Sampling and Summarization for Social Networks

SDM 2013 Tutorial

Shou-De Lin[*], Mi-Yen Yeh[#], and Cheng-Te Li[*]

[*] Computer Science and Information Engineering, National Taiwan University

[#] Institute of Information Science, Academic Sinica

sdlin@csie.ntu.edu.tw, miyen@iis.sinica.edu.tw, d98944005@csie.ntu.edu.tw

Tutorial slides: http://mslab.csie.ntu.edu.tw/tut-pakdd13/samsum-pakdd13.pdf

(note that this version is very different from the slides in the SDM conference CD)

About This Tutorial

- It is a two-hour tutorial for SDM2013 on social network sampling and summarization
  - >50 papers are surveyed and organized in this talk, but they are by no means complete.
  - We will highlight the trend, categorize different types of strategies, and describe some ongoing works of us
- Agenda
  - Introduction + Sampling (50 min +10 min Q/A)
  - Summarization + conclusion (50 min+10 min Q/A)

Big Social Network➔ Billions of different types of nodes and links

What can be mined from this picture?

Challenges Facing to Mine Big-Network Data

>1 billion

>500 million

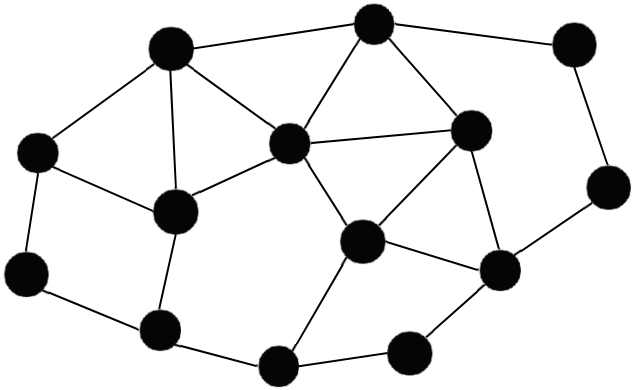
>200 million

Sometimes the full networks are not completely observed in advance

Even they are, loading everything into memory for further analysis might not be feasible

Even it is feasible, performing conventional operations (e.g. average path length) can take a long time, not to mention more complicated ones

An Example on Facebook



- 1+Billion users
- Avg: 130 friends each node

It costs >1TB memory to simply save the raw graph data (without attributes, labels nor content)

This can cause problems for information extraction, processing, and analysis

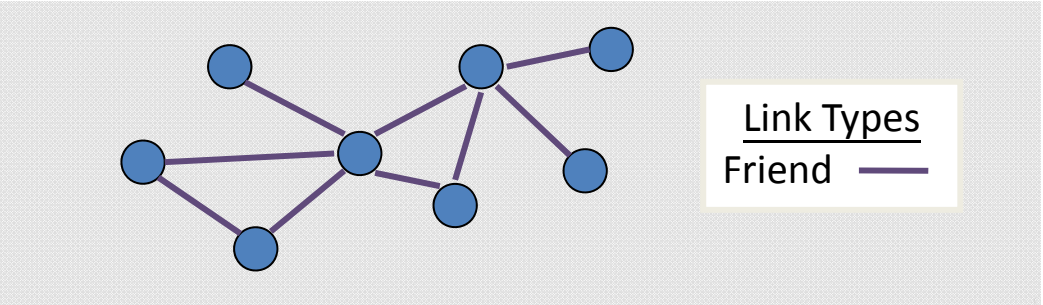
Two possible solutions: **Sampling** and **Summarization**

Sampling Versus Summarization

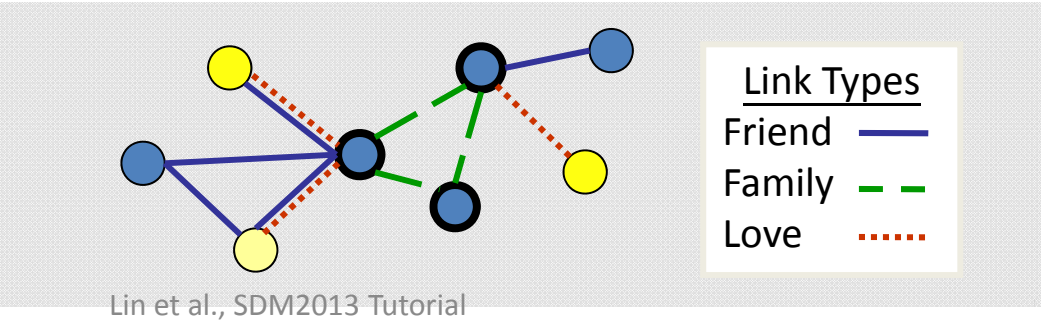
- Sampling for Social Network
  - Assume the information of a node becomes known **only after** it is sampled
  - Goal: gradually identify a small set of **representative** nodes and links of a social network, usually given little prior information about this network
- Summarization Social Network
  - The entire social network is **known in prior**
  - Goal: condense the social network as much as possible without losing too much information

Homogeneous & Heterogeneous Social Networks

- **Homogeneous → Single Relational Network**
  - **Single** object type & Link type
- **Heterogeneous → Multi-Relational Network**
  - **Multiple** object type & Link type
- Example
  - Homogeneous



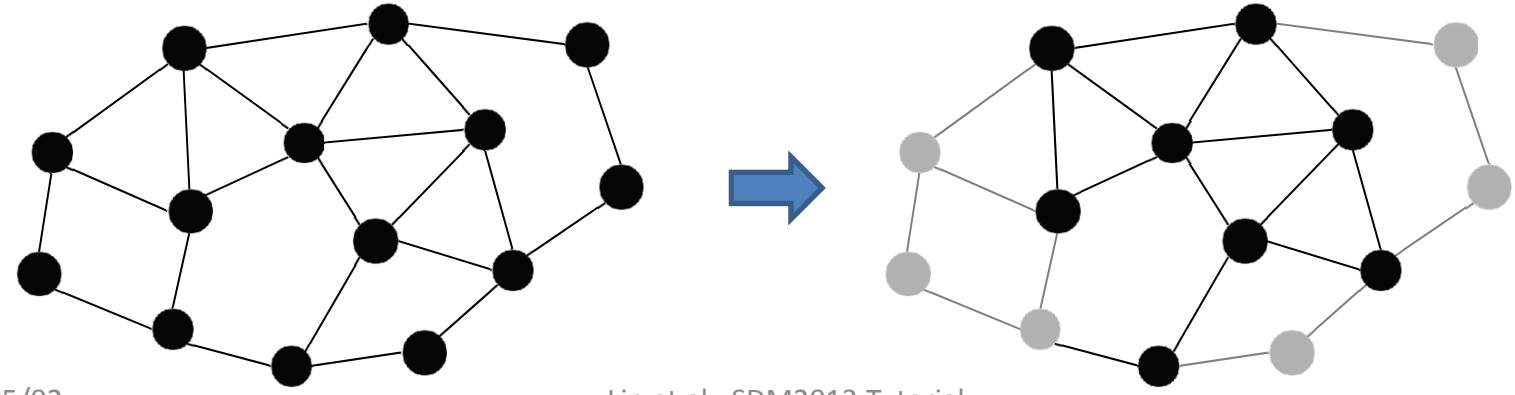
  - Heterogeneous



Sampling for Social Networks

Sampling Social Networks

- Assume that the detailed information of a node can only be seen after it is sampled
  - Entire social network is not known in advance
- Goal
  - Sample (i.e. gradually observe nodes and links) a sub-network that represents the whole network
    - To preserve certain properties of the original network



Properties Preserved (1/3)

- Homogeneous Static Social Network
  - In/Out Degree Distribution
  - Path Length Distribution
  - Clustering Coefficient Distribution
  - Eigenvalues
  - Weakly/Strongly Connected Component Size Distribution
  - Community Structure
  - Etc..

Properties Preserved (2/3)

- Homogeneous Dynamic Social Networks
  (Graphs are time-evolving)
  - Densification Power Law
    - Number of edges vs. number of nodes over time
  - Shrinking diameter
    - Observed that shrinks and stabilizes over time
  - Average clustering coefficient over time
  - Largest singular value of graph adjacency matrix over time
  - Etc…

Properties Preserved (3/3)

- Heterogeneous Social Network
  - Note type Distribution
  - Intra-link and Inter-link type Distribution
  - Distribution on Higher-order types connection

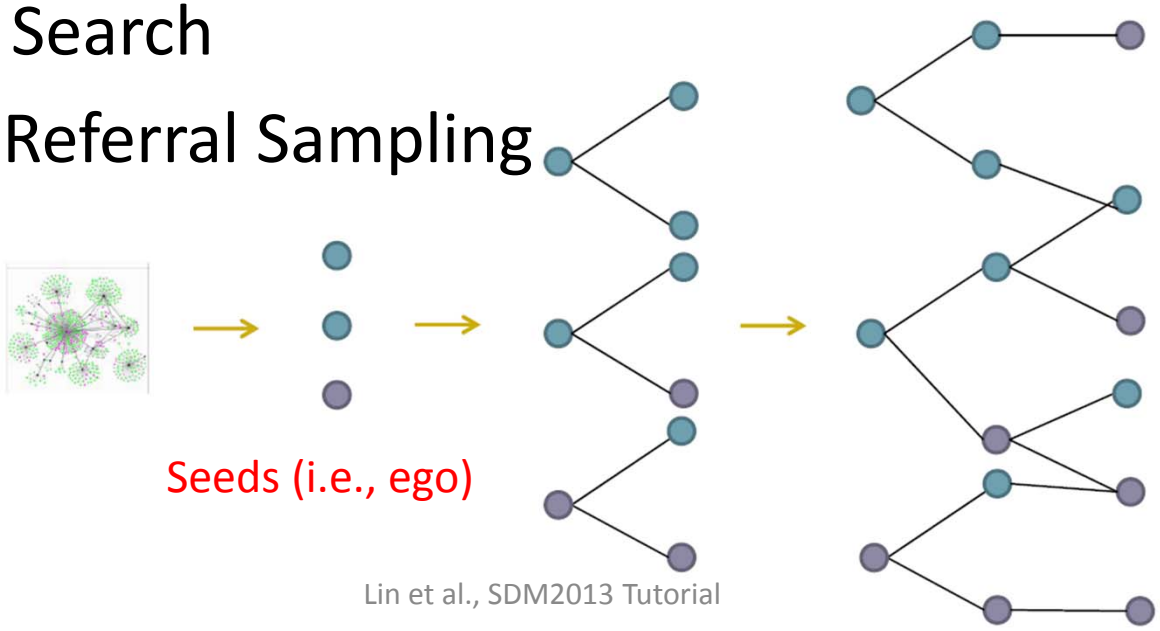
Evaluating the Sampling Quality

- How to measure the quality of the sampling algorithm?
- A sampling algorithm is effective if
  - The sampled social network can preserve certain network properties
  - Using the sampled network to perform an ultimate task (e.g. centrality analysis, link prediction, etc), one can produce similar results as if this task were performed on the fully observed network
  - It can produce a small sampled sub-network to achieve the above two goals

Sampling for **Homogeneous** Social Networks

Three Main Strategies

- Node Selection
- Edge Selection
- Sampling by Exploration
  - Random Walk
  - Graph Search
  - Chain-Referral Sampling

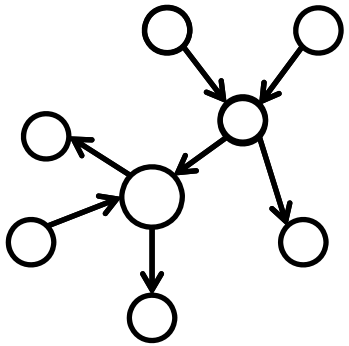


Node Selection

- Random Node Sampling
  - Uniformly select a set of nodes
- Degree-based Sampling [Adamic'01]
  - the probability of a node being selected is proportional to its degree (assuming known)
- PageRank-based Sampling [Leskovec'06]
  - the probability of a node being selected is proportional to its PageRank value (assuming known)

Edge Selection

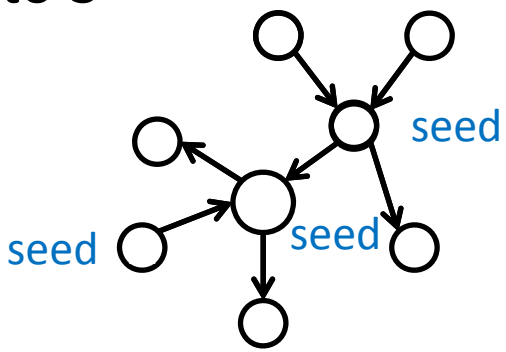
- Random Edge (RE) Sampling
  - Uniformly select edges at random, and then include the associated nodes
- Random Node-Edge (RNE) Sampling
  - Uniformly select a node, then uniformly select an edge incident to it



- Hybrid Sampling [Leskovec'06]
  - With probability *p* perform RE sampling, with probability *1-p* perform RNE sampling

Edge Selection (cont.)

- Induced Edge Sampling [Ahmed'12]
  - Step 1: Uniformly select edges (and consequently nodes) for several rounds
  - Step 2: Add edges that exist between sampled nodes
- Frontier Sampling [Ribeiro'10]
  - Step 0: Randomly select a set of nodes L as seeds
  - Step 1: Select a seed u from L using degree-based sampling
  - Step 2: Select an edge of u, (u, v), uniformly
  - Step 3: Replace u by v in L and add (u, v) to the sequence of sampled edges
  - * Repeat Step 1 to 3



Seed={A,B}

Degree(B)>Degree(A)

Randomly pick (B,C) into the sampled sequence

Replace B by C as a new seed

Sampling by Exploration

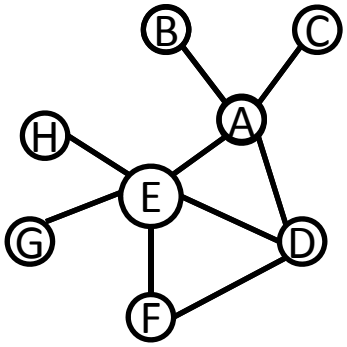
- Random Walk [Gjoka'10]
  - The next-hop node is chosen uniformly among the neighbors of the current node
- Random Walk with Restart [Leskovec'06]
  - Uniformly select a random node and perform a random walk with restarts
- Random Jump [Ribeiro'10]
  - Same as random walk but with a probability p we jump to any node in the network
- Forest Fire [Leskovec'06]
  - Choose a node *u* uniformly
  - Generate a random number *z* and select *z* out links of *u* that are not yet visited
  - Apply this step recursively for all newly added nodes

Sampling by Exploration (cont.)

- Ego-Centric Exploration (ECE) Sampling
  - Similar to random walk, but each neighbor has p probability to be selected
  - Multiple ECE (starting with multiple seeds)
- Depth-First / Breadth-First Search [Krishnamurthy'05]
  - Keep visiting neighbors of *earliest* / *most recently* visited nodes
- Sample Edge Count [Maiya'11]
  - Move to neighbor with the highest degree, and keep going
- Expansion Sampling [Maiya'11]
  - Construct a sample with the maximal expansion. Select the neighbor v based on $argmax_{v \in N(S)} |N(\{v\}) - (N(S) \cup S)|$

S: the set of sampled nodes, N(S): the 1st neighbor set of S

Example: Expansion Sampling

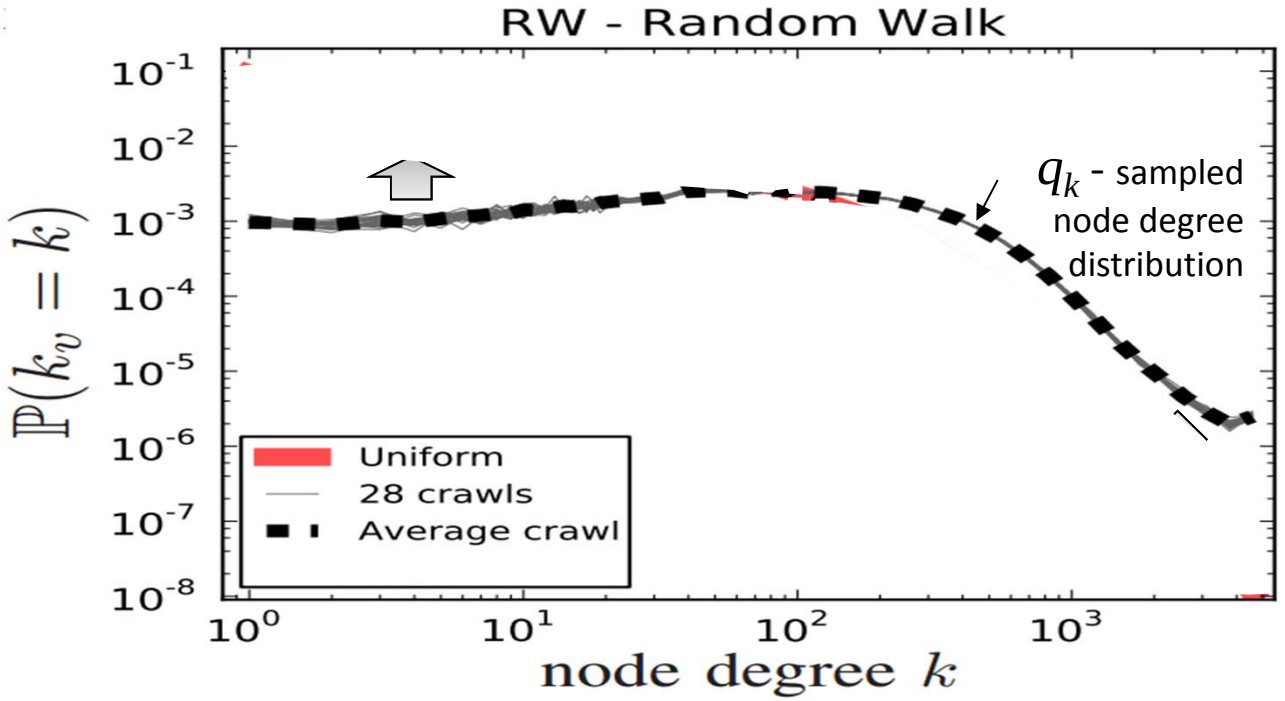


|N({A})|=4

|N({E}) – N({A}) U{A}|=|{F,G,H}|=3

|N({D}) – N({A}) U{A}|=|{F}|=1

Drawback of Random Walk: Degree Bias!



- Real average node degree ~ 94, Sampled average node degree ~ 338
- Solution: modify the transition probability :

$$P_{v,w} = \begin{cases} \frac{1}{k_v} * \min(1, \frac{k_v}{k_w}) & \text{If w is a neighbor of v} \\ 1 - \sum_{y \neq v} P_{v,y} & \text{If w = v} \\ 0 & \text{otherwise} \end{cases}$$

Metropolis Graph Sampling [Hubler'08]

- Step 1: Initially pick one subgraph sample S with n' nodes randomly
- Step 2: Iterate the following steps until convergence
  - 2.1: Remove one node from S
  - 2.2: Randomly add a new node to S → S'
  - 2.3: Compute the likelihood ratio $a = \frac{\rho^*(S')}{\rho^*(S)}$

    $if\ a \geq 1: accept\ replacement: S := S'$

    $if\ a < 1: accept\ replacement: S := S'$ with probability a

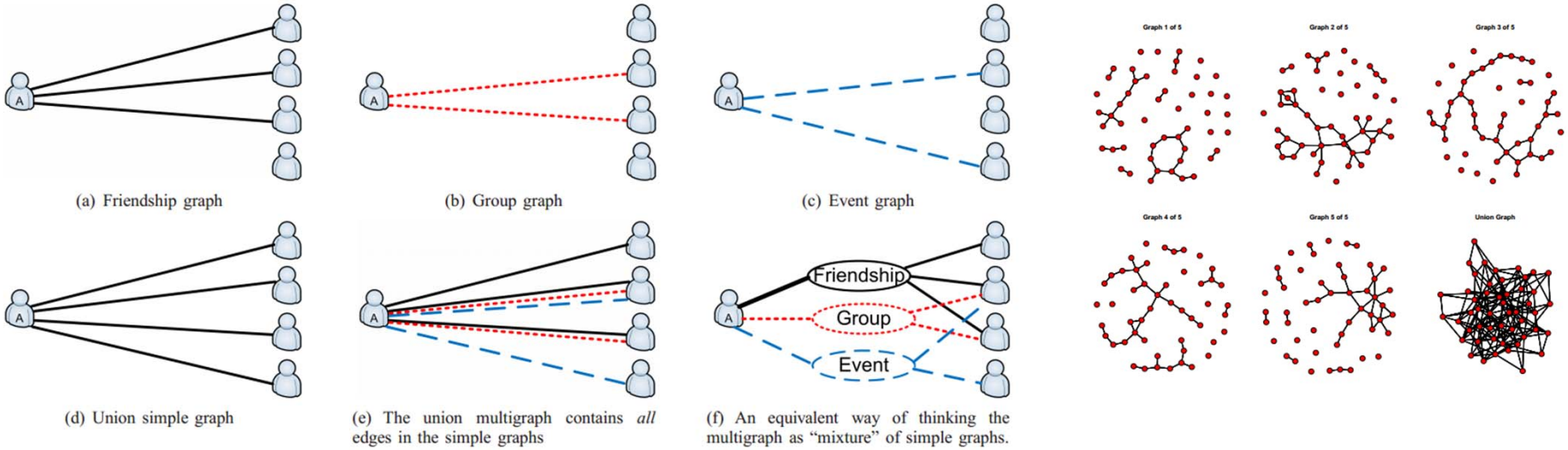
    $reject\ replacement: S := S'$ with probability $1 - a$
  - $\rho^*(S)$ measures the similarity of a certain property between the sample S and the original network G
    - Be derived approximately using Simulated Annealing

Sampling for Heterogeneous Social Networks

Sampling on Heterogeneous Social Networks

- Heterogeneous Social Networks (HSN)
  - A graph G=<V, E> has n nodes (v_1,v_2, …, v_n), m directed edges (e_1, …, e_m) and k different types
  - Each node/edge belongs to a type
    - Given a finite set L = {L_1, …, L_k} denoting k types
- Sampling methods for HSN
  - Multi-graph sampling [Gjoka'10]
  - Type-distribution preserving sampling (Li' 11)
  - Relational-profile preserving sampling (Yang'13)

Multigraph Sampling



(a) Friendship graph (b) Group graph (c) Event graph

(d) Union simple graph (e) The union multigraph contains *all* edges in the simple graphs (f) An equivalent way of thinking the multigraph as "mixture" of simple graphs.

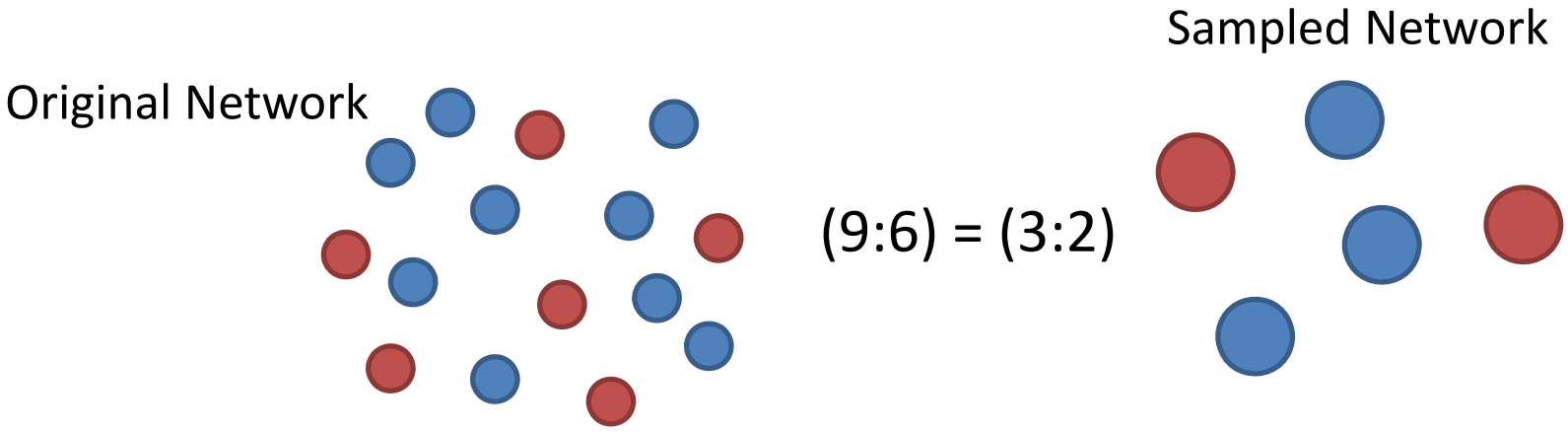
- Random walk sampling on the **union multiple graph** to avoid stopping on the disconnected graph.

Sampling Heterogeneous Social Networks

- Sampling methods for HSN
  - Multi-graph sampling [Gjoka'10]
  - Type-distribution preserving sampling (Li' 11)
  - Relational-profile preserving sampling (Yang'13)

Node Type Distribution Preserving Sampling

- Given a graph G and a sampled subgraph G_S
- The node type distribution of G_S is expected to be the same as G, i.e., $d(Dist(Gs), Dist(G)) = 0$
  - d() denotes the difference between two distributions

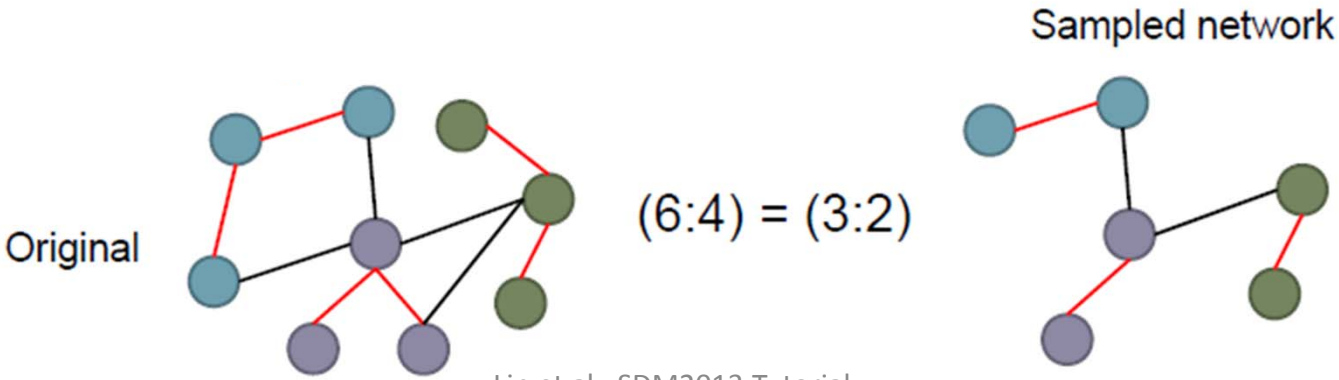


Connection-type Preserving Sampling

- Heterogeneous Connection
  - For an edge $E[v_i,v_j]$
  - **Intra**-connection edge: $Type(v_i) = Type(v_j)$
  - **Inter**-connection edge: $Type(v_i) != Type(v_j)$
- **Intra**-Relationship preserving
  - The ratio of the intra-connection should be preserved, that is:

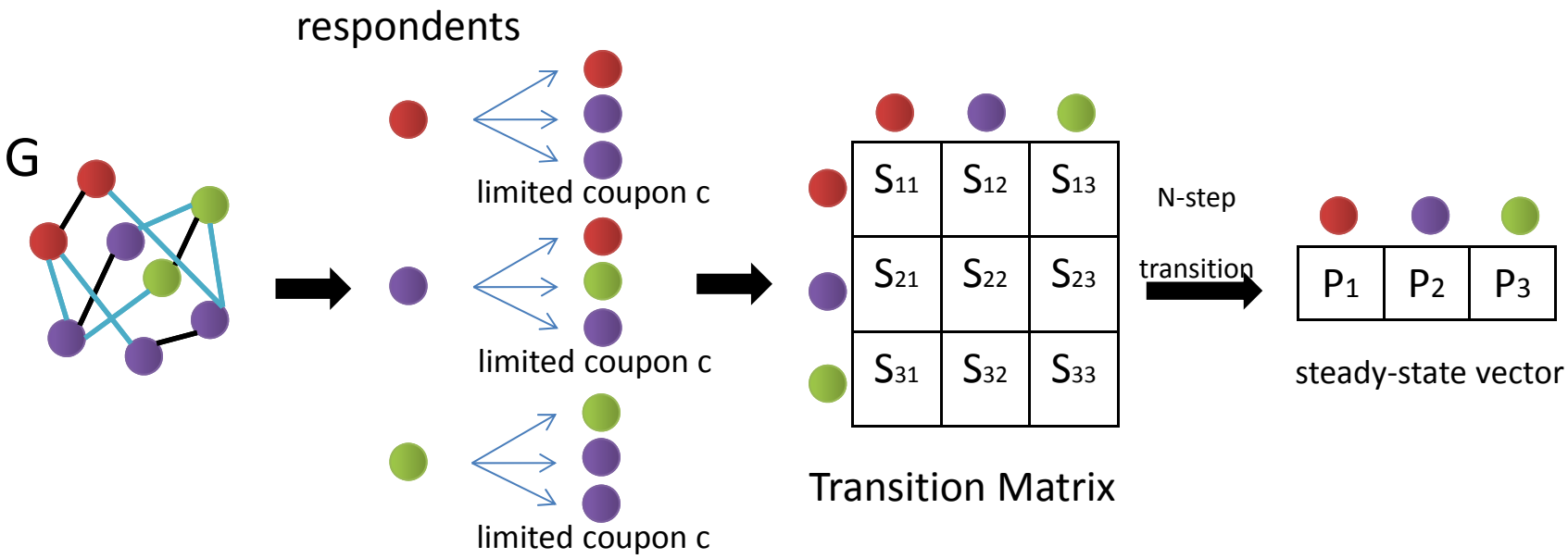
$$d(IR(G_S),IR(G)) = 0$$

  - If the intra-relationship is preserved, the **inter**-relationship is also preserved



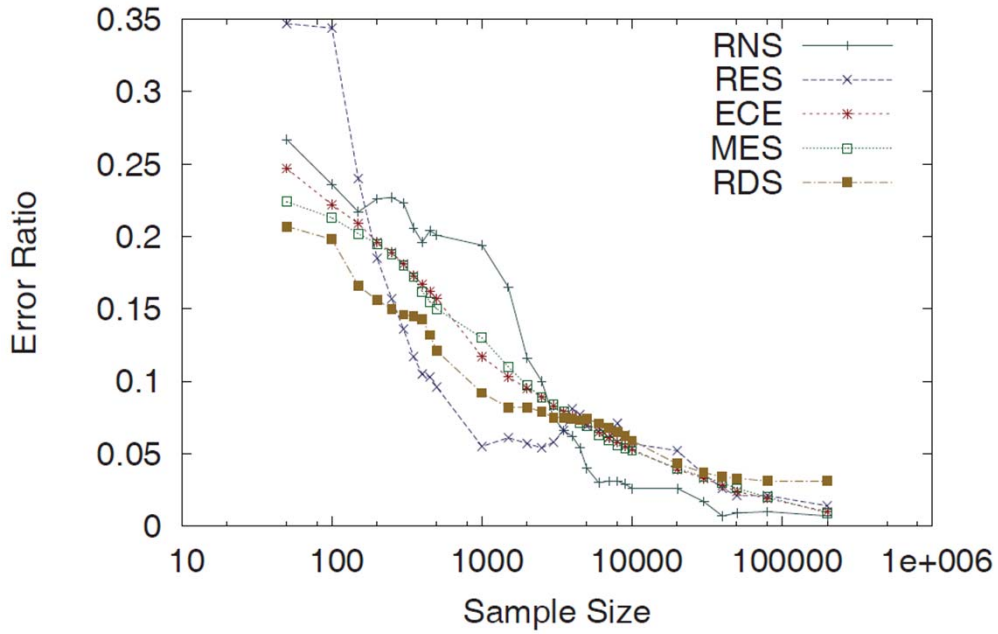
Respondent-driven Sampling

- First proposed in social science[Heck'99] to solve the hidden population in surveying.
- Two Main Phases:
  Snowball sampling ➔ Finding steady-state in the transition matrix

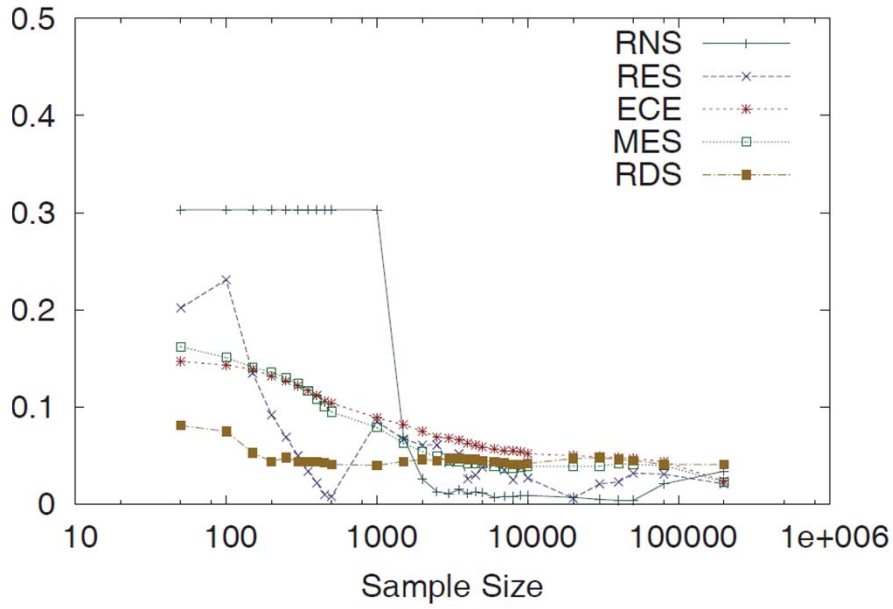


Comparing Different Sampling algorithms

Similarity of node type-distribution



Similarity of Intra-link distribution



- Respondent-driven Sampling does a good job with small node size, but saturate to mediocre afterwards
- Random node sampling performs poorly in the beginning, but reaches the best results after sufficient amount of nodes are sampled.

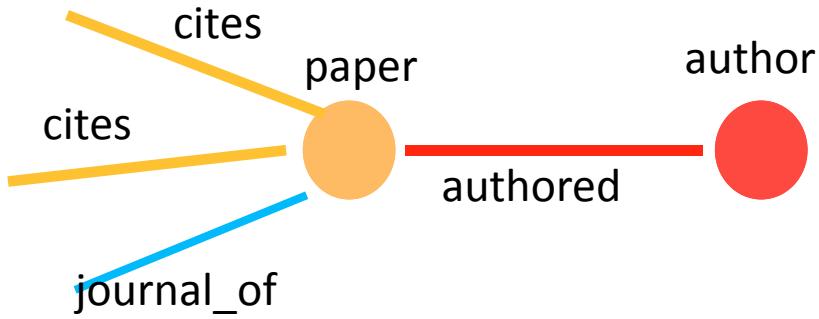
Heterogeneous Social Networks

- Sampling methods for HSN
  - Multi-graph sampling [Gjoka'10]
  - Type-distribution preserving sampling (Li' 11)
  - Relational-profile preserving sampling (Yang'13)

Relational Profile Preserving Sampling

- Node-type/intra-type preservation considers the **semantics** of nodes, but not the **structure** of the networks
- Homogeneous network sampling considers the **structure** but not the **semantics** of the networks
- Propose the **Relational Profile** to consider semantics and structure all together
  - Capture the dependency between each Node Type(NT) and Edge Type(ET) of a **directed** Heterogeneous Network
  - Consists of 4 Relational Matrices
    - Conditional probabilities $P(T_j|T_i)$ (e.g. P(LT=cites|NT=paper))
    - Node to node, node to edge, edge to node, edge to edge

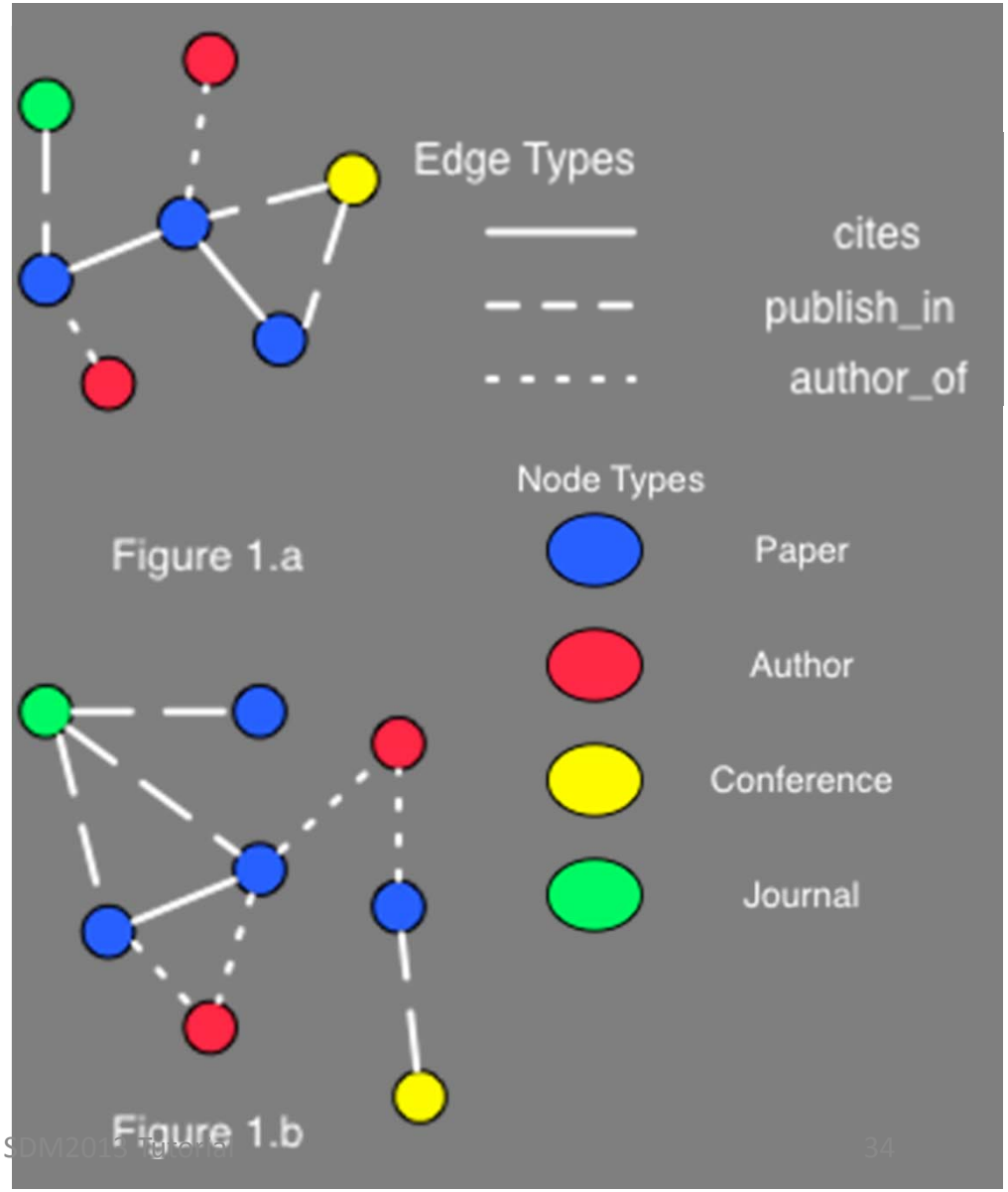
	NT	ET
NT	Transition Matrix	Transition Matrix
ET	Transition Matrix	Transition Matrix



Example of Relational Profile (RP)

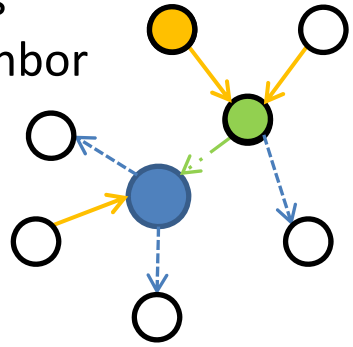
	P	A	C	J	c	p	a
P	0.44	0.22	0.22	0.11	0.44	0.33	0.22
A	1						1
C	1					1	
J	1					1	
c	1				0.22	0.44	0.33
p	0.5		0.33	0.17	0.66		0.33
a	0.5	0.5			0.6	0.4	

	P	A	C	J	c	p	a
P	0.182	0.364	0.091	0.273	0.182	0.364	0.364
A	1						1
C	1					1	
J	1					1	
c	1					0.5	0.5
p	0.5		0.125	0.375	0.17	0.5	0.33
a	0.5	0.5			0.22	0.33	0.44



Challenge: How to approximate RP when the true RP is unknown

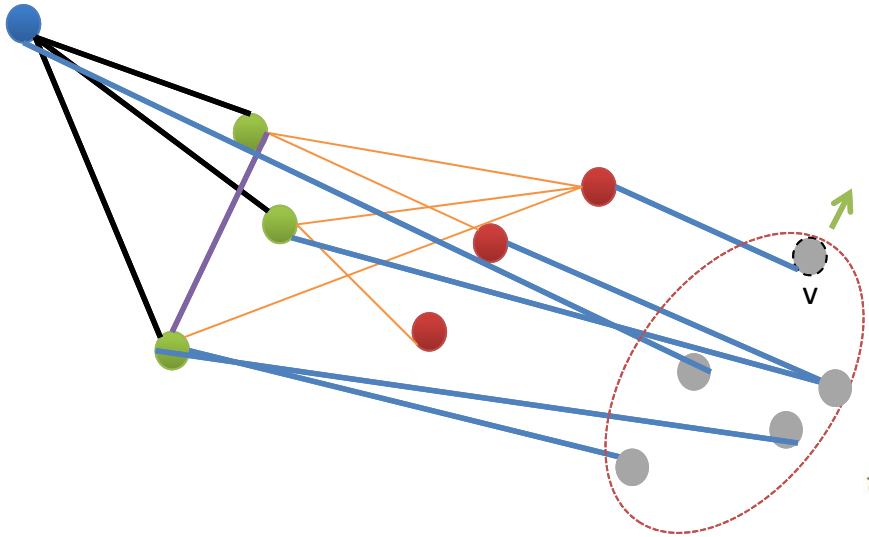
- We propose Exploration by Expectation Sampling
- Aim to preserve the unknown relational profile while adding new sample nodes
  1. Randomly choose a starting node and the corresponding edges
  2. Based on current RP, **select** a next node from all 1 degree neighbor
  3. Add the new node and all its edges
  4. Update RP of the sub-sampled graph
  5. Repeat step 3, 4 & 5 until the converge of RP



- Which node should be **selected**?
  - Select the node whose inclusion can potentially lead to the largest change to the existing RP
    - Use the partially observed ***RP*** to generate the 'expected amount of change' of each candidate node as its score
    - Weighted sampling based on the score

Relational Profile Sampling (RPS)

Idea: Sample to increase the diversity

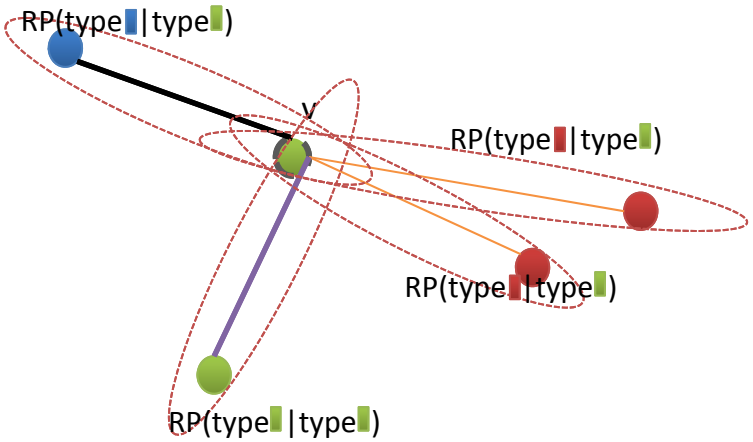


$D(v, G_s)$ = estimated change of RP given sampling v on the current graph G_s

$=E[\Delta_P(G_s, G_s+v)|G_s]$, where $\Delta_P = RMSE_{RP}$

which can be calculated as

$$\sum_{t \in NT} P(type(v) = t| \quad G_s \qquad)\Delta RP(G_s, G_s + v)$$



Exploiting the existing RP, P(type(v)=t|G_s) can be obtained using the observed types of v's neighbors

$$\prod_{i \in N(v)} \left(\frac{RP(type(i)|type(v) = t)P(type(v) = t)}{Z}\right)$$

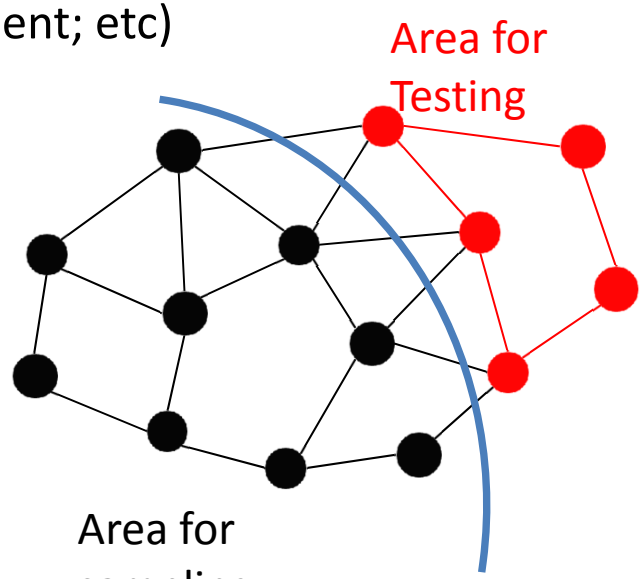
P(type|type) can be obtained from the existing RP

Goal: maximize expected property (Relational Profile distribution) change

Evaluation

- **Evaluation I** (**Property Preservation**): see how well the sampled network approximates two properties of the full network
- **Evaluation II** (**Prediction**): training a prediction model using the sampled network to infer out-of-sampled network status:
  - **Node Type Prediction**: Predict the type of unseen nodes in the network using a sub-sampled network
  - **Missing Relations Prediction**: Recover/predict the missing links
  - Features:
    - f_{deg} = (in/out deg; avg in/out deg of neighbors)
    - f_{topo} = (Common Neighbors; Jaccard's Coefficient; etc)
    - $f_{nt} = P(type(v)|G_c) = \frac{\#type(v)=t \forall v \in N(n)}{|N(n)|}$
    - $f_{RPnode} = \prod_{i \in N(n)} \frac{1}{Z} RP(type(i)|type(v)=t) P(type(v)=t)$
    - $f_{RPpath} = \sum_{p \in Path(s,t)} \prod_{(p_1,p_2) \in p} P(type(p_2)|type(p_1))$

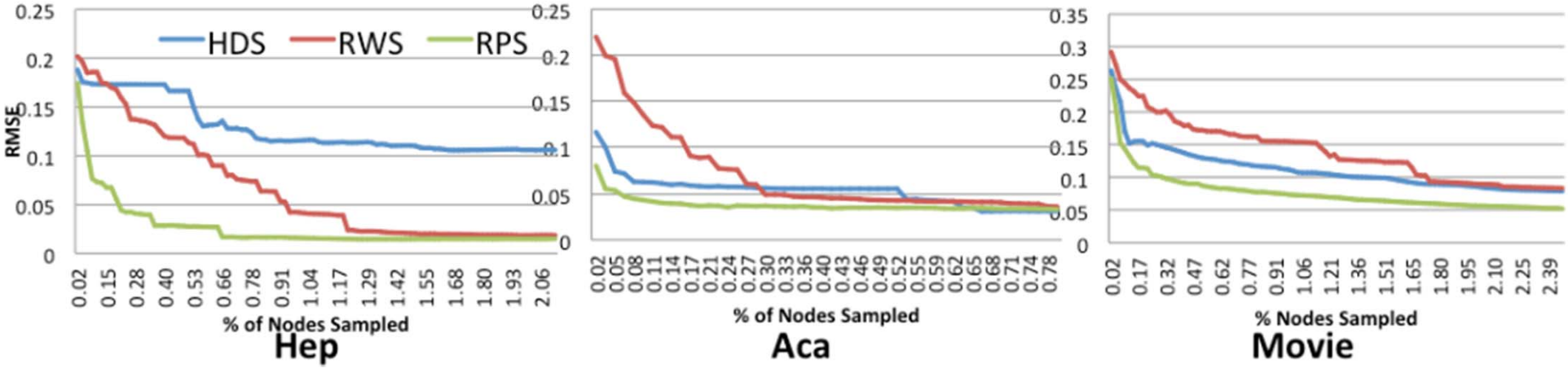
- Datasets: 3 real-life large scale HSN
- Baselines:
  - Random Walk Sampling (RW)
  - Degree-based sampling (HDS)



Experiments (Property Preservation)

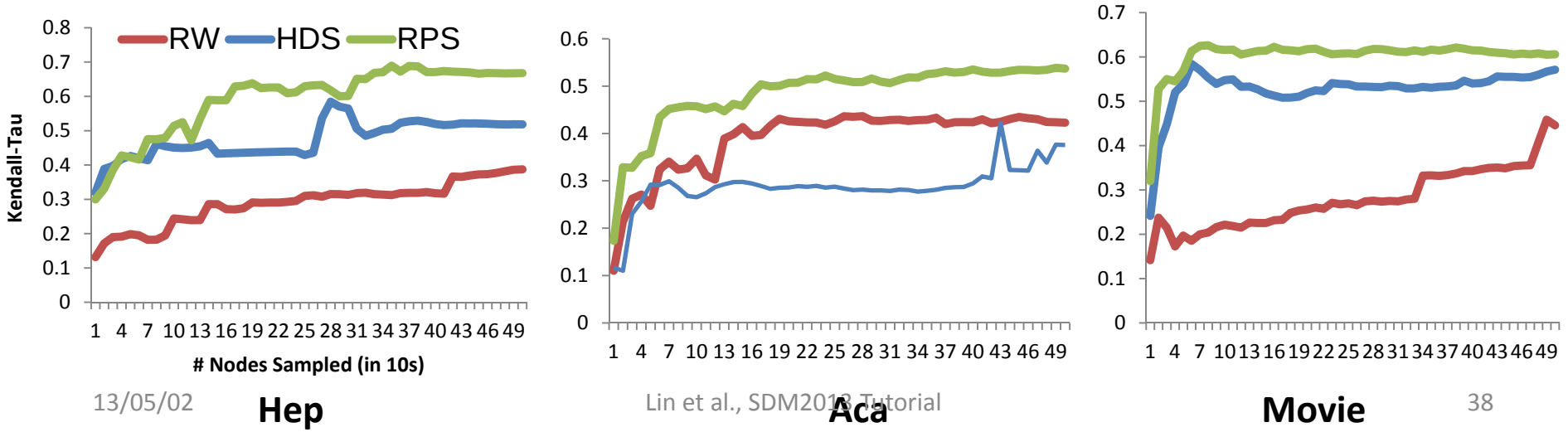
- RMSE (for RP)

Type dependency preservation



- Weighted PageRank

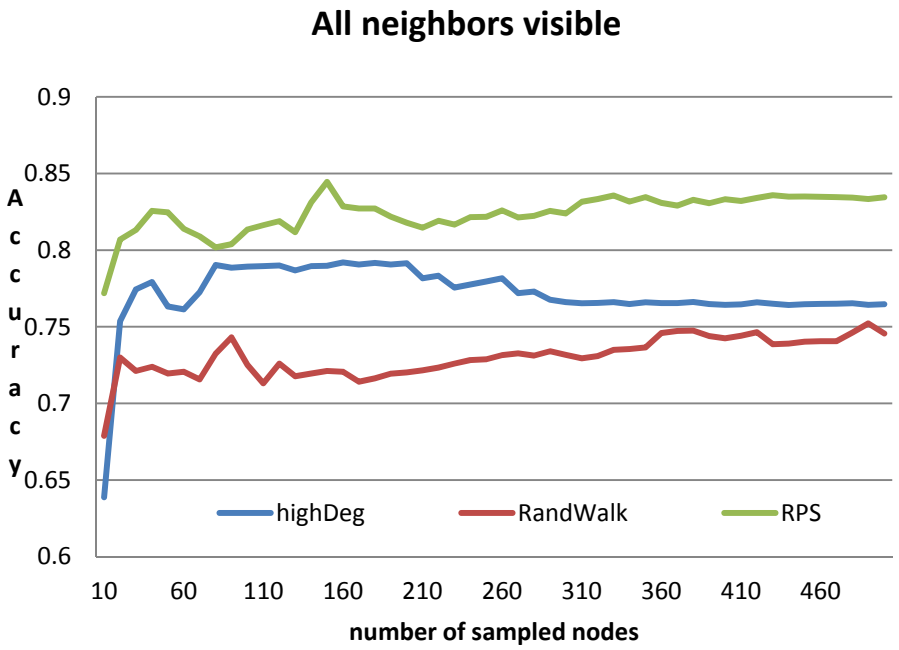
Preserving relative node weights propagated throughout entire network



Experiments (Prediction)

- We show Academic Network for brevity.

Node Type Prediction



Missing Relation Prediction

Remarks: These evaluations provide a general idea how one can evaluate a SN sampling algorithm

Task-driven Network Sampling

- Sampling Community Structure [Maiya'10][Satuluri'11]
- Sampling Network Backbone for Influence Maximization [Mathioudakis'11]
- Sampling High Centrality Individuals [Maiya'10]
- Sampling Personalized PageRank Values [Vattani'11]
- Sampling Network for Link/Label Prediction [Ahmed'12]

Take Home Points

- **Why sampling a social network?**
  - **the full network (e.g. Facebook) cannot be fully observed**
  - **crawling can be costly in terms of resource and time consumption (therefore a smart sampling strategy is needed)**

	Homogeneous SN	Heterogeneous SN
Node and Edge Selection	[Leskovec'06] [Adamic'01] [Ahmed'12][Ribeiro'10]	[Kurant'12]
Sampling by Exploration	[Krishnamurthy'05] [Leskovec'06][Hubler'08] [Gjoka'10][Ribeiro'10] [Maiya'11][Kurant'11]	[Gjoka'11][Li'11][Kurant'12] [Yang'13]
Task-driven Sampling	[Maiya'10][Satuluri'11][Mathioudakis'11] [Vattani'11][Ahmed'12]	

The 2nd part of this tutorial:

Social Network Summarization

Goals of Social Network Summarization

- Find a condensed representation of a given social network to
  - produce a succinct overview of the social network,
  - save the storage,
  - enable efficient mining / query processing

Beyond Graph Summarization

- To summarize not only the structure or topology information such as:
  - Neighbor set / adjacency
  - Reachability
  - Connectivity
- but also the semantic information such as:
    - Attributes of an entity and a relationship.
    - Relationships of entity-entity, entity-community, community-community.

Issues for Summarization

- Purpose
  - Are there certain properties to preserve? What types of queries / mining tasks are the summaries for?
- Precision of the summary
  - Lossless: can reconstruct the exact original social networks
  - Lossy: cannot fully recover, usually for a better compression ratio
- Evaluation
  - Space saving: Reduction of # node/edge, total data size in bytes, bit per edges, etc.
  - Quality: reconstruction errors, interestingness, query errors (degree, centrality, connectivity), etc.
  - Efficiency: time for summarization and time savings of querying or mining on the summaries.

Main Approaches for Summarization

- Aggregation based
  - Creating a summary graph with supernodes and superedges.
  - For efficient storage, analysis, and visualization.
- Abstraction based
  - Extracting a subgraph given certain criteria for abstraction and various visualizations.
- Compression based
  - Encoding the network in a space-efficient way based on the structure information.
- Application-oriented
  - Designing specifically for different kinds of applications.

Web/Graph -> Social Network

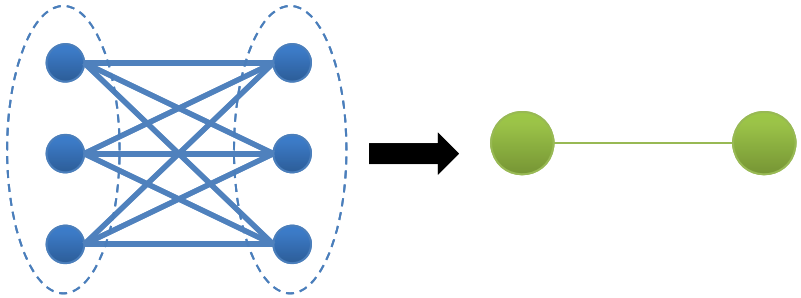
- from the homogeneous to the heterogeneous structure

Main Approaches for Summarization

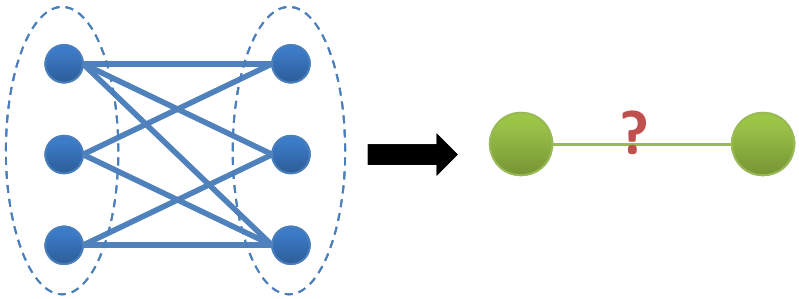
- Aggregation based
- Abstraction based
- Compression based
- Application-oriented

Aggregation based on Node/Link Structures

- The basic idea:
  - Merge nodes with similar neighbors into a supernode.
  - Add a superedge between two supernodes conditionally.
  - E.g., **complete bipartite graph**



- What if a subgraph is not complete?
  - E.g.,



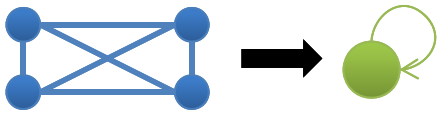
Supernode graph + edge corrections!

Aggregation based on Node/Link Structures (cont'd)

- S-node representation for Web graphs [Raghavan'03]:
  - Partition web pages
    - URL split (domain) + Cluster split (adjacency list of out-links)
  - Supernode graph: a node represents a partition and a link between two partitions if there exists any link between two pages, one from each partition.
  - Positive/Negative superedge graphs: used to annotate the actual linkage between web pages.
  - Lossless representation.

Aggregation based on Node/Link Structures (cont'd)

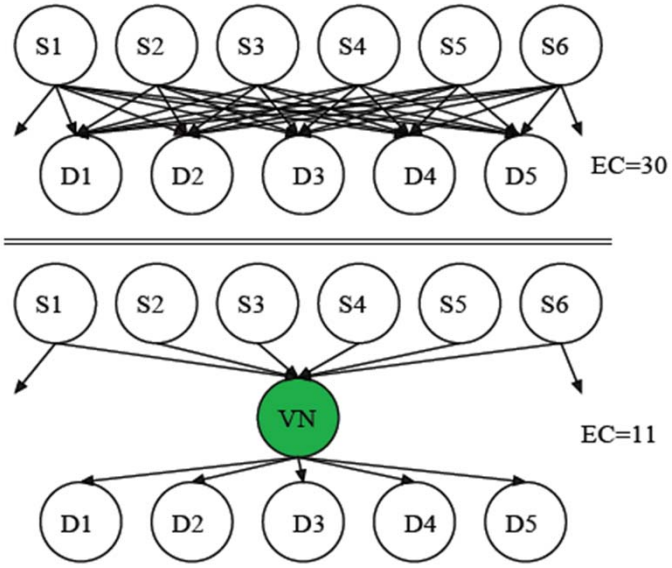
- A two-part representation $R(S,C)$ is proposed [Navlakha'08]:
  - Graph summary S: an aggregated graph.
    - Merge nodes with more common neighbors.
    - A link is added between two supernodes if the nodes in one supernode are ***densely*** connected to those in the other.
    - Allow supernodes to have a self-edge.



  - Edge corrections C: to be used while recovering the original graph.
  - Both lossless/lossy methods are proposed.

Aggregations of Links by Frequent Patterns

- Leverage pattern mining to compress the Web graph, which supports community discovery and random access [Buehrer'08].
- Two phases:
  - Clustering phase: nodes with similar out-links are grouped together.
  - Mining phase (for each cluster): mine virtual nodes to aggregate edges
- A lossless and more compact structure

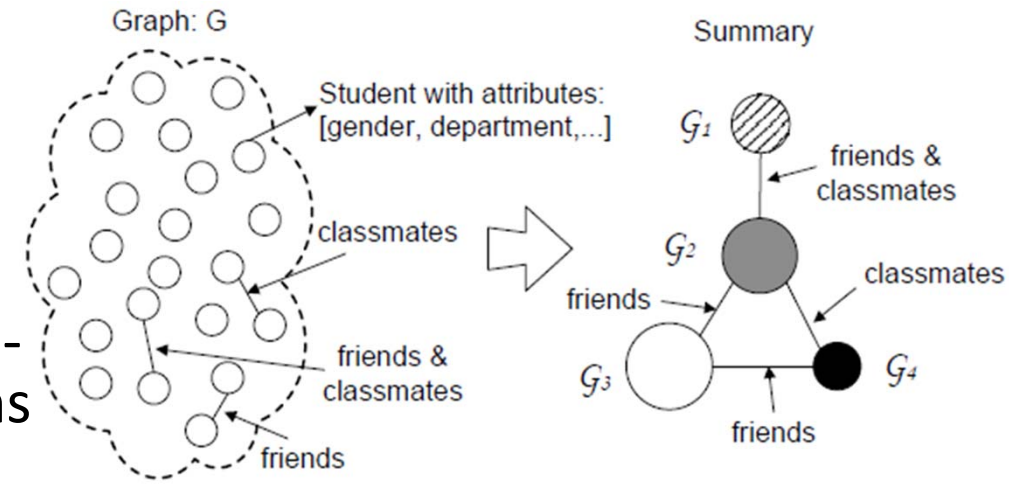


Vertex Id	Outlink List
23	1,2,3,5,6,10,12,15
55	1,2,3,5
102	1,2,3,20
204	1,7,8,9
13	1,2,3,8
64	1,2,3,5,6,10,12,15
43	1,2,3,5,6,10,22,31
431	1,2,3,5,6,10,21,31,67

Aggregation by Node Attributes and Relations

- SNAP [Tian'08]: Summarizing by Grouping Nodes on Attributes and Pairwise Relationships.

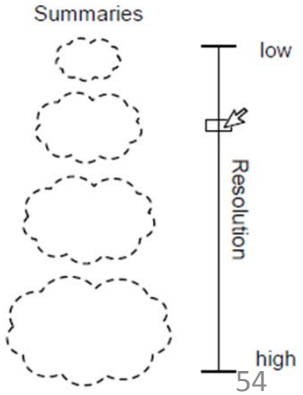
- User can specify a node-attribute set and a relation-type set, the system returns an (A,R)-compatible grouping.
  - E.g., A={gender, department}, R={friends, classmates}
- Lossless.



Each student in group G_1 has at least a friend and a classmate in group G_2.

Aggregation by Node Attributes and Relations (cont'd)

- *k*-SNAP [Tian'08]: relaxes the homogeneity requirement for the relationships and allows users to control (drill-down, roll-up) the sizes of the summaries.
  - *k* is the user-specified number of grouping nodes.
  - Not requiring that every node participates in a group relationship.
  - Lossy



Aggregation by Node Attributes and Relations (cont'd)

- [Zhang'10] Improves two limitations of ***k*-SNAP** in practice
  - Limitation 1: Only handles categorical node attribute
    - Sol: Provide cutoffs to categorize numerical attributes
  - Limitation 2: The search space is too large for manually identifying interesting summaries
    - Sol: An **interestingness measure** (diversity, coverage, conciseness) is introduced to evaluate the interestingness of a summary

Main Approaches for Summarization

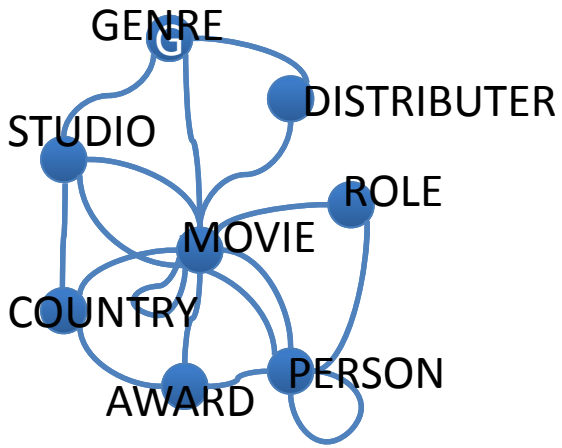
- Aggregation based
- Abstraction based
- Compression based
- Application-oriented

Visual Analysis of Large Heterogeneous Social Networks

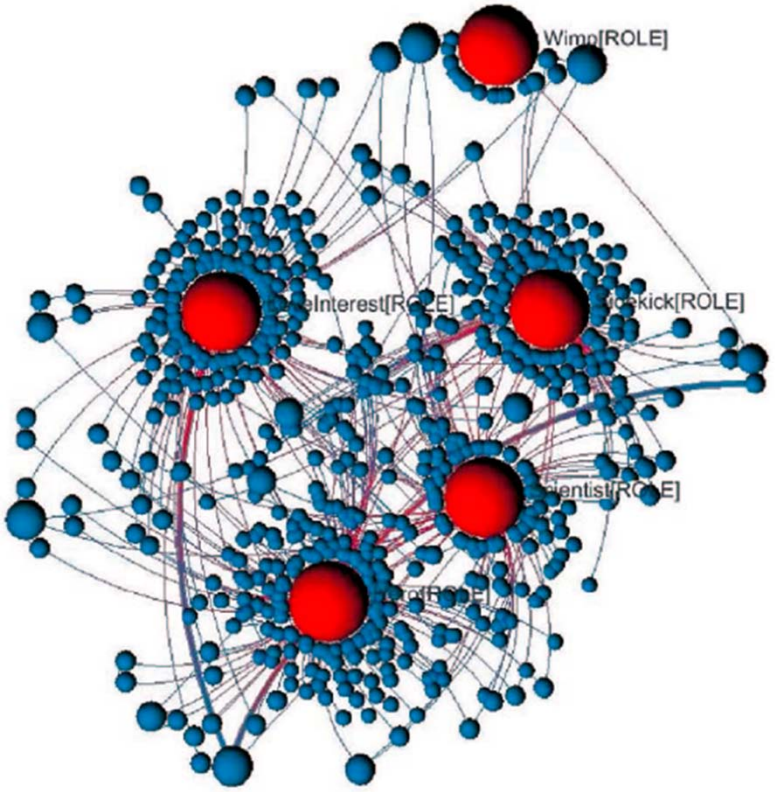
- **OntoVis** [Shen'06] is a visual analysis tool for heterogeneous social network based on the given **ontology** graph
  - **Semantic abstraction**: generate an induced graph of node types selected by users
  - **Structural abstraction**: remove one-degree nodes and duplicate paths for reducing visual complexity
  - **Importance filtering**: using statistics such as node degree, dispersion and disparity per type to determine the important node types for emphasizing.

Visual Analysis of Large Heterogeneous Social Networks (cont'd)

[Shen'06]



Ontology graph of the movie Dataset from the UCI KDD Archive[1]

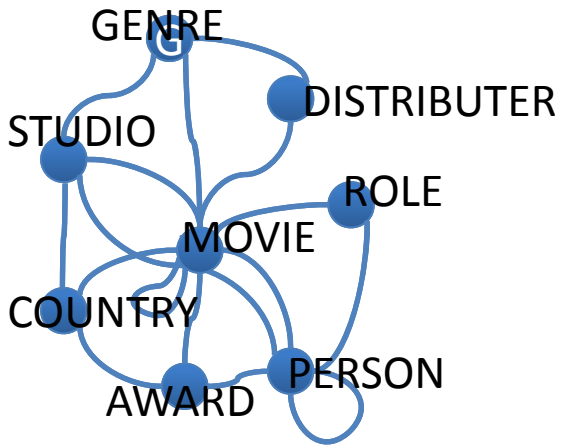


Semantic abstraction on "role-actor" relationships
Red nodes: role
Blue nodes: actor

1. http://kdd.ics.uci.edu/databases/movies/movies.data.html

Visual Analysis of Large Heterogeneous Social Networks (cont'd)

[Shen'06]



Ontology graph of the movie Dataset from the UCI KDD Archive[1]

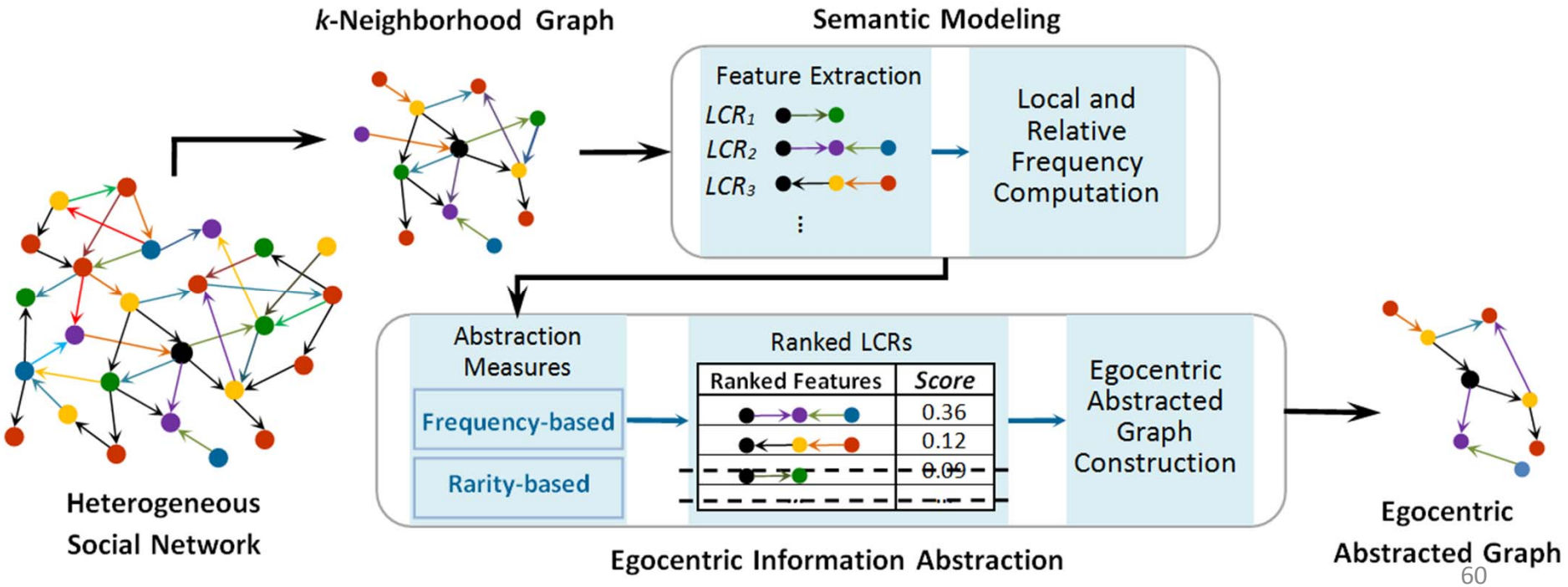
Importance filtering on "node type disparity"
Node size: disparity of connected types
on edge: frequencies of links between two types

1. http://kdd.ics.uci.edu/databases/movies/movies.data.html

Egocentric Abstraction on Heterogeneous Social Networks

[Li'09]

- Construct the abstracted graph of an **ego** node for a heterogeneous social network
  - Identify each unique *k*-step linear combination of relations as a feature
  - Counting the frequency of each unique feature
  - Several criteria are introduced to decide which features are important for the ego
    - E.g., abstraction by showing only rare/frequent features.



Main Approaches for Summarization

- Aggregation based
- Abstraction based
- Compression based
- Application-oriented

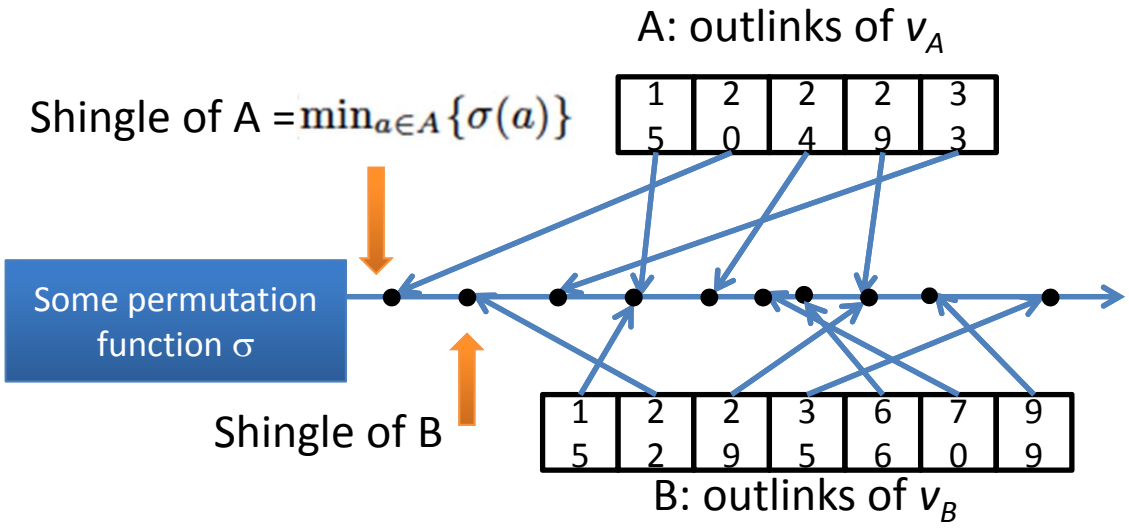
Ordering-based Compression

- URLs of web pages have two features:
  - **Similarity**: Usually the source and the target of a link are close to each other (in lexicographical order).
  - **locality**: pages close to each other (in lexicographical order) tend to have many common successors, i.e., many navigational links are the same for pages in the same local cluster and with the same host.
- By leveraging the **lexicographical order**, the BV scheme [Boldi'04] needs only 3bits per edge to encode the Web graphs.

- However, nodes in social networks have no natural orders.

Ordering for Nodes in Social Networks

- The **shingle ordering** based on Jaccard coefficient to find locality in social networks [Chierichetti'09].



$$M_\sigma(A) = \sigma^{-1}(\min_{a \in A}\{\sigma(a)\})$$

$$\Pr[M_\sigma(A) = M_\sigma(B)] = \frac{|A \cap B|}{|A \cup B|}$$

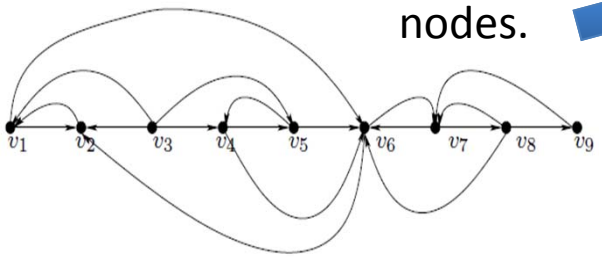
$$= J(A, B).$$

- If two nodes share a lot of common out-neighbors, with high probability they will be close to each other in a shingle-based ordering.

Neighbor Query Friendly Compression of Social Networks

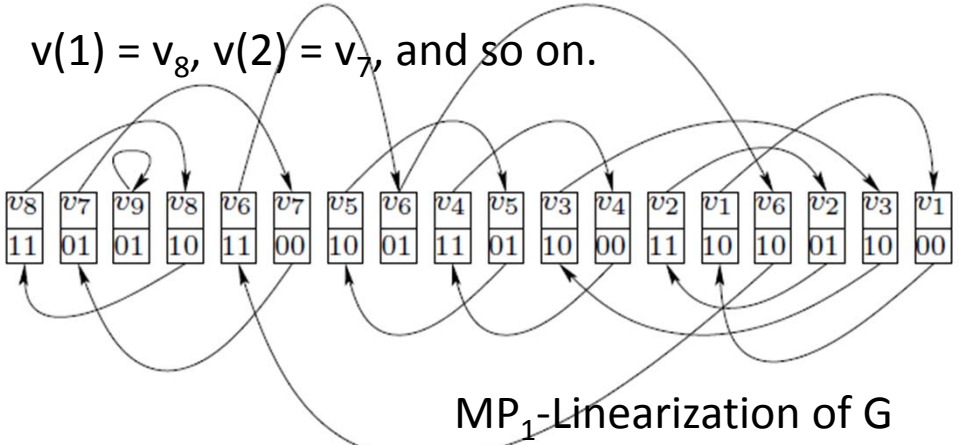
- A novel **Eulerian data structure** using **multi-position linearizations** (MP) of directed graphs is proposed to compress social networks while both **out/in neighbor queries** can be answered in **sublinear** time [Maserrat'10].

Find S, a cover that contains all nodes in G with S-distance = 1 by duplicating necessary nodes.



Original graph G

v(1) = v_8, v(2) = v_7, and so on.



MP_1-Linearization of G

(Min. distance among all pairs of (u,v) is 1)

- Local information: 2 bits to encode if (v(i-1), v(i)) and (v(i),v(i+1)) exists in E(G).
- Pointers: next appearance of v(i).

Other Graph/Network Compressions

- Community-based (hubs and spokes) Compression [Kang'10]
- MP-Linearization for for lossy compression to preserve communities in social networks [Maserrat'12]
- Mix clusterings and orders for Compressing Social Networks [Boldi'11]
- Encoding based on the newly defined **structural entropy** for Erdös-Rényi graphs [Choi'12].

Main Directions

- Aggregation based
- Abstraction
- Compression based
- Application-oriented

Application-Oriented Summarization

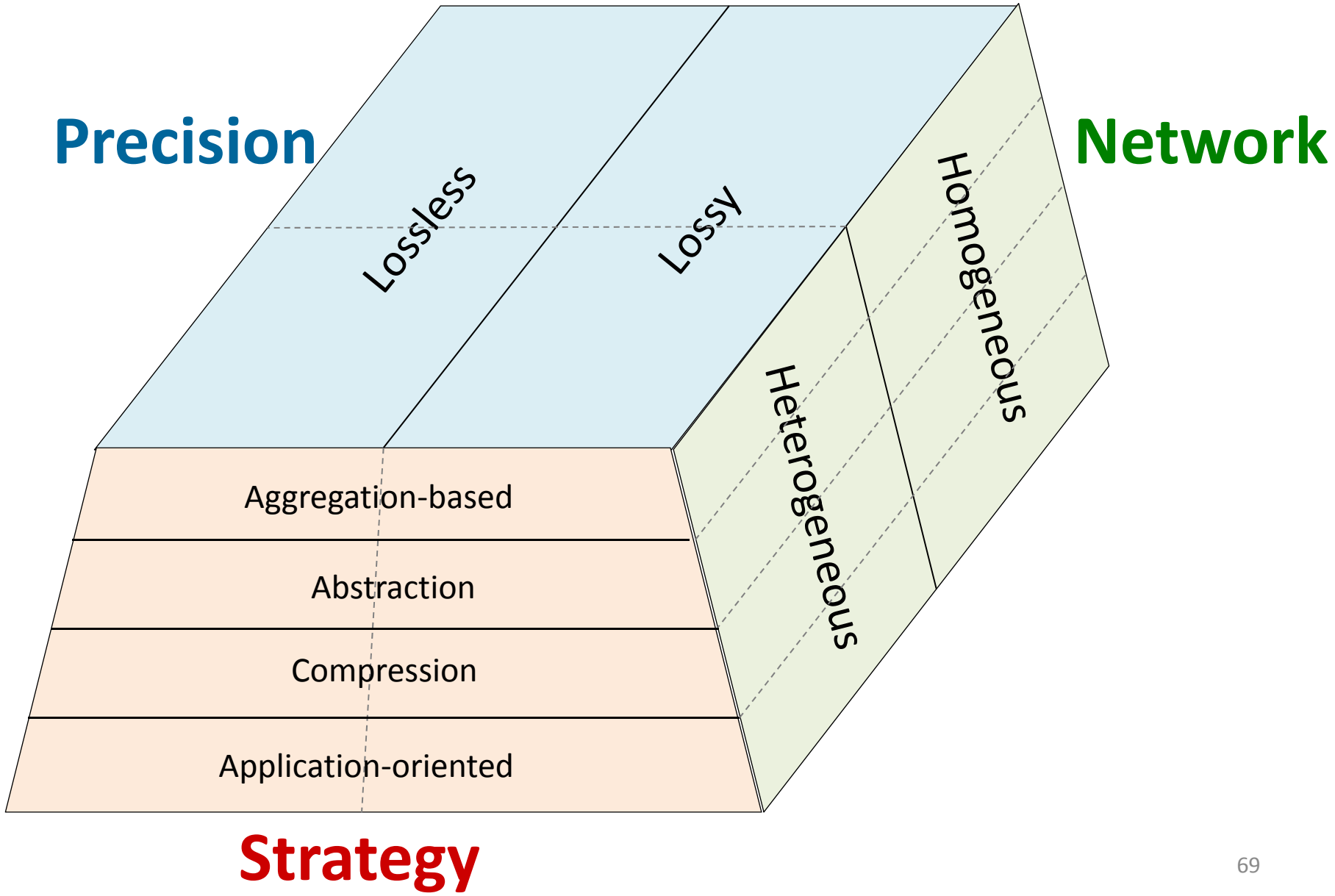
- Summarization for query-answering and pattern mining
  - Adjacency, degree, centrality [LeFevre'10]
  - Connectivity [Zhou'10][Toivonen'11]
  - Graph pattern mining/search [Chen'09][Kang'10][Fan'12]
- Graph management system [Kang'11]

and so on.

What We Have not Covered in this Tutorial Yet

- Comparisons of the performance among all these works in terms of
  - Efficiency
  - Space saving
  - Quality

Social Network Summarization Overview



Summarization Categories

	Homogeneous	Heterogeneous
Aggregation-based	[Raghavan'03][Navlakha'08] [Buehrer'08]	[Tian'08] [Zhang'10][Liu'11]
Abstraction		[Shen'06][Li'09]
Compression	[Chierichetti'09][Maserrat'10] [Maserrat'12] [Kang'10][Choi'12]	
Application-oriented	[Zhou'10][LeFevre'10] [Toivonen'11][Kang'11]	[Chen'09][Fan'12]

Summarization Strategies: Lossless / Lossy

Opportunities for Future Research

- Advanced techniques to sample/summarize more complex graph structures
  - E.g. location-based social networks, diffusion networks, dynamic social networks, social network with activity information, etc.
- Should we focus on task-driven sampling and summarization or do we need a general framework across tasks?
- Sampling/Summarization on noisy data
- Standard evaluation metrics and benchmark data are in high demand.
- And many others…

Final Remarks

- Sampling and summarization have immediate practical values in the big data era
  - Allow data miners to perform advanced mining tasks in large graphs
  - Achieve scalable storage and querying
  - Facilitate the development of real-world applications
- Existing works are rich, but by no means complete to handle every aspect of the problem.

Acknowledgements

- This tutorial is partially sponsored by National Science Council, National Taiwan University and Intel Corporation under Grants NSC101-2911-I-002-001, NSC101-2628-E-002-028-MY2 and NTU102R7501
- Special thanks to Shu-Ming Hsu @ Academia Sinica for his inputs

Reference – Homogeneous Sampling

- J. Leskovec and C. Faloutsos. Sampling from large graphs. In KDD 2006.
- A. S. Maiya and T. Y. Berger-Wolf. Benefits of bias: towards better characterization of network sampling. In KDD 2011.
- B. Ribeiro and D. Towsley. Estimating and sampling graphs with multidimensional random walks. In ACM SIGCOMM IMC 2010.
- M. Gjoka, M. Kurant, C. T. Butts, and A. Markopoulou. Walking in facebook: a case study of unbiased sampling of OSNs. In IEEE INFOCOM 2010.
- V. Krishnamurthy, M. Faloutsos, M. Chrobak, L. Lao, J.-H. Cui, and A. G. Percus. Reducing large internet topologies for faster simulations. In Networking, 2005.
- N. K. Ahmed, J. Neville, and R. Kompella. Network Sampling: From Static to Streaming Graphs. arXiv:1211.3412, 2012.
- C. Hubler, H.-P. Kriegel, K. M. Borgwardt, and Z. Ghahramani. Metropolis Algorithms for Representative Subgraph Sampling. In IEEE ICDM 2008.
- M. Kurant, M. Gjoka, C. T. Butts, and A. Markopoulou. Walking on a graph with a magnifying glass: stratified sampling via weighted random walks. SIGMETRICS Perform. Eval. Rev. 2011.

Reference – Heterogeneous Sampling

- M. Gjoka, C. T. Butts, M. Kurant, and A. Markopoulou. Multigraph Sampling of Online Social Networks. IEEE Journal on Selected Areas in Communications, 2011.
- M. Kurant, M. Gjoka, Y. Wang, Z. W. Almquist, C. T. Butts, and A. Markopoulou. Coarse-grained topology estimation via graph sampling. ACM WOSN 2012.
- J.-Y. Li and M.-Y. Yeh. On Sampling Type Distribution from Heterogeneous Social Networks. In PAKDD 2011.
- Cheng-Lun Yang, Perng-Hwa Kung, Chun-An Chen, Shou-De Lin. Semantically Sampling in Heterogeneous Social Networks in WWW 2013
- D. Heckathorn. Respondent-driven sampling: a new approach to the study of hidden populations. Social problems, 1997.

Reference – Task-driven Sampling

- A. S. Maiya and T. Y. Berger-Wolf. Sampling community structure. In WWW 2010.
- M. Mathioudakis, F. Bonchi, C. Castillo, A. Gionis, and A. Ukkonen. Sparsification of Influence Networks. In KDD 2011.
- A.S. Maiya and T.Y. Berger-Wolf. Online Sampling of High Centrality Individuals in Social Networks. In PAKDD 2010.
- V. Satuluri, S. Parthasarathy, and Y. Ruan. Local Graph Sparsification for Scalable Clustering. In SIGMOD 2011.
- A. Vattani, D. Chakrabarti, and M. Gurevich. Preserving Personalized Pagerank in Subgraphs. In ICML 2011.
- N. K. Ahmed, J. Neville, and R. Kompella. Network Sampling Designs for Relational Classification. In AAAI ICWSM 2012.

References: Aggregation-based Summarization

- S. Navlakha, R. Rastogi, N. Shrivastava. Graph Summarization with Bounded Error. In Proc. of ACM SIGMOD International Conference on Management of Data (SIGMOD'08), 2008.
- Y. Tian, R. A. Hankins and J. M. Patel. Efficient Aggregation for Graph Summarization. In Proc. of ACM SIGMOD International Conference on Management of Data (SIGMOD'08), 2008.
- G. Buehrer and K. Chellapilla. A Scalable Pattern Mining Approach to Web Graph Compression with Communities. In Proceedings of the 2008 International Conference on Web Search and Data Mining (WSDM'08), pages 95–106, 2008.
- N. Zhang, Y. Tian, and J. M. Patel. Discovery-driven Graph Summarization. In Proc. of IEEE International Conference on Data Engineering (ICDE'10), 2010.

References: Abstraction-based Summarization

- Z. Shen, K. L. Ma and T. Eliassi-Rad. Visual Analysis of Large Heterogeneous Social Networks by Semantic and Structural Abstraction. IEEE Transactions on Visualization and Computer Graphics, 12(6), 1427–1439, 2006.
- C.-T. Li and S.-D. Lin. Egocentric Information Abstraction for Heterogeneous Social Networks, In Proc. of International Conference on Advances in Social Network Analysis and Mining (ASONAM'09), 2009.

References: Compression-based Summarization

- P. Boldi and S. Vigna. The Webgraph Framework I: Compression Techniques. In the 13th international conference on World Wide Web (WWW'04), pages 595–602, 2004.
- F. Chierichetti, R. Kumar, S. Lattanzi, M. Mitzenmacher, A. Panconesi, and P. Raghavan. On Compressing Social Networks, In Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'09), 2009.
- H. Maserrat and J. Pei. Neighbor Query Friendly Compression of Social Networks, In Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'10), 2010.
- H. Maserrat and J. Pei. Community Preserving Lossy Compression of Social Networks, In Proc. ICDM, 2012.
- P. Boldi, Marco Rosa, Massimo Santini, and Sebastiano Vigna. Layered label propagation: a multiresolution coordinate-free ordering for compressing social networks. In WWW'11.
- Y. Choi and W. Szpankowski. Compression of Graphical Structures: Fundamental Limits, Algorithms, and Experiments. Information Theory, IEEE Transactions on, 58(2):620–638, February 2012

References: Application-oriented Summarization

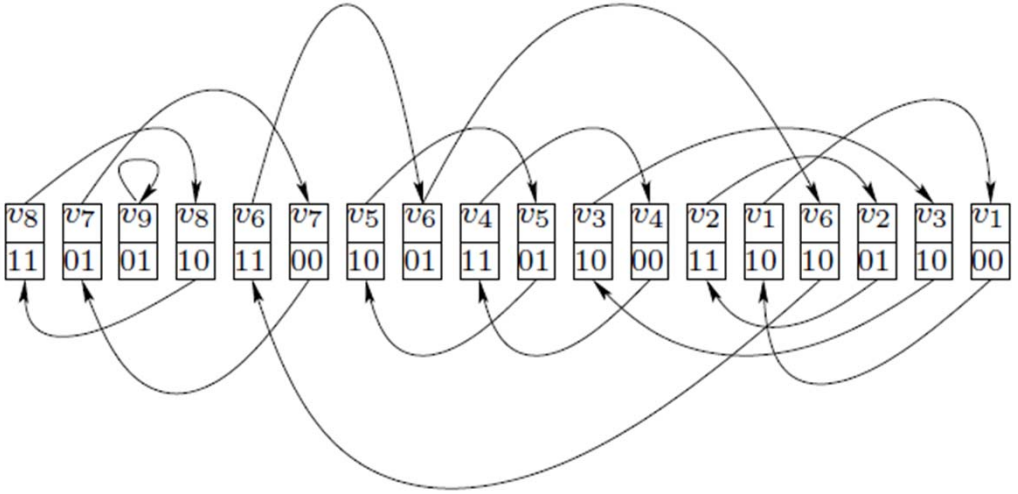
- F. Zhou, S. Malher, and H. Toivonen. Network Simplification with Minimal Loss of Connectivity. In Proc. of IEEE International Conference on Data Mining (ICDM'10), 2010.
- H. Toivonen, F. Zhou, A. Hartikainen, and A. Hinkka. Compression of Weighted Graphs, In Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11), 2011.
- U. Kang, H. Tong, J. Sun, C. Y. Lin, and C. Faloutsos. GBASE: A Scalable and General Graph Management System, In Proc. of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD'11), 2011.
- K. LeFevre and E. Terzi. GraSS: Graph Structure Summarization. In Proc. of SIAM International Conference on Data Mining (SDM'10), 2010.
- U. Kang and C. Faloutsos. Beyond 'Caveman Communities': Hubs and Spokes for Graph Compression and Mining. In Proc. of IEEE International Conference on Data Mining (ICDM'10), 2010.
- C. Chen, C. X. Lin, M. Fredrikson, M. Christodorescu, X. Yan, and J. Han. Mining Graph Patterns Efficiently via Randomized Summaries. Proc. VLDB Endow., 2(1):742–753, August 2009.
- W. Fan, J. Li, X. Wang, and Y. Wu. Query Preserving Graph Compression, In Proc. of ACM SIGMOD International Conference on Management of Data (SIGMOD'12), 2012.

Thank you!

Q&A

Neighbor Query Friendly Compression of Social Networks (cont'd)

[Maserrat'10]



Neighbor query of v in $O(\sum_{u \in N_v} deg(u) \log |V(G)|)$

The upper bound of bits used for encoding a graph is asymptotically about ½ log(|V(G)|), which is the number of bit used for encoding an edge by baseline.

* Similar ideas are also used for lossy compression to preserve communities in social networks [Maserrat'12].