information Networks

---

- Integrated Clustering and Ranking of Heterogeneous Information Networks
  - Clustering of Homogeneous Information Networks
    - LinkClus: Clustering with link-based similarity measure
    - SCAN: Density-based clustering of networks
    - Others
      - Spectral clustering
      - Modularity-based clustering
      - Probabilistic model-based clustering
  - User-Guided Clustering of Information Networks
- 

# Link-Based Clustering: Why Useful?

---



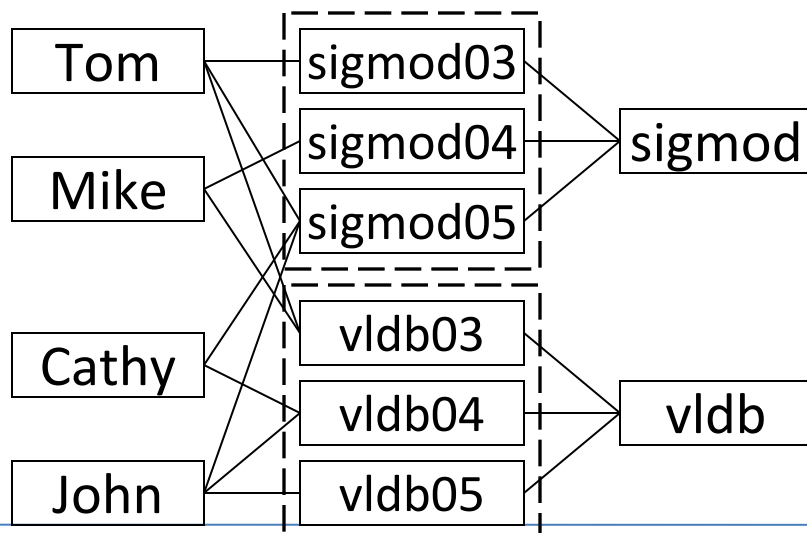
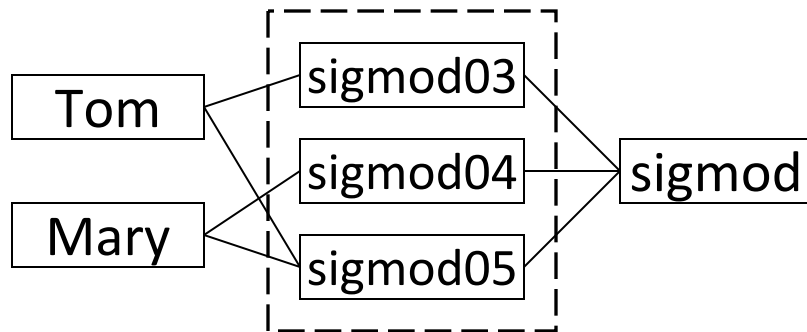
Questions:

Q1: How to cluster each type of objects?

Q2: How to define similarity between each type of objects?

# SimRank: Link-Based Similarities

- Two objects are similar if linked with the same or similar objects



$$\mathbf{S}(a, b) = \begin{cases} \frac{c}{|\mathcal{I}(a)||\mathcal{I}(b)|} \sum_{i=1}^{|\mathcal{I}(a)|} \sum_{j=1}^{|\mathcal{I}(b)|} \mathbf{S}(\mathcal{I}_i(a), \mathcal{I}_j(b)), & a \neq b \\ 1, & a = b \end{cases}$$

Jeh & Widom, 2002 - *SimRank*

Similarity between two objects  $a$  and  $b$ ,  $\mathbf{S}(a, b)$  = the average similarity between objects linked with  $a$  and those with  $b$ :

where  $\mathcal{I}(v)$  is the set of in-neighbors of the vertex  $v$ .

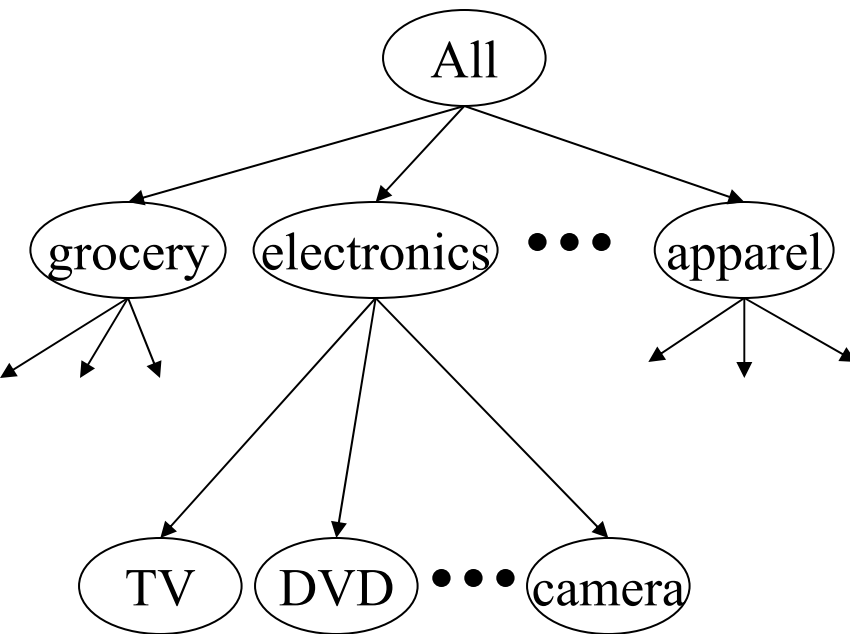
But: It is expensive to compute:

For a dataset of  $N$  objects and  $M$  links, it takes  $O(N^2)$  space and  $O(M^2)$  time to compute all similarities.

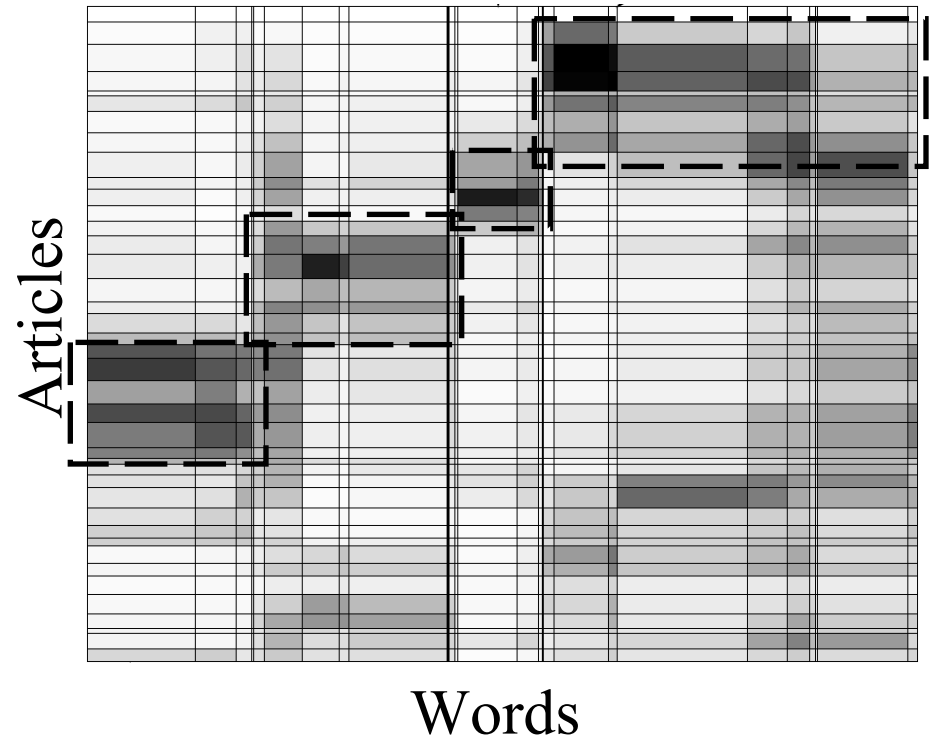
# Observation 1: Hierarchical Structures

- Hierarchical structures often exist naturally among objects (e.g., taxonomy of animals)

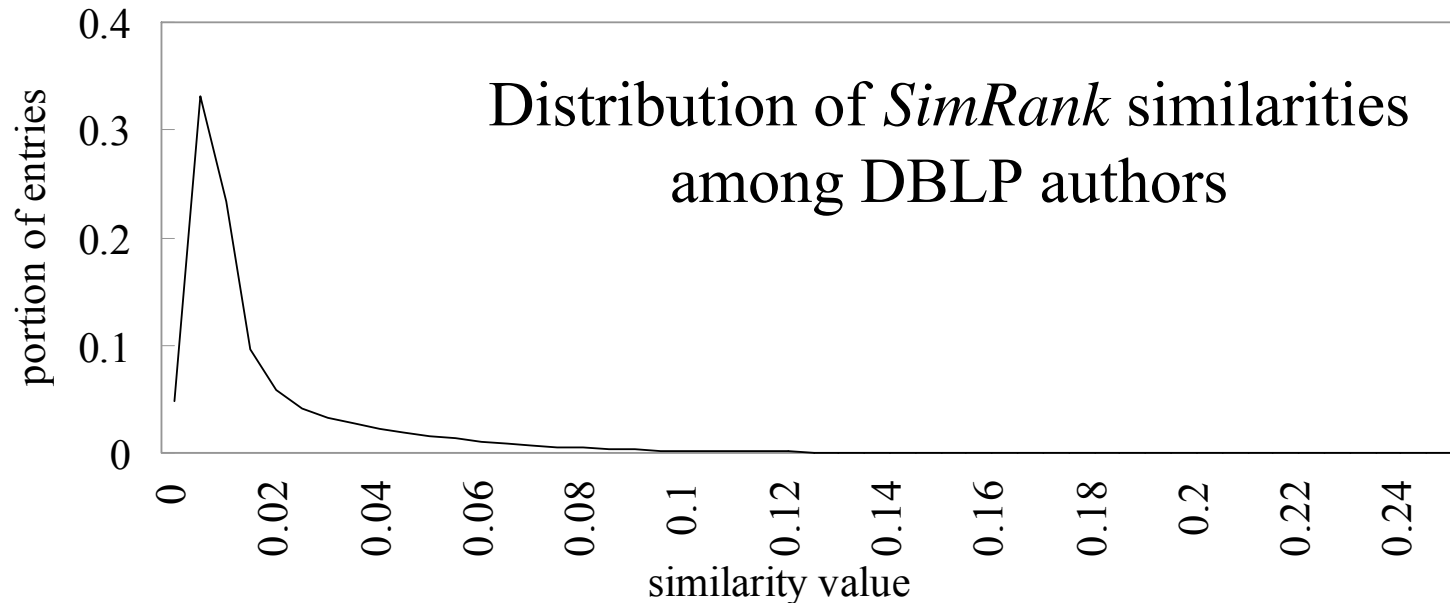
A hierarchical structure of products in Walmart



Relationships between articles and words (Chakrabarti, Papadimitriou, Modha, Faloutsos, 2004)

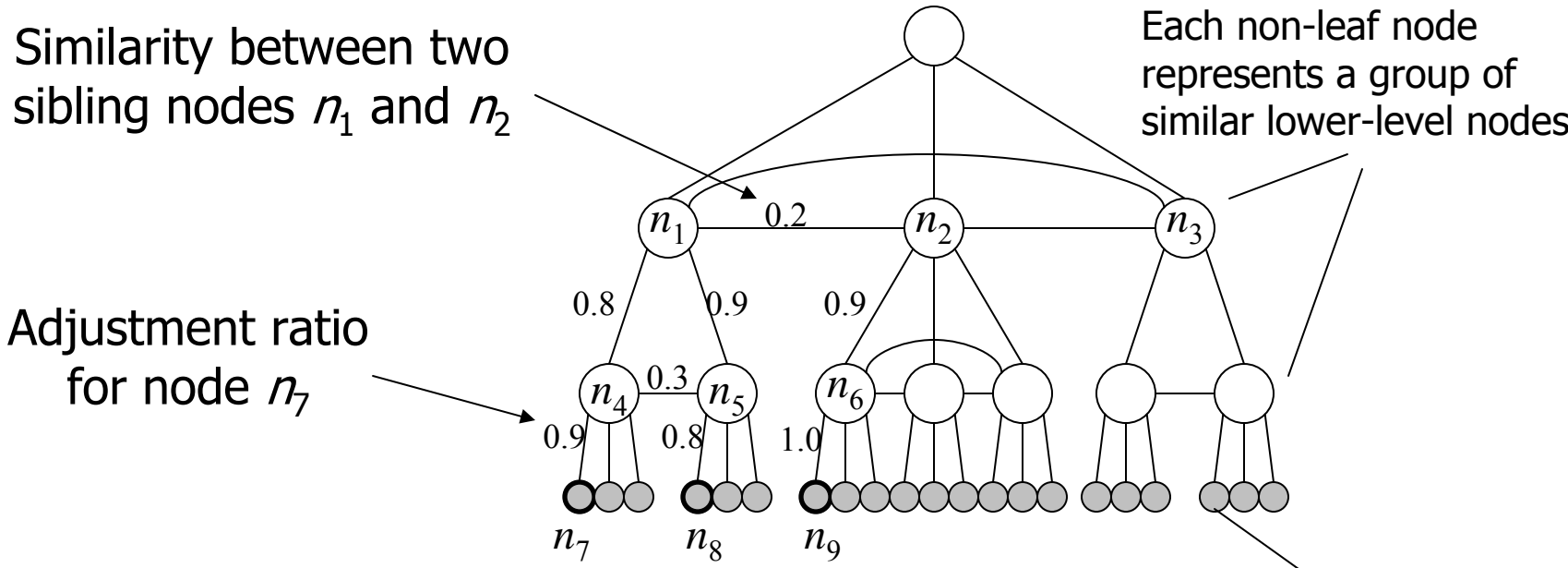


# Observation 2: Distribution of Similarity



- Power law distribution exists in similarities
  - 56% of similarity entries are in  $[0.005, 0.015]$
  - 1.4% of similarity entries are larger than 0.1
  - Our goal: Design a data structure that stores the significant similarities and compresses insignificant ones

# Our Data Structure: SimTree



- $sim_p(n_7, n_8) = s(n_7, n_4) \times s(n_4, n_5) \times s(n_5, n_8)$ 
  - Path-based node similarity
- Similarity between two nodes is the average similarity between objects linked with them in other SimTrees
- Adjustment ratio for  $x = \frac{\text{Average similarity between } x \text{ and all other nodes}}{\text{Average similarity between } x\text{'s parent and all other nodes}}$



# LinkClus: SimTree-Based Hierarchical Clustering

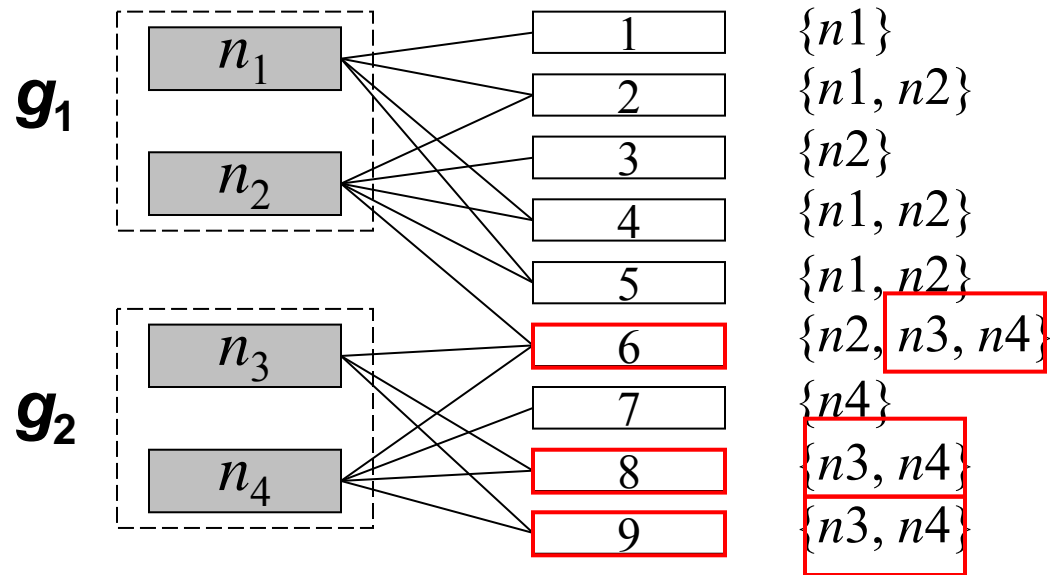
---

- Initialize a SimTree for objects of each type
- Repeat
  - For each SimTree, update the similarities between its nodes using similarities in other SimTrees
    - Similarity between two nodes  $a$  and  $b$  is the average similarity between objects linked with them
  - Adjust the structure of each SimTree
    - Assign each node to the parent node that it is most similar to

# Initialization of SimTrees

- Finding tight groups  $\longrightarrow$  Frequent pattern mining  
*Reduced to* Transactions

The tightness of a group of nodes is the support of a frequent pattern



- Initializing a tree:
  - Start from leaf nodes (level-0)
  - At each level  $l$ , find non-overlapping groups of similar nodes with frequent pattern mining

# Complexity: LinkClus vs. SimRank

- After initialization, iteratively (1) for each SimTree update the similarities between its nodes using similarities in other SimTrees, and (2) Adjust the structure of each SimTree
- Computational complexity:
  - For two types of objects,  $N$  in each, and  $M$  linkages between them

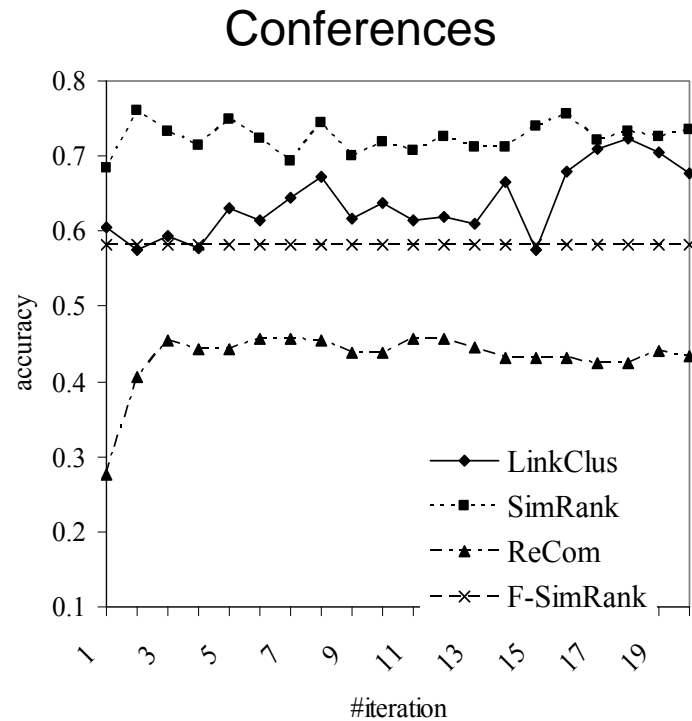
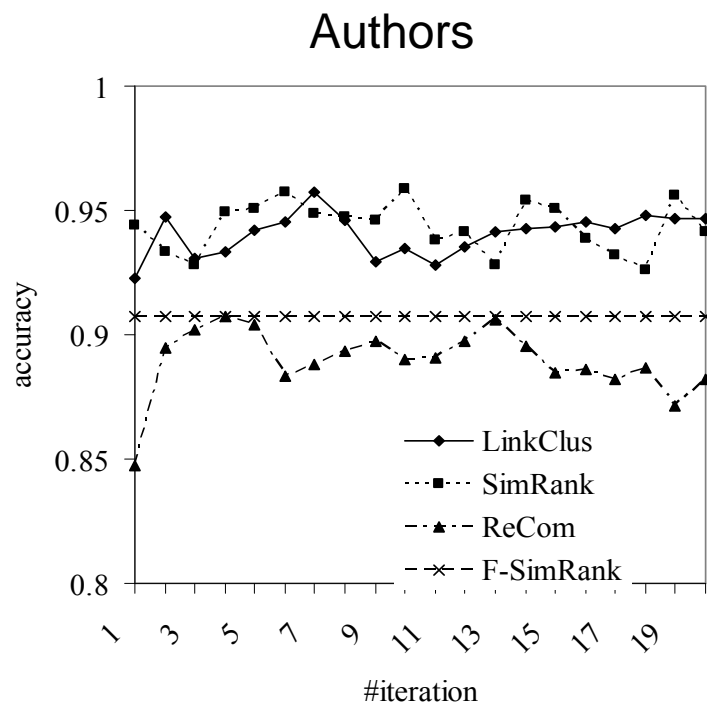
	Time	Space
Updating similarities	$O(M(\log M)^2)$	$O(M+N)$
Adjusting tree structures	$O(M)$	$O(M)$
<i>LinkClus</i>	$O(M(\log M)^2)$	$O(M+N)$
<i>SimRank</i>	$O(M^2)$	$O(N^2)$

# Performance Comparison: Experiment Setup

---

- DBLP dataset: 4170 most productive authors, and 154 well-known conferences with most proceedings
  - Manually labeled research areas of 400 most productive authors according to their home pages (or publications)
  - Manually labeled areas of 154 conferences according to their call for papers
- Approaches Compared:
  - SimRank (Jeh & Widom, KDD 2002)
    - Computing pair-wise similarities
  - SimRank with FingerPrints (F-SimRank)
    - Fogaras & R´acz, WWW 2005
    - pre-computes a large sample of random paths from each object and uses samples of two objects to estimate SimRank similarity
  - ReCom (Wang et al. SIGIR 2003)
    - Iteratively clustering objects using cluster labels of linked objects

# DBLP Data Set: Accuracy and Computation Time



Approaches	Accr-Author	Accr-Conf	average time
<b>LinkClus</b>	<b>0.957</b>	0.723	<b>76.7</b>
<b>SimRank</b>	<b>0.958</b>	<b>0.760</b>	1020
<b>ReCom</b>	0.907	0.457	43.1
<b>F-SimRank</b>	0.908	0.583	83.6

# Email Dataset: Accuracy and Time


---

- F. Nielsen. Email dataset.  
<http://www.imm.dtu.dk/~rem/data/Email-1431.zip>
- 370 emails on conferences, 272 on jobs, and 789 spam emails
- Why is LinkClus even better than SimRank in accuracy?
  - Noise filtering due to frequent pattern-based preprocessing

<i>Approach</i>	<i>Accuracy</i>	<i>Total time (sec)</i>
LinkClus	<b>0.8026</b>	1579.6
SimRank	0.7965	39160
ReCom	0.5711	74.6
F-SimRank	0.3688	479.7
CLARANS	0.4768	8.55

# Clustering and Ranking in Information Networks

---

- Integrated Clustering and Ranking of Heterogeneous Information Networks
- Clustering of Homogeneous Information Networks
  - LinkClus: Clustering with link-based similarity measure
  - SCAN: Density-based clustering of networks 
  - Others
    - Spectral clustering
    - Modularity-based clustering
    - Probabilistic model-based clustering
- User-Guided Clustering of Information Networks

# SCAN: Density-Based Network Clustering

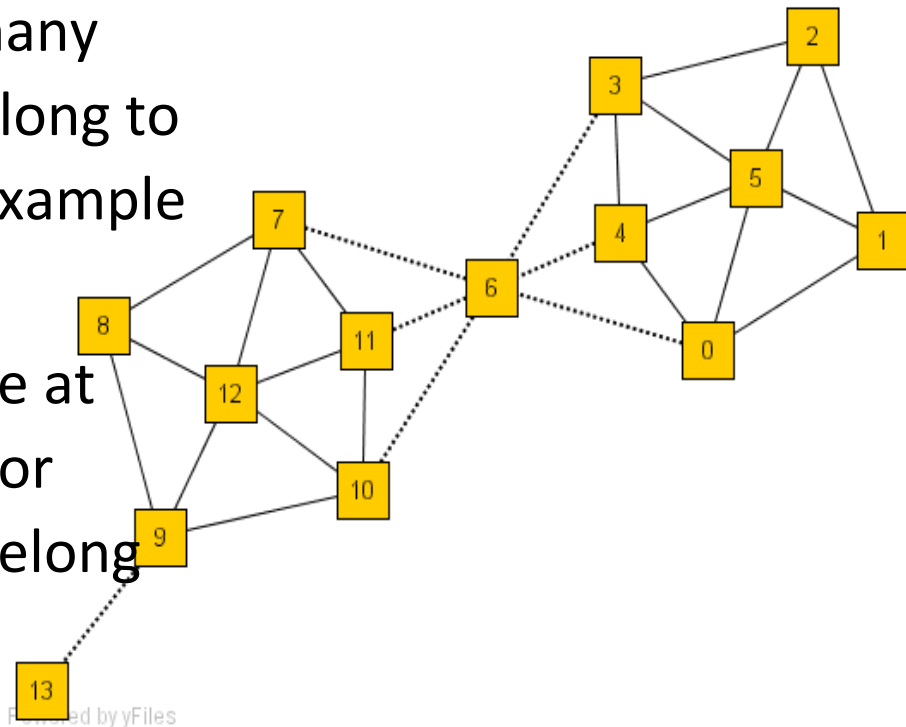
---

- Networks made up of the mutual relationships of data elements usually have an underlying structure. Clustering: A structure discovery problem
- Given simply information of who associates with whom, could one identify clusters of individuals with common interests or special relationships (families, cliques, terrorist cells)?
- Questions to be answered: How many clusters? What size should they be? What is the best partitioning? Should some points be segregated?
- Scan: An interesting density-based algorithm
  - X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger, “SCAN: A Structural Clustering Algorithm for Networks”, Proc. 2007 ACM SIGKDD Int. Conf. Knowledge Discovery in Databases (KDD'07), San Jose, CA, Aug. 2007



# Social Network and Its Clustering Problem

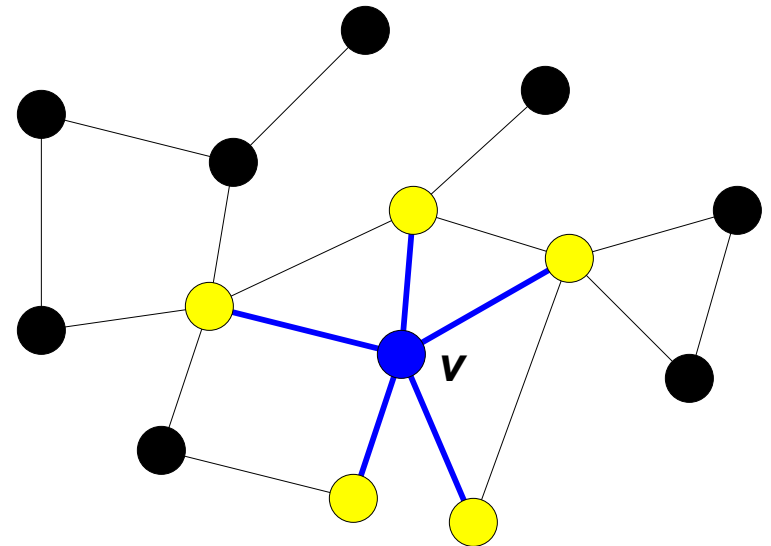
- Individuals in a tight social group, or **clique**, know many of the same people, regardless of the size of the group.
- Individuals who are **hubs** know many people in different groups but belong to no single group. Politicians, for example bridge multiple groups.
- Individuals who are **outliers** reside at the margins of society. Hermits, for example, know few people and belong to no group.



# Structure Similarity

- Define  $\Gamma(v)$  as immediate neighbor of a vertex  $v$ .
- The desired features tend to be captured by a measure  $\sigma(u, v)$  as Structural Similarity

$$\sigma(v, w) = \frac{|\Gamma(v) \cap \Gamma(w)|}{\sqrt{|\Gamma(v)| |\Gamma(w)|}}$$



- Structural similarity is large for members of a clique and small for hubs and outliers.

# Structural Connectivity [1]

---

▪  $\varepsilon$ -Neighborhood:  $N_\varepsilon(v) = \{w \in \Gamma(v) \mid \sigma(v, w) \geq \varepsilon\}$

▪ Core:  $CORE_{\varepsilon, \mu}(v) \Leftrightarrow |N_\varepsilon(v)| \geq \mu$

▪ Direct structure reachable:

$$DirRECH_{\varepsilon, \mu}(v, w) \Leftrightarrow CORE_{\varepsilon, \mu}(v) \wedge w \in N_\varepsilon(v)$$

▪ Structure reachable: transitive closure of direct structure reachability

▪ Structure connected:

$$CONNECT_{\varepsilon, \mu}(v, w) \Leftrightarrow \exists u \in V : RECH_{\varepsilon, \mu}(u, v) \wedge RECH_{\varepsilon, \mu}(u, w)$$

[1] M. Ester, H. P. Kriegel, J. Sander, & X. Xu (KDD'96) "A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases"

# Structure-Connected Clusters

- Structure-connected cluster  $C$

- Connectivity:  $\forall v, w \in C : CONNECT_{\varepsilon, \mu}(v, w)$

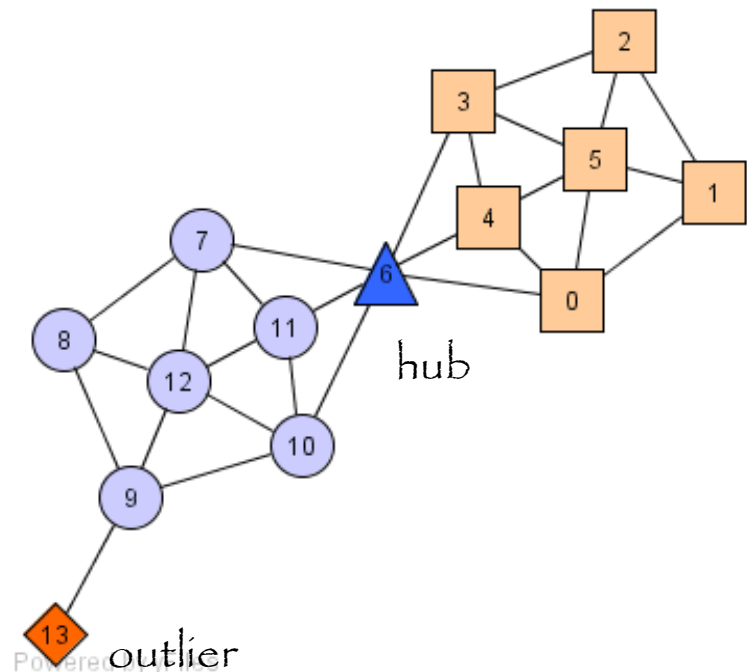
- Maximality:  $\forall v, w \in V : v \in C \wedge REACH_{\varepsilon, \mu}(v, w) \Rightarrow w \in C$

- Hubs:

- Not belong to any cluster
- Bridge to many clusters

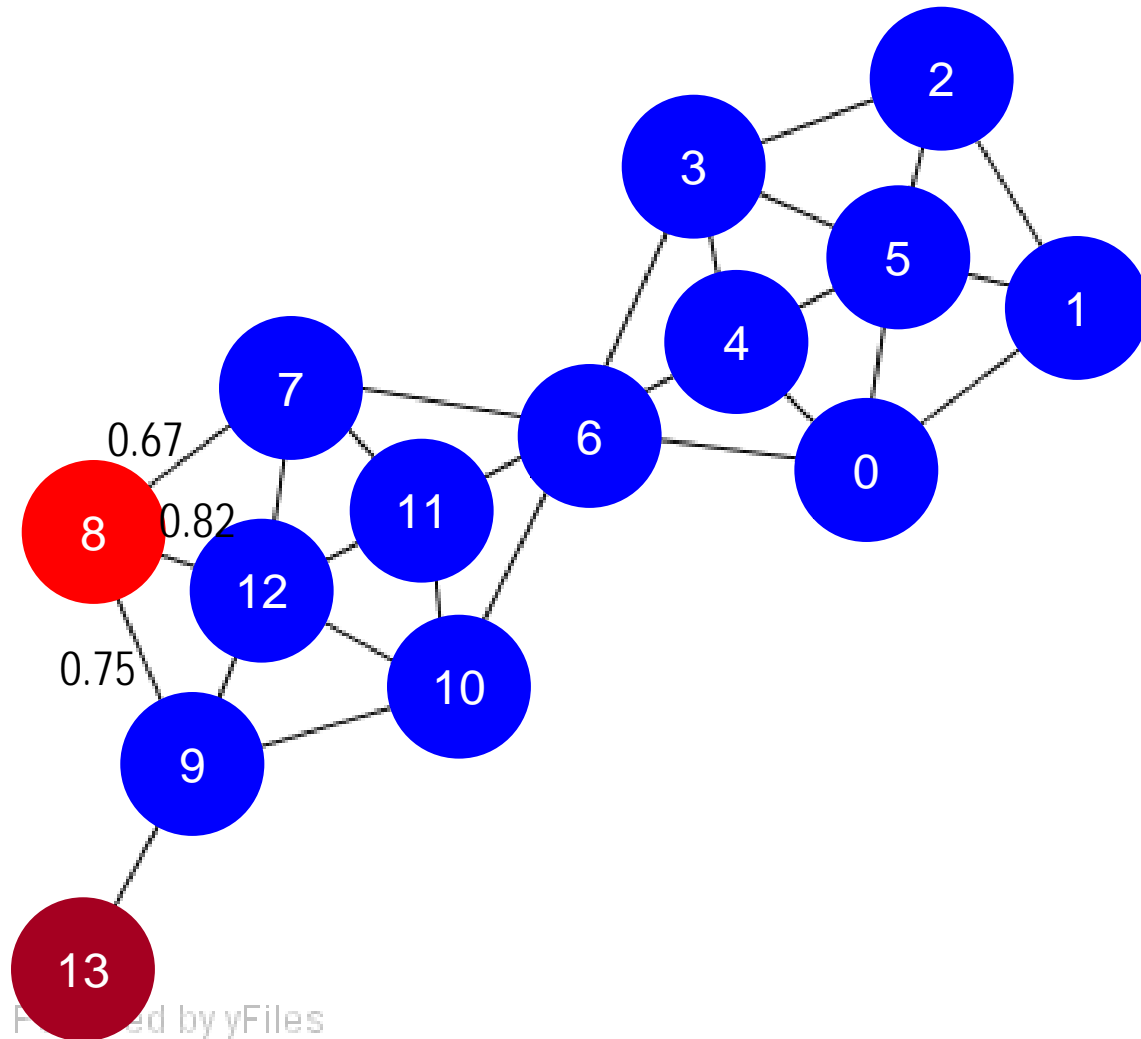
- Outliers:

- Not belong to any cluster
- Connect to less clusters



# Algorithm

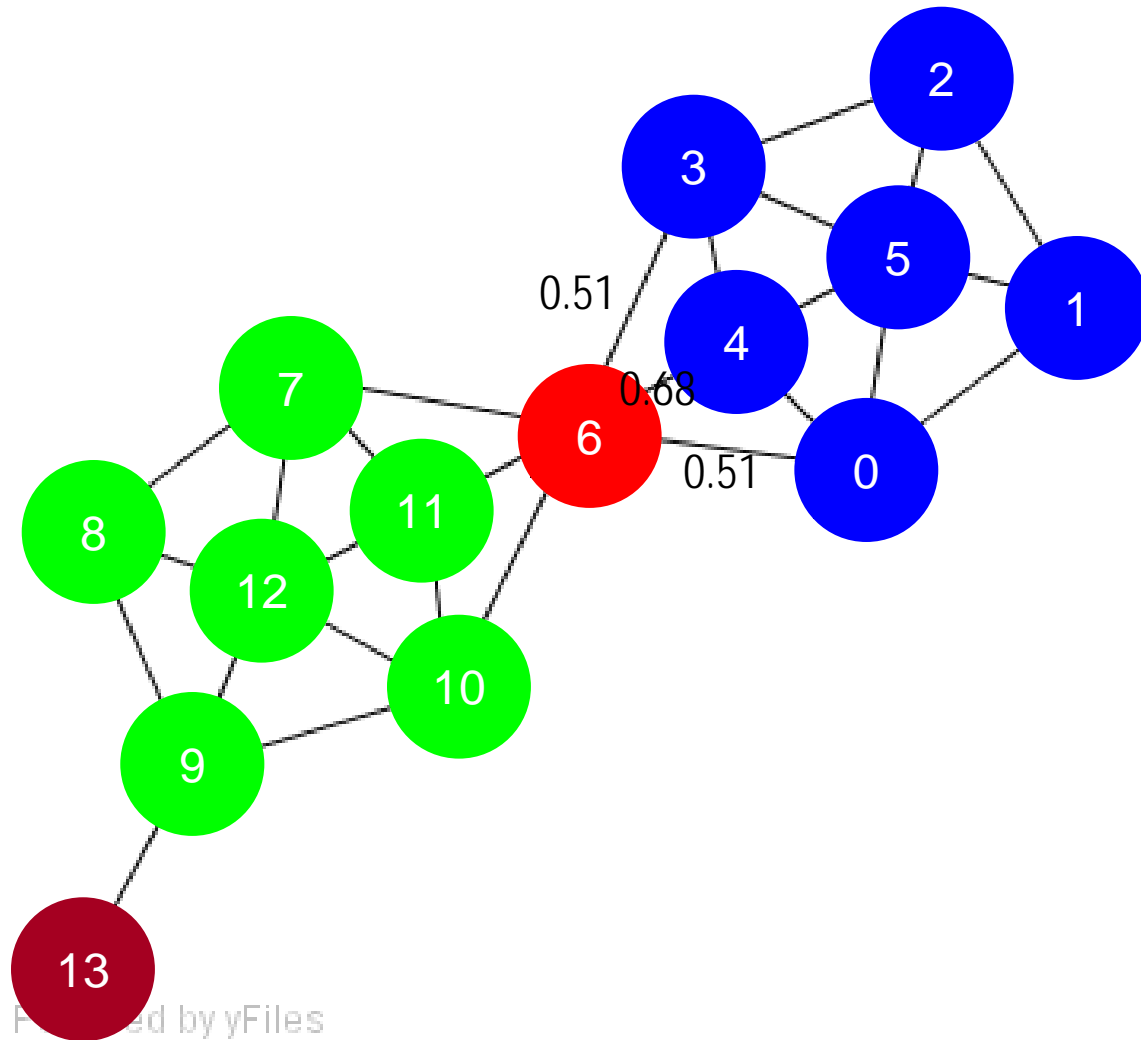
$\mu = 2$   
 $\varepsilon = 0.7$



Powered by yFiles

# Algorithm

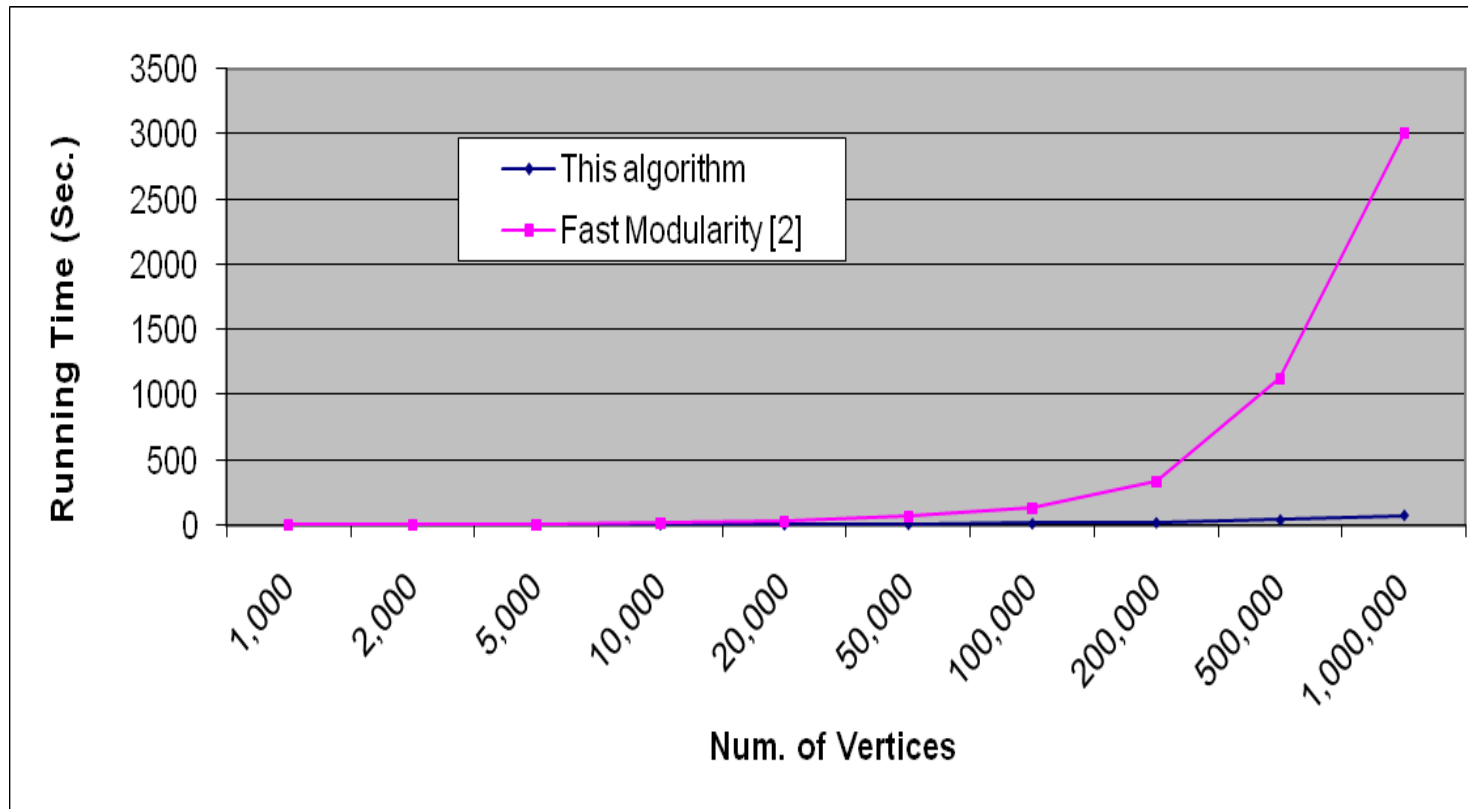
$\mu = 2$   
 $\varepsilon = 0.7$



Rendered by yFiles

# Running Time


- Running time:  $O(|E|)$
- For sparse networks:  $O(|V|)$



[2] A. Clauset, M. E. J. Newman, & C. Moore, *Phys. Rev. E* **70**, 066111 (2004).

# Clustering and Ranking in Information Networks

---

- Integrated Clustering and Ranking of Heterogeneous Information Networks
- Clustering of Homogeneous Information Networks
  - LinkClus: Clustering with link-based similarity measure
  - SCAN: Density-based clustering of networks
  - Others 
    - Spectral clustering
    - Modularity-based clustering
    - Probabilistic model-based clustering
- User-Guided Clustering of Information Networks



# Spectral Clustering

- Spectral clustering: Find the best cut that partitions the network
  - Different criteria to decide “best”
    - Min cut, ratio cut, normalized cut, Min-Max cut
- Using Min cut as an example [Wu et al. 1993]

- Assign each node  $i$  an indicator variable

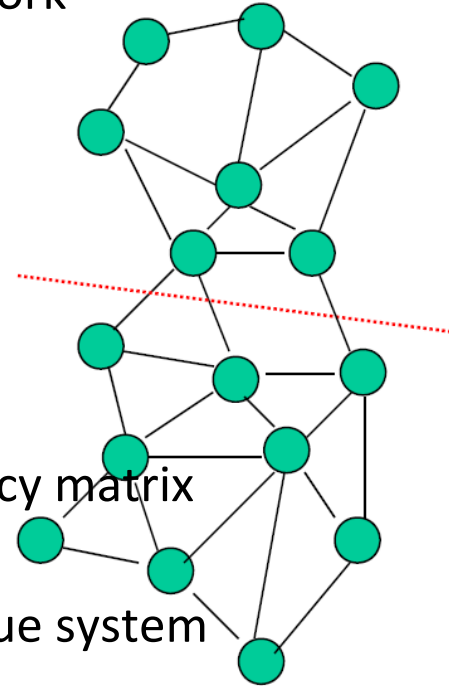
$$q_i = \begin{cases} 1 & \text{if } i \in A \\ -1 & \text{if } i \in B \end{cases}$$

- Represent the cut size using indicator vector and adjacency matrix
  - Cutsizesize =  $\frac{1}{2}q^T(D-W)q$
- Minimize the objective function through solving eigenvalue system
  - Relax the discrete value of  $q$  to continuous value

- $(D-W)q = \lambda q$

- Map continuous value of  $q$  into discrete ones to get cluster labels
  - Use second smallest eigenvector for  $q$

- $A = \{i \mid q_2(i) < 0\}, B = \{i \mid q_2(i) \geq 0\}$



# Modularity-Based Clustering

---

- Modularity-based clustering
  - Find the best clustering that maximizes the modularity function
- Q-function [Newman et al., 2004]
  - Let  $e_{ij}$  be a half of the fraction of edges between group  $i$  and group  $j$ 
    - $e_{ii}$  is the fraction of edges within group  $i$
  - Let  $a_i$  be the fraction of all ends of edges attached to vertices in group  $i$ 
    - $a_i = \sum_j e_{ij}$
  - $Q$  is then defined as sum of difference between within-group edges and expected within-group edges
    - $$Q = \sum_i (e_{ii} - a_i^2)$$
- Minimize  $Q$ 
  - One possible solution: hierarchically merge clusters resulting in greatest increase in  $Q$  function [Newman et al., 2004]


# Probabilistic Model-Based Clustering

---

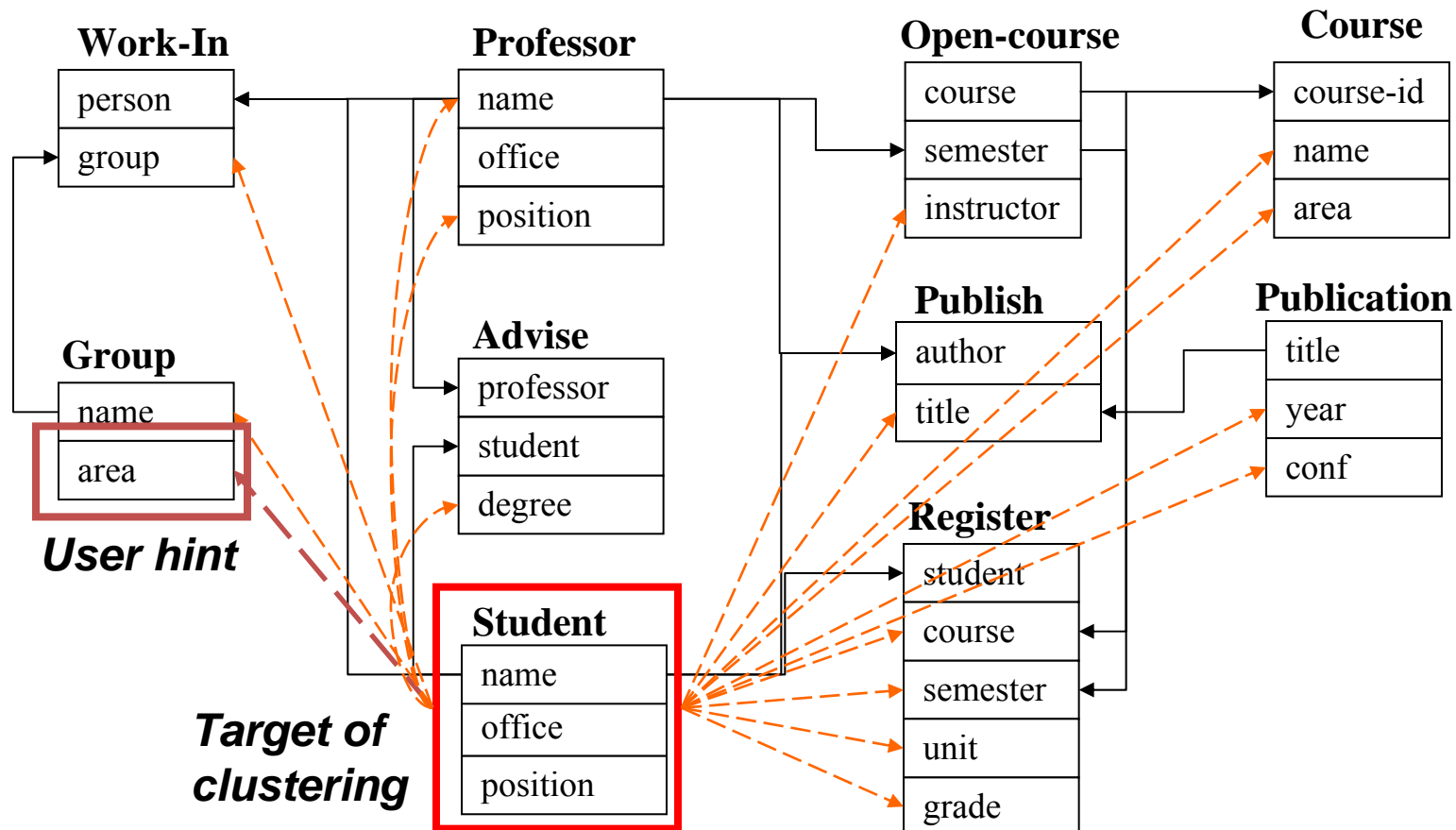
- Probabilistic model-based clustering
  - Build generative models for links based on hidden cluster labels
  - Maximize the log-likelihood of all the links to derive the hidden cluster membership
- An example: Airoldi et al., Mixed membership stochastic block models, 2008
  - Define a group interaction probability matrix  $B_{(K \times K)}$ 
    - $B(g,h)$  denotes the probability of link generation between group  $g$  and group  $h$
  - Generative model for a link
    - For each node, draw a membership probability vector from a Dirichlet prior
    - For each pair of nodes, draw cluster labels according to their membership probability (supposing  $g$  and  $h$ ), then decide whether to have a link according to probability  $B(g, h)$
  - Derive the hidden cluster label by maximize the likelihood given  $B$  and the prior

# Clustering and Ranking in Information Networks

---

- Integrated Clustering and Ranking of Heterogeneous Information Networks
- Clustering of Homogeneous Information Networks
  - LinkClus: Clustering with link-based similarity measure
  - SCAN: Density-based clustering of networks
  - Others
    - Spectral clustering
    - Modularity-based clustering
    - Probabilistic model-based clustering
- User-Guided Clustering of Information Networks 

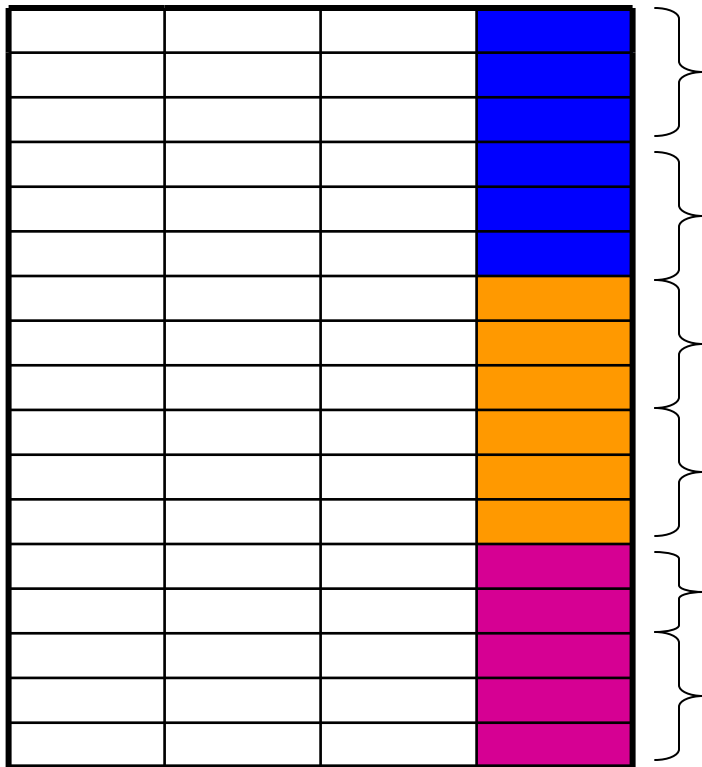
# User-Guided Clustering in DB InfoNet



- User usually has a goal of clustering, e.g., clustering students by research area
- User specifies his clustering goal to a DB-InfoNet cluster: CrossClus

# Classification vs. User-Guided Clustering

User hint



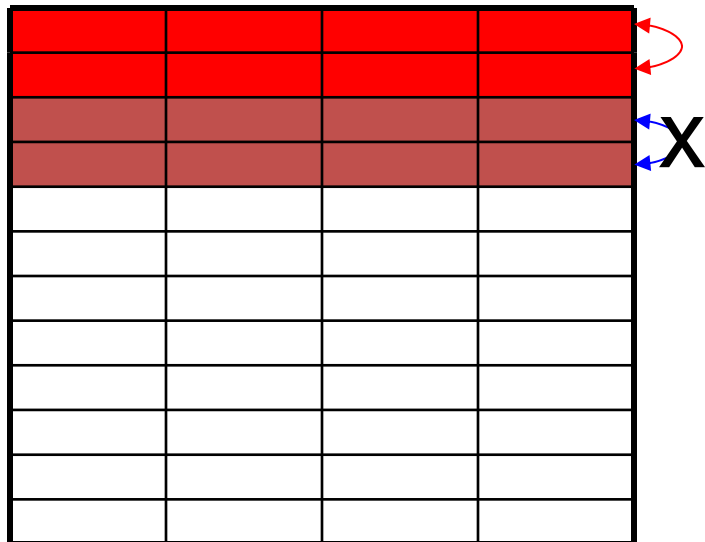
All tuples for clustering

- User-specified *feature* (in the form of *attribute*) is used as a hint, not class labels
  - The attribute may contain too many or too few distinct values
    - E.g., a user may want to cluster students into 20 clusters instead of 3
  - Additional features need to be included in cluster analysis

# User-Guided Clustering vs. Semi-supervised Clustering

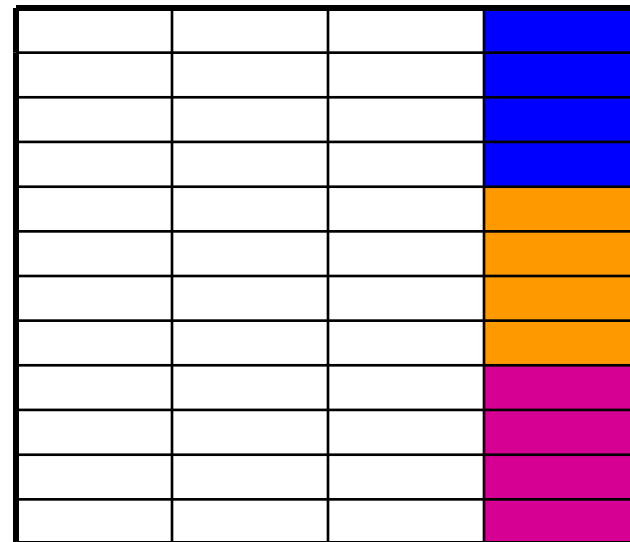
- Semi-supervised clustering [Wagstaff, et al' 01, Xing, et al.'02]
  - User provides a training set consisting of “similar” and “dissimilar” pairs of objects
- User-guided clustering
  - User specifies an attribute as a hint, and more relevant features are found for clustering

Semi-supervised clustering



All tuples for clustering

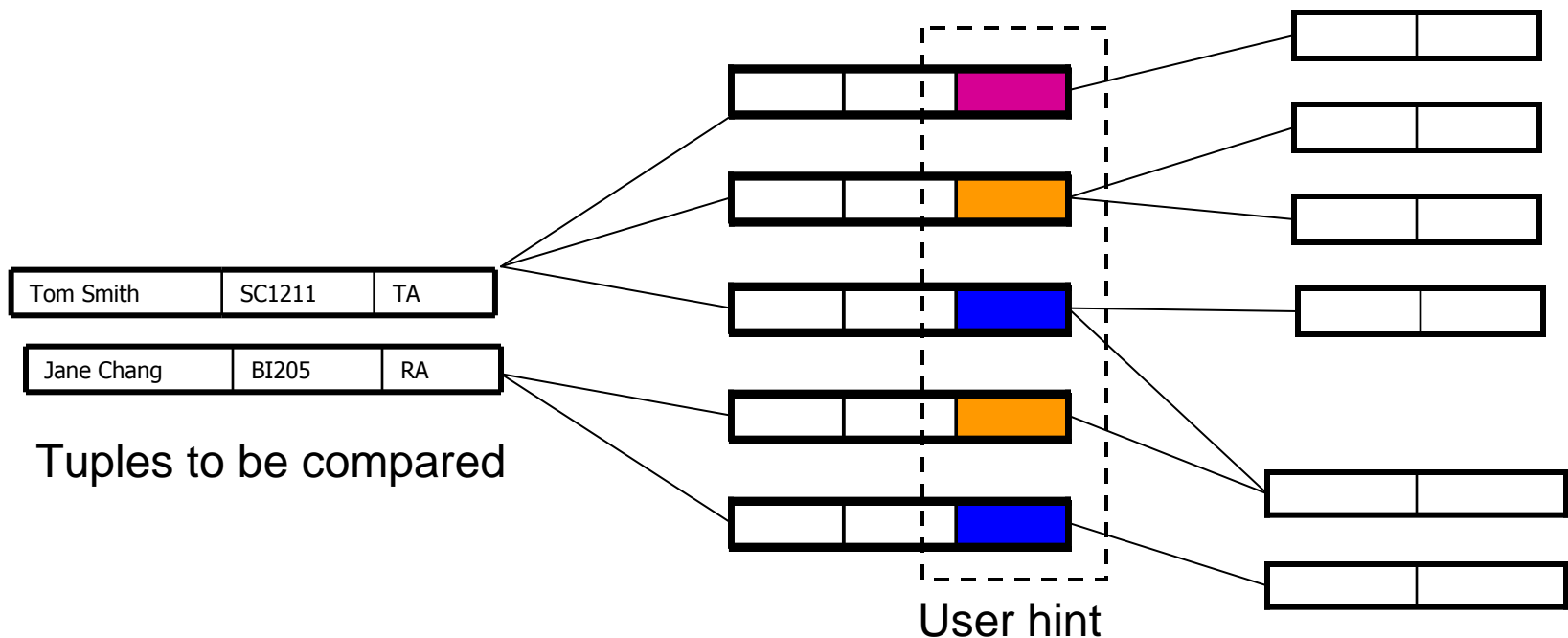
User-guided clustering



All tuples for clustering

# Why Not Typical Semi-Supervised Clustering?

- Why not do typical semi-supervised clustering?
  - Much information (in multiple relations) is needed to judge whether two tuples are similar
  - A user may not be able to provide a good training set
- It is much easier for a user to specify an attribute as a hint, such as a student's *research area*





# CrossClus: An Overview

---

- CrossClus: Framework
  - Search for good multi-relational features for clustering
  - Measure similarity between features based on how they cluster objects into groups
  - User guidance + heuristic search for finding pertinent features
  - Clustering based on a  $k$ -medoids-based algorithm
- CrossClus: Major advantages
  - User guidance, even in a very simple form, plays an important role in multi-relational clustering
  - CrossClus finds *pertinent features* by computing similarities between features

# Selection of Multi-Relational Features

- A multi-relational feature is defined by:
  - A join path. E.g.,  $Student \rightarrow Register \rightarrow OpenCourse \rightarrow Course$
  - An attribute. E.g.,  $Course.area$
  - (For numerical feature) an aggregation operator. E.g., sum or average
- Categorical Feature  $f = [Student \rightarrow Register \rightarrow OpenCourse \rightarrow Course, Course.area, null]$

areas of courses of each student

Tuple	Areas of courses		
	<i>DB</i>	<i>AI</i>	<i>TH</i>
$t_1$	5	5	0
$t_2$	0	3	7
$t_3$	1	5	4
$t_4$	5	0	5
$t_5$	3	3	4

Values of feature  $f$

Tuple	Feature $f$		
	<i>DB</i>	<i>AI</i>	<i>TH</i>
$t_1$	0.5	0.5	0
$t_2$	0	0.3	0.7
$t_3$	0.1	0.5	0.4
$t_4$	0.5	0	0.5
$t_5$	0.3	0.3	0.4

$f(t_1)$



$f(t_2)$



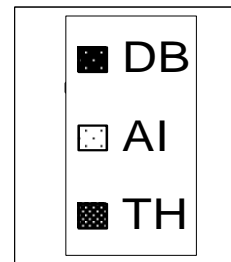
$f(t_3)$



$f(t_4)$



$f(t_5)$



- Numerical Feature, e.g., average grades of students
  - $h = [Student \rightarrow Register, Register.grade, average]$
  - E.g.  $h(t_1) = 3.5$

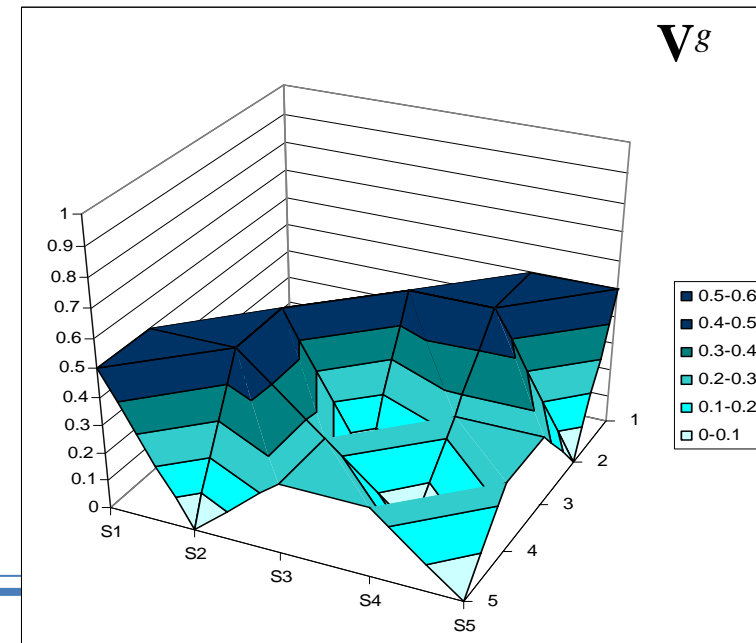
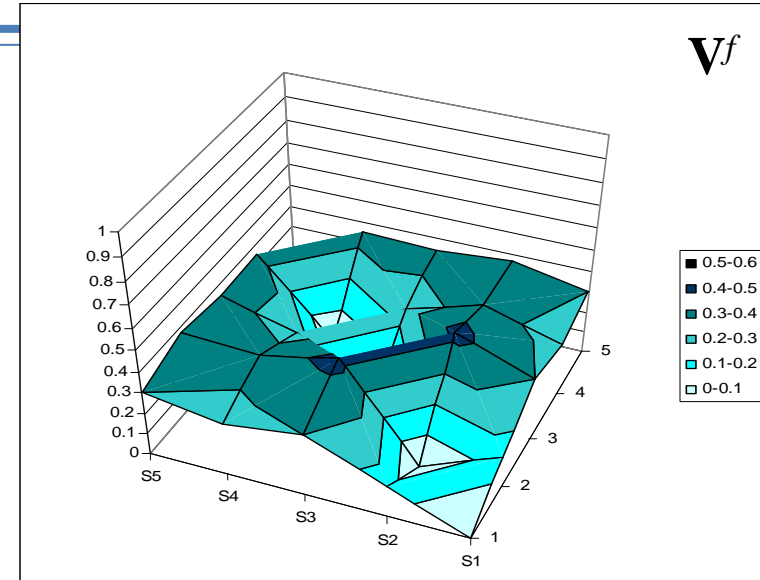
# Similarity Between Features

Values of Feature  $f$  and  $g$

	Feature $f$ (course)			Feature $g$ (group)		
	DB	AI	TH	Info sys	Cog sci	Theory
$t_1$	0.5	0.5	0	1	0	0
$t_2$	0	0.3	0.7	0	0	1
$t_3$	0.1	0.5	0.4	0	0.5	0.5
$t_4$	0.5	0	0.5	0.5	0	0.5
$t_5$	0.3	0.3	0.4	0.5	0.5	0

Similarity between two features –  
cosine similarity of two vectors

$$Sim(f, g) = \frac{V^f \cdot V^g}{|V^f| |V^g|}$$

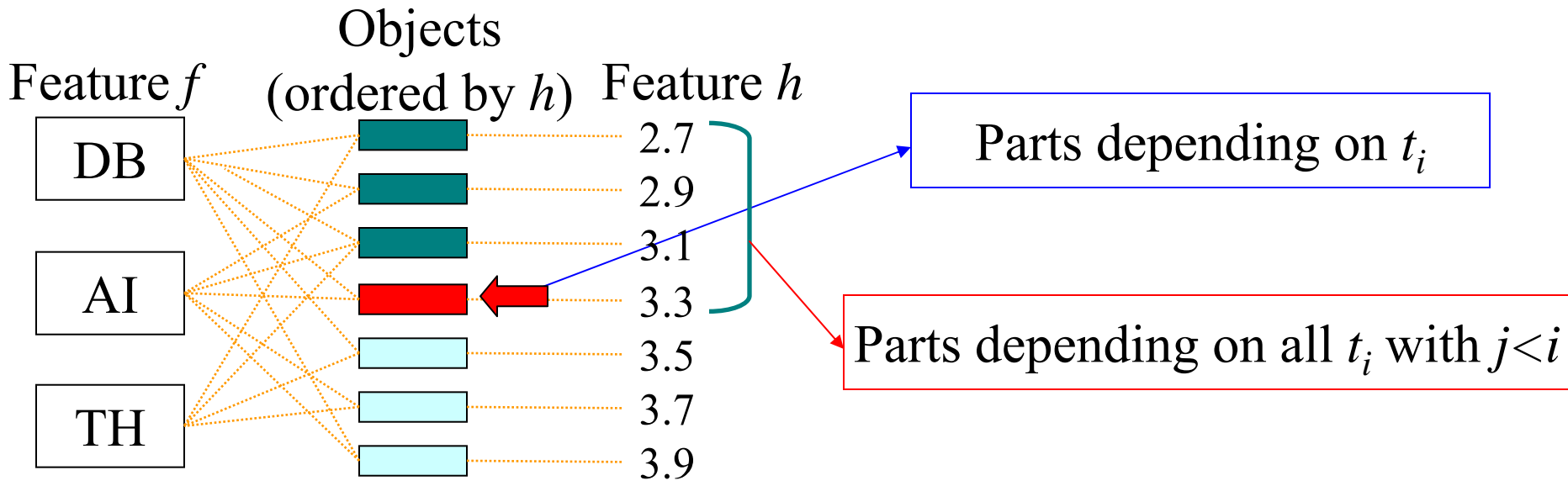


# Similarity between Categorical & Numerical Features

$$\begin{aligned}
 V^h \cdot V^f &= 2 \sum_{i=1}^N \sum_{j<i} sim_h(t_i, t_j) \cdot sim_f(t_i, t_j) \\
 &= 2 \sum_{i=1}^N \sum_{k=1}^l f(t_i) \cdot p_k (1 - h(t_i)) \left( \sum_{j<i} f(t_j) \cdot p_k \right) + 2 \sum_{i=1}^N \sum_{k=1}^l f(t_i) \cdot p_k \left( \sum_{j<i} h(t_j) \cdot f(t_j) \cdot p_k \right)
 \end{aligned}$$

*Only depend on  $t_i$*

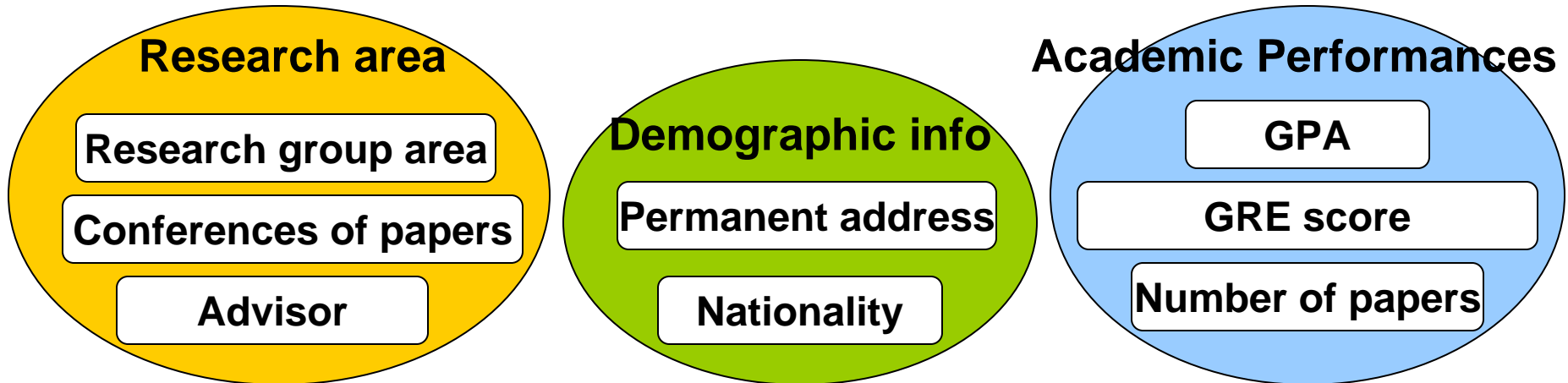
*Depend on all  $t_j$  with  $j<i$*



# Searching for Pertinent Features

---

- Different features convey different aspects of information

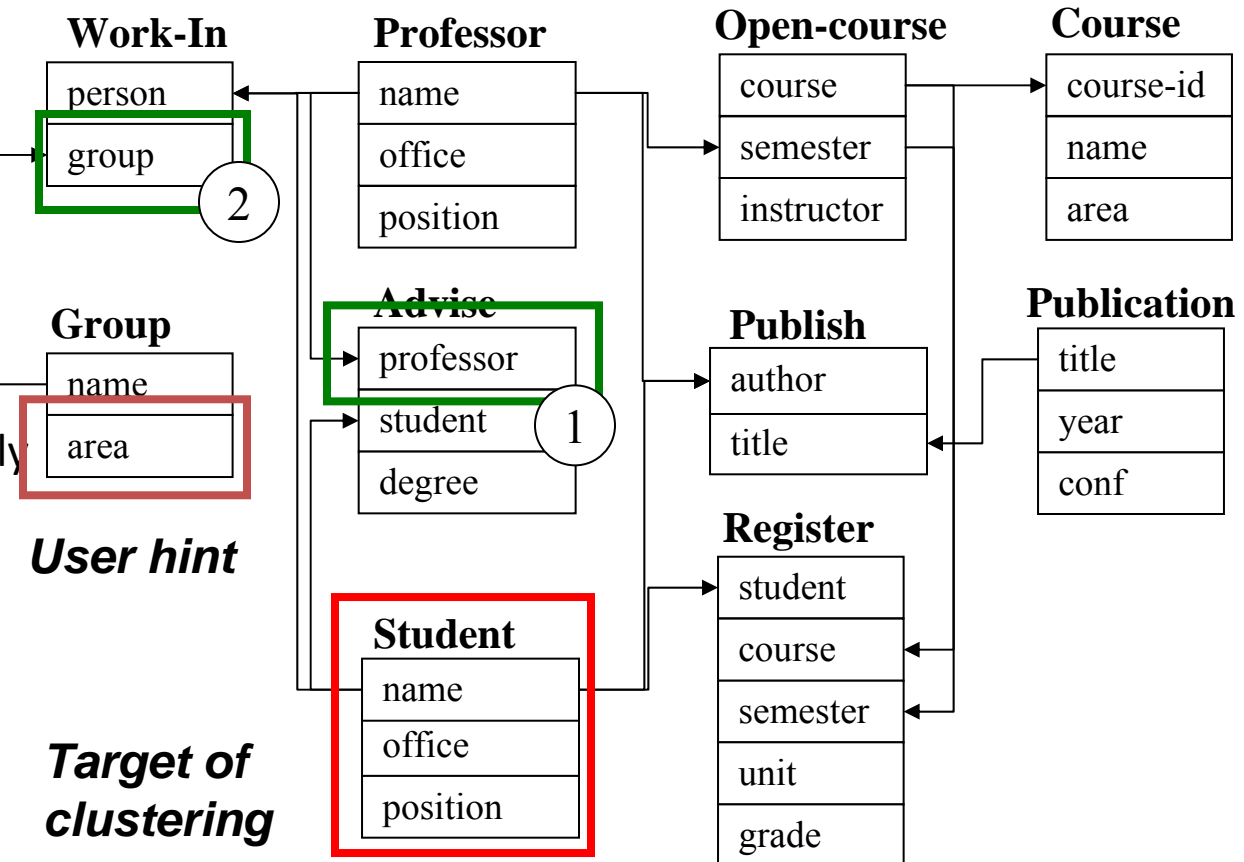


- Features conveying same aspect of information usually cluster objects in more similar ways
  - research group areas vs. conferences of publications
- Given user specified feature
  - Find pertinent features by computing feature similarity

# Heuristic Search for Pertinent Features

## Overall procedure

1. Start from the user-specified feature
2. Search in neighborhood of existing pertinent features
3. Expand search range gradually



- Tuple ID propagation [Yin, et al.'04] is used to create multi-relational features
  - IDs of target tuples can be propagated along any join path, from which we can find tuples joinable with each target tuple

# Clustering with Multi-Relational Feature

---

- Given a set of  $L$  pertinent features  $f_1, \dots, f_L$ , similarity between two objects

$$\text{sim}(t_1, t_2) = \sum_{i=1}^L \text{sim}_{f_i}(t_1, t_2) \cdot f_i.\text{weight}$$

- Weight of a feature is determined in feature search by its similarity with other pertinent features
- For clustering, we use CLARANS, a scalable  $k$ -medoids [Ng & Han'94] algorithm

# Experiments: Compare CrossClus with Existing Methods

---

- Baseline: Only use the user specified feature
- PROCLUS [Aggarwal, et al. 99]: a state-of-the-art subspace clustering algorithm
  - Use a subset of features for each cluster
  - We convert relational database to a table by propositionalization
  - User-specified feature is forced to be used in every cluster
- RDBC [Kirsten and Wrobel'00]
  - A representative ILP clustering algorithm
  - Use neighbor information of objects for clustering
  - User-specified feature is forced to be used



# Measuring Clustering Accuracy

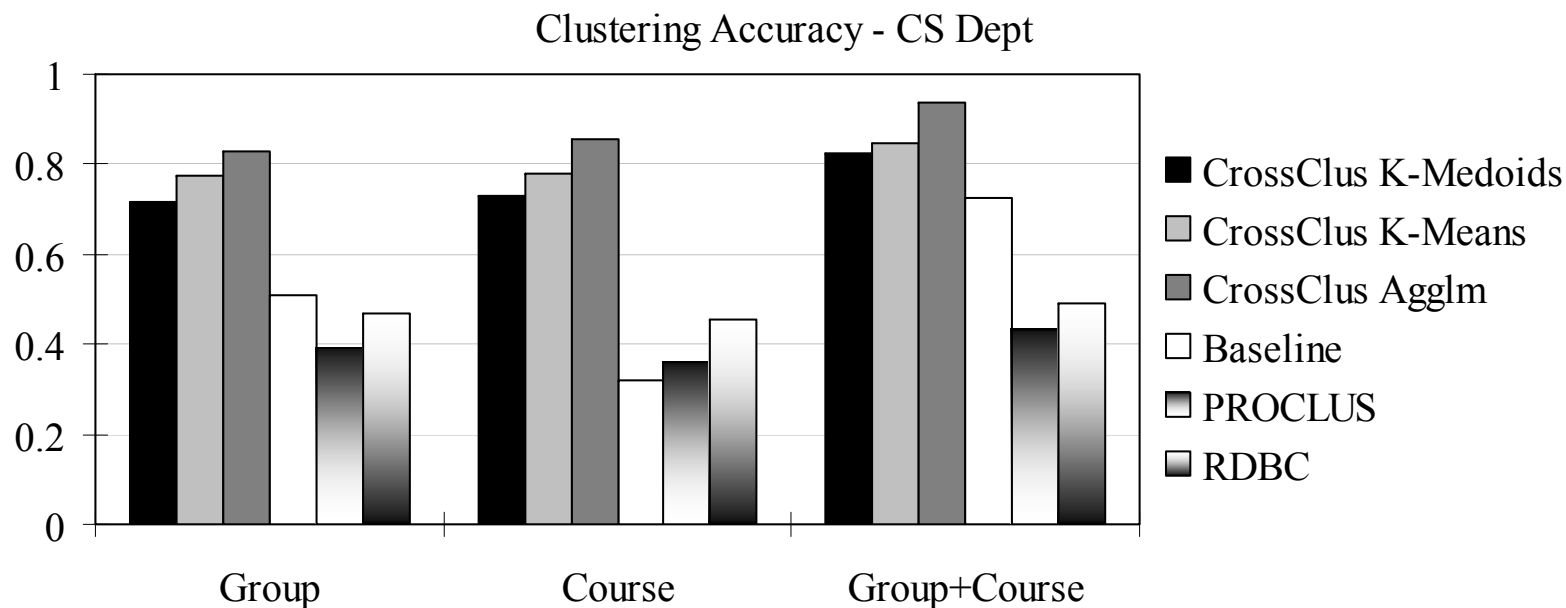
---

- To verify that CrossClus captures user's clustering goal, we define “accuracy” of clustering
- Given a clustering task
  - Manually find all features that contain information directly related to the clustering task – standard feature set
    - E.g., Clustering students by research areas
    - Standard feature set: research group, group areas, advisors, conferences of publications, course areas
  - Accuracy of clustering result: how similar it is to the clustering generated by standard feature set

$$\text{deg}(C \subset C') = \frac{\sum_{i=1}^n \max_{1 \leq j \leq n'} (|c_i \cap c'_j|)}{\sum_{i=1}^n |c_i|}$$

$$\text{sim}(C, C') = \frac{\text{deg}(C \subset C') + \text{deg}(C' \subset C)}{2}$$

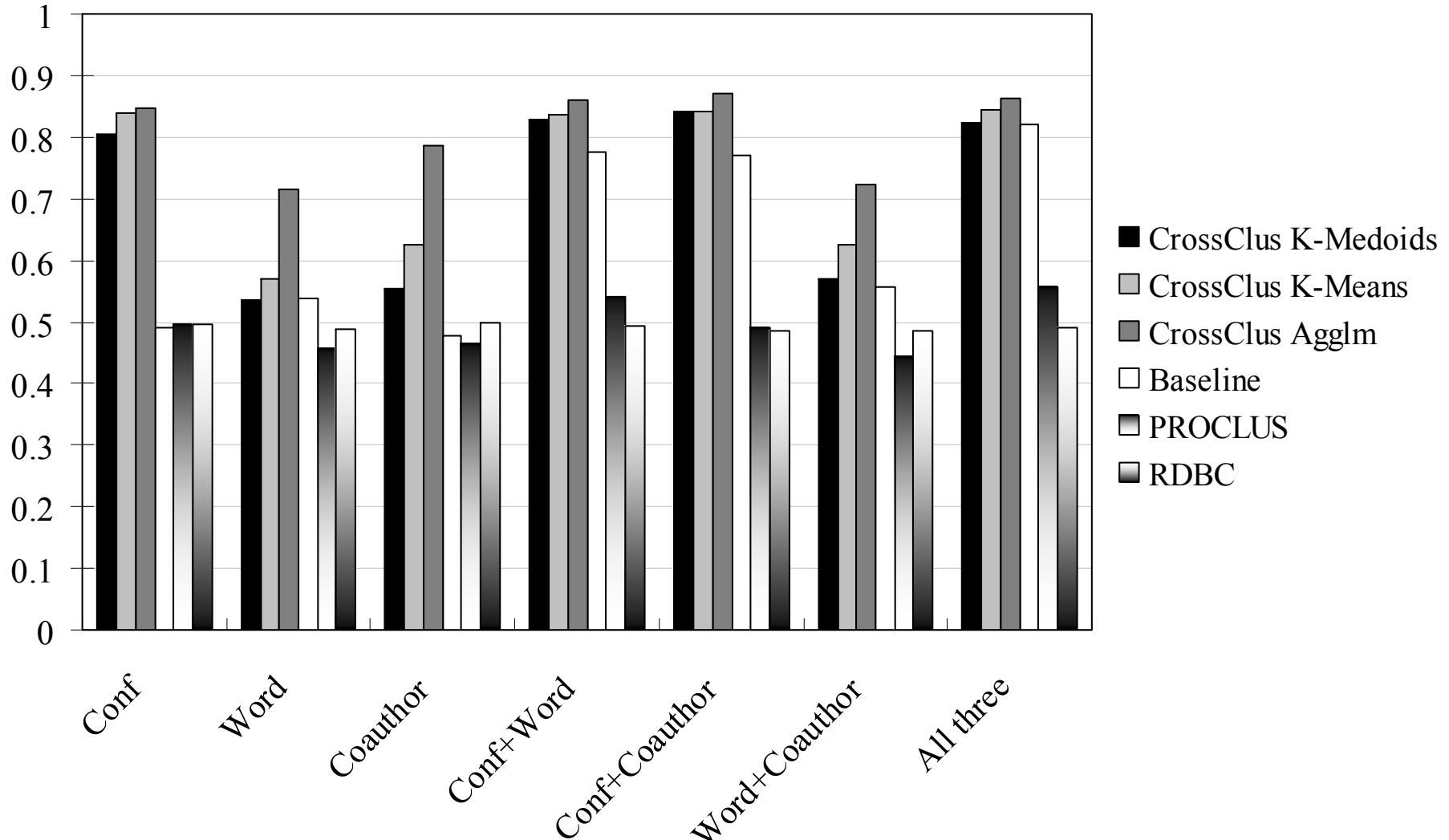
# Clustering Professors: CS Dept Dataset



- (Theory): J. Erickson, S. Har-Peled, L. Pitt, E. Ramos, D. Roth, M. Viswanathan
- (Graphics): J. Hart, M. Garland, Y. Yu
- (Database): K. Chang, A. Doan, J. Han, M. Winslett, C. Zhai
- (Numerical computing): M. Heath, T. Kerkhoven, E. de Sturler
- (Networking & QoS): R. Kravets, M. Caccamo, J. Hou, L. Sha
- (Artificial Intelligence): G. Dejong, M. Harandi, J. Ponce, L. Rendell
- (Architecture): D. Padua, J. Torrellas, C. Zilles, S. Adve, M. Snir, D. Reed, V. Adve
- (Operating Systems): D. Mickunas, R. Campbell, Y. Zhou


# DBLP Dataset

Clustering Accuracy - DBLP




# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
  - **Part I: Clustering, Ranking and Classification**
    - Clustering and Ranking in Information Networks
    - **Classification of Information Networks** 
  - **Part II:** Data Quality and Search in Information Networks
    - Data Cleaning and Data Validation by InfoNet Analysis
    - Similarity Search in Information Networks
  - **Part III:** Advanced Topics on Information Network Analysis
    - Role Discovery and OLAP in Information Networks
    - Mining Evolution and Dynamics of Information Networks
  - **Conclusions**
-

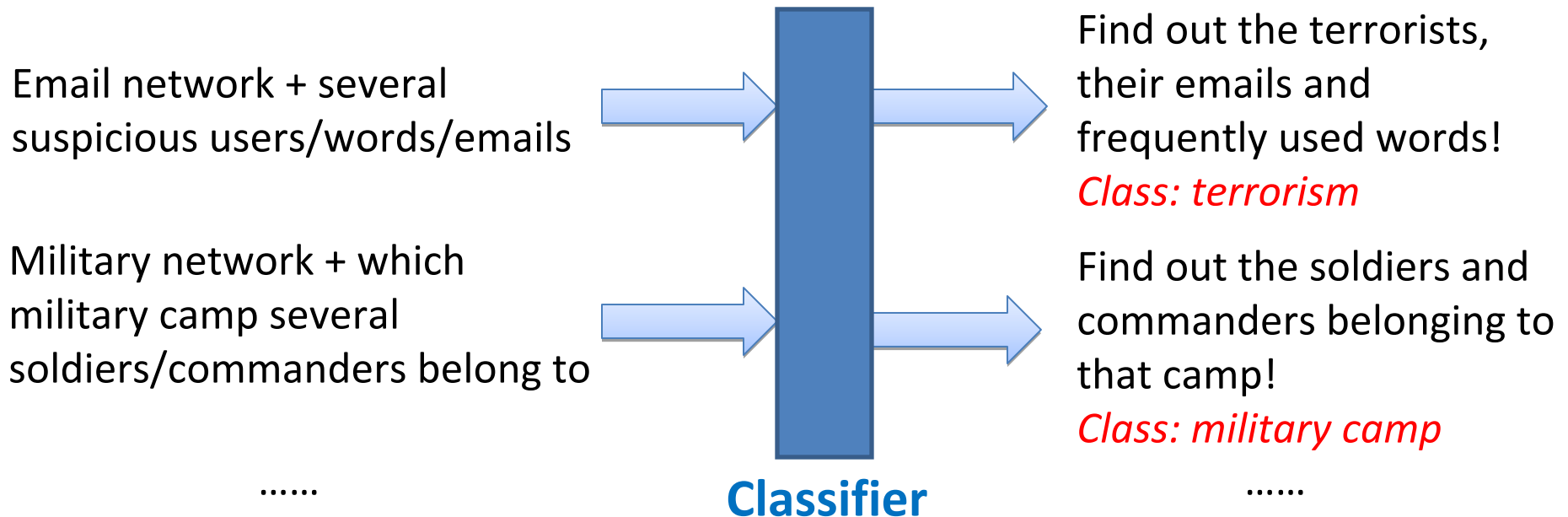
# Classification of Information Networks

---

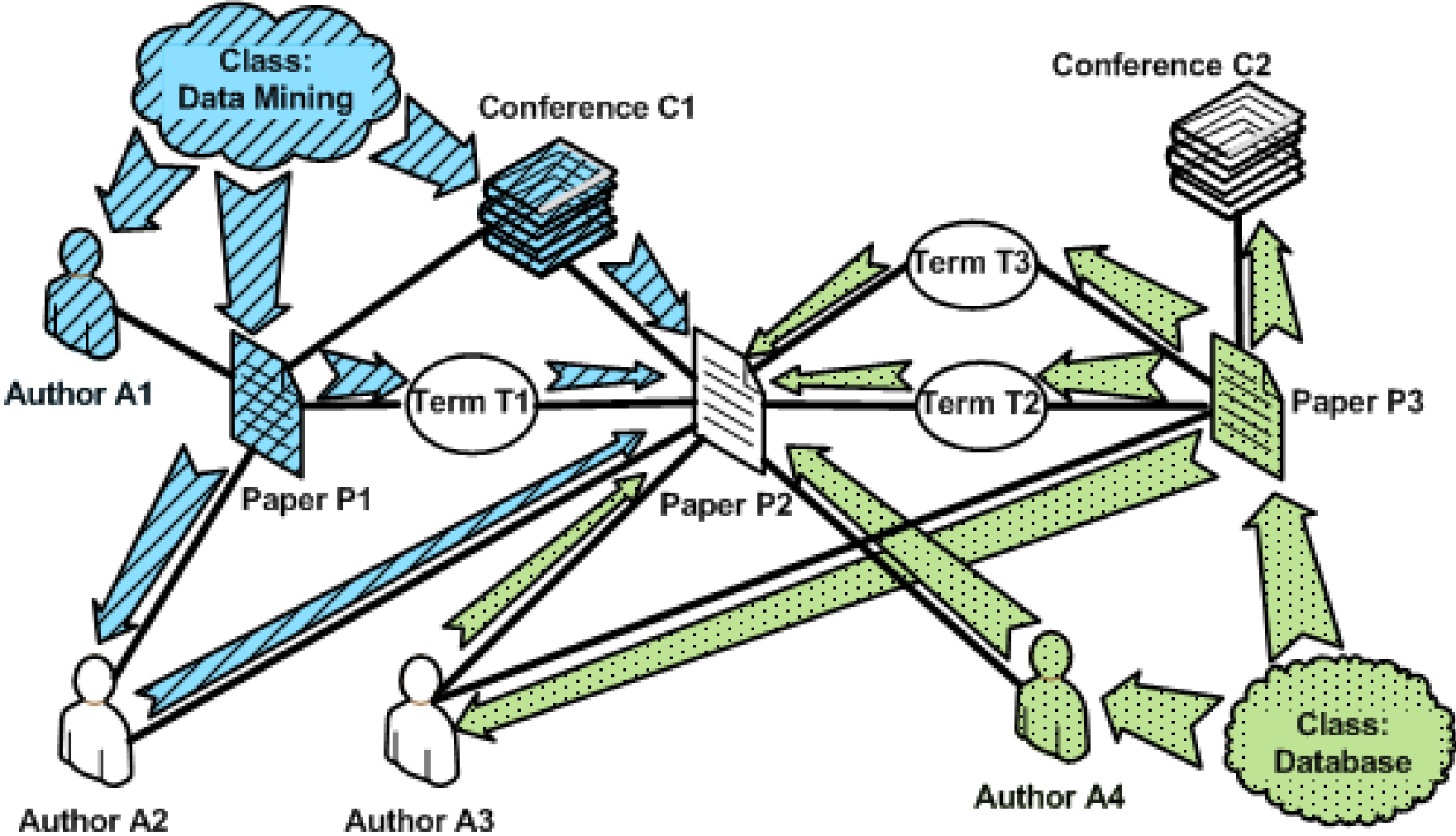
- Classification of Heterogeneous Information Networks:
  - Graph-regularization-Based Method (GNetMine) 
  - Multi-Relational-Mining-Based Method (CrossMine)
  - Statistical Relational Learning-Based Method (SRL)
- Classification of Homogeneous Information Networks

# Why Classifying Heterogeneous InfoNet?

- ❑ Sometimes, we do have *prior knowledge* for part of the nodes/objects!
- ❑ **Input:** Heterogeneous information network structure + class labels for some objects/nodes
- ❑ **Goal:** Classify the heterogeneous networked data into classes, each of which is composed of multi-typed data objects sharing a common topic.
  - Natural generalization of classification on homogeneous networked data



# Classification: Knowledge Propagation



# GNetMine: Methodology

---

- ❑ Classification of networked data can be essentially viewed as a process of *knowledge propagation*, where information is propagated from labeled objects to unlabeled ones through links until a stationary state is achieved.
- ❑ A novel graph-based regularization framework to address the classification problem on heterogeneous information networks.
- ❑ Respect the link type differences by preserving consistency over each relation graph corresponding to each type of links separately
  - ❑ Mathematical intuition: Consistency assumption
    - The confidence ( $f$ ) of two objects ( $x_{ip}$  and  $x_{jq}$ ) belonging to class  $k$  should be similar if  $x_{ip} \leftrightarrow x_{jq}$  ( $R_{ij,pq} > 0$ )
    - $f$  should be similar to the given ground truth



# GNetMine: Graph-Based Regularization

- Minimize the objective function

$$\begin{aligned}
 & J(\mathbf{f}_1^{(k)}, \dots, \mathbf{f}_m^{(k)}) \\
 &= \sum_{i,j=1}^m \lambda_{ij} \sum_{p=1}^{n_i} \sum_{q=1}^{n_j} R_{ij,pq} \left( \frac{1}{\sqrt{D_{ij,pp}}} f_{ip}^{(k)} - \frac{1}{\sqrt{D_{ji,qq}}} f_{jq}^{(k)} \right)^2 \\
 &+ \sum_{i=1}^m \alpha_i (\mathbf{f}_i^{(k)} - \mathbf{y}_i^{(k)})^T (\mathbf{f}_i^{(k)} - \mathbf{y}_i^{(k)})
 \end{aligned}$$

User preference: how much do you value this relationship / ground truth?

*Smoothness constraints:* objects linked together should share similar estimations of confidence belonging to class  $k$

Normalization term applied to each type of link separately:  
reduce the impact of popularity of nodes

Confidence estimation on labeled data and their pre-given labels should be similar

# Experiments on DBLP

---

- ❑ Class: Four research areas (communities)
    - Database, data mining, AI, information retrieval
  - ❑ Four types of objects
    - Paper (14376), Conf. (20), Author (14475), Term (8920)
  - ❑ Three types of relations
    - Paper-conf., paper-author, paper-term
  - ❑ Algorithms for comparison
    - Learning with Local and Global Consistency (LLGC) [Zhou et al. NIPS 2003] – also the homogeneous version of our method
    - Weighted-vote Relational Neighbor classifier (wvRN) [Macskassy et al. JMLR 2007]
    - Network-only Link-based Classification (nLB) [Lu et al. ICML 2003, Macskassy et al. JMLR 2007]
-

# Classification Accuracy: Labeling a Very Small Portion of Authors and Papers

(a%, p%)	nLB		wvRN		LLGC		GNetMine
	A-A	A-C-P-T	A-A	A-C-P-T	A-A	A-C-P-T	A-C-P-T
(0.1%, 0.1%)	25.4	26.0	40.8	34.1	41.4	61.3	<b>82.9</b>
(0.2%, 0.2%)	28.3	26.0	46.0	41.2	44.7	62.2	<b>83.4</b>
(0.3%, 0.3%)	28.4	27.4	48.6	42.5	48.8	65.7	<b>86.7</b>
(0.4%, 0.4%)	30.7	26.7	46.3	45.6	48.7	66.0	<b>87.2</b>
(0.5%, 0.5%)	29.8	27.3	49.0	51.4	50.6	68.9	<b>87.5</b>

Comparison of classification accuracy on authors (%)

(a%, p%)	nLB		wvRN		LLGC		GNetMine
	P-P	A-C-P-T	P-P	A-C-P-T	P-P	A-C-P-T	A-C-P-T
(0.1%, 0.1%)	49.8	31.5	62.0	42.0	67.2	62.7	<b>79.2</b>
(0.2%, 0.2%)	73.1	40.3	71.7	49.7	72.8	65.5	<b>83.5</b>
(0.3%, 0.3%)	77.9	35.4	77.9	54.3	76.8	66.6	<b>83.2</b>
(0.4%, 0.4%)	79.1	38.6	78.1	54.4	77.9	70.5	<b>83.7</b>
(0.5%, 0.5%)	80.7	39.3	77.9	53.5	79.0	73.5	<b>84.1</b>

Comparison of classification accuracy on papers (%)

(a%, p%)	nLB	wvRN	LLGC	GNetMine
	A-C-P-T	A-C-P-T	A-C-P-T	A-C-P-T
(0.1%, 0.1%)	25.5	43.5	79.0	<b>81.0</b>
(0.2%, 0.2%)	22.5	56.0	83.5	<b>85.0</b>
(0.3%, 0.3%)	25.0	59.0	<b>87.0</b>	<b>87.0</b>
(0.4%, 0.4%)	25.0	57.0	86.5	<b>89.5</b>
(0.5%, 0.5%)	25.0	68.0	90.0	<b>94.0</b>

Comparison of classification accuracy on conferences(%)

# Knowledge Propagation: List Objects with the Highest Confidence Measure Belonging to Each Class

No.	Database	Data Mining	Artificial Intelligence	Information Retrieval
1	data	mining	learning	retrieval
2	database	data	knowledge	information
3	query	clustering	Reinforcement	web
4	system	learning	reasoning	search
5	xml	classification	model	document

Top-5 terms related to each area

No.	Database	Data Mining	Artificial Intelligence	Information Retrieval
1	Surajit Chaudhuri	Jiawei Han	Sridhar Mahadevan	W. Bruce Croft
2	H. V. Jagadish	Philip S. Yu	Takeo Kanade	Iadh Ounis
3	Michael J. Carey	Christos Faloutsos	Andrew W. Moore	Mark Sanderson
4	Michael Stonebraker	Wei Wang	Satinder P. Singh	ChengXiang Zhai
5	C. Mohan	Shusaku Tsumoto	Thomas S. Huang	Gerard Salton


Top-5 authors concentrated in each area

No.	Database	Data Mining	Artificial Intelligence	Information Retrieval
1	VLDB	KDD	IJCAI	SIGIR
2	SIGMOD	SDM	AAAI	ECIR
3	PODS	PAKDD	CVPR	WWW
4	ICDE	ICDM	ICML	WSDM
5	EDBT	PKDD	ECML	CIKM

Top-5 conferences concentrated in each area

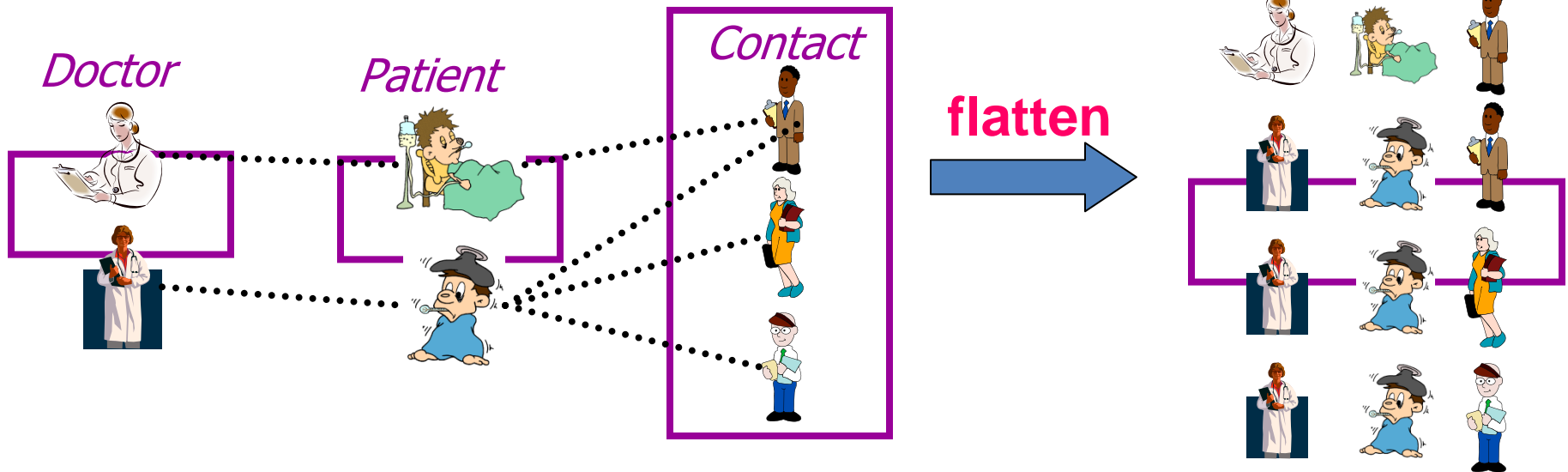
# Classification of Information Networks

---

- Classification of Heterogeneous Information Networks:
  - Graph-regularization-Based Method (GNetMine)
  - Multi-Relational-Mining-Based Method (CrossMine) 
  - Statistical Relational Learning-Based Method (SRL)
- Classification of Homogeneous Information Networks

# Multi-Relation to Flat Relation Mining?

- Folding multiple relations into a single “flat” one for mining?



- Cannot be a solution due to problems:
  - Lose information of linkages and relationships, no semantics preservation
  - Cannot utilize information of database structures or schemas (e.g., E-R modeling)

# One Approach: Inductive Logic Programming (ILP)

---

- Find a hypothesis that is consistent with background knowledge (training data)
  - FOIL, Golem, Progol, TILDE, ...
- Background knowledge
  - Relations (predicates), Tuples (ground facts)
- Inductive Logic Programming (ILP)
  - Hypothesis: The hypothesis is usually a set of rules, which can predict certain attributes in certain relations
    - $\text{Daughter}(X, Y) \leftarrow \text{female}(X), \text{parent}(Y, X)$

Training examples

Daughter(mary, ann)	+
Daughter(eve, tom)	+
Daughter(tom, ann)	-
Daughter(eve, ann)	-

Background knowledge

Parent(ann, mary)
Parent(ann, tom)
Parent(tom, eve)
Parent(tom, ian)

Female(ann)
Female(mary)
Female(eve)

# Inductive Logic Programming Approach to Multi-Relation Classification

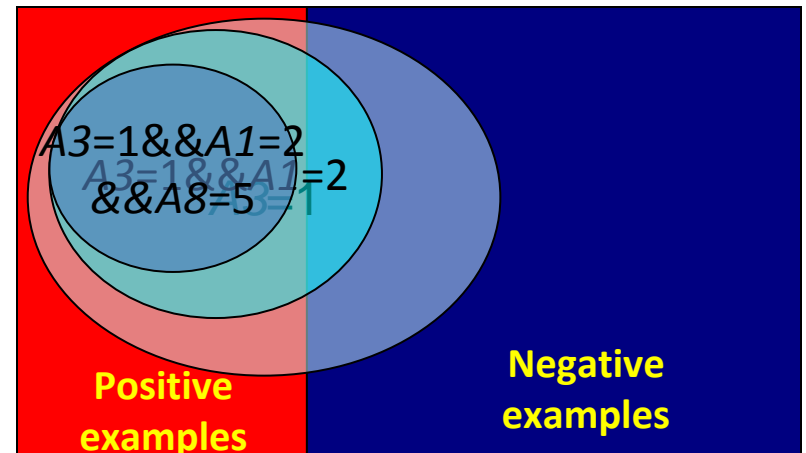
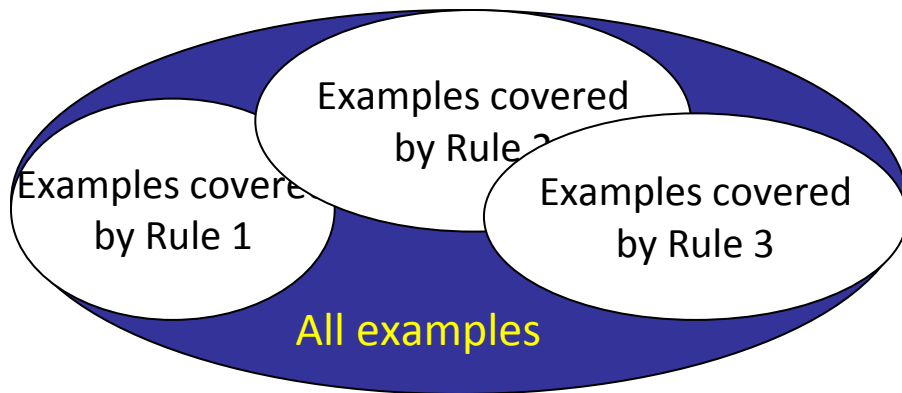
---

- ILP Approached to Multi-Relation Classification
  - Top-down Approaches (e.g., FOIL)
    - while**(enough examples left)
      - generate a rule
      - remove examples satisfying this rule
  - Bottom-up Approaches (e.g., Golem)
    - Use each example as a rule
    - Generalize rules by merging rules
  - Decision Tree Approaches (e.g., TILDE)
- ILP Approach: Pros and Cons
  - Advantages: Expressive and powerful, and rules are understandable
  - Disadvantages: Inefficient for databases with complex schemas, and inappropriate for continuous attributes



# FOIL: First-Order Inductive Learner (Rule Generation)

- Find a set of rules consistent with training data
- A top-down, sequential covering learner
- Build each rule by heuristics
  - Foil gain – a special type of information gain



- To generate a rule
  - while**(true)
    - find the best predicate  $p$
    - if** foil-gain( $p$ ) > threshold **then** add  $p$  to current rule
    - else** break

# Find the Best Predicate: Predicate Evaluation

---

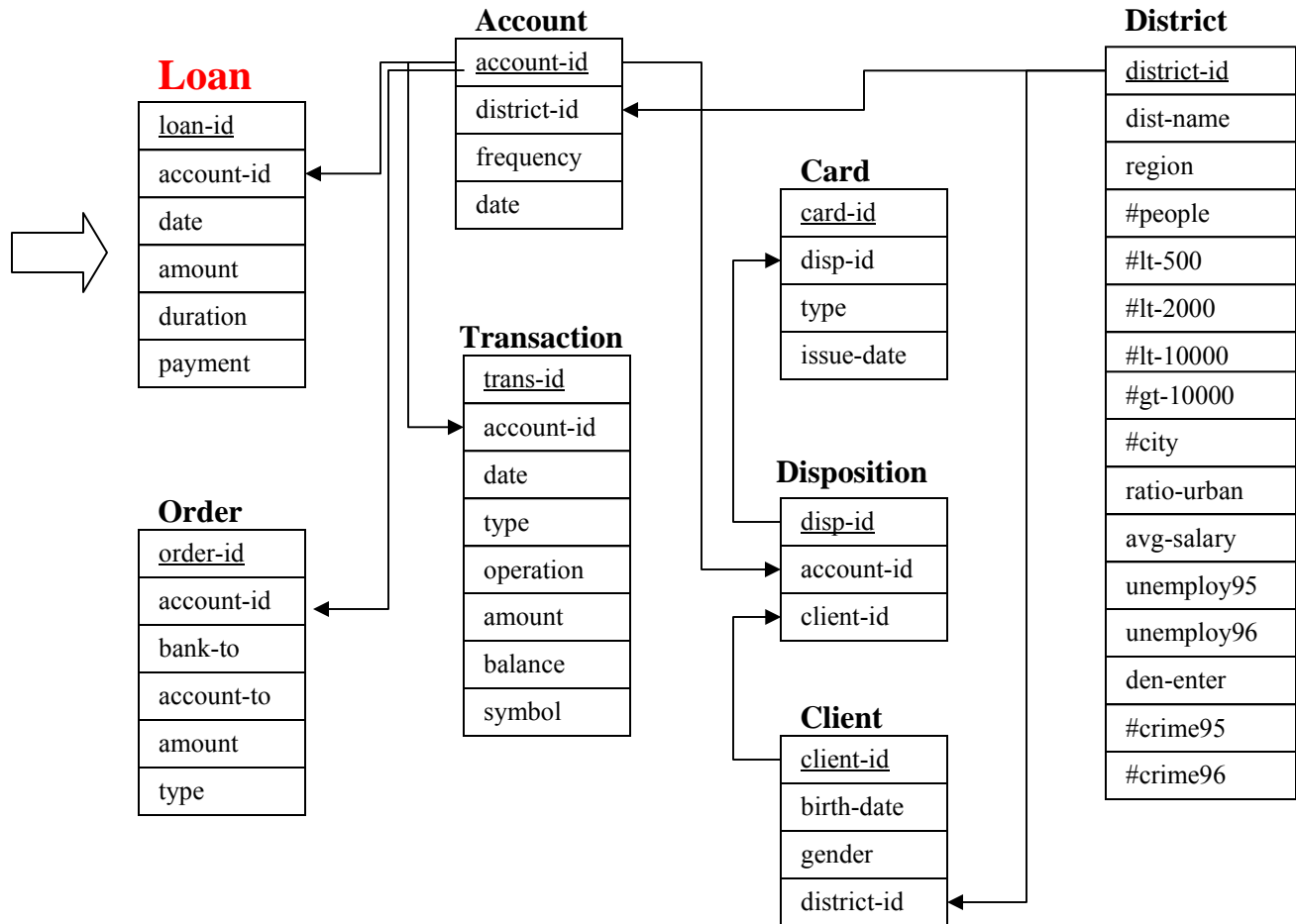
- All predicates in a relation can be evaluated based on propagated IDs
- Use *foil-gain* to evaluate predicates
  - Suppose current rule is  $r$ . For a predicate  $p$ ,

$$\text{foil-gain}(p) = P(r+p) \times \left[ -\log \frac{P(r)}{P(r)+N(r)} + \log \frac{P(r+p)}{P(r+p)+N(r+p)} \right]$$

- Categorical Attributes
  - Compute foil-gain directly
- Numerical Attributes
  - Discretize with every possible value

# Loan Applications: Backend Database

Target relation:  
Each tuple has a class label, indicating whether a loan is paid on time.



How to make decisions to loan applications?

# CrossMine: An Effective Multi-relational Classifier

---

- Methodology
  - Tuple-ID propagation: an efficient and flexible method for virtually joining relations
  - Confine the rule search process in promising directions
  - Look-one-ahead: a more powerful search strategy
  - Negative tuple sampling: improve efficiency while maintaining accuracy

# Tuple ID Propagation



Applicant #1



Applicant #2



Applicant #3



Applicant #4

Loan ID	Account ID	Amount	Duration	Decision
1	124	1000	12	Yes
2	124	4000	12	Yes
3	108	10000	24	No
4	45	12000	36	No

Account ID	Frequency	Open date	Propagated ID	Labels
124	monthly	02/27/93	1, 2	2+, 0-
108	weekly	09/23/97	3	0+, 1-
45	monthly	12/09/96	4	0+, 1-
67	weekly	01/01/97	Null	0+, 0-

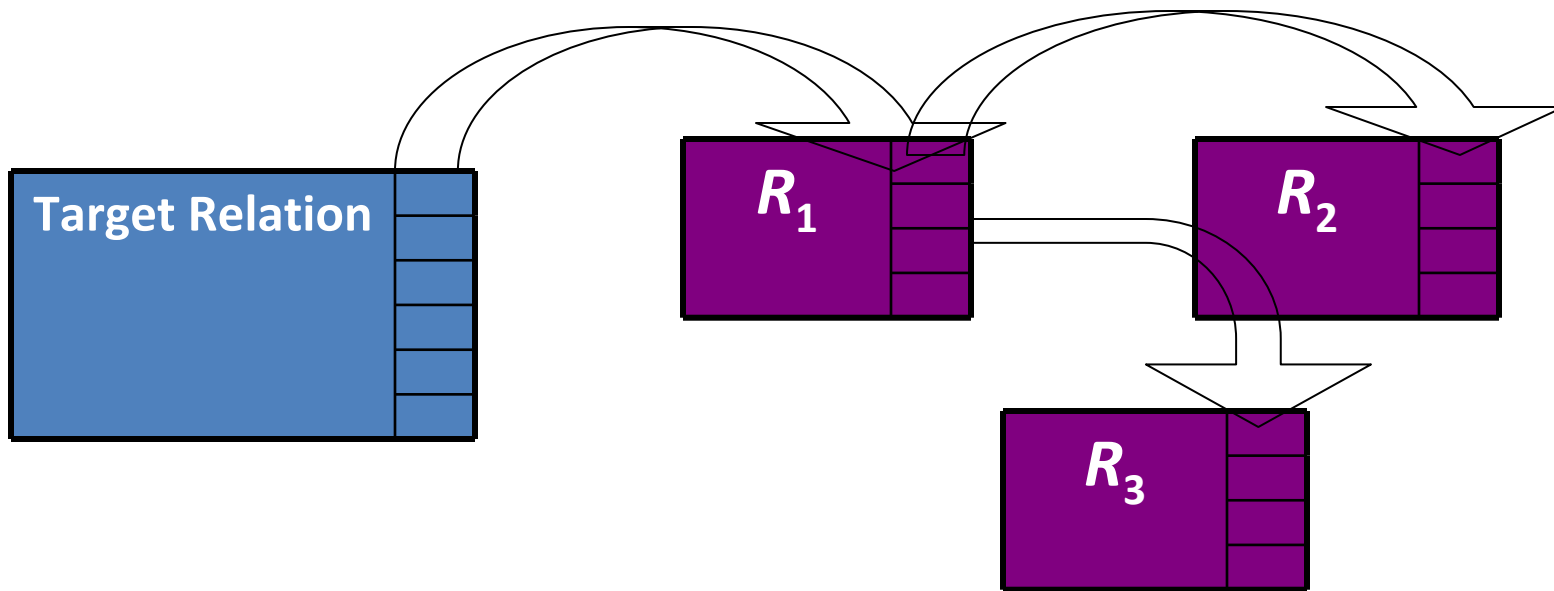
Possible predicates:

- Frequency='monthly': 2 +, 1 -
- Open date < 01/01/95: 2 +, 0 -

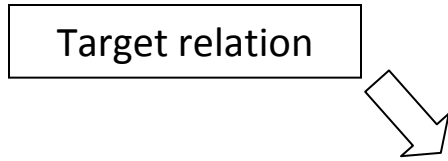
- Propagate tuple IDs of target relation to non-target relations
- Virtually join relations to avoid the high cost of physical joins

# Tuple ID Propagation (Idea Outlined)

- Efficient
  - Only propagate the tuple IDs
  - Time and space usage is low
- Flexible
  - Can propagate IDs among non-target relations
  - Many sets of IDs can be kept on one relation, which are propagated from different join paths



# Rule Generation: Example



## Rule Generation

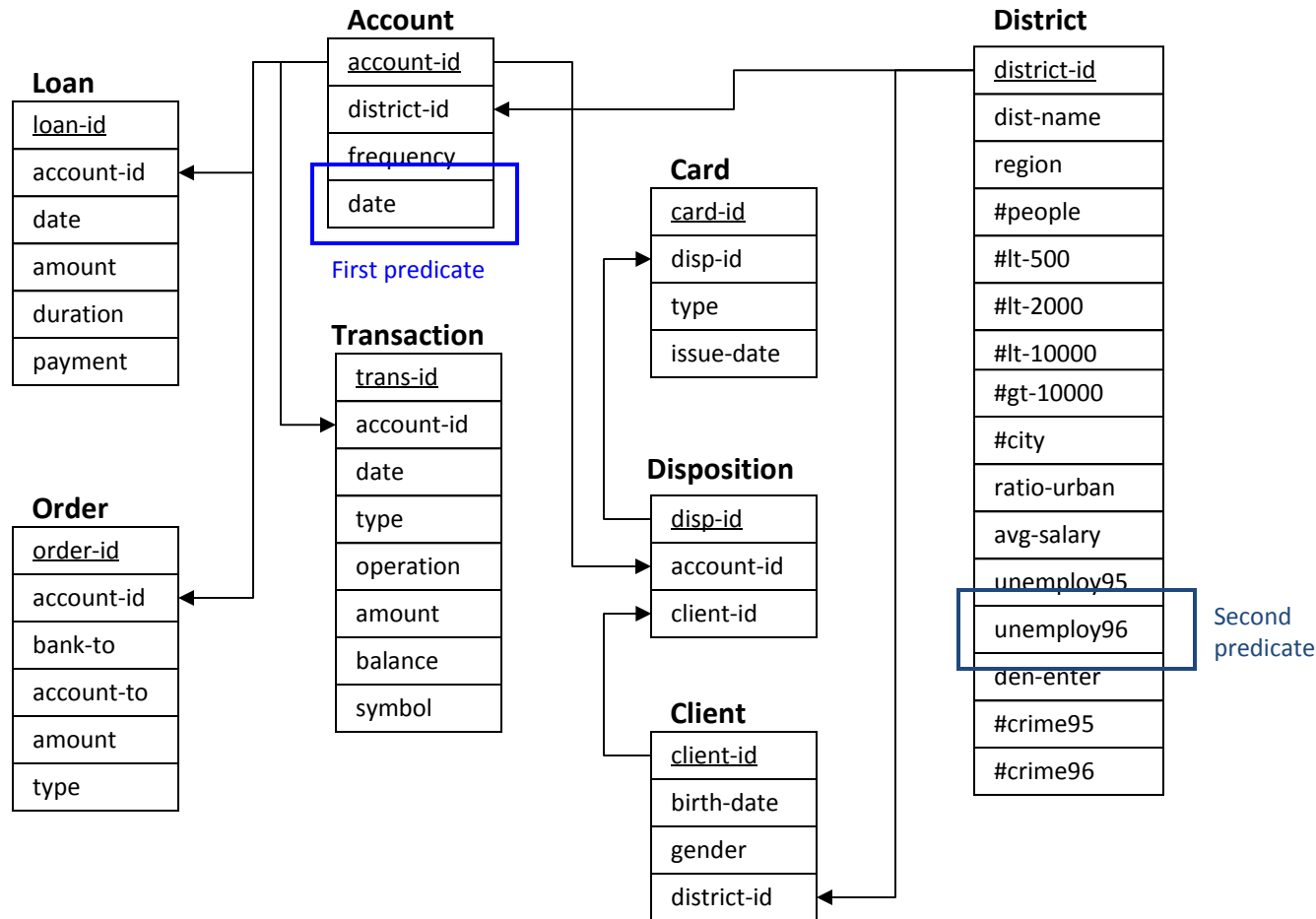
Start at the target relation

Repeat

- Search in all active relations
- Search in all relations joinable to active relations
- Add the best predicate to the current rule
- Set the involved relation to active

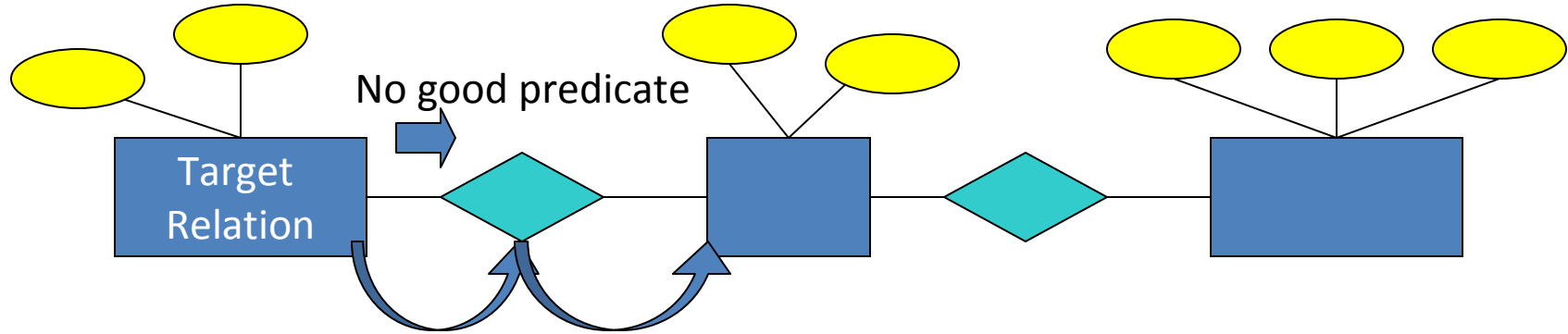
Until

- The best predicate does not have enough gain
- Current rule is too long



# Look-one-ahead in Rule Generation

- Two types of relations: Entity and Relationship
- Often cannot find useful predicates on relations of relationship

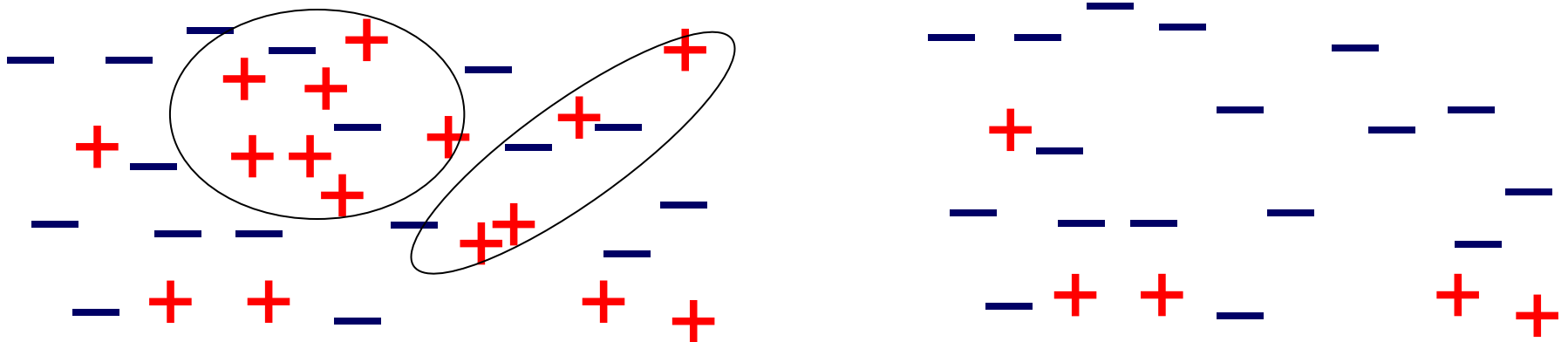


- Solution of CrossMine:
  - When propagating IDs to a relation of relationship, propagate one more step to next relation of entity



# Negative Tuple Sampling

- Each time a rule is generated, covered positive examples are removed
- After generating many rules, there are much less positive examples than negative ones
  - Cannot build good rules (low support)
  - Still time consuming (large number of negative examples)
- Solution: Sampling on negative examples
  - Improve efficiency without affecting rule quality



# Real Dataset

---

- PKDD Cup 99 dataset – Loan Application


	Accuracy	Time (per fold)
FOIL	74.0%	3338 sec
TILDE	81.3%	2429 sec
CrossMine	90.7%	15.3 sec

- Mutagenesis dataset (4 relations): Only 4 relations, so TILDE does a good job, though slow

	Accuracy	Time (per fold)
FOIL	79.7%	1.65 sec
TILDE	89.4%	25.6 sec
CrossMine	87.7%	0.83 sec

# Classification of Information Networks

---

- Classification of Heterogeneous Information Networks:
  - Graph-regularization-Based Method (GNetMine)
  - Multi-Relational-Mining-Based Method (CrossMine)
  - Statistical Relational Learning-Based Method (SRL) 
- Classification of Homogeneous Information Networks

# Probabilistic Relational Models in Statistical Relational Learning

---

- Goal: model distribution of data in relational databases
  - Treat both entities and relations as classes
  - Intuition: objects are no longer independent with each other
    - Build statistical networks according to the dependency relationships between attributes of different classes
- A Probabilistic Relational Models (PRM) consists of
  - Relational schema (from databases)
  - Dependency structure (between attributes)
  - Local probability model (conditional probability distribution)
- Three major methods of probabilistic relational models
  - Relational Bayesian Network (RBN, Lise Getoor et al.)
  - Relational Markov Networks (RMN, Ben Taskar et al.)
  - Relational Dependency Networks (RDN, Jennifer Neville et al.)

# Relational Bayesian Networks (RBN)

---

- Extend Bayesian network to consider entities, properties and relationships in a DB scenario
- Three different uncertainties
  - Attribute uncertainty
    - Model conditional probability for an attribute given its parent variables
  - Structural uncertainty
    - Model conditional probability for a reference or link existence given its parent variables
  - Class uncertainty
    - Refine the conditional probability by considering subclasses or hierarchy of classes


# Rel. Markov Networks & Rel. Dependency Networks

---

- Similar ideas to Relational Bayesian Networks
- Relational Markov Networks
  - Extend from Markov Network
  - Undirected link to model dependency relation instead of directed links as in Bayesian networks
- Relational Dependency Networks
  - Extend from Dependency Network
  - Undirected link to model dependency relation
  - Use pseudo-likelihood instead of exact likelihood
    - Efficient in learning

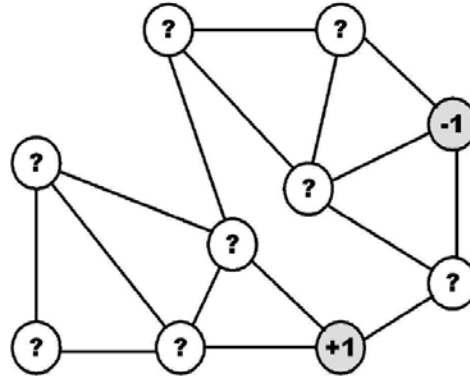
# Classification of Information Networks

---

- Classification of Heterogeneous Information Networks:
  - Graph-regularization-Based Method (GNetMine)
  - Multi-Relational-Mining-Based Method (CrossMine)
  - Statistical Relational Learning-Based Method (SRL)
- Classification of Homogeneous Information Networks 

# Transductive Learning in the Graph

- Problem: for a set of nodes in the graph, the class labels are given for partial of the nodes, the task is to learn the labels of the unlabeled nodes



- Methods
  - Label propagation algorithm [Zhu et al. 2002, Zhou et al. 2004, Szummer et al. 2001]
    - Iteratively propagate labels to its neighbors, according to the transition probability defined by the network
  - Graph regularization-based algorithm [Zhou et al. 2004]
    - Intuition: trade off between (1) consistency with the labeling data and (2) consistency between linked objects
    - An quadratic optimization problem



# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
- **Part I:** Clustering, Ranking and Classification
  - Clustering and Ranking in Information Networks
  - Classification of Information Networks
- **Part II: Data Quality and Search in Information Networks**
  - **Data Cleaning and Data Validation by InfoNet Analysis**
  - Similarity Search in Information Networks
- **Part III:** Advanced Topics on Information Network Analysis
  - Role Discovery and OLAP in Information Networks
  - Mining Evolution and Dynamics of Information Networks
- **Conclusions**

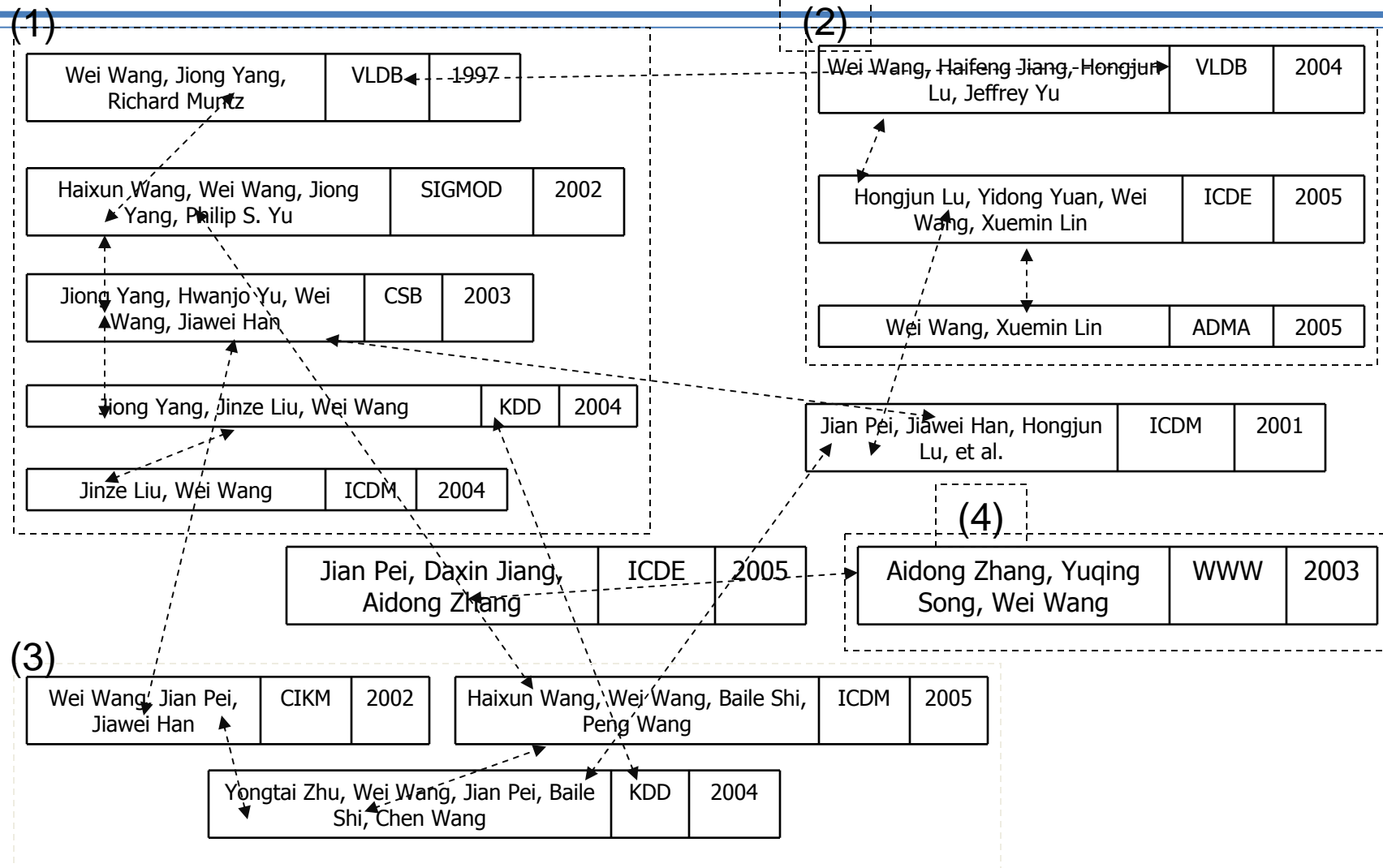


# Data Cleaning by Link Analysis

---

- Object reconciliation vs. object distinction as data cleaning tasks
- Link analysis may take advantages of redundancy and make facilitate entity cross-checking and validation
- Object distinction: Different people/objects do share names
  - In AllMusic.com, 72 songs and 3 albums named “Forgotten” or “The Forgotten”
  - In DBLP, 141 papers are written by at least 14 “Wei Wang”
- New challenges of object distinction:
  - Textual similarity cannot be used
- Distinct: Object distinction by information network analysis
  - X. Yin, J. Han, and P. S. Yu, “Object Distinction: Distinguishing Objects with Identical Names by Link Analysis”, ICDE'07

# Entity Distinction: The “Wei Wang” Challenge in DBLP



(1) Wei Wang at UNC

(2) Wei Wang at UNSW, Australia

(3) Wei Wang at Fudan Univ., China

(4) Wei Wang at SUNY Buffalo

# The DISTINCT Methodology

---

- Measure similarity between references
  - Link-based similarity: Linkages between references
    - References to the same object are more likely to be connected (Using random walk probability)
  - Neighborhood similarity
    - Neighbor tuples of each reference can indicate similarity between their contexts
- Self-boosting: Training using the “same” bulky data set
- Reference-based clustering
  - Group references according to their similarities

# Training with the “Same” Data Set

---

- Build a training set automatically
  - Select distinct names, e.g., Johannes Gehrke
  - The collaboration behavior within the same community share some similarity
  - Training parameters using a typical and large set of “unambiguous” examples
- Use SVM to learn a model for combining different join paths
  - Each join path is used as two attributes (with link-based similarity and neighborhood similarity)
  - The model is a weighted sum of all attributes

# Clustering: Measure Similarity between Clusters

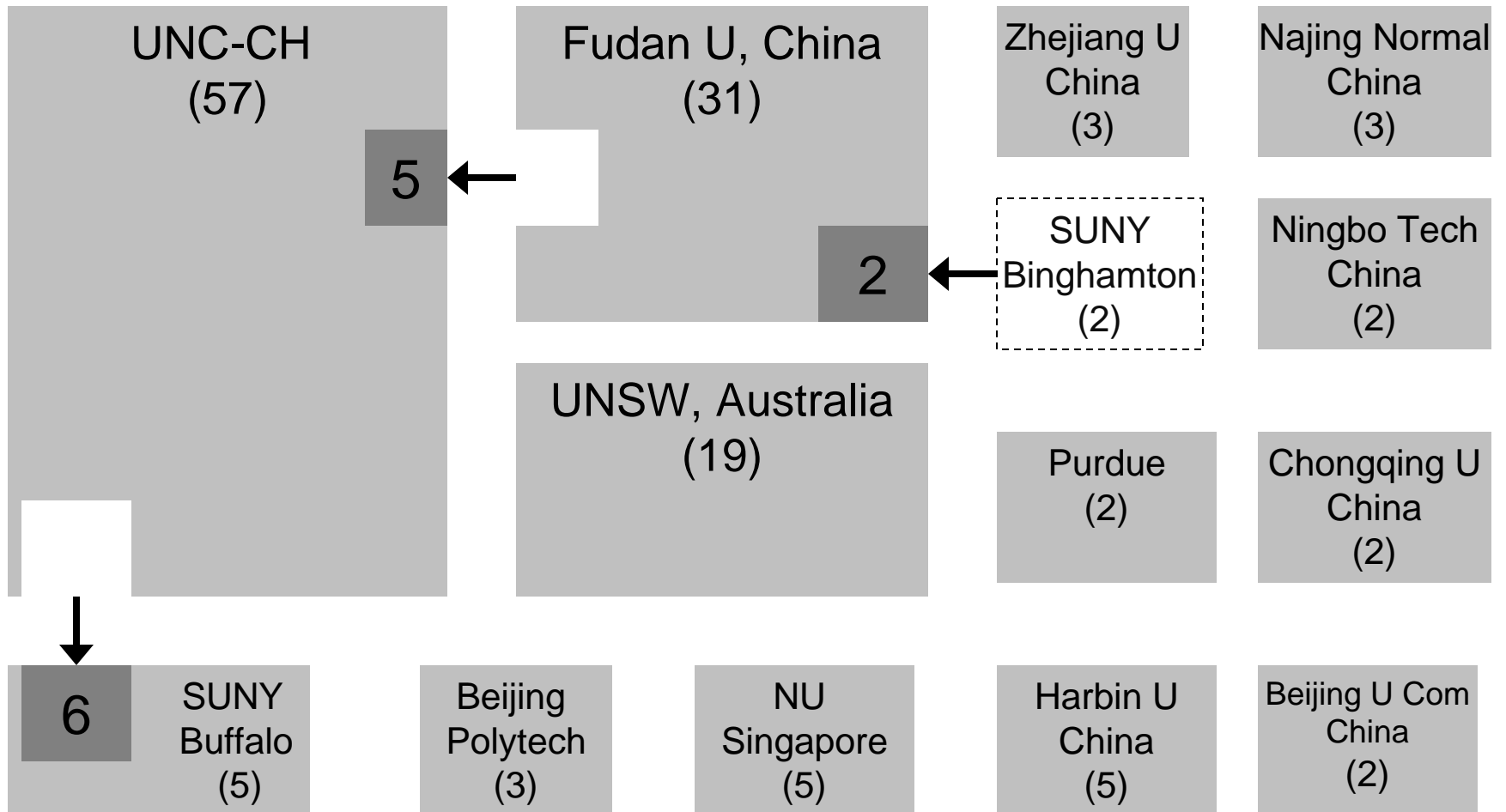
---

- Single-link (highest similarity between points in two clusters) ?
  - No, because references to different objects can be connected.
- Complete-link (minimum similarity between them)?
  - No, because references to the same object may be weakly connected.
- Average-link (average similarity between points in two clusters)?
  - A better measure
  - *Refinement: Average neighborhood similarity and collective random walk probability*

# Real Cases: DBLP Popular Names

<i>Name</i>	<i>Num_authors</i>	<i>Num_refs</i>	<i>accuracy</i>	<i>precision</i>	<i>recall</i>	<i>f-measure</i>
Hui Fang	3	9	1.0	1.0	1.0	1.0
Ajay Gupta	4	16	1.0	1.0	1.0	1.0
Joseph Hellerstein	2	151	0.81	1.0	0.81	0.895
Rakesh Kumar	2	36	1.0	1.0	1.0	1.0
Michael Wagner	5	29	0.395	1.0	0.395	0.566
Bing Liu	6	89	0.825	1.0	0.825	0.904
Jim Smith	3	19	0.829	0.888	0.926	0.906
Lei Wang	13	55	0.863	0.92	0.932	0.926
Wei Wang	14	141	0.716	0.855	0.814	0.834
Bin Yu	5	44	0.658	1.0	0.658	0.794
<i>Average</i>			0.81	0.966	0.836	0.883


# Distinguishing Different “Wei Wang”s





# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
- **Part I:** Clustering, Ranking and Classification
  - Clustering and Ranking in Information Networks
  - Classification of Information Networks
- **Part II: Data Quality and Search in Information Networks**
  - **Data Cleaning and Data Validation by InfoNet Analysis** 
  - Similarity Search in Information Networks
- **Part III:** Advanced Topics on Information Network Analysis
  - Role Discovery and OLAP in Information Networks
  - Mining Evolution and Dynamics of Information Networks
- **Conclusions**

# Truth Validation by Info. Network Analysis

---

- The trustworthiness problem of the web (according to a survey):
  - 54% of Internet users trust news web sites most of time
  - 26% for web sites that sell products
  - 12% for blogs
- TruthFinder: Truth discovery on the Web by link analysis
  - Among multiple conflict results, can we automatically identify which one is likely the true fact?
- Veracity (conformity to truth):
  - Given a large amount of conflicting information about many objects, provided by multiple web sites (or other information providers), how to discover the true fact about each object?
- Our work: Xiaoxin Yin, Jiawei Han, Philip S. Yu, “Truth Discovery with Multiple Conflicting Information Providers on the Web”, TKDE’08

# Conflicting Information on the Web

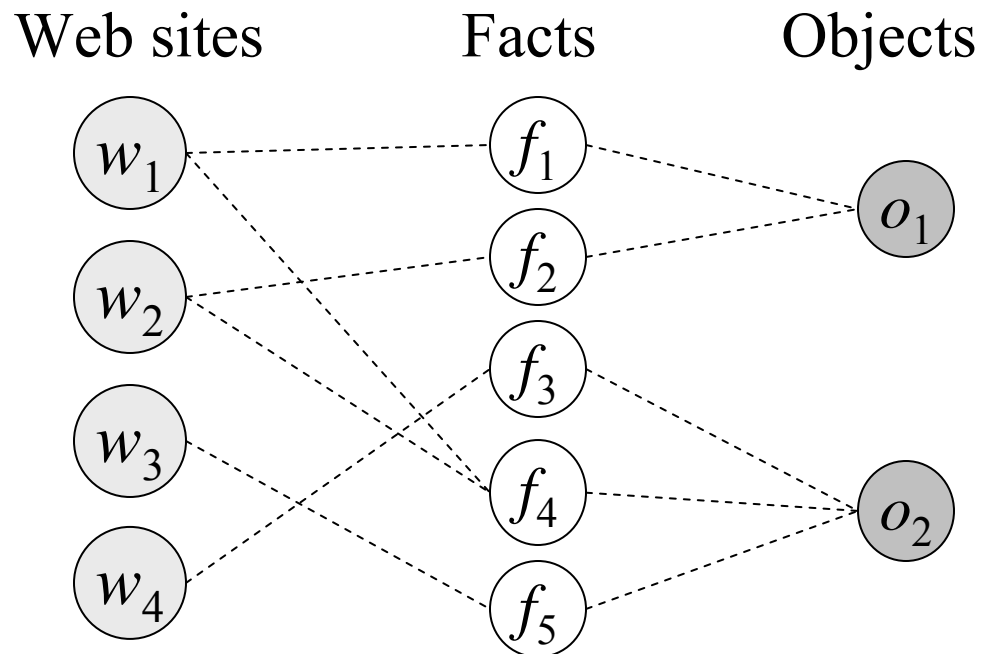
- Different websites often provide conflicting info. on a subject, e.g., Authors of “*Rapid Contextual Design*”

<i>Online Store</i>	<i>Authors</i>
Powell’s books	Holtzblatt, Karen
Barnes & Noble	Karen Holtzblatt, Jessamyn Wendell, Shelley Wood
A1 Books	Karen Holtzblatt, Jessamyn Burns Wendell, Shelley Wood
Cornwall books	Holtzblatt-Karen, Wendell-Jessamyn Burns, Wood
Mellon’s books	Wendell, Jessamyn
Lakeside books	WENDELL, JESSAMYNHOLTZBLATT, KARENWOOD, SHELLEY
Blackwell online	Wendell, Jessamyn, Holtzblatt, Karen, Wood, Shelley

# Our Setting: Info. Network Analysis

---

- Each object has a set of *conflictive* facts
  - E.g., different author names for a book
- And each web site provides some facts
- How to find the true fact for each object?



# Basic Heuristics for Problem Solving

---

1. There is usually only one true fact for a property of an object
2. This true fact appears to be the same or similar on different web sites
  - E.g., “Jennifer Widom” vs. “J. Widom”
3. **The false facts on different web sites are less likely to be the same or similar**
  - False facts are often introduced by random factors
4. **A web site that provides mostly true facts for many objects will likely provide true facts for other objects**

# Overview of the TruthFinder Method

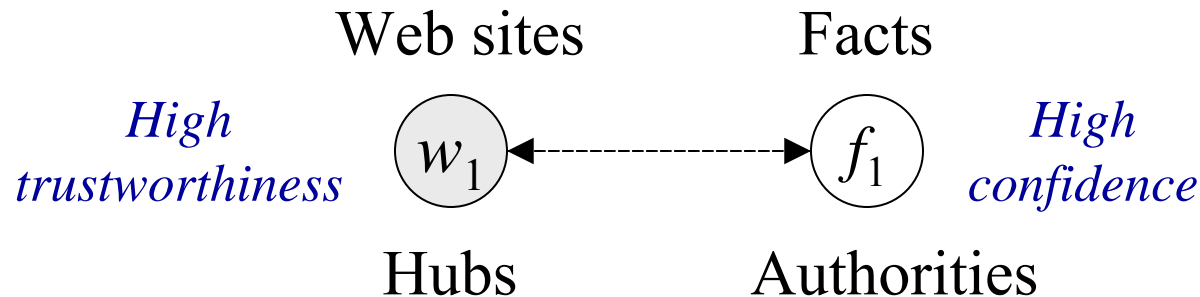
---

- Confidence of facts  $\leftrightarrow$  Trustworthiness of web sites
  - A fact has *high confidence* if it is provided by (many) trustworthy web sites
  - A web site is *trustworthy* if it provides many facts with high confidence
- The TruthFinder mechanism, an overview:
  - Initially, each web site is equally trustworthy
  - Based on the above four heuristics, infer fact confidence from web site trustworthiness, and then backwards
  - Repeat until achieving stable state

# Analogy to Authority-Hub Analysis

---

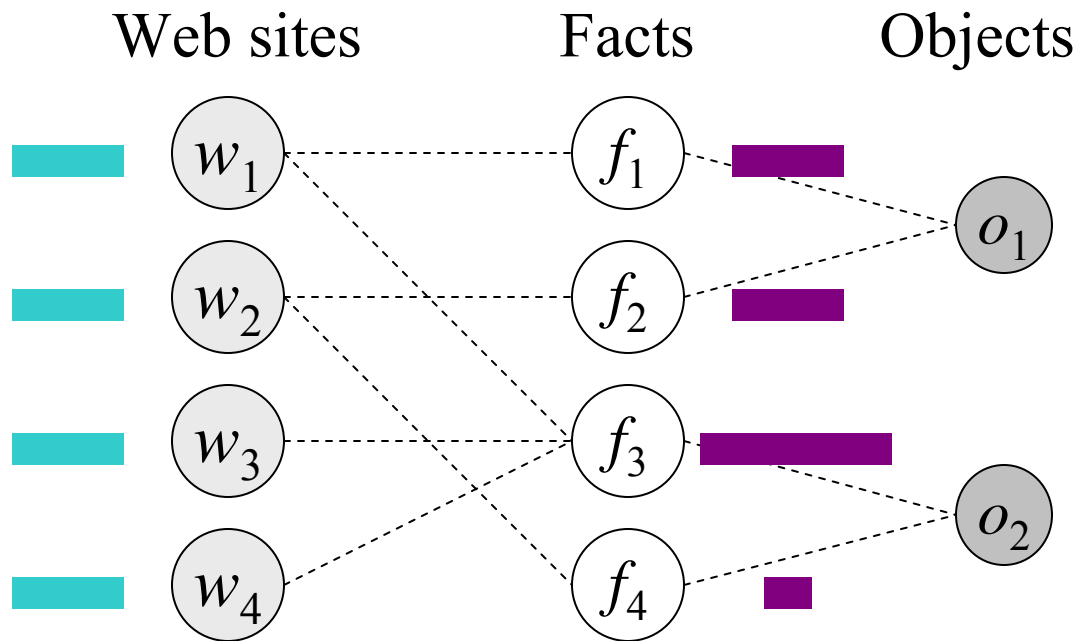
- Facts  $\leftrightarrow$  Authorities, Web sites  $\leftrightarrow$  Hubs



- Difference from authority-hub analysis
  - Linear summation cannot be used
    - A web site is trustable if it provides accurate facts, instead of many facts
    - Confidence is the probability of being true
  - Different facts of the same object influence each other

# Inference on Trustworthiness

- Inference of web site trustworthiness & fact confidence



True facts and trustable web sites will become apparent after some iterations



# Computational Model: $t(w)$ and $s(f)$

- The trustworthiness of a web site  $w$ :  $t(w)$ 
  - Average confidence of facts it provides

$$t(w) = \frac{\sum_{f \in F(w)} s(f)}{|F(w)|}$$

*Sum of fact confidence* (points to the numerator)

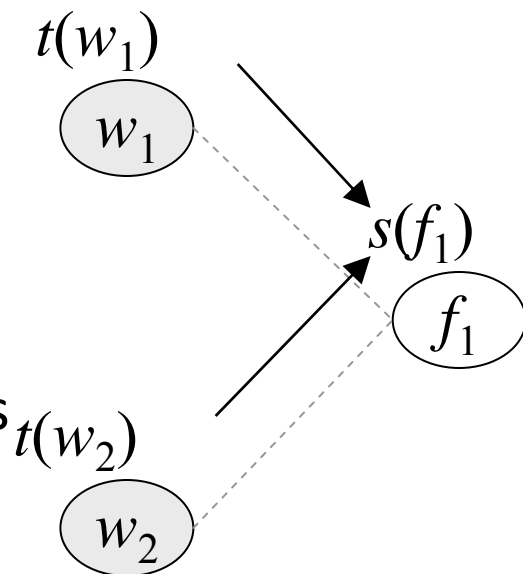
*Set of facts provided by  $w$*  (points to the denominator)

- The confidence of a fact  $f$ :  $s(f)$ 
  - One minus the probability that all web sites providing  $f$  are wrong

$$s(f) = 1 - \prod_{w \in W(f)} (1 - t(w))$$

*Probability that  $w$  is wrong* (points to  $1 - t(w)$ )

*Set of websites providing  $f$*  (points to the product set)



# Experiments: Finding Truth of Facts

---

- Determining authors of books
  - Dataset contains 1265 books listed on abebooks.com
  - We analyze 100 random books (using book images)

<b>Case</b>	<b><i>Voting</i></b>	<b><i>TruthFinder</i></b>	<b><i>Barnes &amp; Noble</i></b>
Correct	71	85	64
Miss author(s)	12	2	4
Incomplete names	18	5	6
Wrong first/middle names	1	1	3
Has redundant names	0	2	23
Add incorrect names	1	5	5
No information	0	0	2

# Experiments: Trustable Info Providers

---

- Finding trustworthy information sources
  - Most trustworthy bookstores found by TruthFinder vs. Top ranked bookstores by Google (query “bookstore”)

## TruthFinder


<b>Bookstore</b>	<i>trustworthiness</i>	<i>#book</i>	<i>Accuracy</i>
TheSaintBookstore	0.971	28	0.959
MildredsBooks	0.969	10	1.0
Alphacraze.com	0.968	13	0.947

## Google

<b>Bookstore</b>	<i>Google rank</i>	<i>#book</i>	<i>Accuracy</i>
Barnes & Noble	1	97	0.865
Powell’s books	3	42	0.654

# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
  - **Part I:** Clustering, Ranking and Classification
    - Clustering and Ranking in Information Networks
    - Classification of Information Networks
  - **Part II: Data Quality and Search in Information Networks**
    - Data Cleaning and Data Validation by InfoNet Analysis
    - **Similarity Search in Information Networks** 
  - **Part III:** Advanced Topics on Information Network Analysis
    - Role Discovery and OLAP in Information Networks
    - Mining Evolution and Dynamics of Information Networks
  - **Conclusions**
-

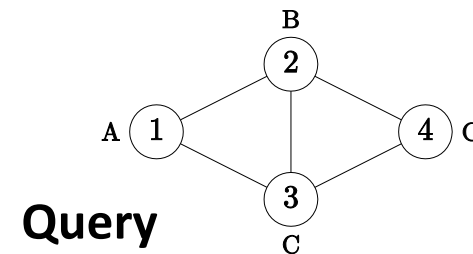
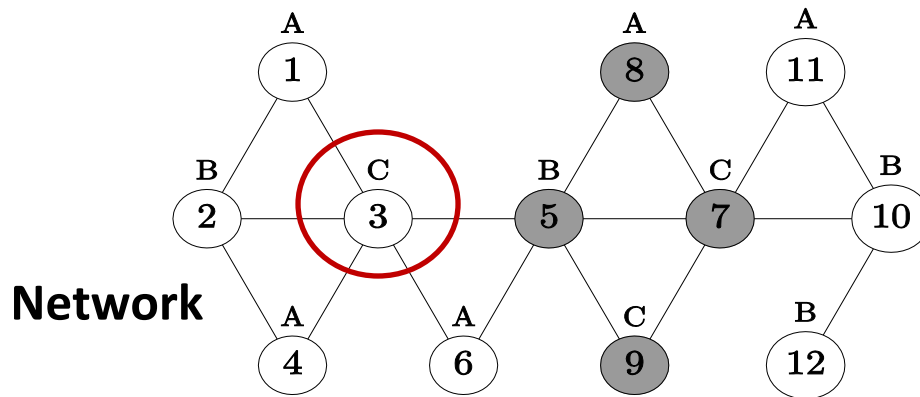
# Similarity Search in Information Networks

---

- Structural similarity vs. semantic similarity
  - Structural similarity: Based on **structural/isomorphic** similarity of sub-graph/sub-network structures
  - Semantic similarity: influenced by similar network structures
- Graph-structure-based indexing and similarity search
  - Structure-based indexing, e.g., gIndex, Spath, ...
  - Use index to search for similar graph/network structures
- Substructure indexing Methods:
  - Key problem: What substructures are good indexing features?
  - **gIndex** [Yan, Yu & Han, SIGMOD'04]: Find *frequent and discriminative subgraphs* (by graph-pattern mining)
  - **Spath** [Zhao & Han, VLDB'10]: Use *decomposed shortest paths* as basic indexing features

# Why S-Path as Indexing Features?

- Shortest paths as neighborhood signatures of vertices (indexing features): scalable and pruning search space effectively
- Processing (by query decomposition):** Decompose the query graph into a set of indexed shortest paths in SPath



**A global lookup table**

label	vid
A	1 4 6 8 11
B	2 5 10 12
C	3 7 9

**Histogram**

distance	label	count
1	A	3
	B	2
2	A	1
	C	2

**ID-List**

vid
1 4 6
2 5
8
7 9

Dashed arrows indicate the mapping from the histogram to the ID-list: (1, A) to [1, 4, 6], (1, B) to [2, 5], (2, A) to [8], and (2, C) to [7, 9].

**Neighborhood signature of  $v_3$**

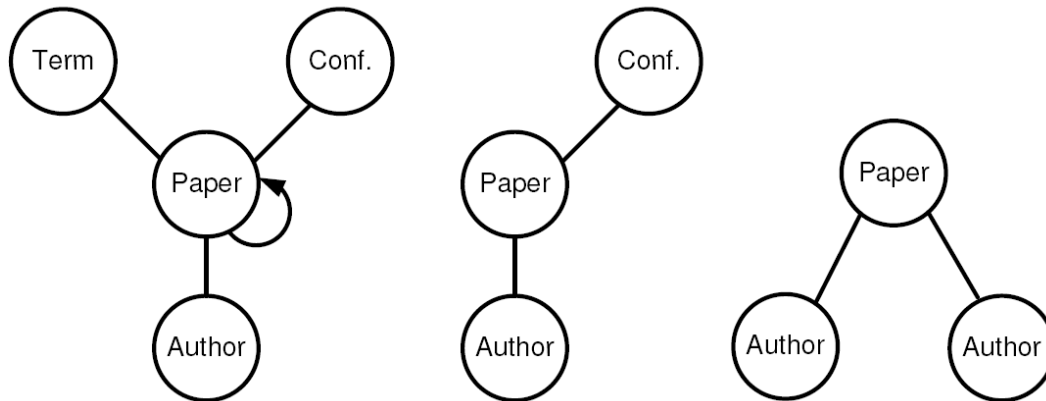
# Semantics-Based Similarity Search in InfoNet

---

- Search top-k similar objects of the same type for a query
  - Find researchers most similar with “Christos Faloutsos”
- Two critical concepts to define a similarity
  - Feature space
    - Traditional data: attributes denoted as numerical value/vector, set, etc.
    - Network data: a relation sequence called “**path schema**”
      - Existing homogeneous network-based similarity does not deal with this problem
  - Measure defined on the feature space
    - Cosine, Euclidean distance, Jaccard coefficient, etc.
    - **PathSim**

# Path Schema for DBLP Queries

- Path schema: A path of InfoNet schema, e.g., APC, APA
- Who are most similar to Christos Faloutsos? (a) Path: *APA*



(a) InfoNet Schema (b) Path Schema: APC/CPA (c) Path Schema: APA

Rank	Author	Score
1	Christos Faloutsos	1
2	Spiros Papadimitriou	0.127
3	Jimeng Sun	0.12
4	Jia-Yu Pan	0.114
5	Agma J. M. Traina	0.110
6	Jure Leskovec	0.096
7	Caetano Traina Jr.	0.096
8	Hanghang Tong	0.091
9	Deepayan Chakrabarti	0.083
10	Flip Korn	0.053

(b) Path: *APCPA*

Rank	Author	Score
1	Christos Faloutsos	1
2	Jiawei Han	0.842
3	Rakesh Agrawal	0.838
4	Jian Pei	0.8
5	Charu C. Aggarwal	0.739
6	H. V. Jagadish	0.705
7	Raghu Ramakrishnan	0.697
8	Nick Koudas	0.689
9	Surajit Chaudhuri	0.677
10	Divesh Srivastava	0.661

(c) Path: *APTPA*

Rank	Author	Score
1	Christos Faloutsos	1
2	Jian Pei	0.661
3	Srinivasan Parthasarathy	0.600
4	Jeffrey Xu Yu	0.587
5	Ming-Syan Chen	0.579
6	Jiawei Han	0.576
7	Mohammed Javeed Zaki	0.571
8	Hans-Peter Kriegel	0.563
9	Yannis Manolopoulos	0.548
10	Rakesh Agrawal	0.545



# Flickr: Which Pictures Are Most Similar?

- Some path schema leads to similarity closer to human intuition
- But some others are not



Figure 5: Top-6 images in Flickr network under path schema *ITI*

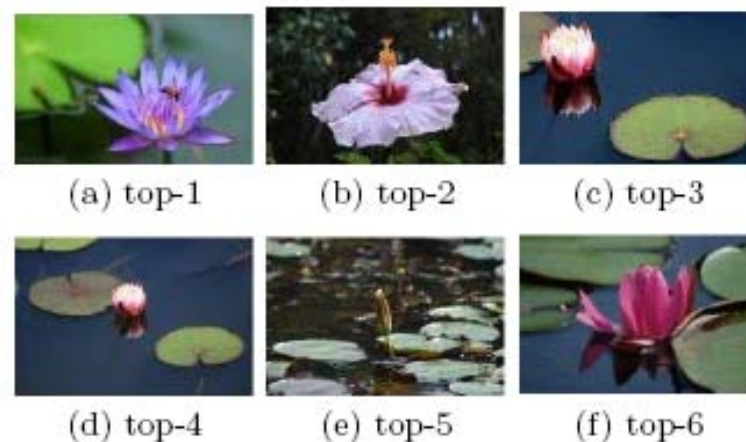


Figure 6: Top-6 images in Flickr network under path schema *ITIGITI*

# Not All Similarity Measures Are Good

(a) P-PageRank: *CPAPC*      (b) PathSim: *CPAPC*

Rank	Conference	Rank	Conference
1	DASFAA	1	DASFAA
2	ICDE	2	DEXA
3	VLDB	3	WAIM
4	SIGMOD Conference	4	APWeb
5	DEXA	5	CIKM
6	TKDE	6	WISE
7	CIKM	7	ICDE
8	Data Knowl. Eng.	8	Data Knowl. Eng.
9	SIGIR	9	PAKDD
10	SIGMOD Record	10	EDBT

Favor highly visible objects

Table 5: P-PageRank vs. PathSim on query: “DASFAA”

(a) SimRank: *CPAPC*      (b) PathSim: *CPAPC*

Rank	Conference	Rank	Conference
1	SIGMOD Conf.	1	SIGMOD Conf.
2	Found. and Trends in DB	2	VLDB
3	ACM SIGMOD D. S. C.	3	ICDE
4	HPTS	4	IEEE Data Eng. Bull.
5	DB for Inter. Des.	5	SIGMOD Rec.
6	IPSJ	6	ACM Trans. DB Syst.
7	CIDR	7	TKDE
8	AFIPS NCC	8	PODS
9	XQuery Impl. Parad	9	VLDB J.
10	CleanDB	10	EDBT

Not reasonable

Table 6: SimRank vs. PathSim on query: “SIGMOD”

# Long Path Schema May Not Be Good

- Repeat the path schema 2, 4, and infinite times for conference similarity query

(a) Path:  $(CPAPC)^2$

Rank	Term	Score
1	SIGMOD Conference	1
2	VLDB	0.981
3	ICDE	0.949
4	TKDE	0.650
5	SIGMOD Record	0.630
6	IEEE Data Eng. Bull.	0.530
7	PODS	0.467
8	ACM Trans. Database Syst.	0.429
9	EDBT	0.420
10	CIKM	0.410

(b) Path:  $(CPAPC)^4$

Rank	Term	Score
1	SIGMOD Conference	1
2	VLDB	0.997
3	ICDE	0.996
4	TKDE	0.787
5	SIGMOD Record	0.686
6	PODS	0.586
7	KDD	0.553
8	CIKM	0.540
9	IEEE Data Eng. Bull.	0.532
10	J. Comput. Syst. Sci	0.463

(c) Path:  $(CPAPC)^\infty$

Rank	Term	Score
1	SIGMOD Conference	1
2	AAAI	0.9999
3	ESA	0.9999
4	IEEE Trans. on Commun.	0.9999
5	STACS	0.9997
6	PODC	0.9996
7	NIPS	0.9993
8	Comput. Geom.	0.9992
9	ICC	0.9991
10	ICDE	0.9984

Table 8: Top-10 similar conferences to “SIGMOD” under path schemas with different lengths

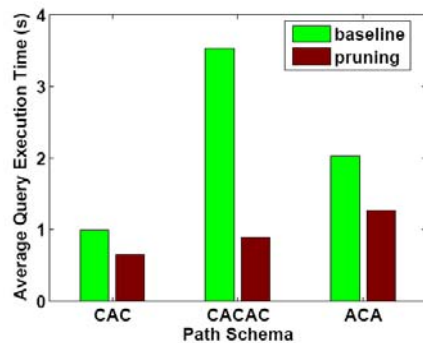
# PathSim: Definition & Properties

---

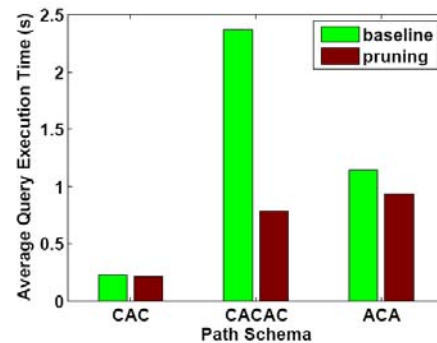
- Commuting matrix corresponding to a path schema  $M_p$ 
  - Product of adjacency matrix of relations in the path schema
  - The element  $M_p(i,j)$  denotes the strength between object  $i$  and object  $j$  in the semantic of path schema  $P$ 
    - If the weight of adjacency matrix is unweighted, it will denote the number of path instances following path schema  $P$
- PathSim
  - $s(i,j) = 2M_p(i,j)/(M_p(i,i) + M_p(j,j))$
  - Properties:
    1. *Symmetric*:  $s(x_i, x_j) = s(x_j, x_i)$ .
    2. *Self-maximum*:  $s(x_i, x_j) \in [0, 1]$ , and  $s(x_i, x_i) = 1$ .

# Co-Clustering-Based Pruning Algorithm

- Store commuting matrices for short path schemas & compute top-k queries on line
- Framework
  - Generate co-clusters for materialized commuting matrices, for feature objects and target objects
  - Derive upper bound for similarity between object and target cluster, and between object and object: Safely pruning target clusters and objects if the upper bound similarity is lower than current threshold
  - Dynamically update top-k threshold:
- Performance: Baseline vs. pruning



(a) Query Set 1 (1-20)



(b) Query Set 2 (1001-1020)

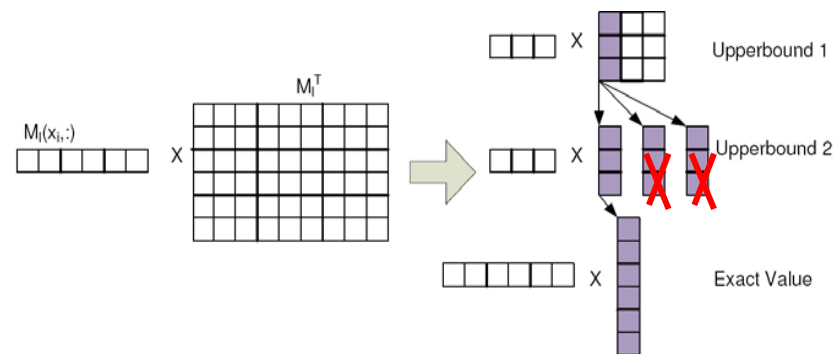



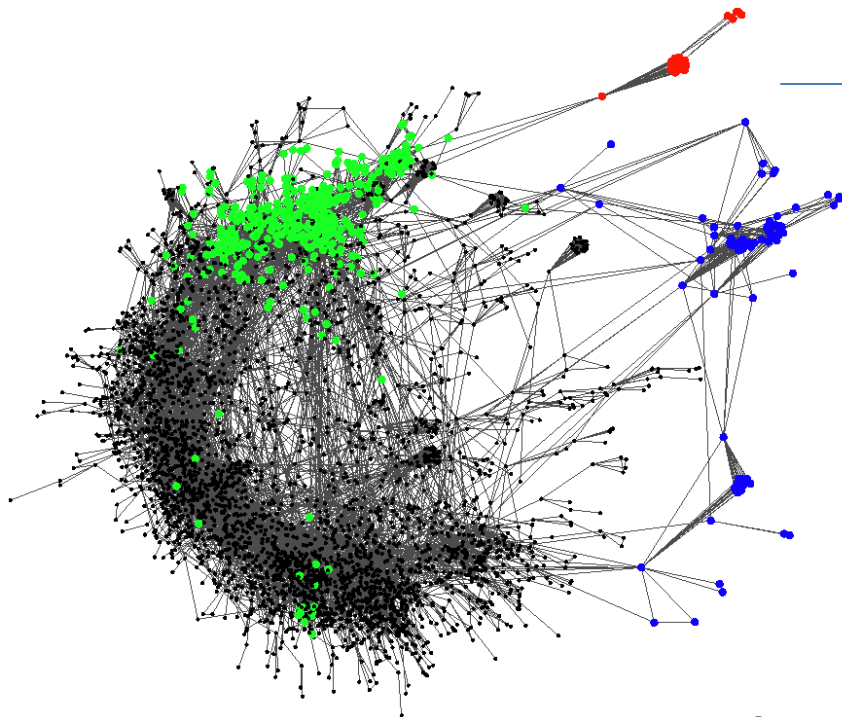
Figure 3: *PathSim*-baseline vs. *PathSim*-pruning

# Outline

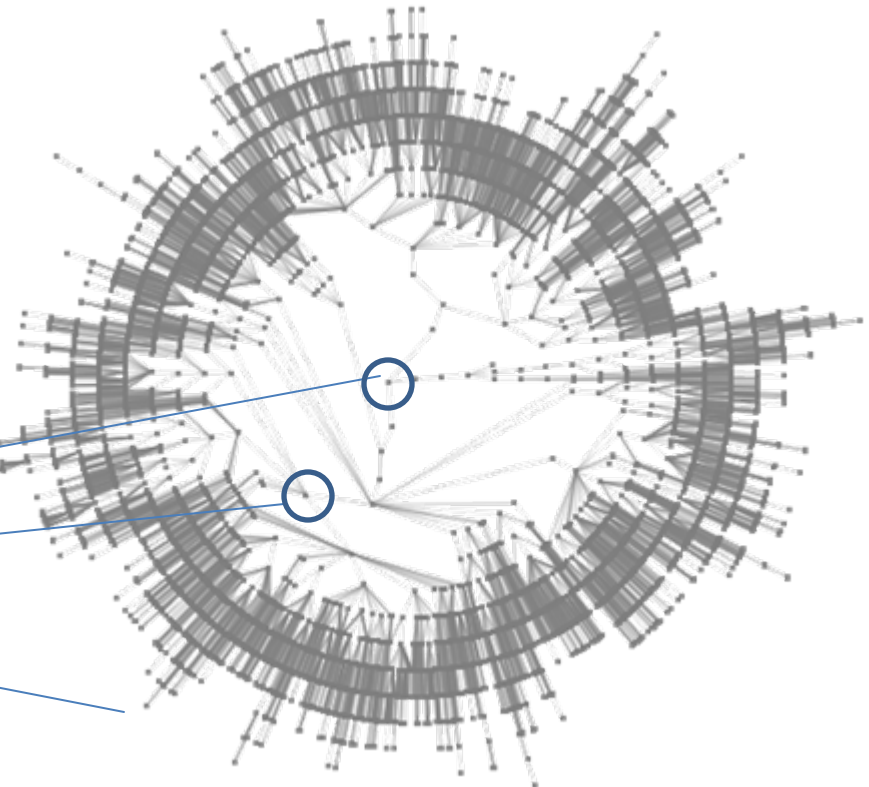
---

- **Motivation:** Why Mining Heterogeneous Information Networks?
- **Part I:** Clustering, Ranking and Classification
  - Clustering and Ranking in Information Networks
  - Classification of Information Networks
- **Part II:** Data Quality and Search in Information Networks
  - Data Cleaning and Data Validation by InfoNet Analysis
  - Similarity Search in Information Networks
- **Part III: Advanced Topics on Information Network Analysis**
  - **Role Discovery and OLAP in Information Networks** 
  - Mining Evolution and Dynamics of Information Networks
- **Conclusions**

# Role Discovery in Network: Why It Matters?



Army communication network (imaginary)



Automatically  
infer

Commander  
Captain  
Soldier

# Role Discovery: Extraction Semantic Information from Links

---

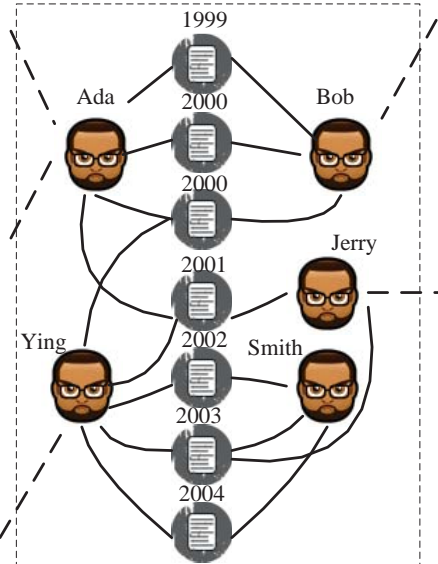
- Objective: Extract semantic meaning from plain links to finely model and better organize information networks
- Challenges
  - Latent semantic knowledge
  - Interdependency
  - Scalability
- Opportunity
  - Human intuition
  - Realistic constraint
  - Crosscheck with collective intelligence
- Methodology: propagate simple intuitive rules and constraints over the whole network



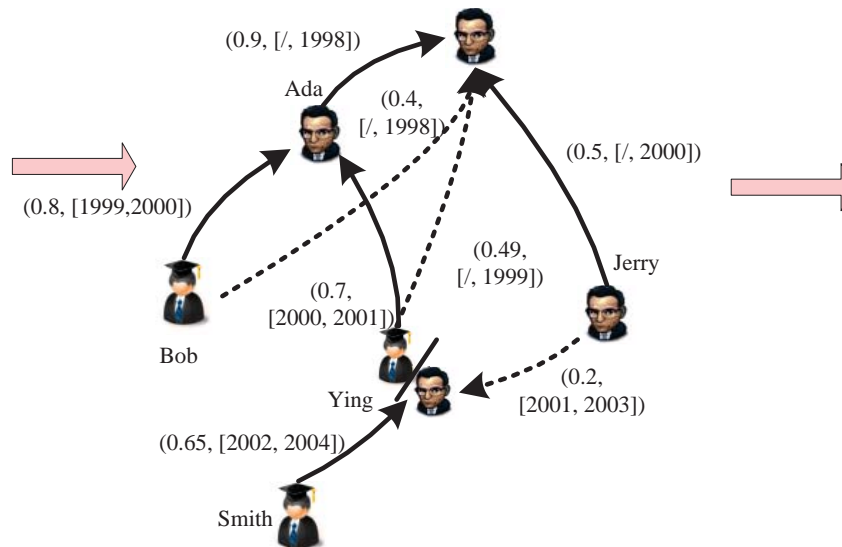
# Discovery of Advisor-Advisee Relationships in DBLP Network

- Input: DBLP research publication network
- Output: Potential advising relationship and its ranking  $(r, [st, ed])$
- Ref. C. Wang, J. Han, et al., "Mining Advisor-Advisee Relationships from Research Publication Networks", SIGKDD 2010

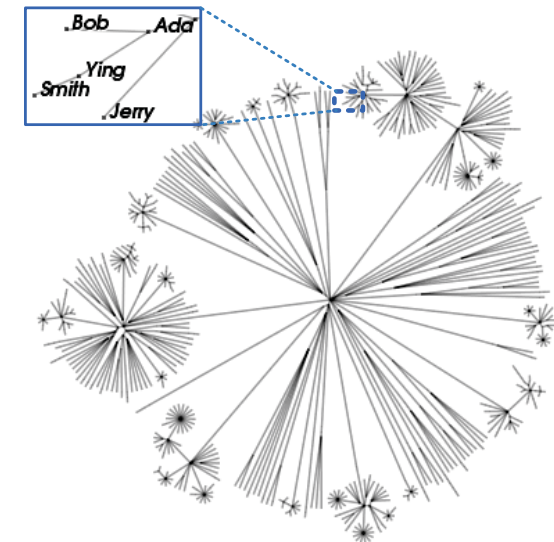
Input: Temporal collaboration network



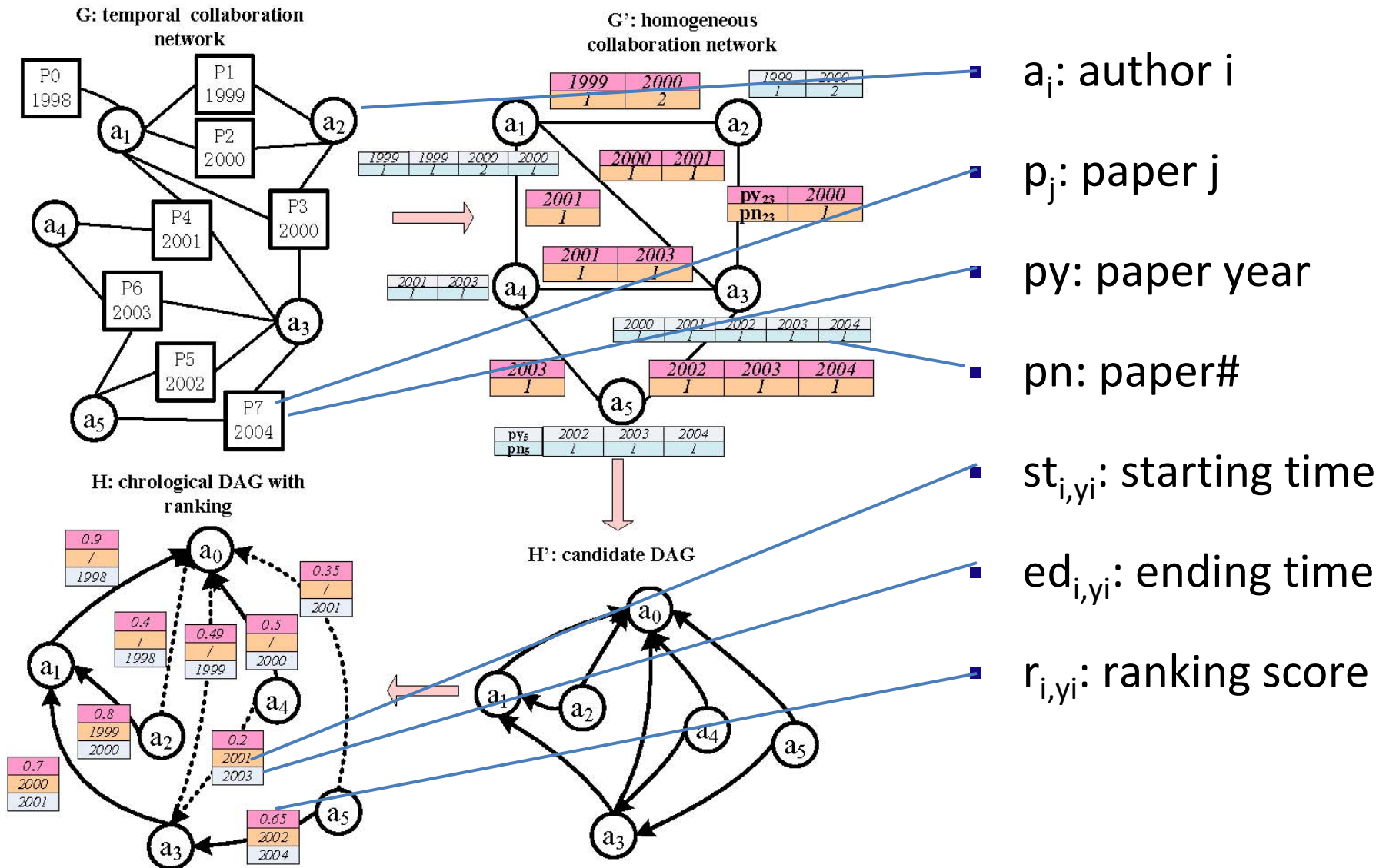
Output: Relationship analysis



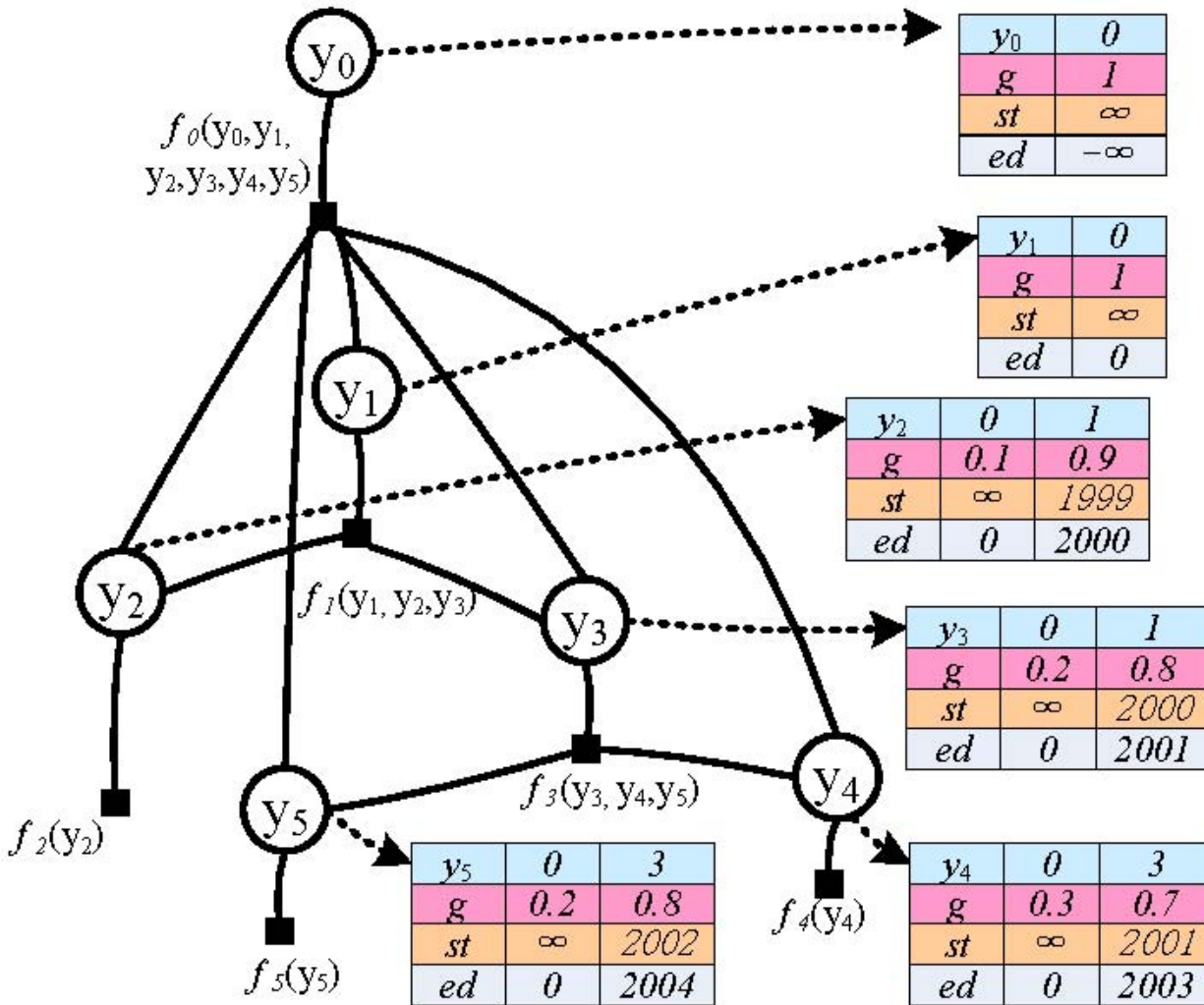
Visualized chorological hierarchies



# Overall Framework



# Time-Constrained Probabilistic Factor Graph (TPFG)

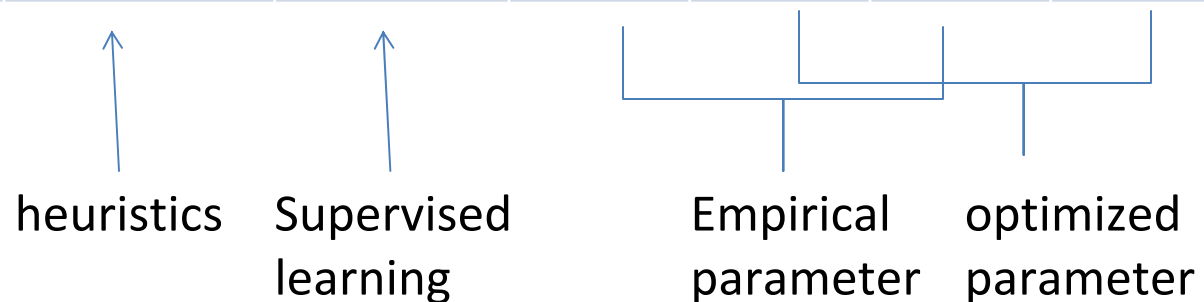


- $y_x$ :  $a_x$ 's advisor
- $st_{x,yx}$ : starting time
- $ed_{x,yx}$ : ending time
- $g(y_x, st_x, ed_x)$  is predefined local feature
- $f_x(y_x, Z_x) = \max g(y_x, st_x, ed_x)$  under time constraint
- Objective function  $P(\{y_x\}) = \prod_x f_x(y_x, Z)$
- $Z = \{z \mid x \in Y_z\}$
- $Y_x$ : set of potential advisors of  $a_x$

# Experiment Results

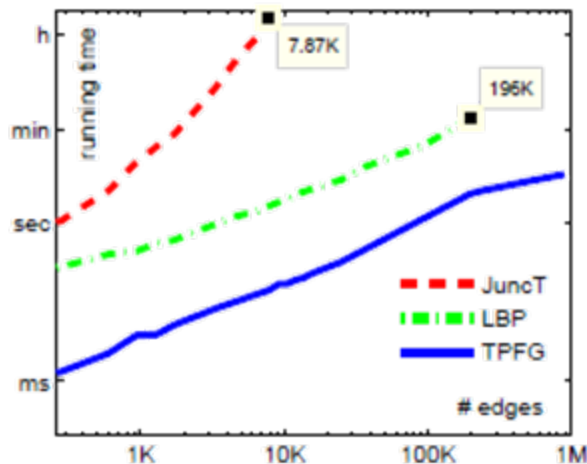
- DBLP data: 654, 628 authors, 1076,946 publications, years provided
- Labeled data: MathGeology Project; AI Geology Project; Homepage

Datasets	RULE	SVM	IndMAX		TPFG	
TEST1	69.9%	73.4%	75.2%	78.9%	80.2%	<b>84.4%</b>
TEST2	69.8%	74.6%	74.6%	79.0%	81.5%	<b>84.3%</b>
TEST3	80.6%	86.7%	83.1%	90.9%	88.8%	<b>91.3%</b>

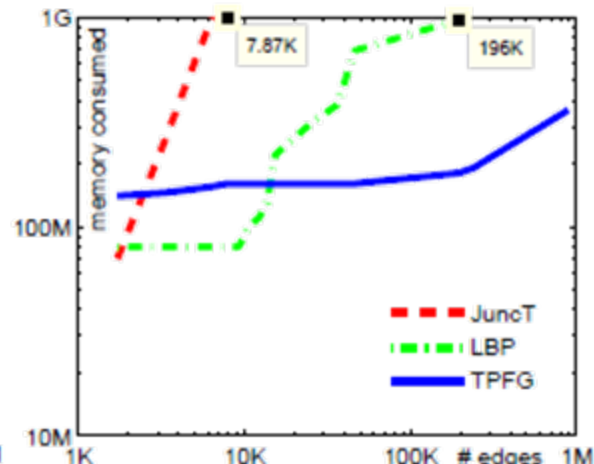


# Case Study & Scalability

Advisee	Top Ranked Advisor	Time	Note
David M. Blei	1. Michael I. Jordan	01-03	PhD advisor, 2004 grad
	2. John D. Lafferty	05-06	Postdoc, 2006
Hong Cheng	1. Qiang Yang	02-03	MS advisor, 2003
	2. Jiawei Han	04-08	PhD advisor, 2008
Sergey Brin	1. Rajeev Motawani	97-98	“Unofficial advisor”



(a) Time

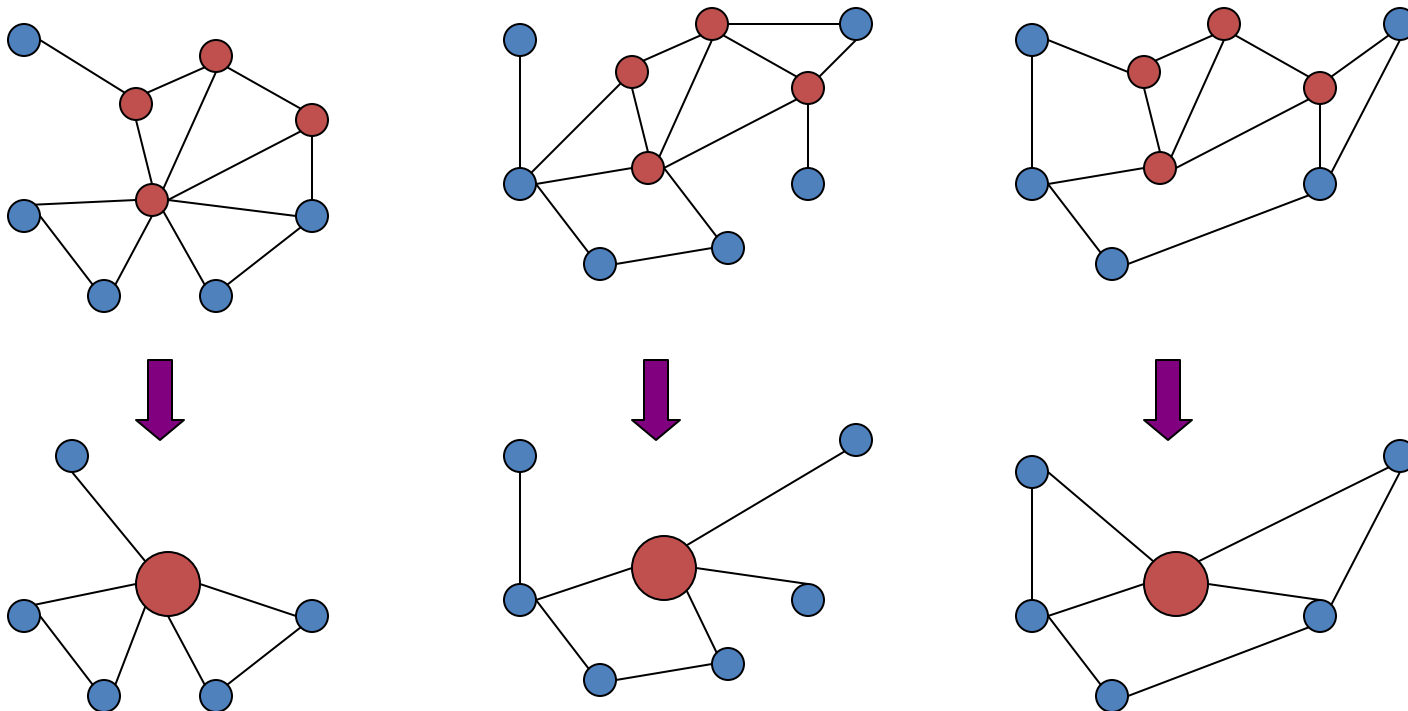


(b) Space

# Graph/Network Summarization: Graph Compression

---

- Extract common subgraphs and simplify graphs by condensing these subgraphs into nodes



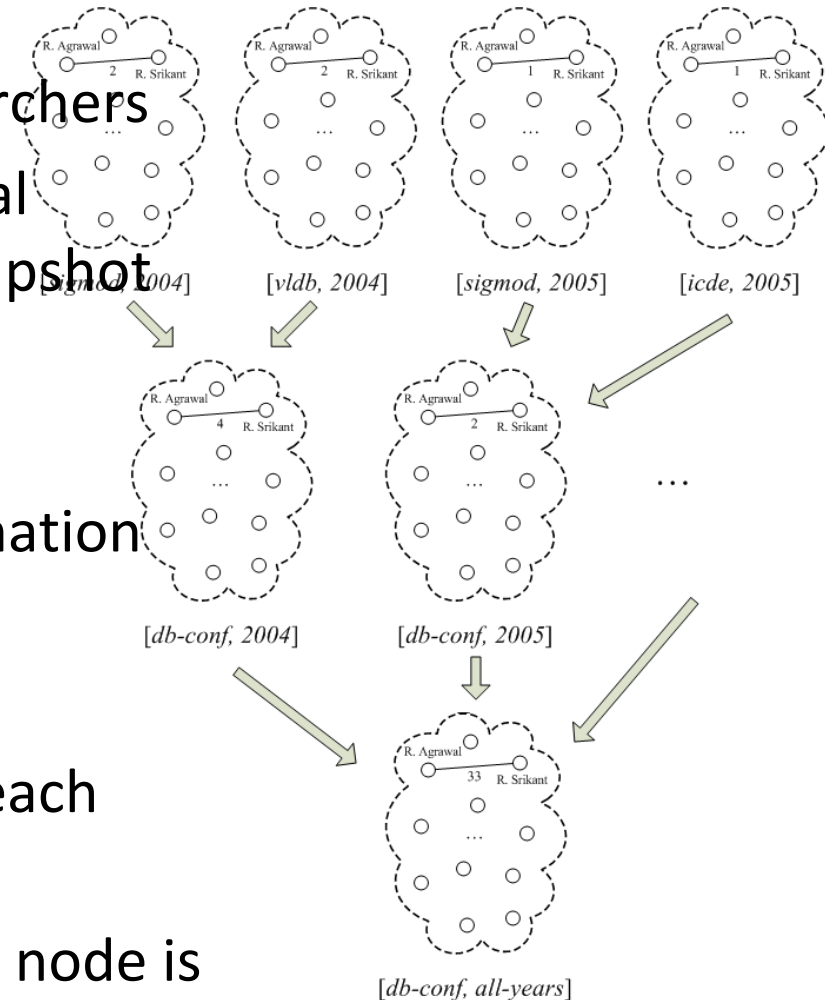
# OLAP on Information Networks

---

- Why OLAP information networks?
- Advantages of OLAP: Interactive exploration of multi-dimensional and multi-level space in a data cube Infonet
  - Multi-dimensional: Different perspectives
  - Multi-level: Different granularities
- InfoNet OLAP: Roll-up/drill-down and slice/dice on information network data
  - Traditional OLAP cannot handle this, because they ignore links among data objects
- Handling two kinds of InfoNet OLAP
  - Informational OLAP
  - Topological OLAP

# Informational OLAP

- In the DBLP network, study the collaboration patterns among researchers
- Dimensions come from informational attributes attached at the whole snapshot level, so-called *Info-Dims*
- I-OLAP Characteristics:
  - Overlay multiple pieces of information
  - No change on the objects whose interactions are being examined
    - In the underlying snapshots, each node is a researcher
    - In the summarized view, each node is still a researcher



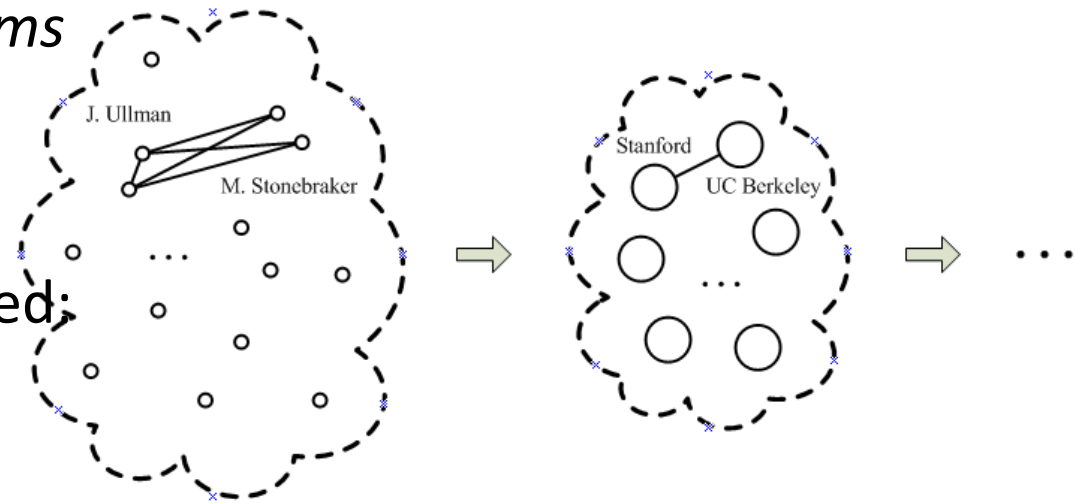


# Topological OLAP

- Dimensions come from the node/edge attributes inside individual networks, so-called *Topo-Dims*

- T-OLAP Characteristics

- Zoom in/Zoom out
- Network topology changed; “generalized” nodes and “generalized” edges
  - In the underlying network, each node is a researcher
  - In the summarized view, each node becomes an institute that comprises multiple researchers



# InfoNet OLAP: Operations & Framework

	InfoNet I-OLAP	InfoNet T-OLAP
<b>Roll-up</b>	Overlay multiple snapshots to form a higher-level summary via I-aggregated network	Shrink the topology & obtain a T-aggregated network that represents a compressed view, with topological elements (i.e., nodes and/or edges) merged and replaced by corresp. higher-level ones
<b>Drill-down</b>	Return to the set of lower-level snapshots from the higher-level overlaid (aggregated) network	A reverse operation of roll-up
<b>Slice/dice</b>	Select a subset of qualifying snapshots based on Info-Dims	Select a subnetwork based on Topo-Dims

- Measure is an aggregated graph & other measures like node count, average degree, etc. can be treated as derived
- Graph plays a dual role: (1) data source, and (2) aggregate measure
- Measures could be complex, e.g., maximum flow, shortest path, centrality
- It is possible to combine I-OLAP and T-OLAP into a hybrid case

# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
- **Part I:** Clustering, Ranking and Classification
  - Clustering and Ranking in Information Networks
  - Classification of Information Networks
- **Part II:** Data Quality and Search in Information Networks
  - Data Cleaning and Data Validation by InfoNet Analysis
  - Similarity Search in Information Networks
- **Part III: Advanced Topics on Information Network Analysis**
  - Role Discovery and OLAP in Information Networks
  - **Mining Evolution and Dynamics of Information Networks**
- **Conclusions**



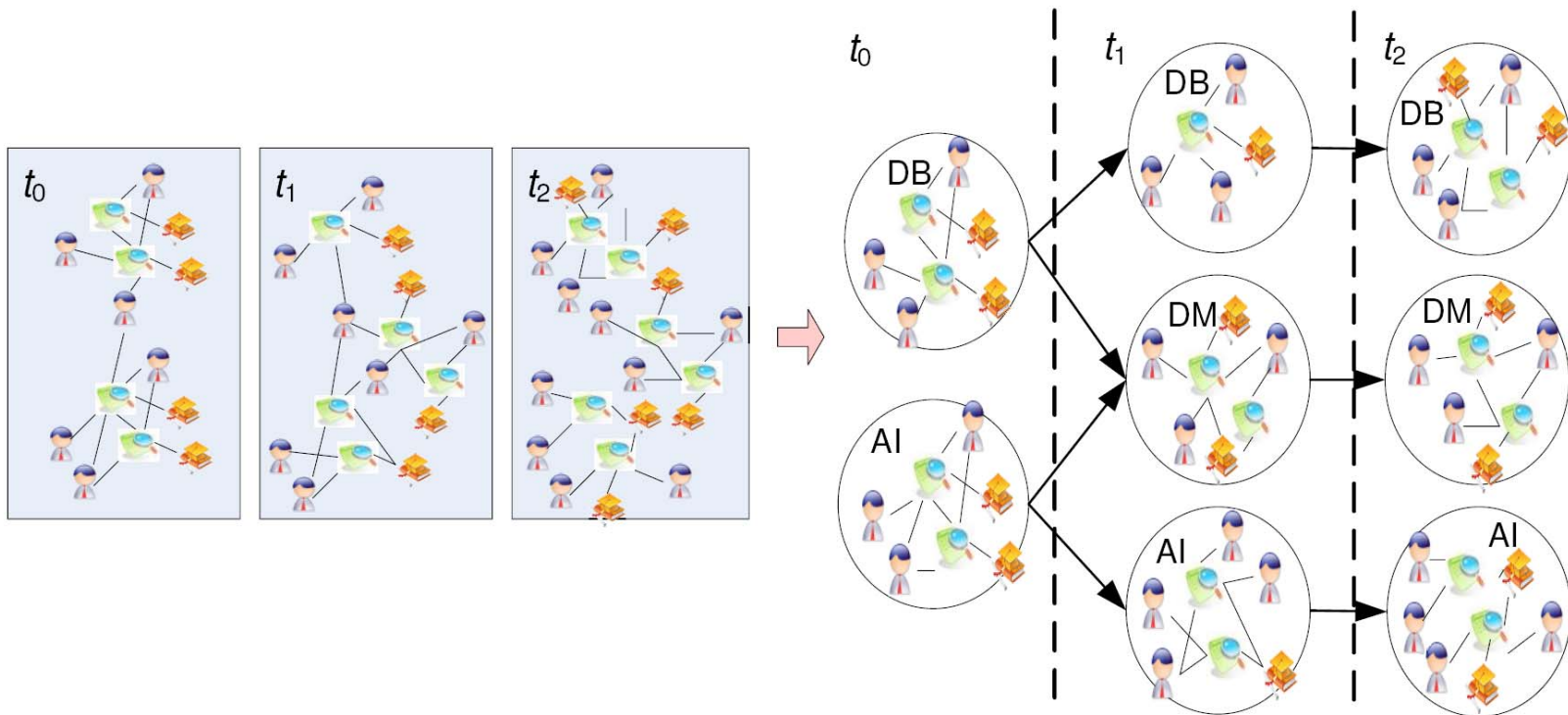
# Mining Evolution and Dynamics of InfoNet

---

- Many networks are with time information
  - E.g., according to paper publication year, DBLP networks can form network sequences
- Motivation: Model evolution of communities in heterogeneous network
  - Automatically detect the best number of communities in each timestamp
  - Model the smoothness between communities of adjacent timestamps
  - Model the evolution structure explicitly
    - Birth, death, split

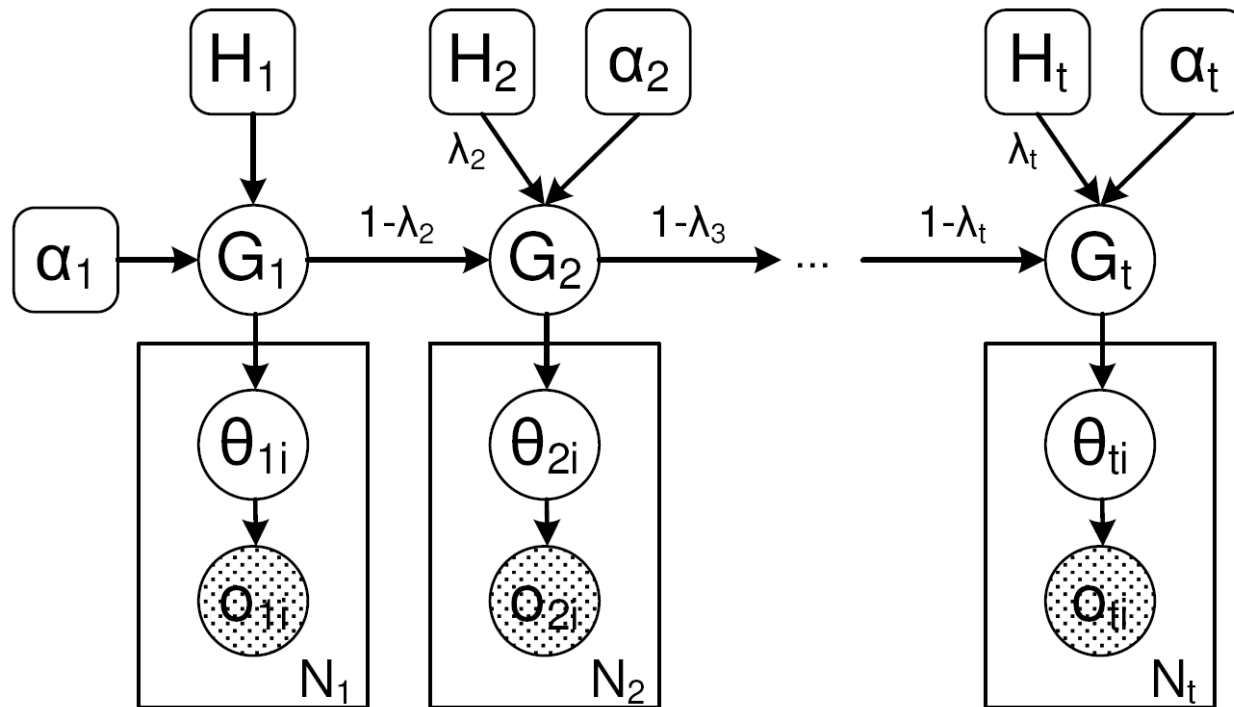
# Evolution: Idea Illustration

- From network sequences to evolutionary communities



# Graphical Model: A Generative Model

- **Dirichlet Process Mixture Model-based generative model**
  - At each timestamp, a community is dependent on historical communities and background community distribution



# Generative Model & Model Inference

- To generate a new paper  $o_i$ 
  - Decide whether to join an existing community or a new one
    - Join an existing community  $k$  with prob.  $n_k / (i-1 + \alpha)$
    - Join a new community  $k$  with prob.  $\alpha / (i-1 + \alpha)$ : Decide its prior, either from a background distribution ( $\lambda$ ) or historical communities ( $(1-\lambda) \pi_k$ ), with different probabilities, draw the attribute distribution from the prior
  - Generate  $o_i$  according to the attribute distribution

$$\begin{aligned} p(o_{i,t} | z_{i,t} = k, \Theta_t) &= p(o_{i,t} | \theta_{k,t}) \\ &= p(\mathbf{a}_{i,t} | \theta_{k,t}^A) p(\mathbf{c}_{i,t} | \theta_{k,t}^C) p(\mathbf{d}_{i,t} | \theta_{k,t}^D) \\ &= \prod_{j=1}^{|A|} \theta_{k,t}^A(j)^{a_{ij,t}} \prod_{j=1}^{|C|} \theta_{k,t}^C(j)^{c_{ij,t}} \prod_{j=1}^{|D|} \theta_{k,t}^D(j)^{d_{ij,t}} \end{aligned}$$

- Greedy inference for each timestamp: Collapse Gibbs sampling, which is trying to sample cluster label for each target object (e.g., paper)

# Accuracy Study

- The more types of objects used, the better accuracy
- Historical prior results in better accuracy

Year	Training Type	Testing Type	Test Size 10% (cluster number $K$ )	Test Size 20% (cluster number $K$ )
1992	Term	Term	1.600 (4)	1.390 (4)
1992	Term+Author	Term+Author	2.205 (8)	1.697 (6)
1992	Term+Author+Conf.	Term+Author	2.434 (8)	2.095 (8)
1992 1991	Term+Author+Conf.	Term+Author	<b>2.8365</b> (8)	<b>2.671</b> (8)

**Table 1: Conference Compactness of Different Models on Test Dataset**

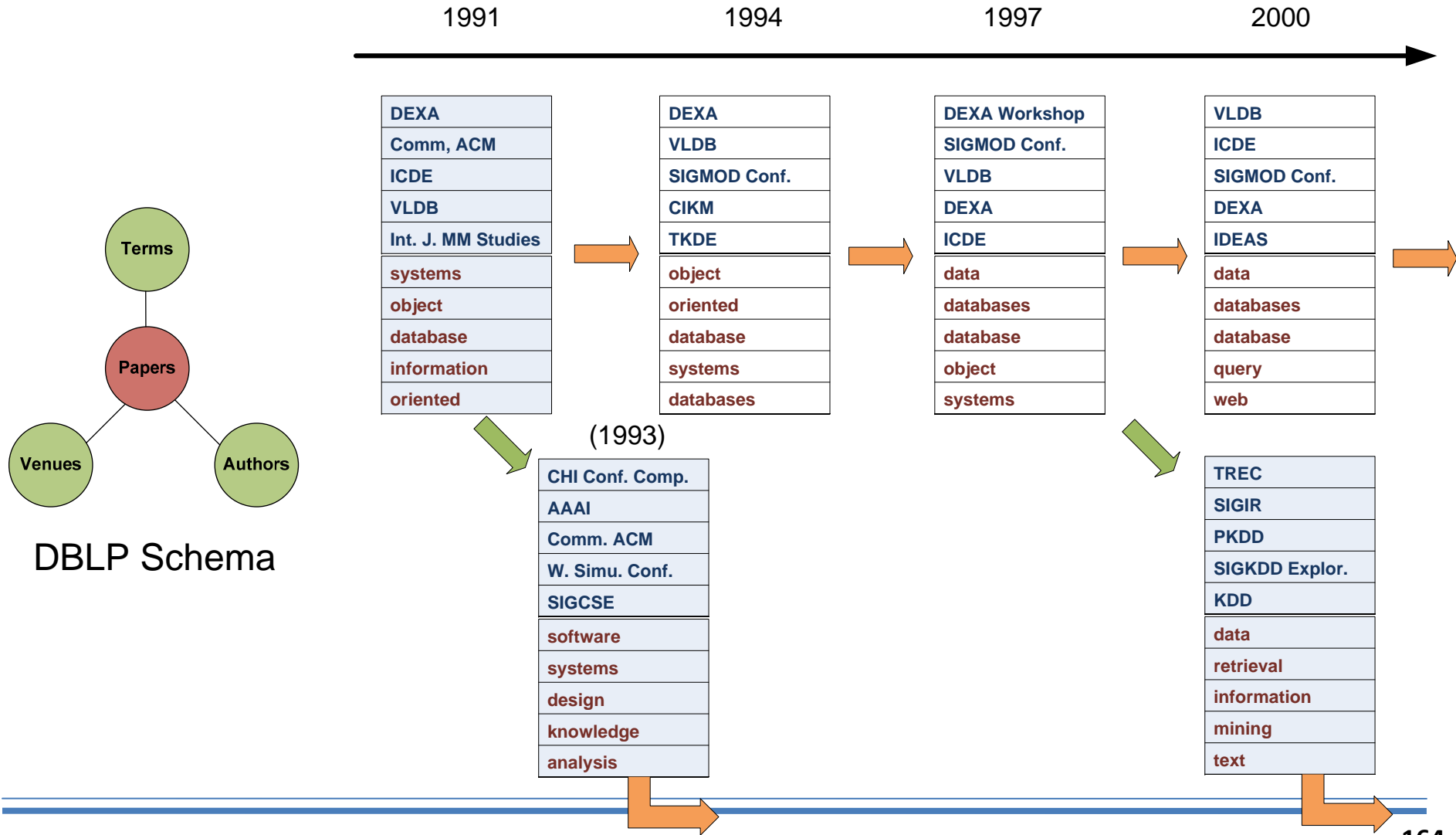
Year	Training Type	Testing Type	Test Size 10%	Test Size 20%
1992	Term+Author+Conf.	Term+Author+Conf.	$3.493 \times 10^{18}$	$4.673 \times 10^{18}$
1992 1991	Term+Author+Conf.	Term+Author+Conf.	<b><math>6.384 \times 10^{17}</math></b>	<b><math>7.106 \times 10^{17}</math></b>

**Table 2: Perplexity Comparison between Models with/without Historical Prior**



# Case Study on DBLP

- Tracking database community evolution



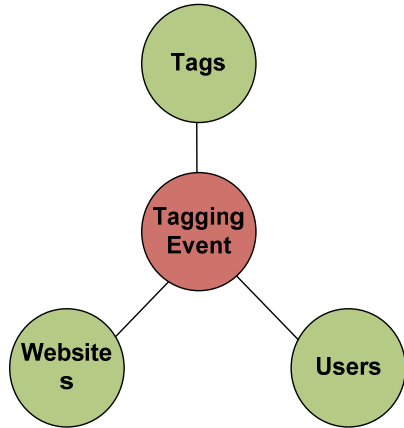
# Case Study on Delicious.com

Jan. 1 - Jan. 7

Jan. 8 - Jan. 14

Jan. 15 - Jan. 21

Jan. 22 - Jan. 28



Delicious Schema

C<sub>1</sub>:

Security
Terrorism
Politics
Travel
Usa
Airport
Israel
Obama
CIA
Afghanistan



Google
China
Security
Internet
Privacy
Politics
Censorship
Facebook
Business
Terrorism



Security
Google
China
Internet
Microsoft
Privacy
Censorship
Politics
Browser
USA



Google
Security
China
Internet
Privacy
Digg
Politics
Datenschutz
Facebook
USA

C<sub>2</sub>:

Mac
Apple
Iphone
Windows
Tablet
Ipod
Tips
Macbook
Tutorial
Drm



Iphone
Apple
Twitter
Mac
Mobile
Apps
Ratio
Blog
Newspapers
Technology



Iphone
Apple
Mac
Mobile
Twitter
Software
Apps
Business
Osx
Radio



Ipad
Apple
Iphone
Technology
Tablet
Mac
Mobile
Newspapers
Kindle
Media

C<sub>3</sub>:

Health
Depression
Sleep
Teenagers
Dubai
Tallest
BBC
Building
Architecture
Mentalhealth



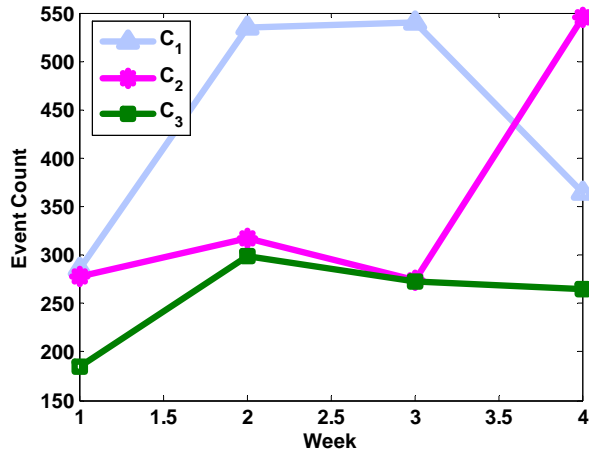
Weather
UK
Photography
Photo
Haiti
Photos
2010
BBC
Snow
Earthquake



Haiti
Photography
BBC
Earthquake
Photos
UK
2010
Disaster
Travel
Wildlife




Haiti
BBC
Photography
Animals
Earthquake
2010
Photos
Nature
Funny
Theonion



# Outline

---

- **Motivation:** Why Mining Heterogeneous Information Networks?
  - **Part I:** Clustering, Ranking and Classification
    - Clustering and Ranking in Information Networks
    - Classification of Information Networks
  - **Part II:** Data Quality and Search in Information Networks
    - Data Cleaning and Data Validation by InfoNet Analysis
    - Similarity Search in Information Networks
  - **Part III: Advanced Topics on Information Network Analysis**
    - Role Discovery and OLAP in Information Networks
    - Mining Evolution and Dynamics of Information Networks
  - **Conclusions** 
-

# Conclusions

---

- Rich knowledge can be mined from information networks
- What is the magic?
  - ***Heterogeneous, structured information networks!***
- Clustering, ranking and classification: Integrated clustering, ranking and classification: RankClus, NetClus, GNetMine, ...
- Data cleaning, validation, and similarity search
- Role discovery, OLAP, and evolutionary analysis
- Knowledge is power, but knowledge is hidden in massive links!
- ***Mining heterogeneous information networks:*** Much more to be explored!!

# Future Research

---

- From mining current single star network schema to ranking, clustering, ..., in multi-star, multi-relational databases
- Mining information networks formed by structured data linking with unstructured data (text, multimedia and Web)
- Mining cyber-physical networks (networks formed by dynamic sensors, image/video cameras, with information networks)
- Enhancing the power of knowledge discovery by transforming massive unstructured data: Incremental information extraction, role discovery, ...  $\Rightarrow$  multi-dimensional structured info-net
- Mining noisy, uncertain, un-trustable massive datasets by information network analysis approach
- Turning Wikipedia and/or Web into structured or semi-structured databases by heterogeneous information network analysis

# References: Books on Network Analysis

---

- A.-L. Barabasi. *Linked: How Everything Is Connected to Everything Else and What It Means*. Plume, 2003.
  - M. Buchanan. *Nexus: Small Worlds and the Groundbreaking Theory of Networks*. W. W. Norton & Company, 2003.
  - D. J. Cook and L. B. Holder. *Mining Graph Data*. John Wiley & Sons, 2007
  - S. Chakrabarti. *Mining the Web: Discovering Knowledge from Hypertext Data*. Morgan Kaufmann, 2003
  - A. Degenne and M. Forse. *Introducing Social Networks*. Sage Publications, 1999
  - P. J. Carrington, J. Scott, and S. Wasserman. *Models and Methods in Social Network Analysis*. Cambridge University Press, 2005.
  - J. Davies, D. Fensel, and F. van Harmelen. *Towards the Semantic Web: Ontology-Driven Knowledge Management*. John Wiley & Sons, 2003.
  - D. Fensel, W. Wahlster, H. Lieberman, and J. Hendler. *Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential*. MIT Press, 2002.
  - L. Getoor and B. Taskar (eds.). *Introduction to statistical learning*. In MIT Press, 2007.
  - B. Liu. *Web Data Mining: Exploring Hyperlinks, Contents, and Usage Data*. Springer, 2006.
  - J. P. Scott. *Social Network Analysis: A Handbook*. Sage Publications, 2005.
  - J. Watts. *Six Degrees: The Science of a Connected Age*. W. W. Norton & Company, 2003.
  - D. J. Watts. *Small Worlds: The Dynamics of Networks between Order and Randomness*. Princeton University Press, 2003.
  - S. Wasserman and K. Faust. *Social Network Analysis: Methods and Applications*. Cambridge University Press, 1994.
-

# References: Some Overview Papers

---

- T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May 2001.
- C. Cooper and A Frieze. A general model of web graphs. *Algorithms*, 22, 2003.
- S. Chakrabarti and C. Faloutsos. Graph mining: Laws, generators, and algorithms. *ACM Comput. Surv.*, 38, 2006.
- T. Dietterich, P. Domingos, L. Getoor, S. Muggleton, and P. Tadepalli. Structured machine learning: The next ten years. *Machine Learning*, 73, 2008
- S. Dumais and H. Chen. Hierarchical classification of web content. *SIGIR'00*.
- S. Dzeroski. Multirelational data mining: An introduction. *ACM SIGKDD Explorations*, July 2003.
- L. Getoor. Link mining: a new data mining challenge. *SIGKDD Explorations*, 5:84{89, 2003.
- L. Getoor, N. Friedman, D. Koller, and B. Taskar. Learning probabilistic models of relational structure. *ICML'01*
- D. Jensen and J. Neville. Data mining in networks. In *Papers of the Symp. Dynamic Social Network Modeling and Analysis*, National Academy Press, 2002.
- T. Washio and H. Motoda. State of the art of graph-based data mining. *SIGKDD Explorations*, 5, 2003.

# References: Some Influential Papers

---

- A. Z. Broder, R. Kumar, F. Maghoul, P. Raghavan, S. Rajagopalan, R. Stata, A. Tomkins, and J. L. Wiener. Graph structure in the web. *Computer Networks*, 33, 2000.
- S. Brin and L. Page. The anatomy of a large-scale hyper-textual web search engine. *WWW'98*.
- S. Chakrabarti, B. E. Dom, S. R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, D. Gibson, and J. M. Kleinberg. Mining the web's link structure. *COMPUTER*, 32, 1999.
- M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. *ACM SIGCOMM'99*
- M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *In Proc. Natl. Acad. Sci. USA* 99, 2002.
- B. A. Huberman and L. A. Adamic. Growth dynamics of world-wide web. *Nature*, 399:131, 1999.
- G. Jeh and J. Widom. SimRank: a measure of structural-context similarity. *KDD'02*
- J. M. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. Tomkins. The web as a graph: Measurements, models, and methods. *COCOON'99*
- D. Kempe, J. Kleinberg, and E. Tardos. Maximizing the spread of influence through a social network. *KDD'03*
- J. M. Kleinberg. Small world phenomena and the dynamics of information. *NIPS'01*
- R. Kumar, P. Raghavan, S. Rajagopalan, D. Sivakumar, A. Tomkins, and E. Upfal. Stochastic models for the web graph. *FOCS'00*
- M. E. J. Newman. The structure and function of complex networks. *SIAM Review*, 45, 2003.



# References: Clustering and Ranking (1)

---

- E. Airoldi, D. Blei, S. Fienberg and E. Xing, “Mixed Membership Stochastic Blockmodels”, JMLR’08
- Liangliang Cao, Andrey Del Pozo, Xin Jin, Jiebo Luo, Jiawei Han, and Thomas S. Huang, “[RankCompete: Simultaneous Ranking and Clustering of Web Photos](#)”, WWW’10
- G. Jeh and J. Widom, “SimRank: a measure of structural-context similarity”, KDD’02
- Jing Gao, Feng Liang, Wei Fan, Chi Wang, Yizhou Sun, and Jiawei Han, “[Community Outliers and their Efficient Detection in Information Networks](#)”, KDD’10
- M. E. J. Newman and M. Girvan, “Finding and evaluating community structure in networks”, Physical Review E, 2004
- M. E. J. Newman and M. Girvan, “Fast algorithm for detecting community structure in networks”, Physical Review E, 2004
- J. Shi and J. Malik, “Normalized cuts and image Segmentation”, *CVPR’97*
- Yizhou Sun, Yintao Yu, and Jiawei Han, "Ranking-Based Clustering of Heterogeneous Information Networks with Star Network Schema", KDD’09
- Yizhou Sun, Jiawei Han, Peixiang Zhao, Zhijun Yin, Hong Cheng, and Tianyi Wu, "RankClus: Integrating Clustering with Ranking for Heterogeneous Information Network Analysis", EDBT’09

# References: Clustering and Ranking (2)

---

- Yizhou Sun, Jiawei Han, Jing Gao, and Yintao Yu, "iTopicModel: Information Network-Integrated Topic Modeling", ICDM'09
- Xiaoxin Yin, Jiawei Han, Philip S. Yu. "[LinkClus: Efficient Clustering via Heterogeneous Semantic Links](#)", VLDB'06.
- Yintao Yu, Cindy X. Lin, Yizhou Sun, Chen Chen, Jiawei Han, Binbin Liao, Tianyi Wu, ChengXiang Zhai, Duo Zhang, and Bo Zhao, "iNextCube: Information Network-Enhanced Text Cube", VLDB'09 (demo)
- A. Wu, M. Garland, and J. Han. Mining scale-free networks using geodesic clustering. KDD'04
- Z. Wu and R. Leahy, "An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation", IEEE Trans. Pattern Anal. Mach. Intell., 1993.
- X. Xu, N. Yuruk, Z. Feng, and T. A. J. Schweiger. SCAN: A structural clustering algorithm for networks. KDD'07
- X. Yin, J. Han, and P. S. Yu. Cross-relational clustering with user's guidance. KDD'05

# References: Network Classification (1)

---

- A. Appice, M. Ceci, and D. Malerba. Mining model trees: A multi-relational approach. ILP'03
- Jing Gao, Feng Liang, Wei Fan, Yizhou Sun, and Jiawei Han, "Bipartite Graph-based Consensus Maximization among Supervised and Unsupervised Models ", NIPS'09
- L. Getoor, N. Friedman, D. Koller and B. Taskar, "Learning Probabilistic Models of Link Structure", JMLR'02.
- L. Getoor, E. Segal, B. Taskar and D. Koller, "Probabilistic Models of Text and Link Structure for Hypertext Classification", IJCAI WS 'Text Learning: Beyond Classification', 2001.
- L. Getoor, N. Friedman, D. Koller, and A. Pfeffer, "Learning Probabilistic Relational Models", chapter in Relation Data Mining, eds. S. Dzeroski and N. Lavrac, 2001.
- M. Ji, Y. Sun, M. Danilevsky, J. Han, and J. Gao. "Graph-based classification on heterogeneous information networks", ECMLPKDD'10.
- Q. Lu and L. Getoor, "Link-based classification", ICML'03
- D. Liben-Nowell and J. Kleinberg, "The link prediction problem for social networks", CIKM'03

# References: Network Classification (2)

---

- J. Neville, B. Gallaher, and T. Eliassi-Rad. Evaluating statistical tests for within-network classifiers of relational data. ICDM'09.
- J. Neville, D. Jensen, L. Friedland, and M. Hay. Learning relational probability trees. KDD'03
- Jennifer Neville, David Jensen, “Relational Dependency Networks”, JMLR'07
- M. Szummer and T. Jaakkola, “Partially labeled classification with markov random walks”, In NIPS, volume 14, 2001.
- M. J. Rattigan, M. Maier, and D. Jensen. Graph clustering with network structure indices. ICML'07
- P. Sen, G. M. Namata, M. Galileo, M. Bilgic, L. Getoor, B. Gallagher, and T. Eliassi-Rad. Collective classification in network data. AI Magazine, 29, 2008.
- B. Taskar, E. Segal, and D. Koller. Probabilistic classification and clustering in relational data. IJCAI'01
- B. Taskar, P. Abbeel, M.F. Wong, and D. Koller, “[Relational Markov Networks](#)”, chapter in L. Getoor and B. Taskar, editors, [Introduction to Statistical Relational Learning](#), 2007
- X. Yin, J. Han, J. Yang, and P. S. Yu, “[CrossMine: Efficient Classification across Multiple Database Relations](#)”, ICDE'04.
- D. Zhou, O. Bousquet, T. N. Lal, J. Weston, and B. Scholkopf, “Learning with local and global consistency”, In *NIPS 16*, Vancouver, Canada, 2004.
- X. Zhu and Z. Ghahramani, “Learning from labeled and unlabeled data with label propagation”, Technical Report, 2002.

# References: Social Network Analysis

---

- B. Aleman-Meza, M. Nagarajan, C. Ramakrishnan, L. Ding, P. Kolari, A. P. Sheth, I. B. Arpinar, A. Joshi, and T. Finin. Semantic analytics on social networks: experiences in addressing the problem of conflict of interest detection. WWW'06
- R. Agrawal, S. Rajagopalan, R. Srikant, and Y. Xu. Mining newsgroups using networks arising from social behavior. WWW'03
- P. Boldi and S. Vigna. The WebGraph framework I: Compression techniques. WWW'04
- D. Cai, Z. Shao, X. He, X. Yan, and J. Han. Community mining from multi-relational networks. PKDD'05
- P. Domingos. Mining social networks for viral marketing. IEEE Intelligent Systems, 20, 2005.
- P. Domingos and M. Richardson. Mining the network value of customers. KDD'01
- P. DeRose, W. Shen, F. Chen, A. Doan, and R. Ramakrishnan. Building structured web community portals: A top-down, compositional, and incremental approach. VLDB'07
- G. Flake, S. Lawrence, C. L. Giles, and F. Coetzee. Self-organization and identification of web communities. IEEE Computer, 35, 2002.
- J. Kubica, A. Moore, and J. Schneider. Tractable group detection on large link data sets. ICDM'03

# References: Data Quality & Search in Networks

---

- I. Bhattacharya and L. Getoor, “Iterative record linkage for cleaning and integration”, Proc. SIGMOD 2004 Workshop on Research Issues on Data Mining and Knowledge Discovery (DMKD'04)
- Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava, “Integrating conflicting data: The role of source dependence”, PVLDB, 2(1):550–561, 2009.
- Xin Luna Dong, Laure Berti-Equille, and Divesh Srivastava, “Truth discovery and copying detection in a dynamic world”, PVLDB, 2(1):562–573, 2009.
- H. Han, L. Giles, H. Zha, C. Li, and K. Tsioutsoulis, “Two supervised learning approaches for name disambiguation in author citations”, ICDL'04.
- Y. Sun, J. Han, T. Wu, X. Yan, and Philip S. Yu, “PathSim: Path Schema-Based Top-K Similarity Search in Heterogeneous Information Networks”, Technical report, CS, UIUC, July 2010.
- X. Yin, J. Han, and P. S. Yu, “Object Distinction: Distinguishing Objects with Identical Names by Link Analysis”, ICDE'07
- X. Yin, J. Han, and P. S. Yu, “Truth Discovery with Multiple Conflicting Information Providers on the Web”, IEEE TKDE, 20(6):796-808, 2008
- P. Zhao and J. Han, “On Graph Query Optimization in Large Networks”, VLDB'10.

# References: Role Discovery, Summarization and OLAP

---

- D. Archambault, T. Munzner, and D. Auber. Topolayout: Multilevel graph layout by topological features. IEEE Trans. Vis. Comput. Graph, 2007.
- Chen Chen, Xifeng Yan, Feida Zhu, Jiawei Han, and Philip S. Yu, "Graph OLAP: Towards Online Analytical Processing on Graphs", ICDM 2008
- Chen Chen, Xifeng Yan, Feida Zhu, Jiawei Han, and Philip S. Yu, "Graph OLAP: A Multi-Dimensional Framework for Graph Data Analysis", KAIS 2009.
- Xin Jin, Jiebo Luo, Jie Yu, Gang Wang, Dhiraj Joshi, and Jiawei Han, "[iRIN: Image Retrieval in Image Rich Information Networks](#)", WWW'10 (demo paper)
- Lu Liu, Feida Zhu, Chen Chen, Xifeng Yan, Jiawei Han, Philip Yu, and Shiqiang Yang, "[Mining Diversity on Networks](#)", DASFAA'10
- Y. Tian, R. A. Hankins, and J. M. Patel. Efficient aggregation for graph summarization. SIGMOD'08
- Chi Wang, Jiawei Han, Yuntao Jia, Jie Tang, Duo Zhang, Yintao Yu, and Jingyi Guo, "[Mining Advisor-Advisee Relationships from Research Publication Networks](#) ", KDD'10
- Zhijun Yin, Manish Gupta, Tim Weninger and Jiawei Han, "[LINKREC: A Unified Framework for Link Recommendation with User Attributes and Graph Structure](#) ", WWW'10

# References: Network Evolution

---

- L. Backstrom, D. Huttenlocher, J. Kleinberg, and X. Lan. Group formation in large social networks: Membership, growth, and evolution. KDD'06
- M.-S. Kim and J. Han. A particle-and-density based evolutionary clustering method for dynamic networks. VLDB'09
- J. Leskovec, J. Kleinberg, and C. Faloutsos. Graphs over time: Densification laws, shrinking diameters and possible explanations. KDD'05
- Yizhou Sun, Jie Tang, Jiawei Han, Manish Gupta, Bo Zhao, “Community Evolution Detection in Dynamic Heterogeneous Information Networks”, KDD-MLG'10