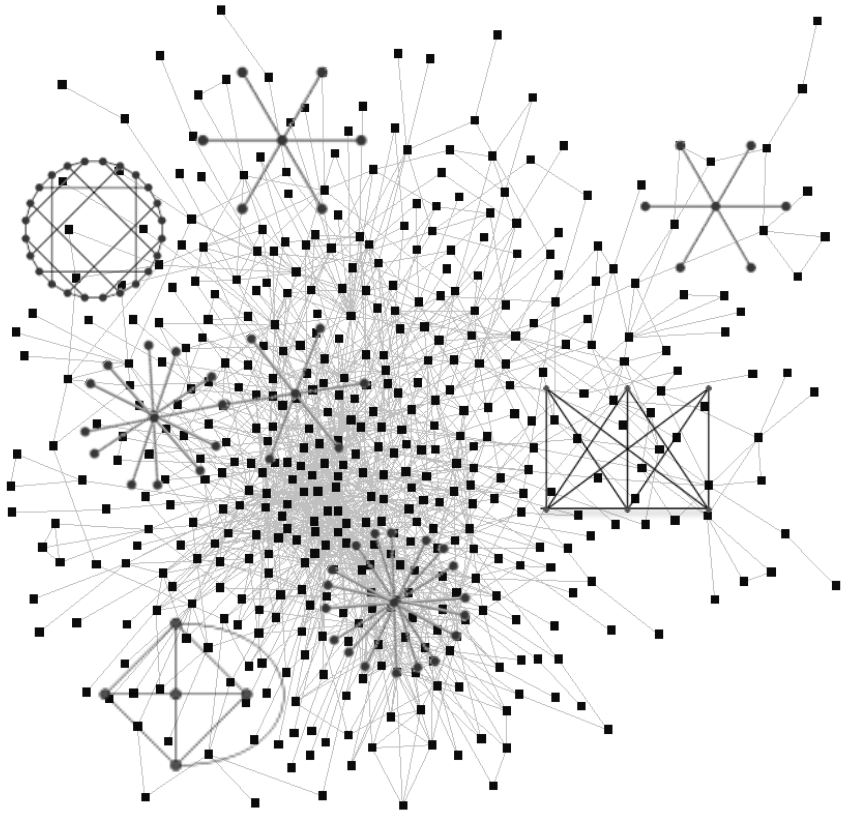
Carnegie Mellon University
KAIST Korea Advanced Institute of Science and Technology
MAX-PLANCK-GESELLSCHAFT
UNIVERSITÄT DES SAARLANDES

VoG: Summarizing and Understanding Large Graphs

Danai Koutra
U Kang
Jilles Vreeken
Christos Faloutsos

SDM, 24-26 April 2014, Philadelphia, USA

Problem Definition: Graph Summarization



Given: a graph

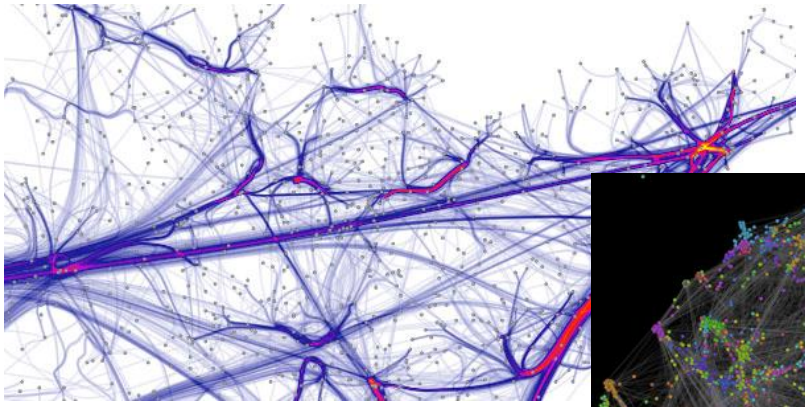
Find: a **succinct** summary with possibly overlapping subgraphs

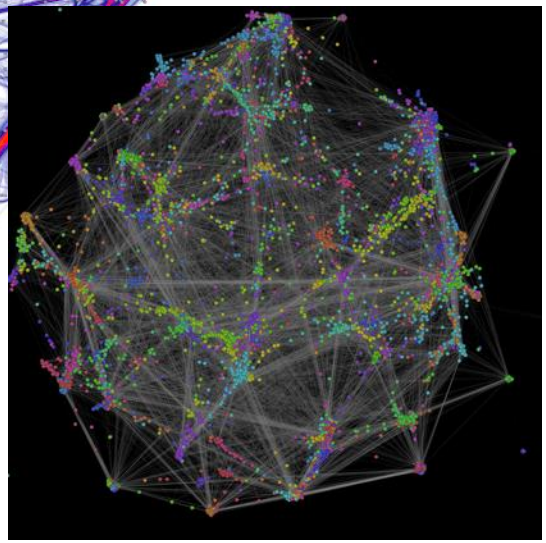
≈

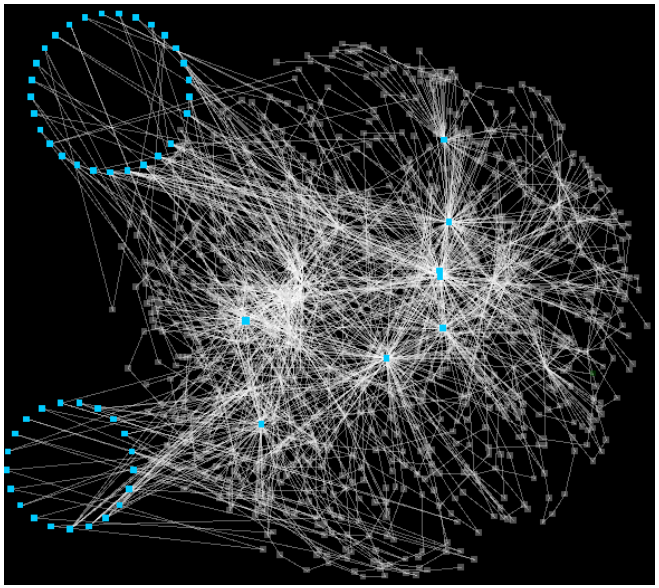
important graph structures

Why graph summarization?

Visualization

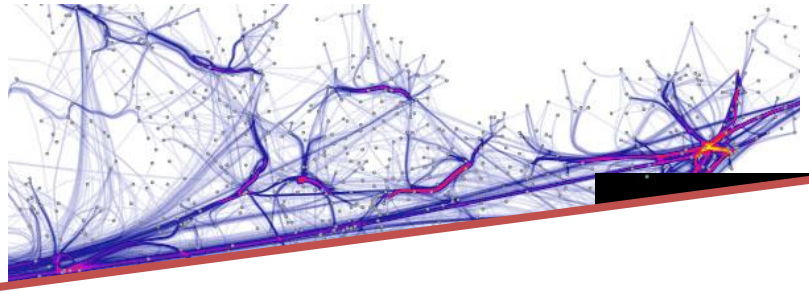




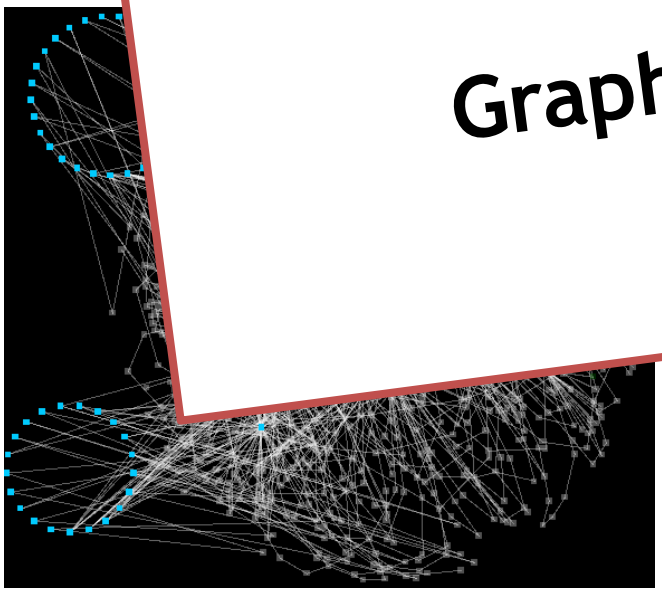


Guiding attention

Why graph summarization?



Graph Understanding



Roadmap

Main Idea

Proposed Algorithm: VoG

VoG: Step-by-Step

Experiments

Conclusions

Main Idea

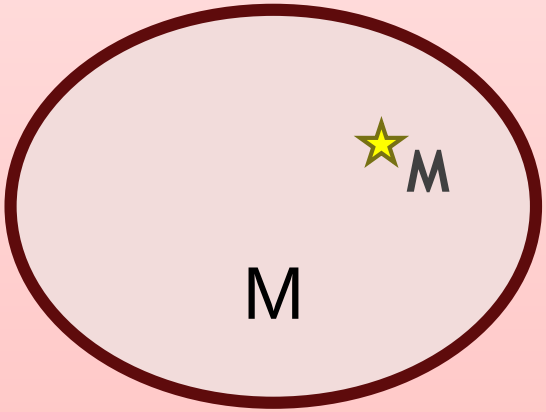
1) Use a graph vocabulary:

2) Best graph summary

➔ structures with the lowest compression cost (MDL)

Minimum Description Length Principle

- Given a set of models M,
- the best model M ε M is



$$\underset{M}{\operatorname{argmin}}\ L(M) + L(D|M)$$

bits for M

bits for the data using M

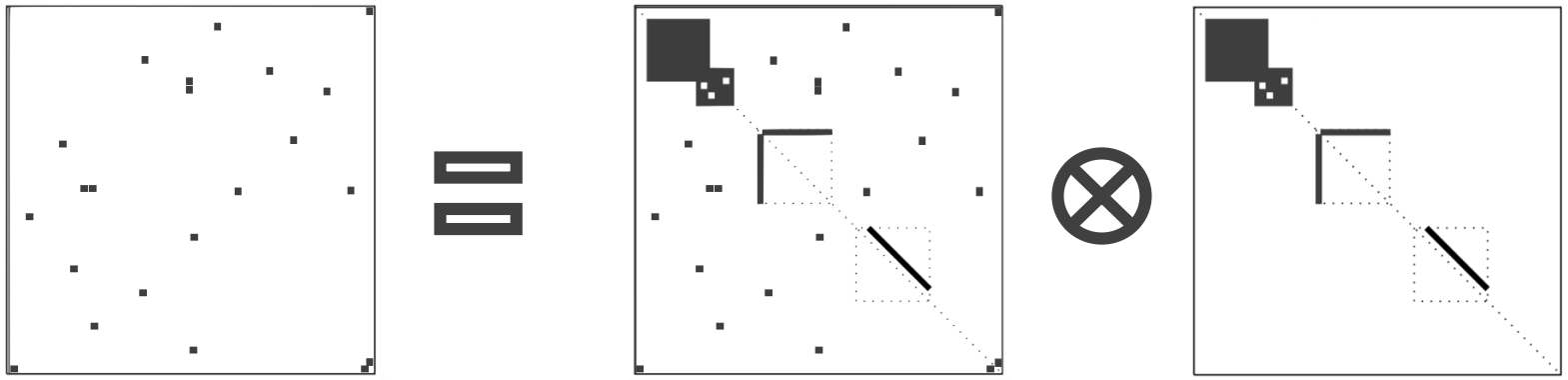
Formally:
Minimum Graph Description

Given: - a graph G with adjacency matrix A
- vocabulary Ω

Find: model M s.t.

$$\min L(G,M) = \min L(M) + L(E)$$

Error E = Adjacency A ⊗ Model M



Roadmap

Main Idea

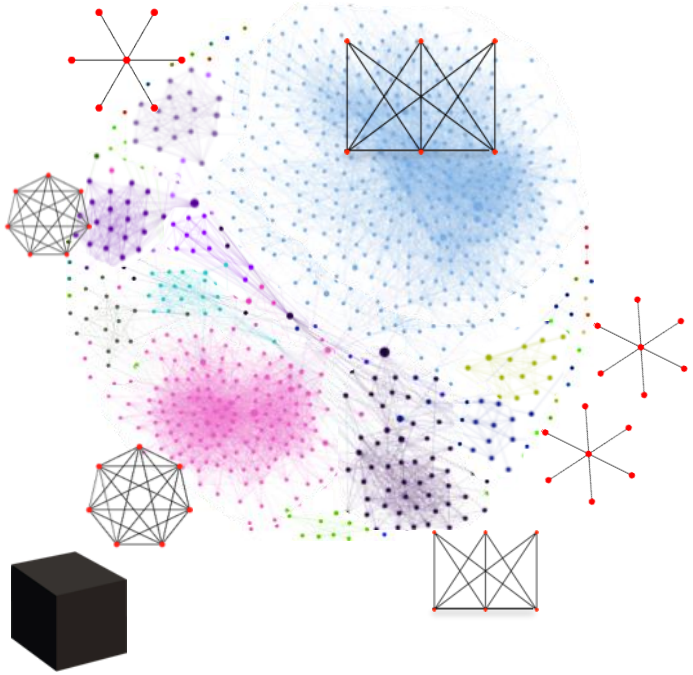
Proposed Algorithm: VoG

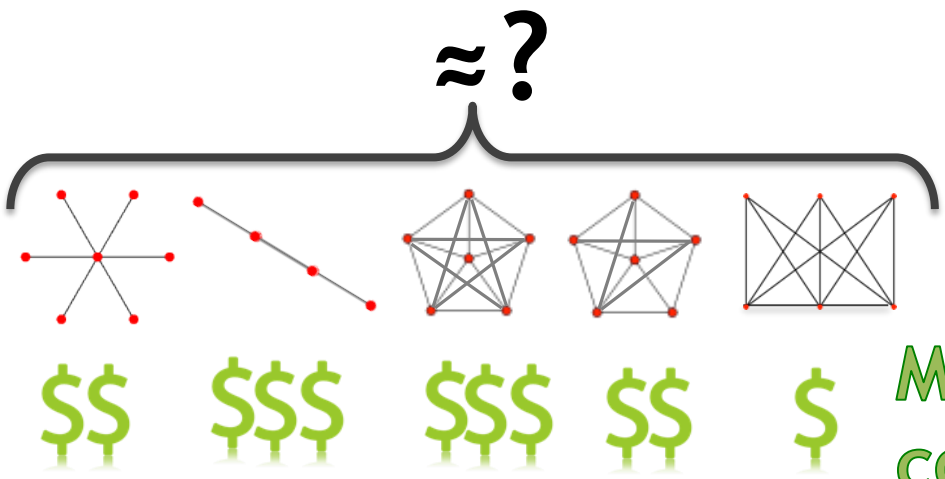
VoG: Step-by-Step

Experiments

Conclusions

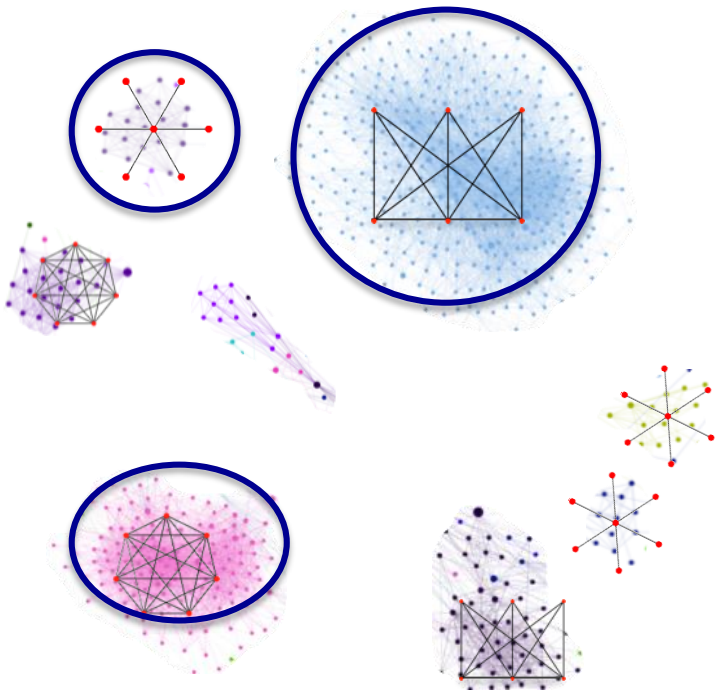
VoG: Overview



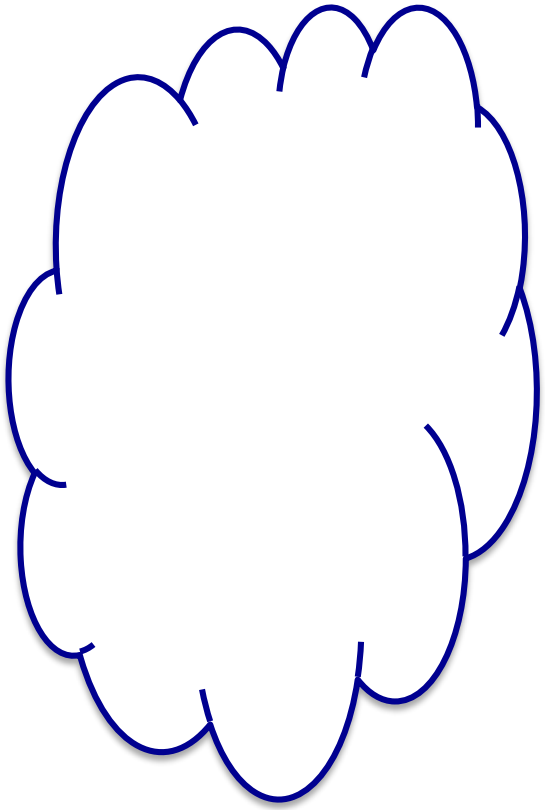


≈

VoG: Overview



some criterion



Summary

Roadmap

Main Idea

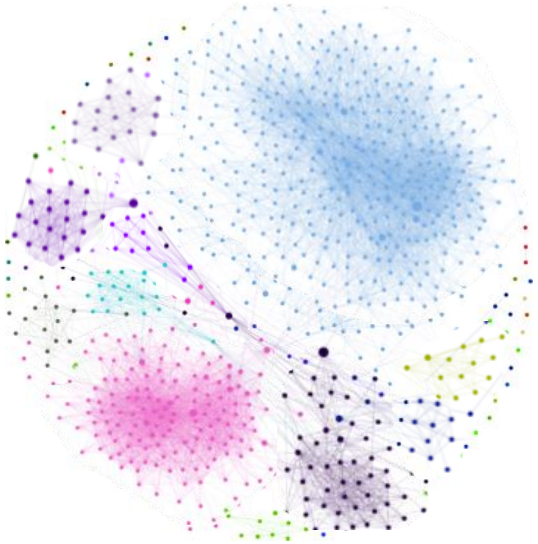
Proposed Algorithm: VoG

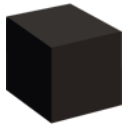
VoG: Step-by-Step

Experiments

Conclusions

Step 1: Graph Decomposition





***Could* use:**

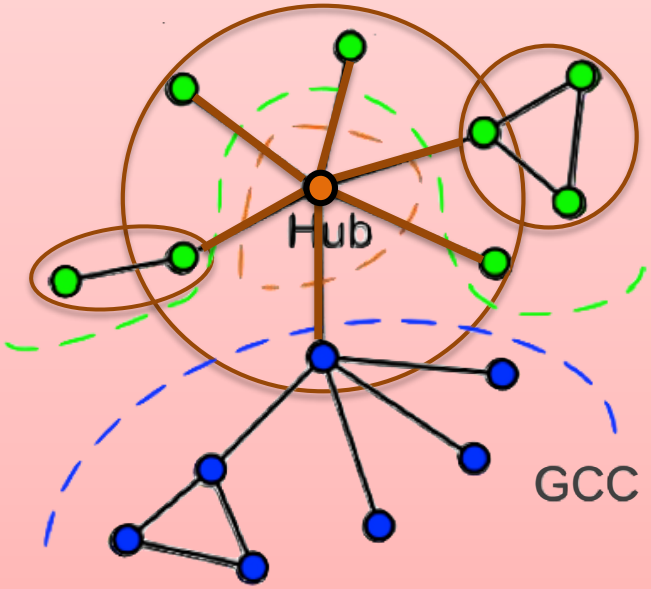
ANY graph decomposition method

***We* use:**

SlashBurn

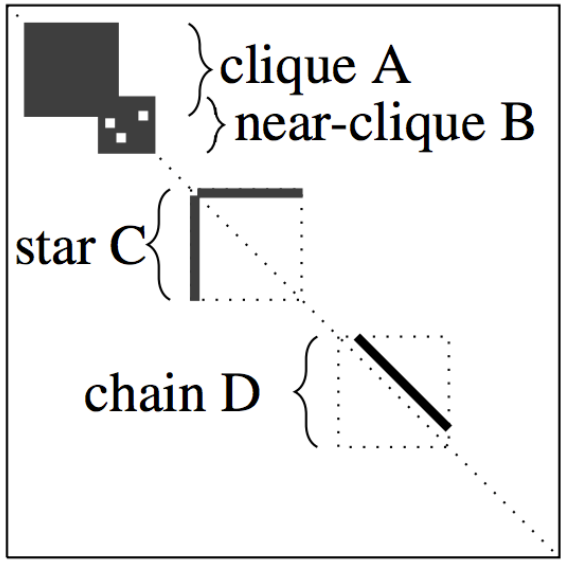
SlashBurn

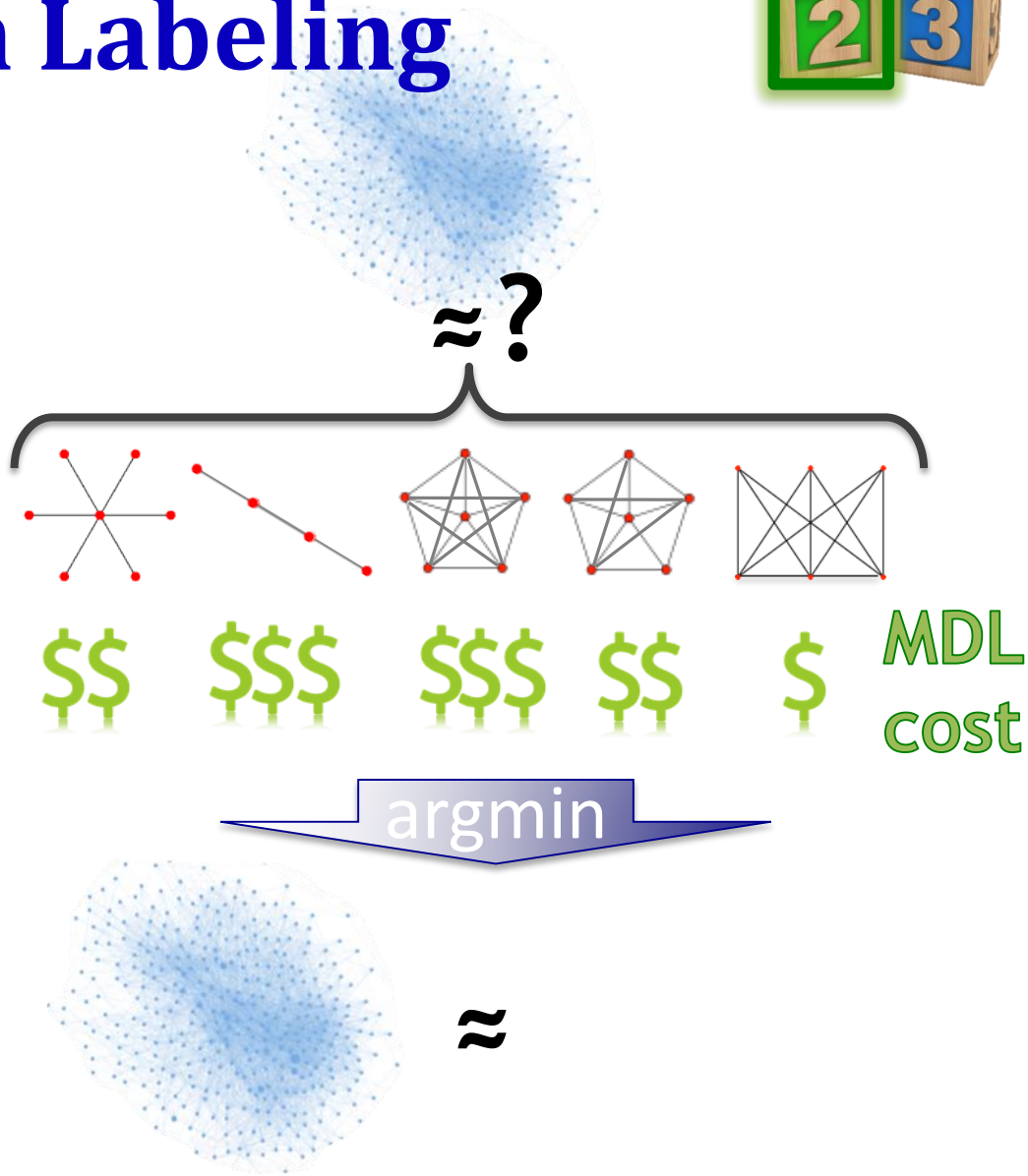
- ***Slash*** top-k hubs, ***burn*** edges
- Repeat on the remaining GCC



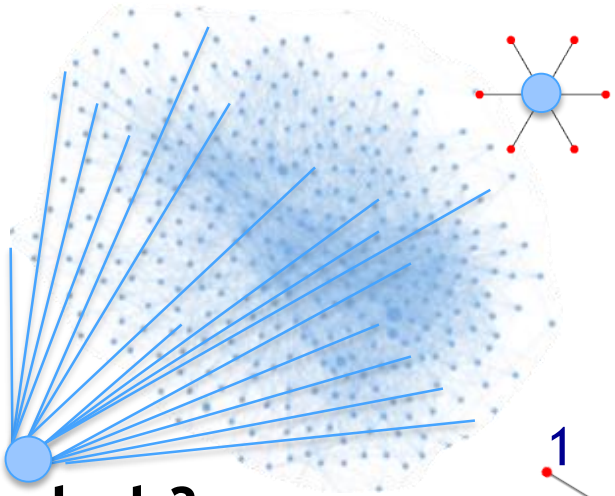
[U Kang and Christos Faloutsos. ICDM'11]

Step 2: Graph Labeling

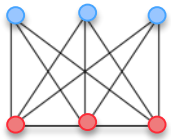


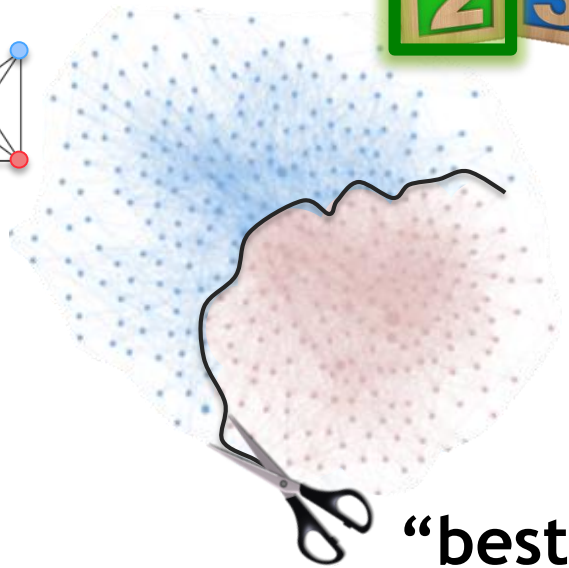


Graph Representation

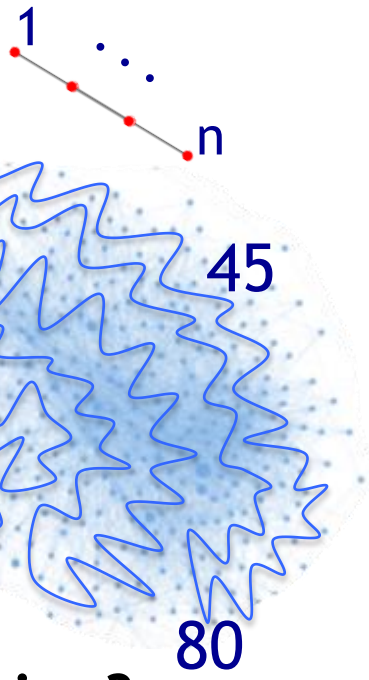


hub?

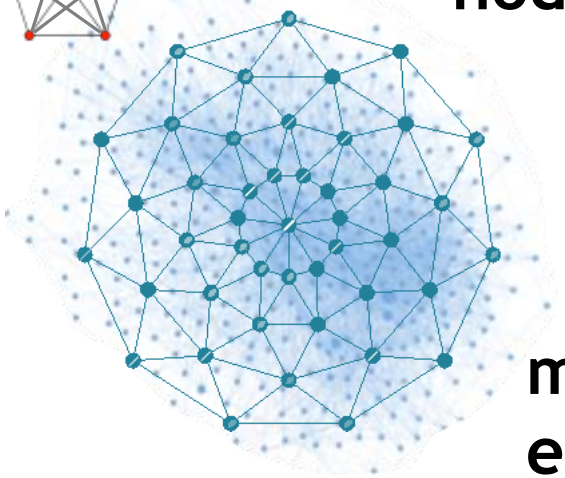




"best"
node split?

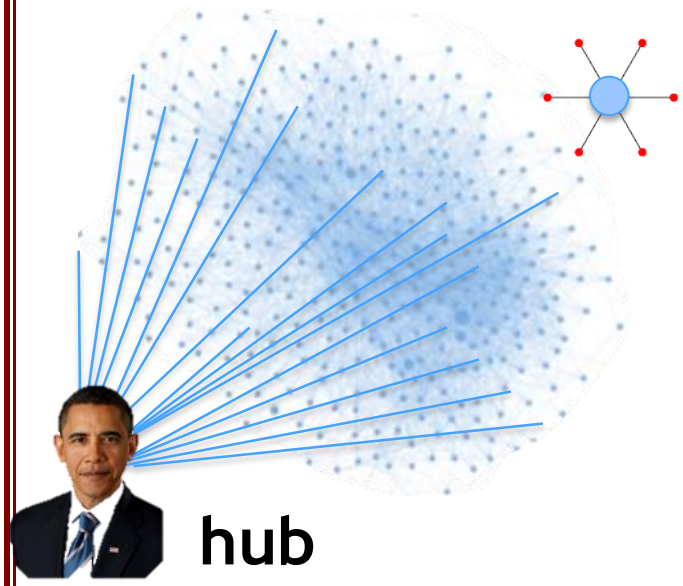


"best"
node ordering?



missing
edges?

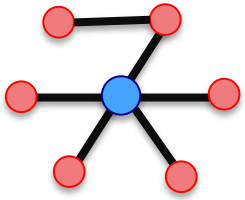
Graph Representation



hub

Hub: top-deg node
Spokes: the rest

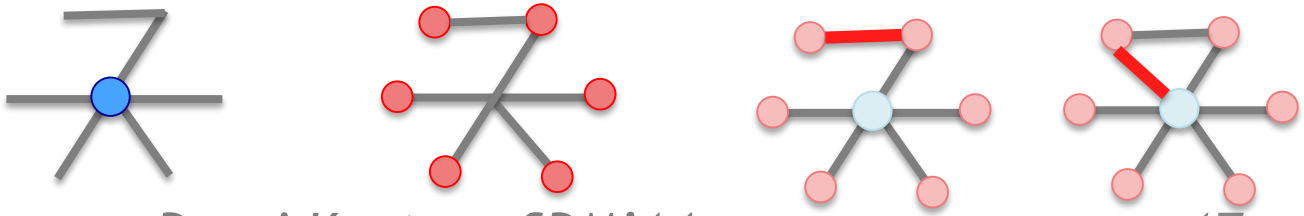
n=7



$$$

Star structure + Errors

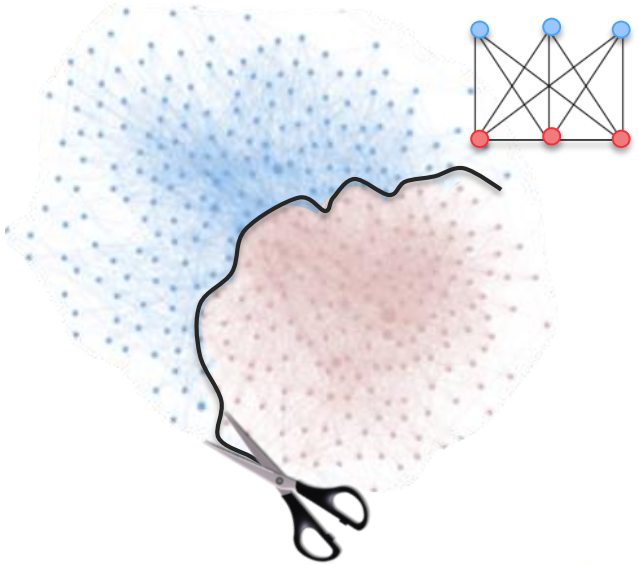
6



Graph Representation

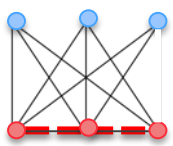
Max bipartite graph: **NP-hard**

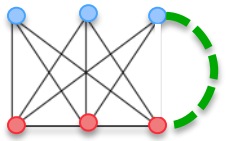
Heuristic: Belief Propagation with heterophily for node classification (blue/red)



$$$

Bipartite graph structure + Errors

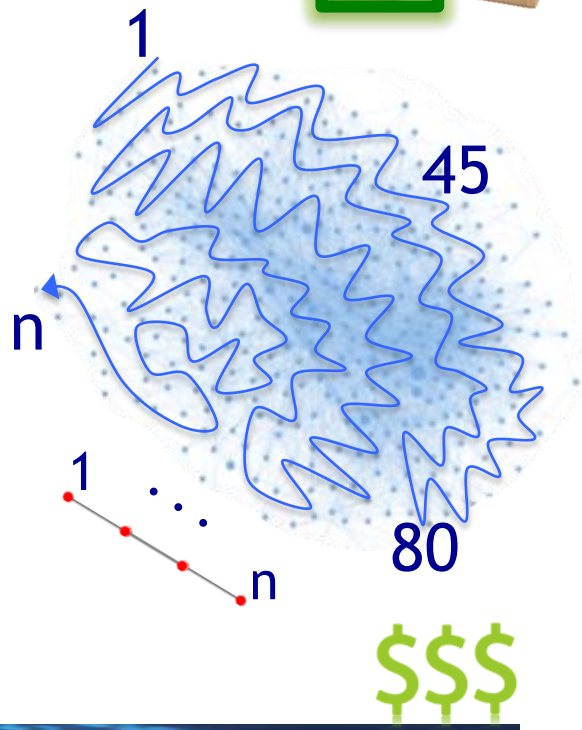




Graph Representation

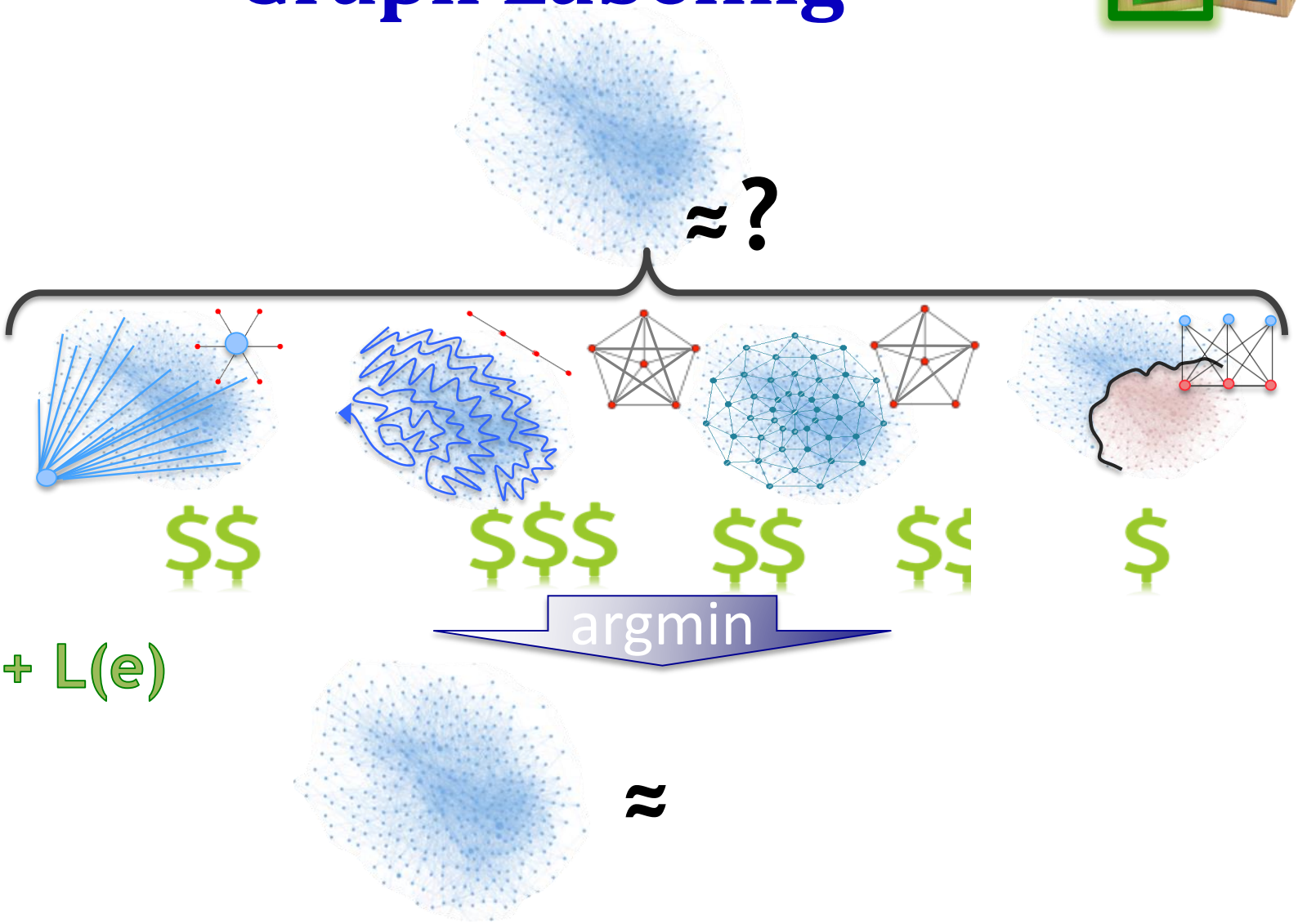
Longest path: **NP-hard**

Heuristic: BFS + local search



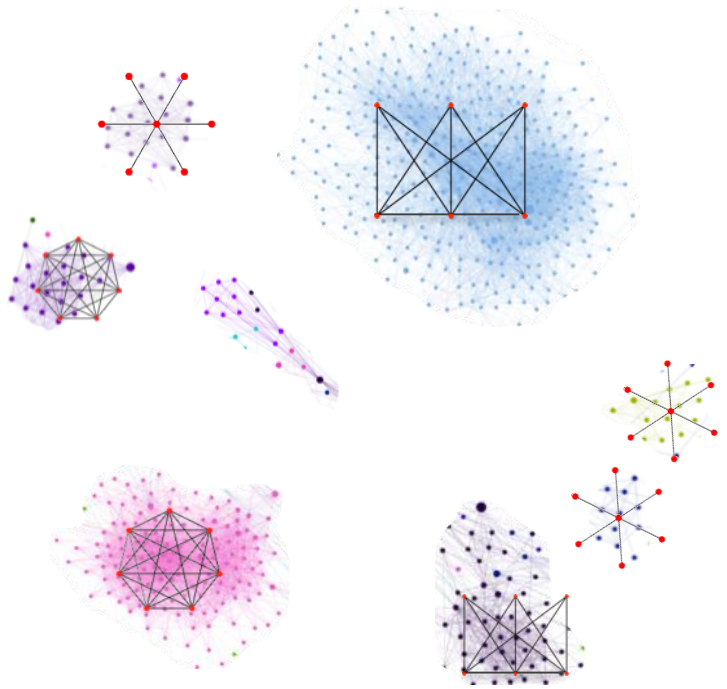
Chain structure + Errors

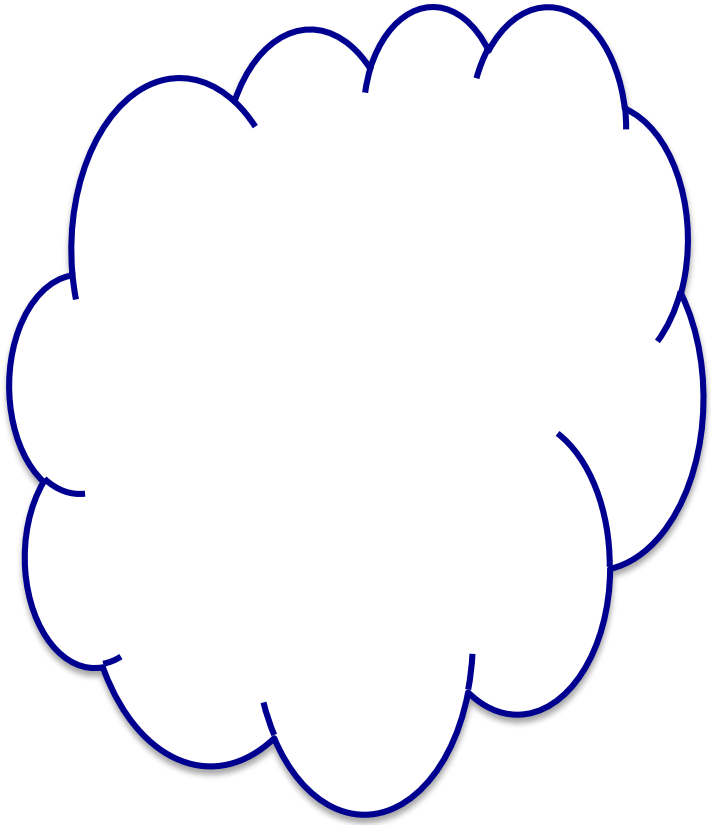
Step 2: Graph Labeling



Step 3: Summary Assembly

PLAIN

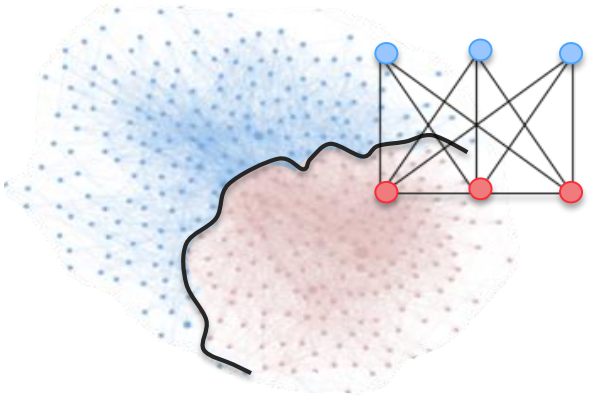


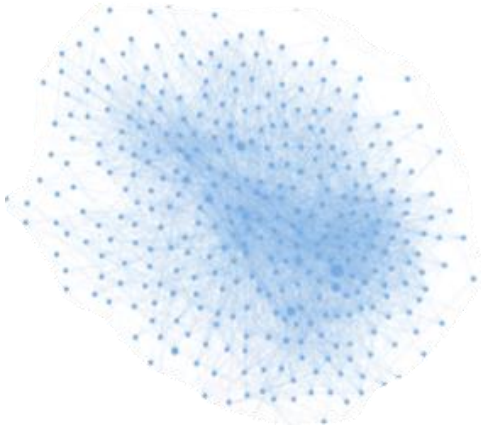


Concepts

Savings = # bits as structure - # bits as noise

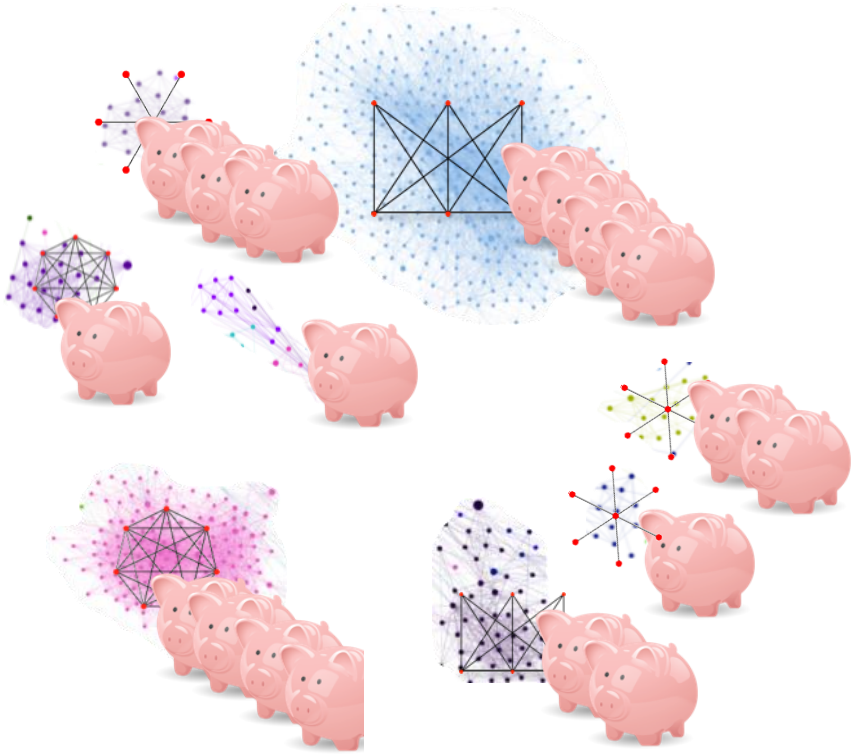
compression gain

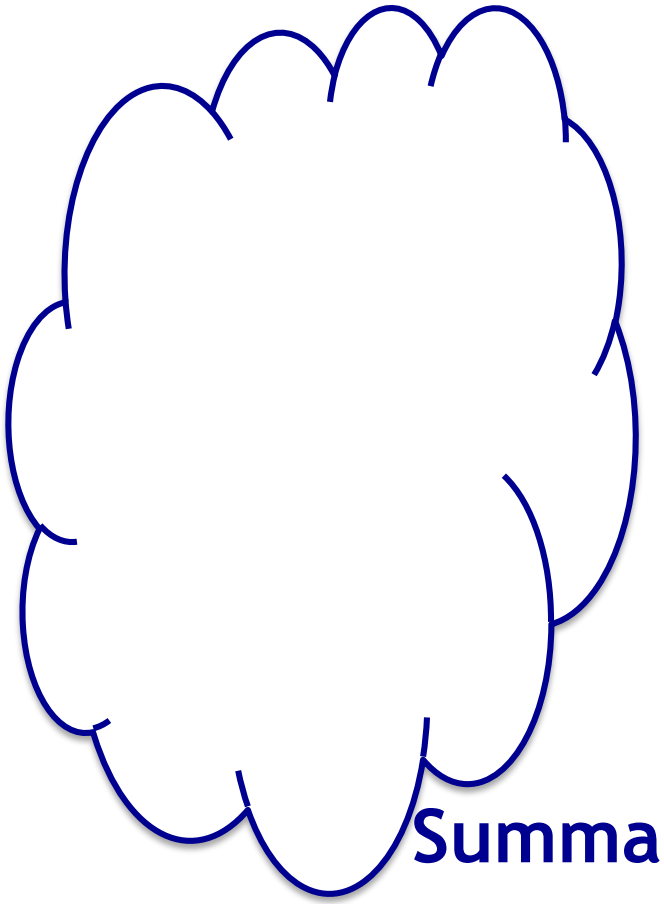




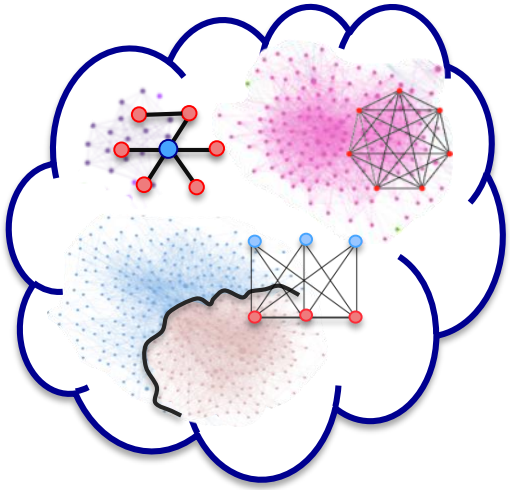
Step 3: Summary Assembly

TOP-k





Concepts



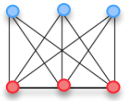
Summary Encoding cost

L(M) = # of structures **+** # of structures per type **+** for each structure its encoding length

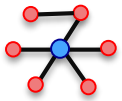
3

 : 1

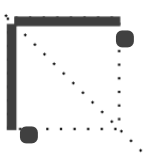
 : 1

 : 1

its type

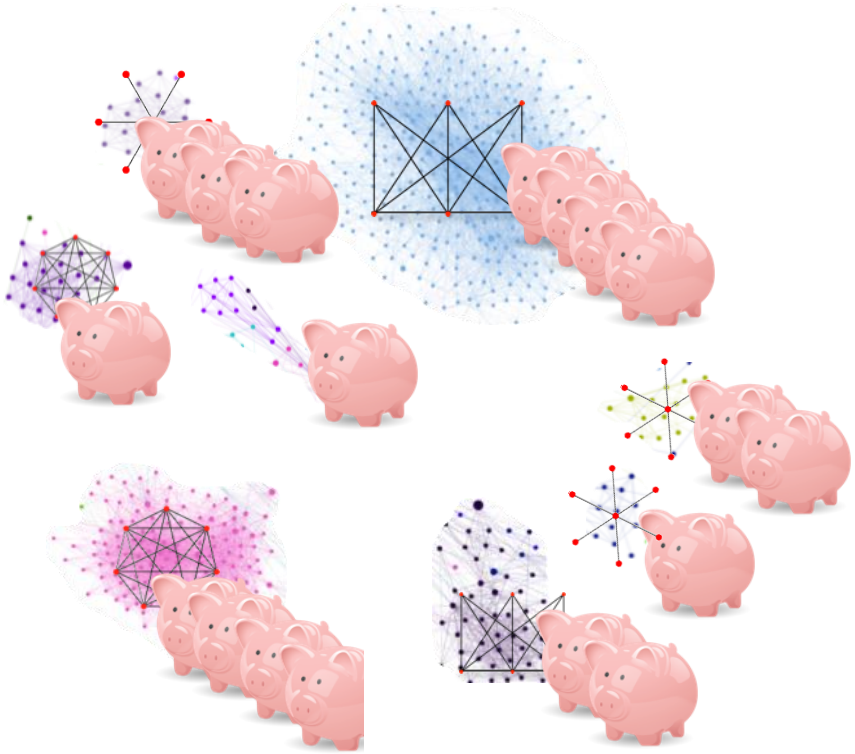


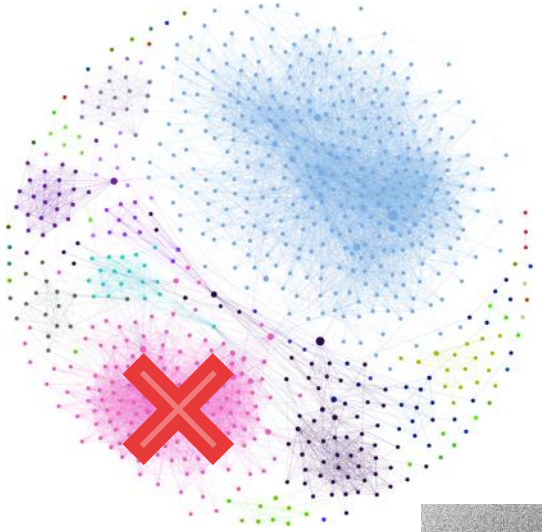
its connectivity

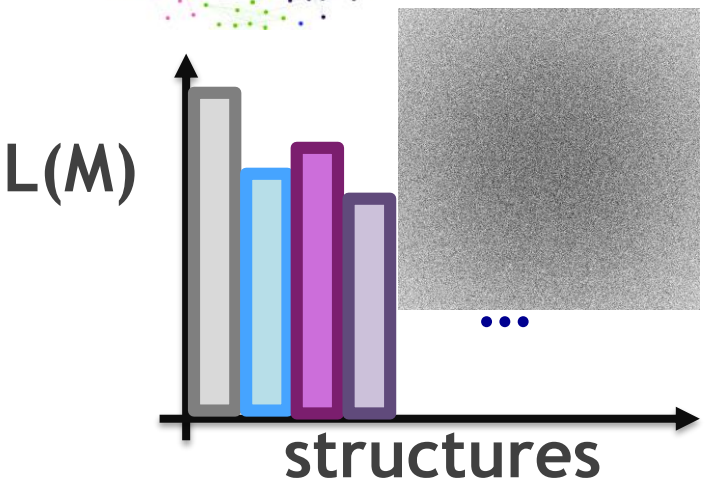


Step 3: Summary Assembly

Greedy&Forget







Roadmap

Main Idea

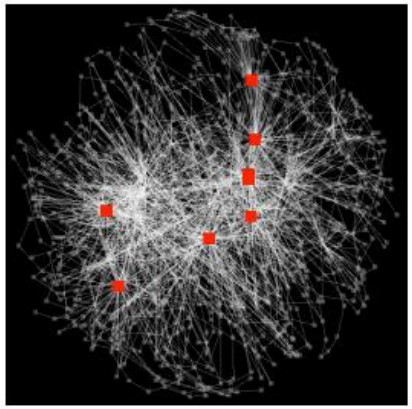
Encoding Schema

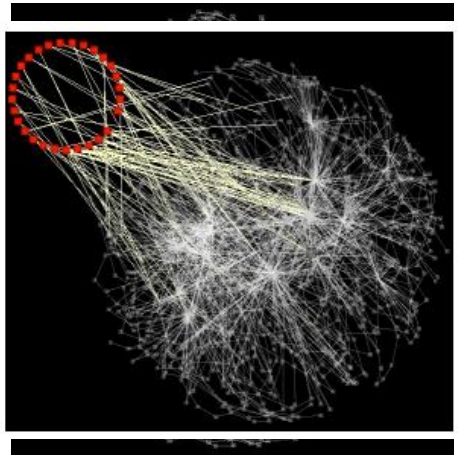
Proposed Algorithm: VoG

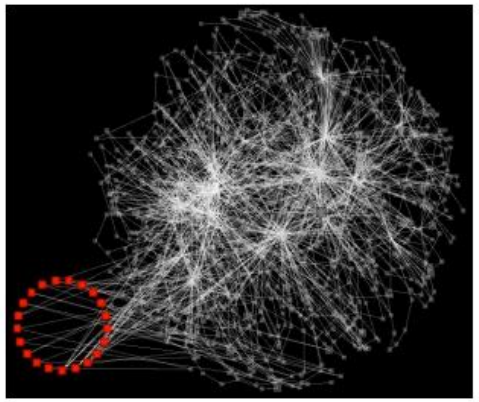
Experiments

Conclusions

Application: Wikipedia controversy







Stars:
admins,
bots,
heavy users

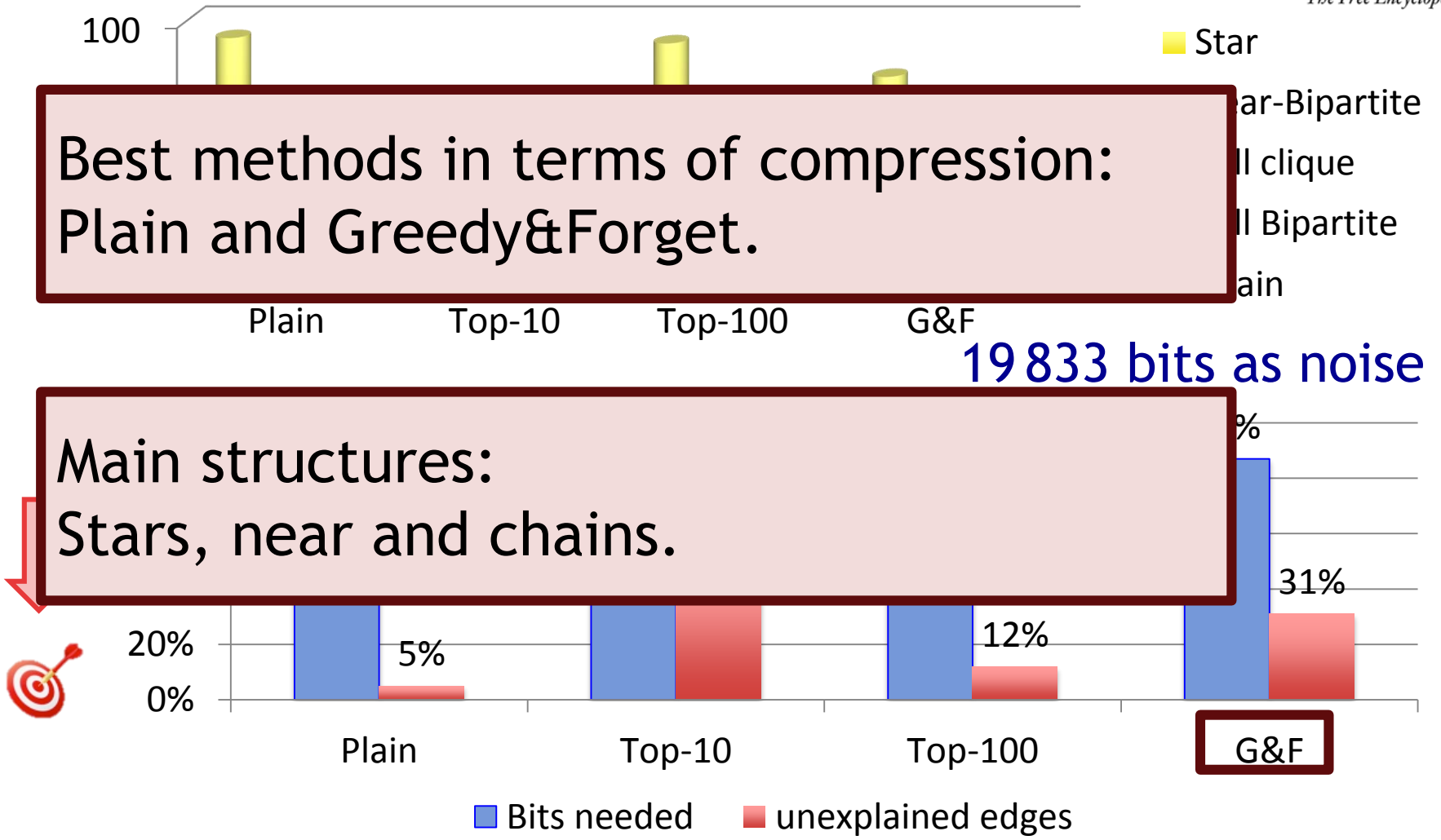
Bipartite cores: edit wars

Kiev vs. Kyiv

vandals

Nodes: wiki editors
Edges: co-edited

Quantitative Analysis



Best methods in terms of compression:
Plain and Greedy&Forget.

Main structures:
Stars, near and chains.

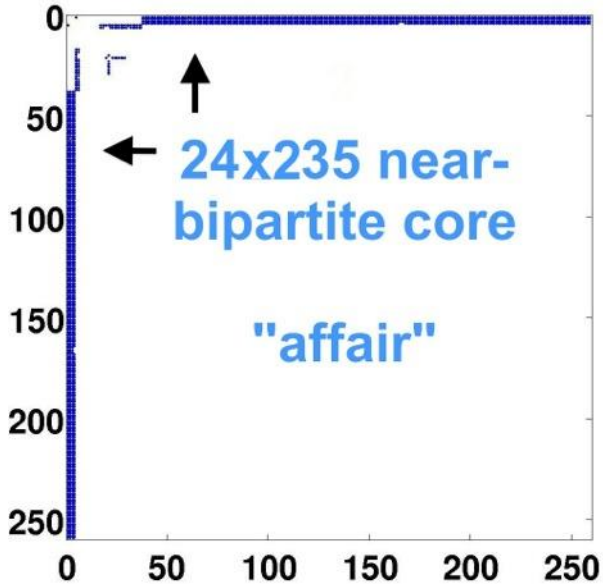
Application: Enron

Top-3 Stars

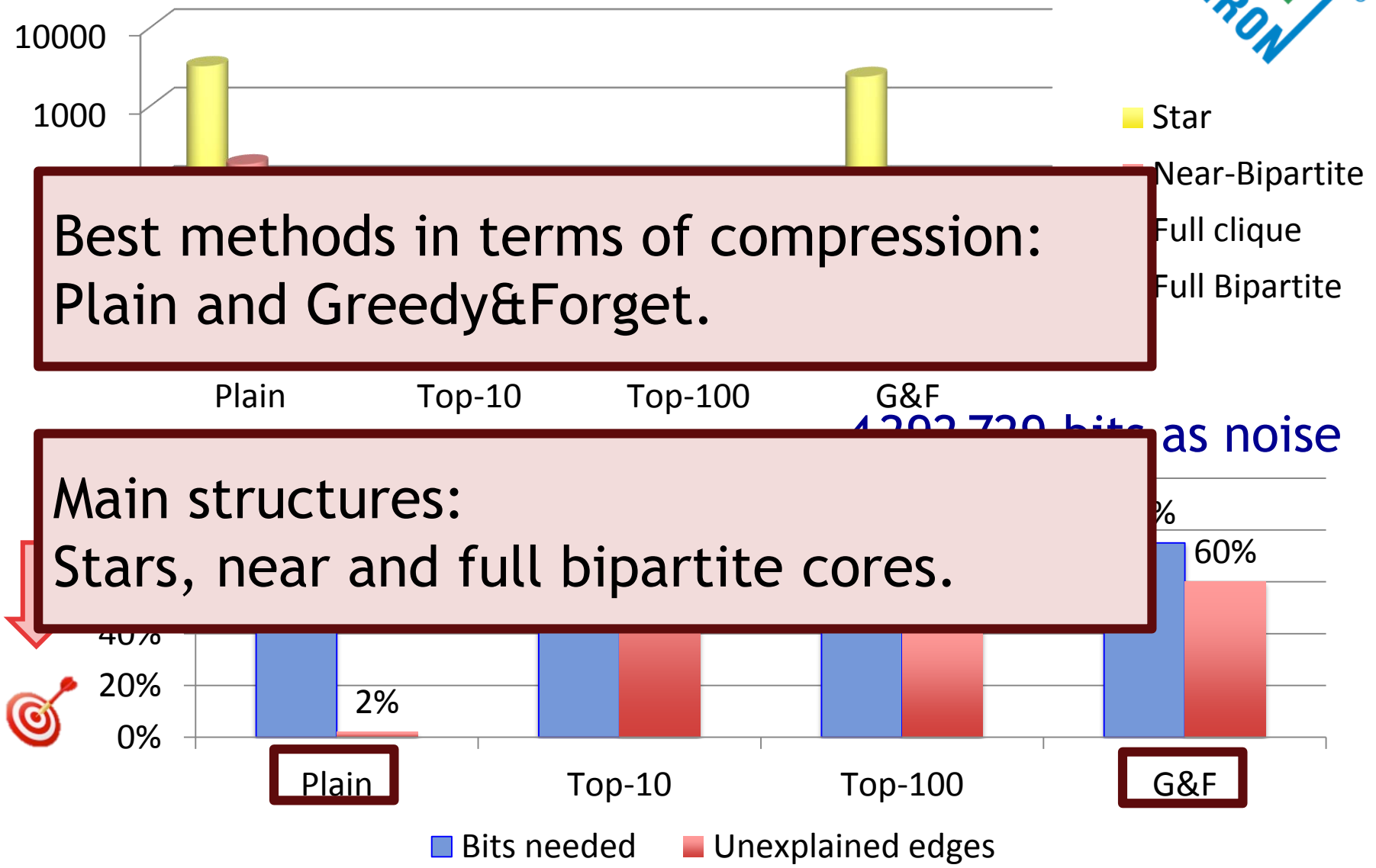
klay
kenneth.lay
@enron.com

jeff.skilling@enron.com

Top-1 NBC



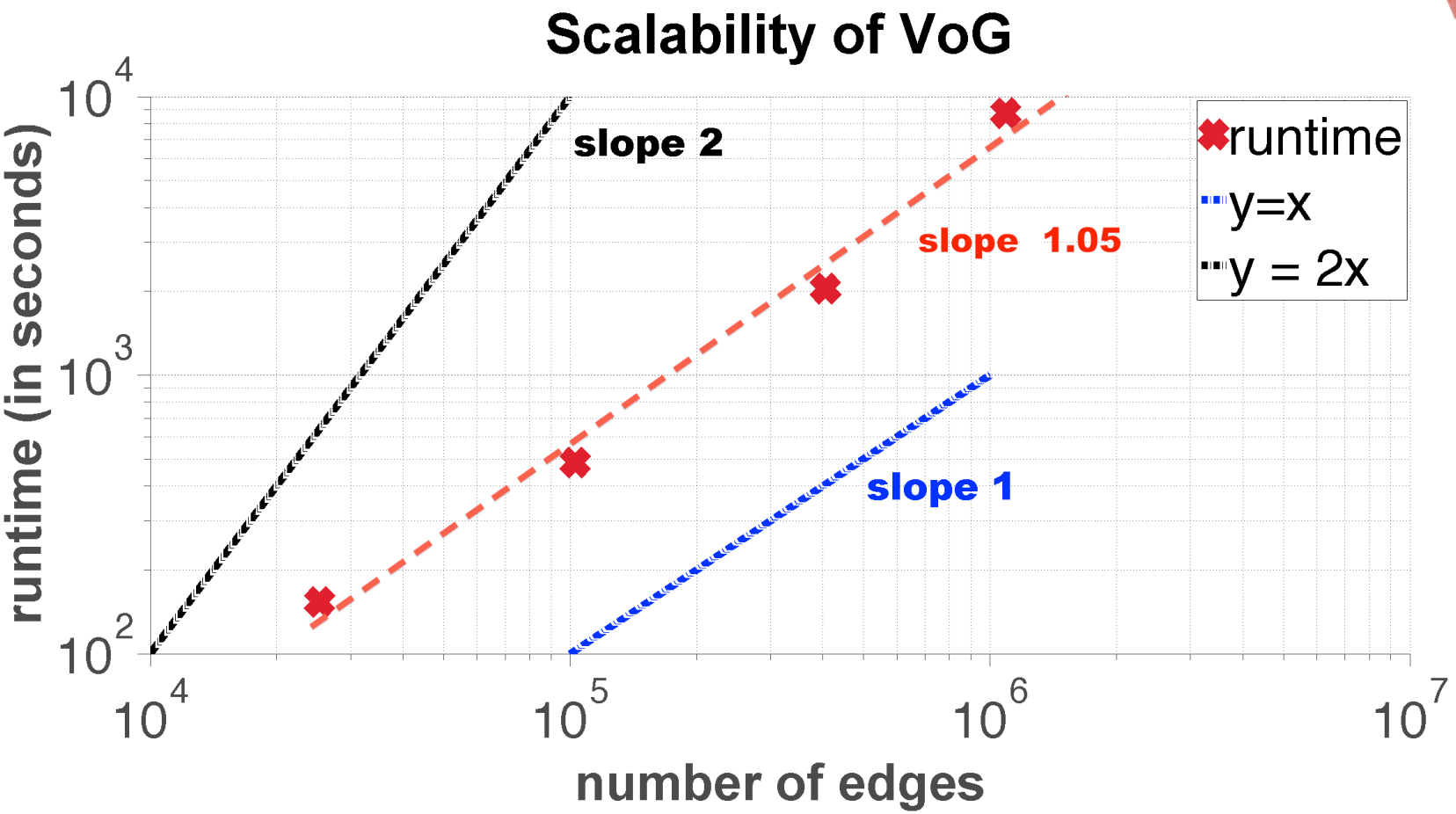
Quantitative Analysis



Best methods in terms of compression:
Plain and Greedy&Forget.

Main structures:
Stars, near and full bipartite cores.

Runtime



VOG is *near-linear* on the number of edges of the input graph.

Roadmap

Main Idea

Encoding Schema

Proposed Algorithm: VoG

Experiments

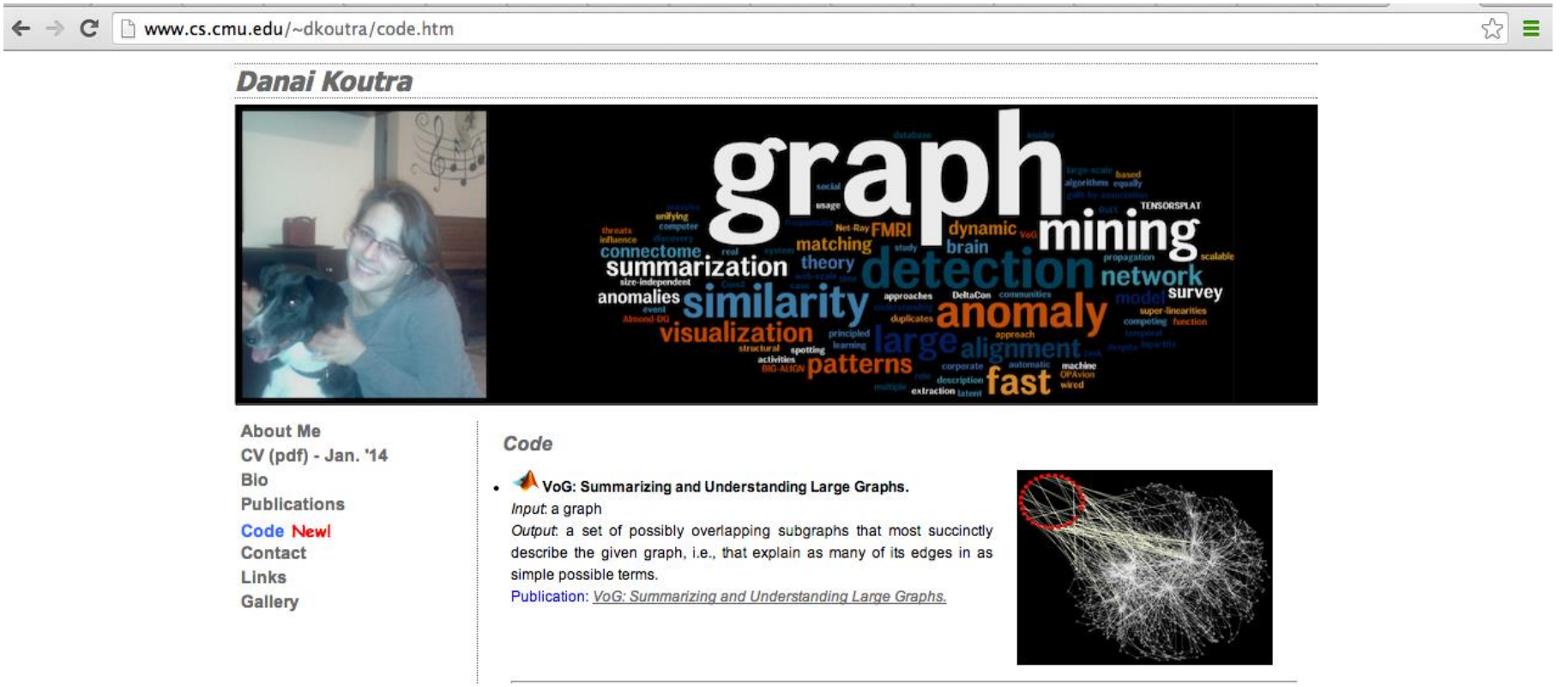
Conclusions

Conclusions

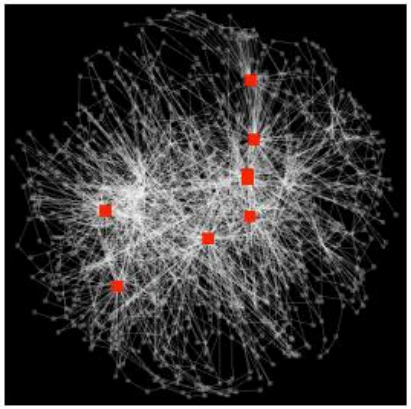
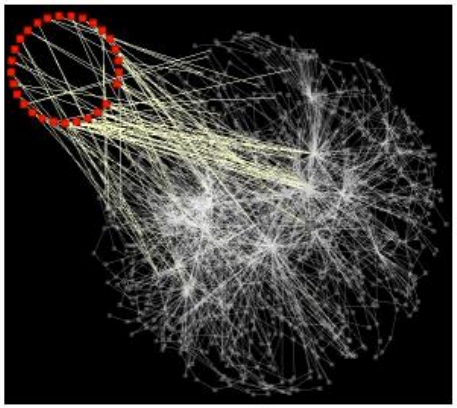
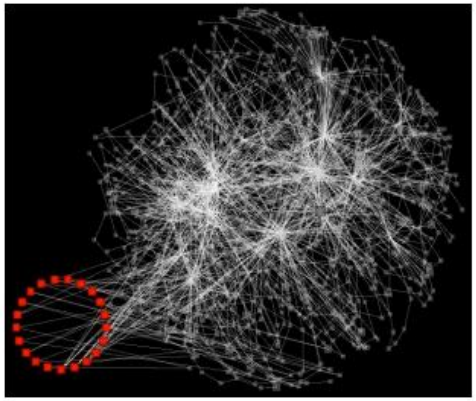
- **Formulation:**
  info-theoretic graph summarization approach
- **Algorithm:**
  VoG is near-linear on the edges
- **Experiments** on real graphs

Code

www.cs.cmu.edu/~dkoutra/SRC/vog.tar



Thank you! Questions?

www.cs.cmu.edu/~dkoutra/pub.htm

danai@cs.cmu.edu