SEATTLE
WA, USA
SIGMOD PODS 2023

EFFECTIVE AND EFFICIENT PAGERANK-BASED POSITIONING FOR GRAPH VISUALIZATION

Shiqi Zhang, Renchi Yang, Xiaokui Xiao, Xiao Yan, Bo Tang

June 2023

NUS National University of Singapore
香港浸會大學 HONG KONG BAPTIST UNIVERSITY
SUSTech Southern University of Science and Technology

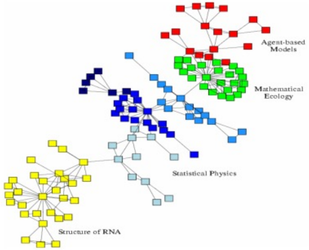
OUTLINE

- Background
- Existing Solutions
- PPRviz
- Experiments
- Conclusion

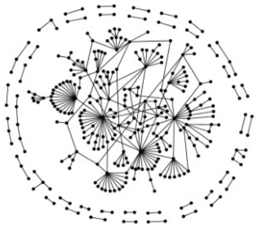
BACKGROUND: GRAPH VISUALIZATION

- Input: a graph G with n nodes and m edges
- Output: a 2D position matrix $\boldsymbol{X}$
- Drawing:
  - Position each node v_i at its coordinate $X[i]$
  - Link two endpoints of each edge with a straight segment
- It helps to understand relational data

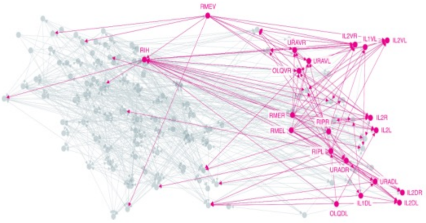
Social networks



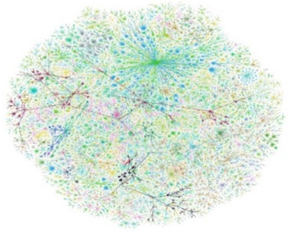
Economic networks



Biomedical networks



Networks of neurons



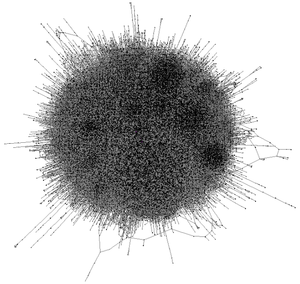
Internet

BACKGROUND: AESTHETIC CRITERIA

- An effective visualization should have good readability
- Evaluate the readability of ***X*** by aesthetic criteria
- Node Distribution (ND):
  - measure the distribution evenness of the nodes on the screen
- Uniform Length Coefficient Variance (ULCV):
  - measure the length skewness of edge segments on the screen

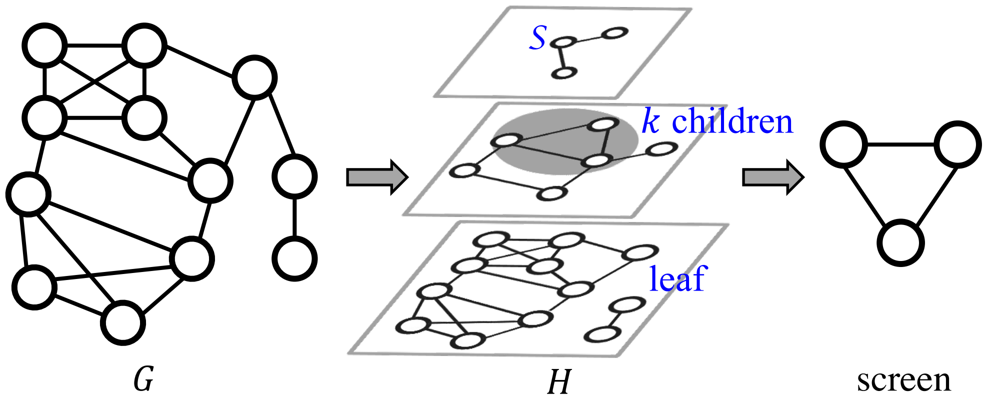
EXISTING SINGLE-LEVEL SOLUTIONS

- Idea:
  - visualize all nodes and edges on the screen
- Steps:
  - compute a graph-theoretical distance matrix ***D***:
    - adjacency-related matrix or the shortest distance matrix
  - embed ***D*** into ***X***:
    - minimize node pair's difference between graph and Euclidean distance
- Cons:
  - Poor readability: aesthetically-unpleasing or hairball-like layout
  - Expensive computational cost



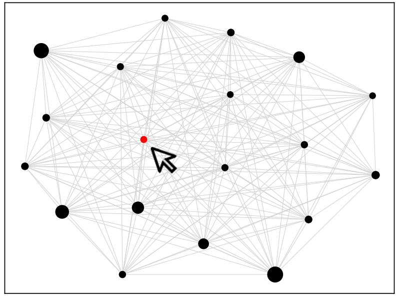
EXISTING MULTI-LEVEL FRAMEWORK

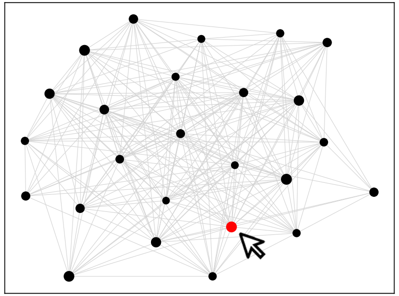
- Idea:
  - interactively show the partial view level by level
- Steps:
  - build a supergraph hierarchy H for G
  - use a single-level solution to visualize children in S
- Pros:
  - avoid hairball
  - reduce embedding overhead
- Cons:
  - the aesthetic issue remains

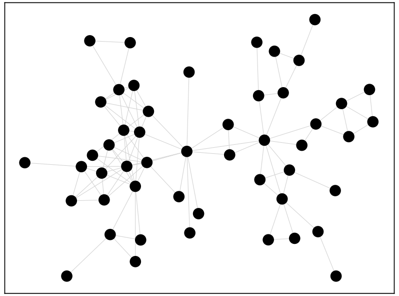


PPRVIZ: OVERVIEW

- Supergraph hierarchy construction :
  - Generate H by Louvain [a] with balanced size
- Node distance computation :
  - Design a new distance measure PDist
  - Compute PDist matrix $\boldsymbol{\Delta}$ for children in S by our Tau-Push
- Position embedding:
  - Compute $\boldsymbol{X}$ by $\boldsymbol{\Delta}$
  - Make node pair's Euclidean distance resemble its PDist



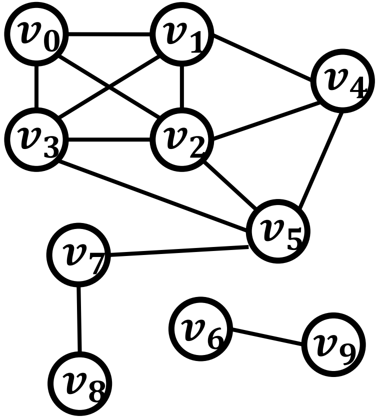




[a] Blondel, Vincent D., et al. "Fast unfolding of communities in large networks." *Journal of statistical mechanics: theory and experiment* (2008).

PPRVIZ: PDIST FOR LEAF NODES

- Personalized PageRank (PPR)
  - Input: a source v_s, a target v_t, and a stopping probability α
  - Random walk with restart (RWR) from v_s:
    - At each step, stops with probability α at the current node,
    - With $1 - \alpha$ probability randomly jumps to one of the neighbors
  - PPR from v_s to v_t: $\pi(v_s, v_t) = \mathbb{P}[\text{RWR from } v_s \text{ stops at } v_t]$

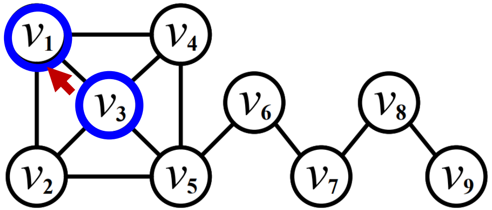


$\pi(v_0, v_8)$	0.01
$\pi(v_2, v_0)$	0.11
$\pi(v_6, v_9)$	0.44

A large $\pi(v_s, v_t)$ indicates v_s and v_t are well-connected, which should be close on graph and screen.

PPRVIZ: PDIST FOR LEAF NODES

- PDist between any nodes v_i, v_j:
  - Degree-normalized PPR (DPPR): $\pi_d(v_i, v_j) = \pi(v_i, v_j) \cdot d(v_i)$
  - Convert DPPR to a distance: $1 - \log\left(\pi_d(v_i, v_j) + \pi_d(v_j, v_i)\right)$
- Pros:
  - Preserve high-order information
  - Guarantee visualization quality in terms of ND and ULCV



RWR from v_3 to v_1:

- $v_3 \to v_1$
- $v_3 \to v_4 \to v_1$
- $v_3 \to v_2 \to v_1$
- $v_3 \to v_5 \to v_4 \to v_1$
- $\ldots$

ignored by shortest distance!

PPRVIZ: TAU-PUSH FOR LEAF NODES

- Tau-Push
  - Compute the tau value τ_j for each v_j and compute a constant τ, where

$$\tau_j = \frac{1}{m} \cdot \sum_i \pi_d(v_i, v_j)$$

  - Estimate $\mathbf{\Delta}[\mathrm{i}, \mathrm{j}]$ for v_j with $\tau_j < \tau$ by a deterministic version of RWR from v_i
  - Estimate $\mathbf{\Delta}[\mathrm{i}, \mathrm{j}]$ for v_j with $\tau_j \geq \tau$ by a reverse traversal from v_j

precompute and store as index

EXPERIMENTS: DATASETS

Dataset	n	m	Description
TwEgo	23	52	Ego network
FbEgo	52	146	Ego network
Wiki-ii	186	632	Authorship network
Physician	241	1.8K	Social network
FilmTrust	874	2.6K	User trust network
SciNet	1.5K	5.4K	Collaboration network
Amazon	334.9K	1.9M	Product network
Youtube	1.1M	6.0M	Social network
Orkut	3.1M	234.4M	Social network
DBLP	5.4M	17.2M	Collaboration network
It-2004	41.3M	2.3B	Crawled network
Twitter	41.7M	3.0B	Social network

Dataset statistics ($K = 10^3, M = 10^6, B = 10^9$)

EXPERIMENTS: COMPETITORS

- Single-level competitors
  - Force-directed methods: FR, LinLog, ForceAtlas
  - Stress methods: CMDS, PMDS
  - Node embedding methods: GFactor, SDNE, LapEig, LLE, Node2vec
  - A variant replacing DPPR in PDist with SimRank
- Multi-level competitors
  - OpenOrd, KDraw
- Most competitors have been applied in software and libraries like Gephi, Graphviz and NetworkX.

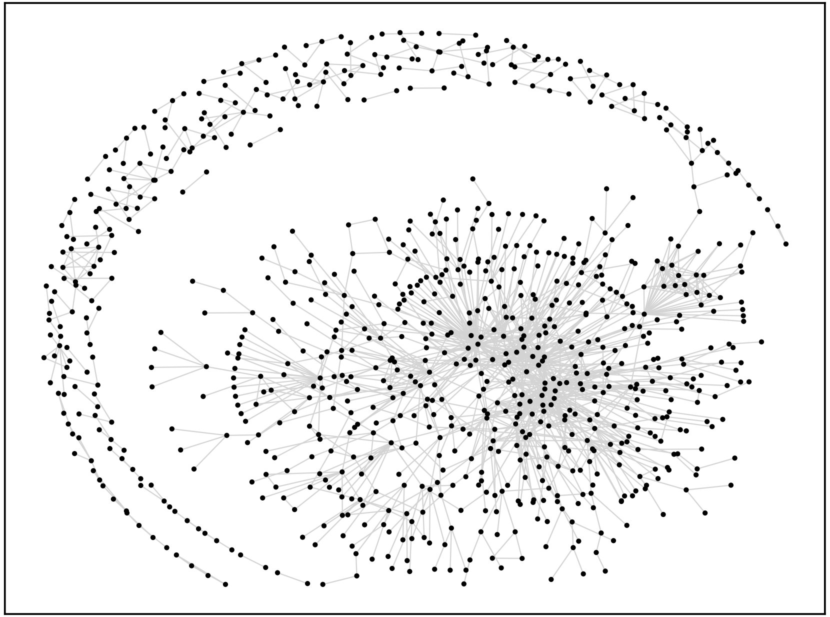
EXPERIMENTS: EFFECTIVENESS OF PPRVIZ

- 6 small datasets and 11 single-level competitors
- ULCV: the smaller the better

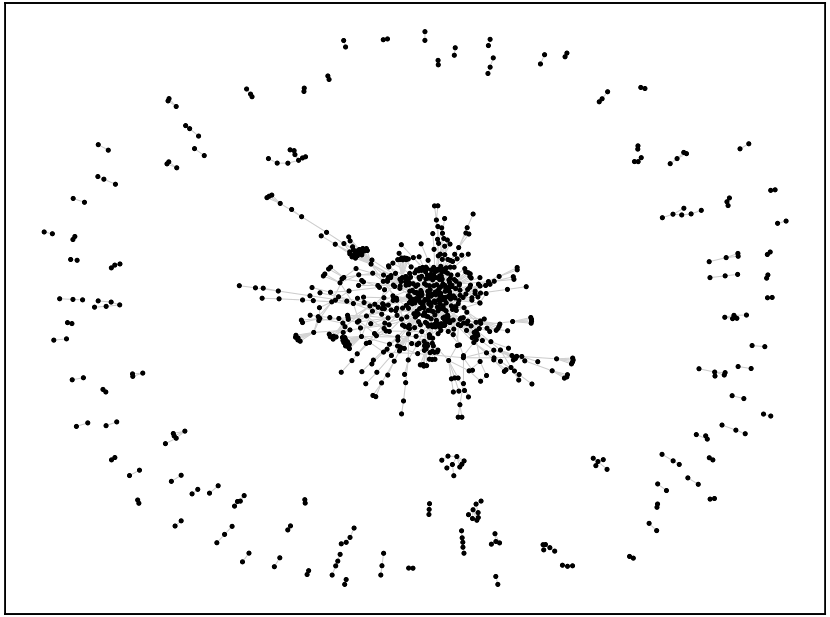
	TwEgo	*FbEgo*	*Wiki-ii*	*Physician*	*FilmTrust*	*SciNet*
PPRviz	**0.22**	**0.39**	**0.35**	**0.45**	**0.48**	**0.34**
FR	0.35	*0.42*	*0.41*	0.53	*0.54*	0.77
LinLog	0.57	0.67	1.09	0.90	1.99	4.70
ForceAtlas	0.37	0.49	0.64	0.55	0.96	1.52
CMDS	0.40	0.46	0.62	0.80	1.05	1.74
PMDS	*0.23*	0.45	0.78	*0.47*	0.69	*0.74*
GFactor	0.45	0.91	0.62	0.95	0.64	0.86
SDNE	1.96	0.94	0.94	1.67	1.31	1.72
LapEig	1.15	0.98	1.04	1.02	1.70	1.26
LLE	0.46	0.77	1.27	0.77	0.87	-
Node2vec	0.80	0.96	0.86	1.41	0.89	1.32
SimRank	0.84	0.75	0.53	0.53	1.78	1.98

EXPERIMENTS: EFFECTIVENESS OF PPRVIZ

- The best competitor FR (in terms of aesthetic criteria)
- Visualizations on FilmTrust



PPRviz



FR

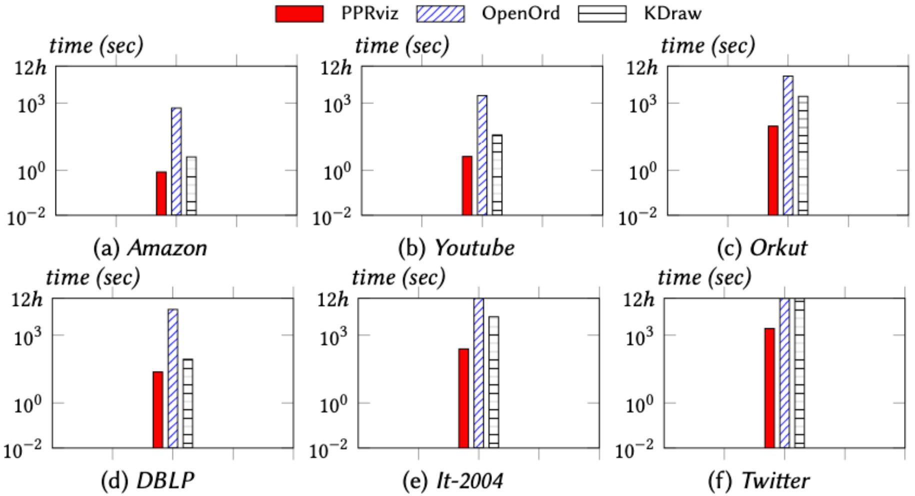
EXPERIMENTS: EFFICIENCY OF PPRVIZ

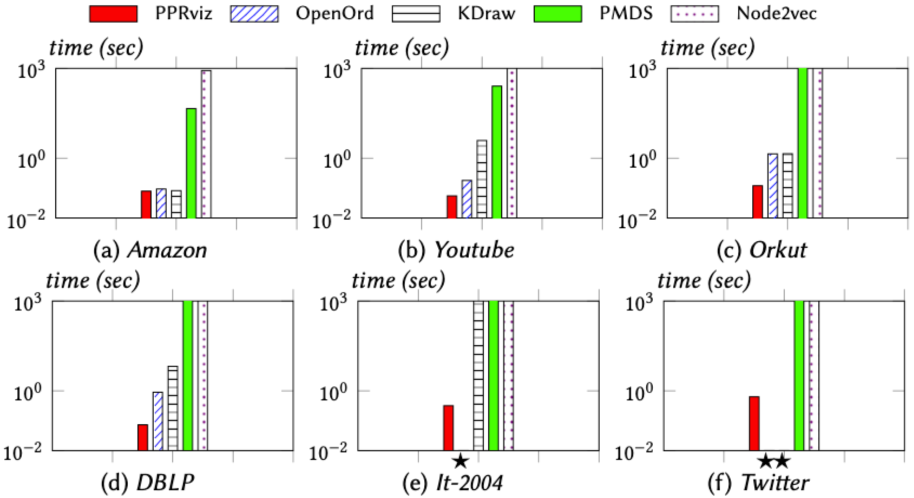
Preprocessing time:

- compute H and index of Tau-Push in PPRviz
- compute H in multi-level methods

Response time:

- visualize S in PPRviz and multi-level methods
- visualize G in single-level methods





CONCLUSION

- PPRviz: graph visualization solution
- PDist: PPR-based distance measure
- Tau-Push: efficient PDist approximation algorithm

SEATTLE
WA, USA
SIGMOD
PODS
2023

THANK YOU!

Q&A

NUS
National University
of Singapore

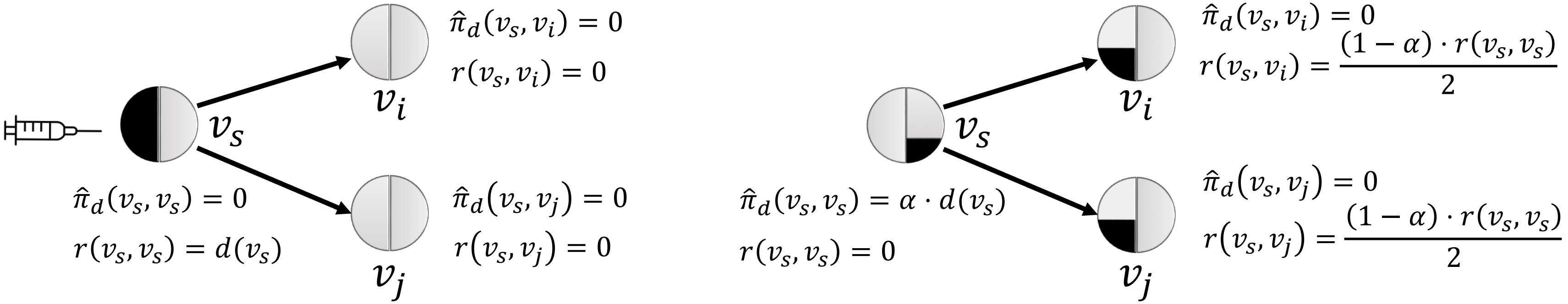
香港浸會大學
HONG KONG BAPTIST UNIVERSITY

SUSTech
Southern University
of Science and
Technology

BACKUP: TAU-PUSH FOR LEAF NODES

- Forward Push [a]
  - Deterministic version of RWR
  - Given a source v_s, each node v_i maintains
    - estimation $\hat{\pi}_d(v_s, v_i)$ and residue $r(v_s, v_i)$
  - Invariant:

$$\pi_d(v_s, v_t) = \hat{\pi}_d(v_s, v_t) + \sum_i \frac{1}{d(v_i)} \cdot r(v_s, v_i) \cdot \pi_d(v_i, v_t)$$



[a] Andersen, Reid, Fan Chung, and Kevin Lang. "Local graph partitioning using pagerank vectors." FOCS. 2006.

BACKUP: TAU-PUSH FOR LEAF NODES

- DPR-guided termination
  - Degree-normalized PageRank (DPR) for v_j:

$$\tau_j = \frac{1}{m} \cdot \sum_i \pi_d(v_i, v_j)$$

  - Given a source v_s and a target v_t, stop Forward Push when each

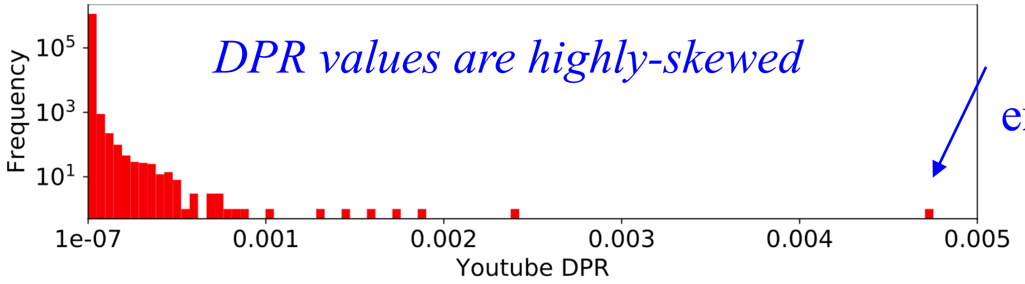
$$r(v_s, v_i) \leq \frac{\epsilon \cdot \delta}{m \cdot \tau_t}$$

  - $\hat{\pi}_d(v_s, v_t)$ is (ϵ, δ)-approximate, since

$$\pi_d(v_s, v_t) - \hat{\pi}_d(v_s, v_t) = \sum_i \frac{1}{d(v_i)} \cdot r(v_s, v_i) \cdot \pi_d(v_i, v_t) \leq \epsilon \cdot \delta$$

BACKUP: TAU-PUSH FOR LEAF NODES

- Refinement
  - Intuition:
    - only using Forward Push incurs redundant overhead
  - Backward Push [a]:
    - perform push operations from a target v_t in a reverse manner



For this v_t, the stop condition is extremely tough even if the estimations of others in S are good

[a] Peter Lofgren and Ashish Goel. Personalized pagerank to a target node. arXiv (2013).

BACKUP: TAU-PUSH FOR LEAF NODES

- Summary
  - DPR Computation and identify large-DPR v_t
  - Forward Push: estimate for most small-DPR v_t
  - Backward Push: estimate for large-DPR v_t

precompute and store as index

τ_1	0.01	τ_7	0.01
τ_2	0.03	τ_8	0.02
τ_3	0.02	τ_9	0.06
τ_4	0.01	τ_{10}	0.07
τ_5	0.04	τ_{11}	0.02
τ_6	0.02	…	…

$\tau = 0.05$