KAIST AI
Kim Jaechul Graduate School

ICDE 2022
Kuala Lumpur | Malaysia

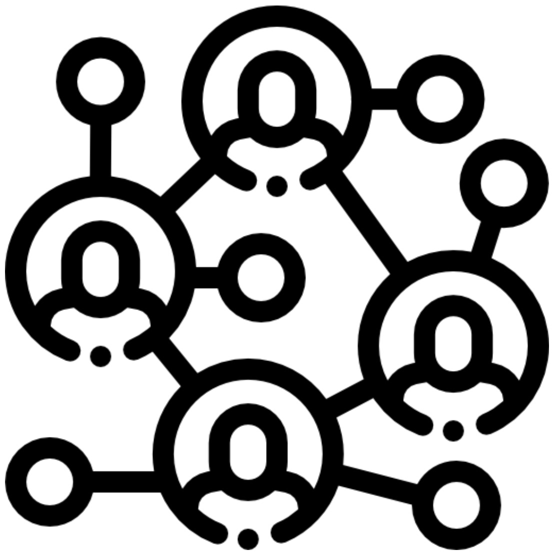
SLUGGER: Lossless Hierarchical Summarization of Massive Graphs

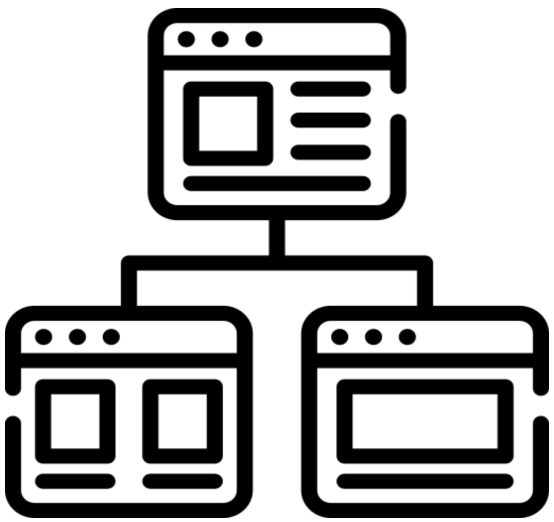
Kyuhan Lee*

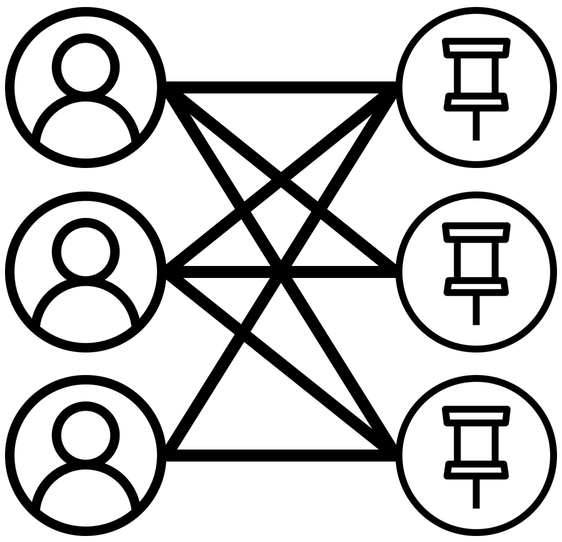
Jihoon Ko*

Kijung Shin

Graph: a Natural and Powerful Abstraction







Social Networks
with over 20B connections

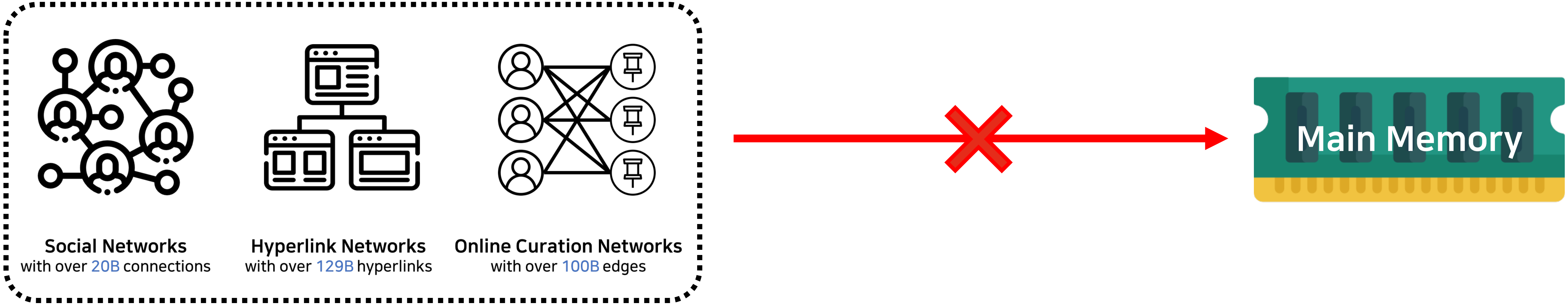
Hyperlink Networks
with over 129B hyperlinks

Online Curation Networks
with over 100B edges

Icon made by Freepik from www.flaticon.com

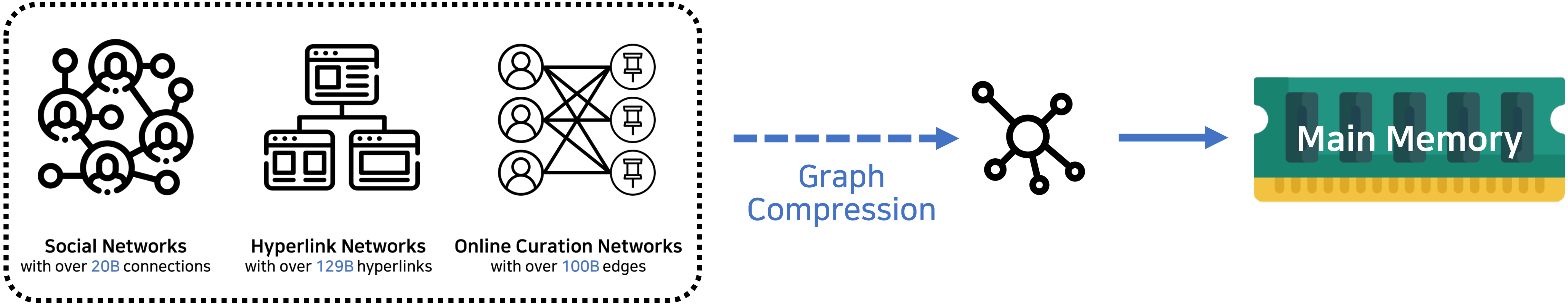
How to Store Large-scale Graphs

- Typical graph algorithms assume that the input graph fits in main memory
  - Large-scale graphs cannot fit in main memory
  - Graph analysis tools are inapplicable to those graphs



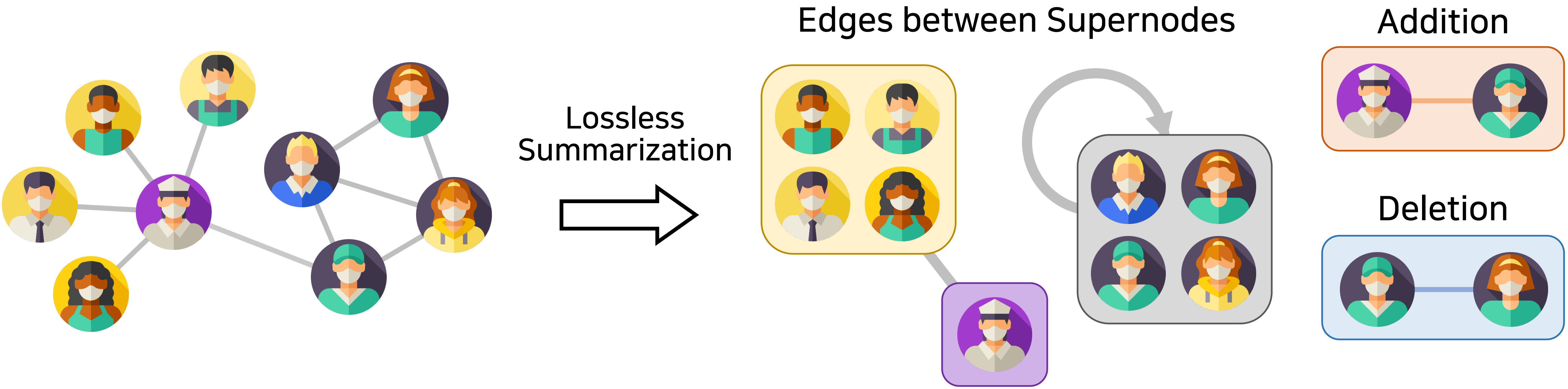
How to Store Large-scale Graphs

- Graph compression [BV04, NRS08, LT10, KKVF14] methods efficiently store the large-scale graphs



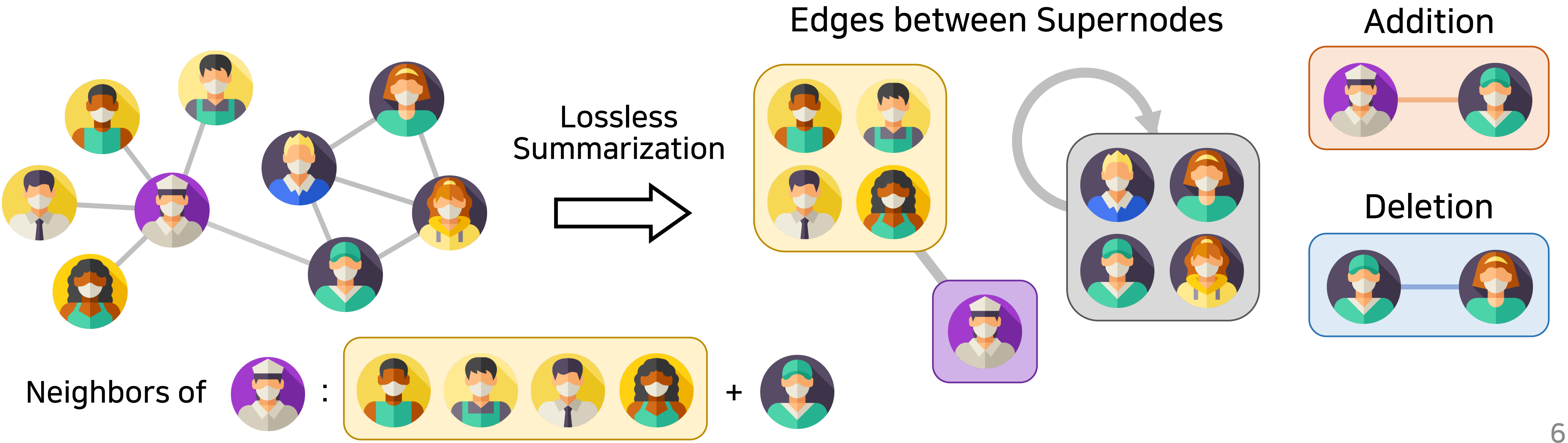
Lossless Graph Summarization

- Main Idea:
  - Nodes with similar connectivity are combined into a supernode so that
  - Connectivity can be encoded together to save bits



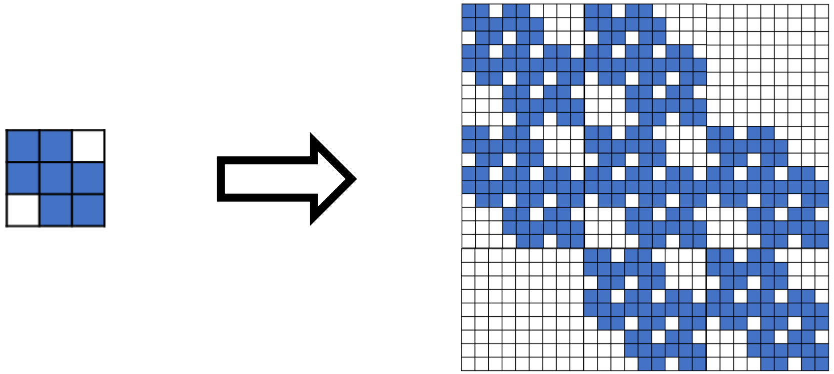
Merits of Graph Summarization

- Combinable
  - the outputs are also graphs [SGKR19, KKS20]
- Queryable
  - retrieving the neighborhood efficiently [SGKR19, KKS20]



Limitations of Graph Summarization

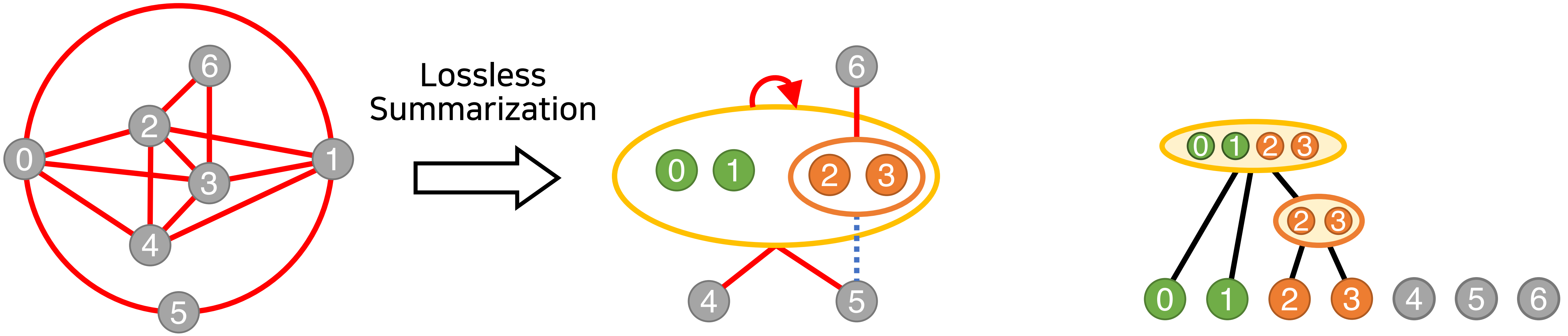
- Hierarchical structures are known to be pervasive
  - Web and biological networks are hierarchically organized [CB97, RB03]
  - Hierarchical structures have been exploited for algorithm design
    - Community Detection [GN02, SGMA07]
    - Realistic graph generation [LCKFG10]



Kronecker Graphs [LCKFG10]

- Graph summarization model cannot express and exploit hierarchy

Hierarchical Graph Summarization



Input Graph | Positive and Negative Edges | Hierarchical Trees of Supernodes

Our Solution:

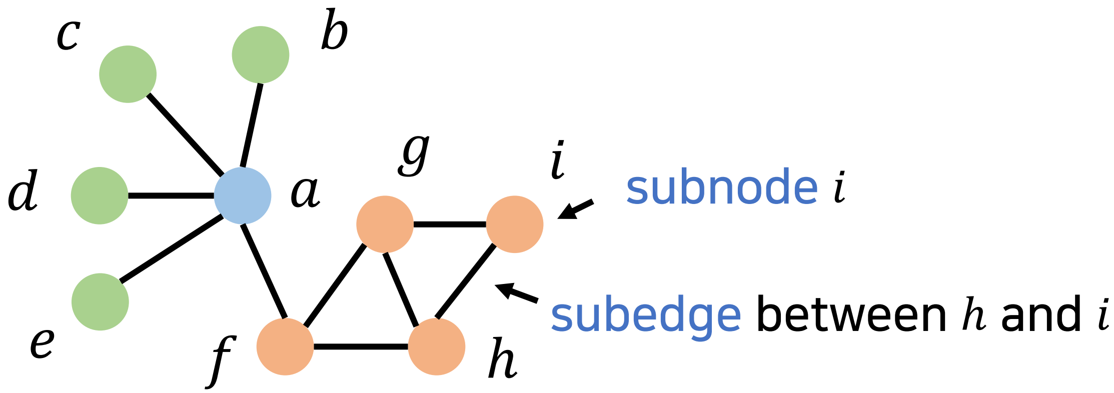
- Propose Hierarchical Graph Summarization Model
- Propose SLUGGER (Scalable Lossless Summarization of Graphs with Hierarchy), a fast and effective algorithm

Outline

- Proposed Model: Hierarchical Graph Summarization Model
- Proposed Algorithm: SLUGGER
- Experimental Results
- Conclusions

Graph

- An undirected graph $G = (V, E)$
  - V: the set of nodes / E: the set of edges
  - (u, v) or (v, u): the undirected edge between $u, v \in V$

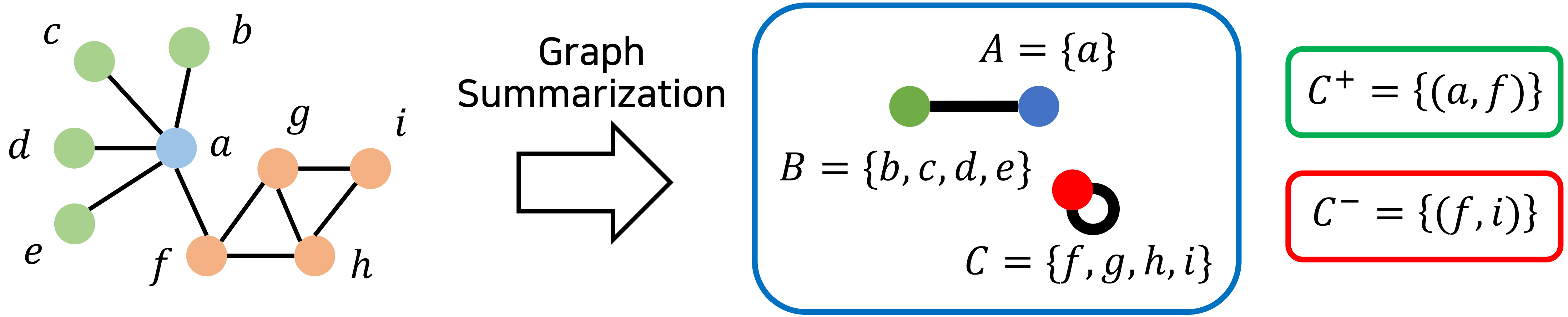


Input graph

$\mathbf{G} = (\boldsymbol{V}, \boldsymbol{E})$

Graph Summarization Model

- The graph summarization model [NRS08] consists of
  - Set P of edges between supernodes S
  - Set C^+ of positive subedges and C^- of negative subedges



Input graph $\mathbf{G} = (\boldsymbol{V}, \boldsymbol{E})$

Summary graph $\boldsymbol{G}^* = (\boldsymbol{S}, \boldsymbol{P})$

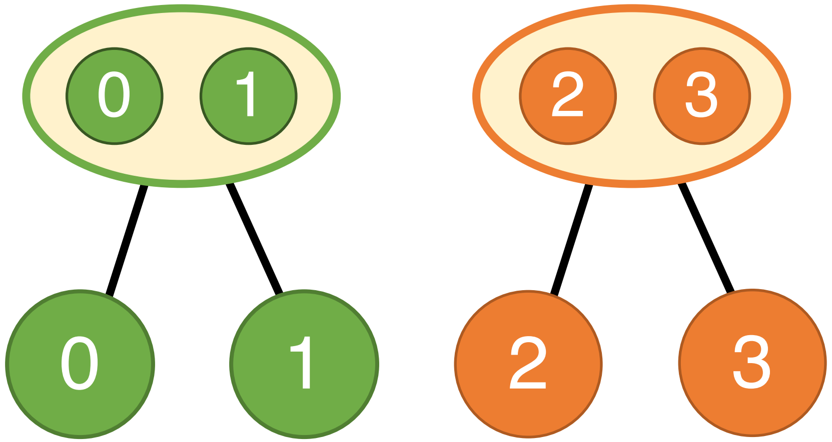
Edge corrections $(\boldsymbol{C}^+, \boldsymbol{C}^-)$

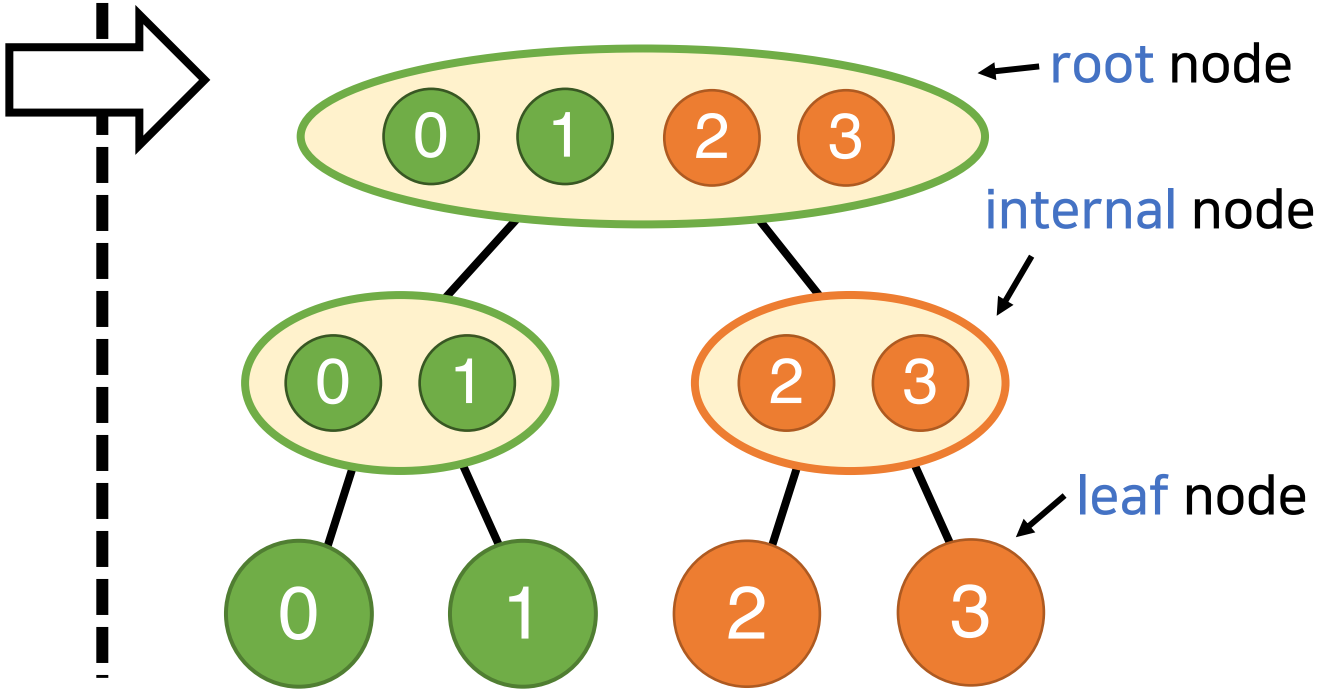
Hierarchical Graph Summarization Model

The main difference from the previous model: supernode

- The Previous Model
  - Supernodes should be disjoint

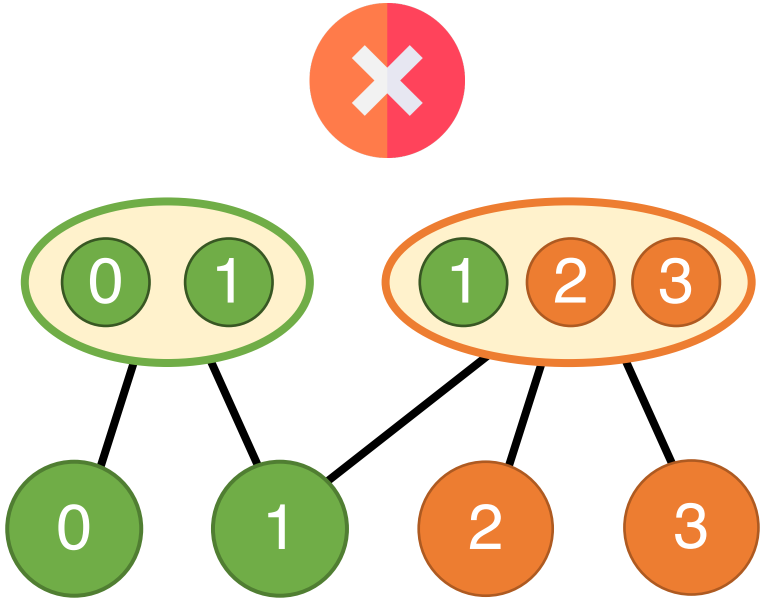
- Hierarchical Model (Proposed)
  - Each supernode may contain smaller supernodes





Hierarchical Graph Summarization Model

In both models, partially overlapping supernodes are not allowed



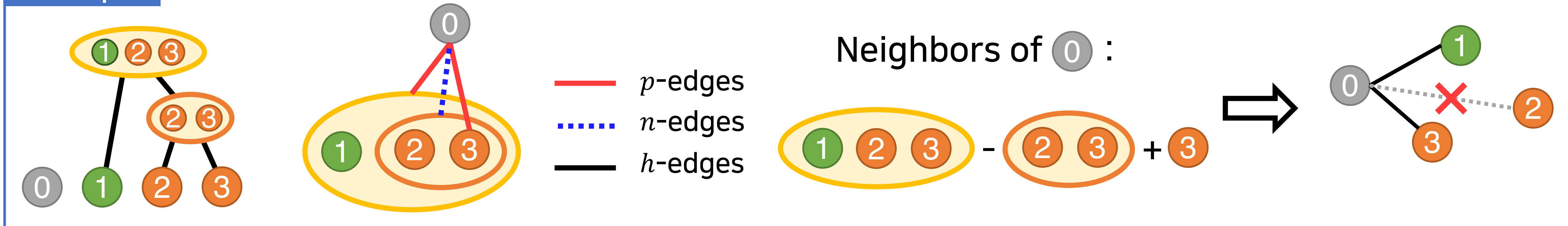
Hierarchical Graph Summarization Model

Parameters are also different from the previous model

- Parameters of the Previous Model
  - P for edges between supernodes
  - C^+ for positive edges and C^- for negative edges between subnodes

- Parameters of the Proposed Model
  - P^+ for positive edges (**p-edges**) between supernodes
  - P^- for negative edges (**n-edges**) between supernodes
  - H for hierarchy edges (**h-edges**)

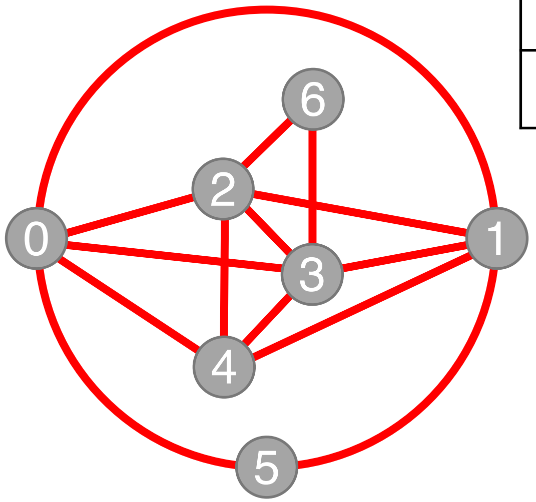
Example



Hierarchical Graph Summarization Model

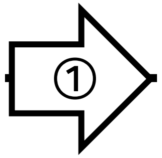
- Example: An undirected graph with 7 nodes and 14 edges

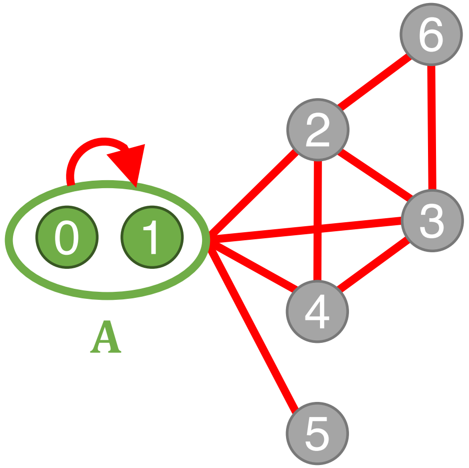
p-edges and n-edges



$\lvert H\rvert$	$\lvert P^+\rvert$	$\lvert P^-\rvert$
0	14	0

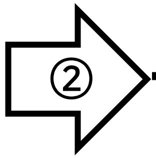
Merge 0 and 1



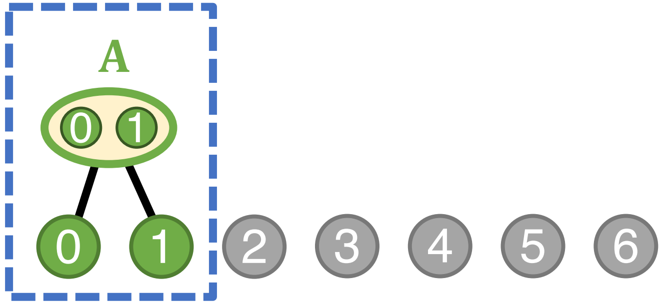


$\lvert H\rvert$	$\lvert P^+\rvert$	$\lvert P^-\rvert$
2	10	0

Merge 2 and 3



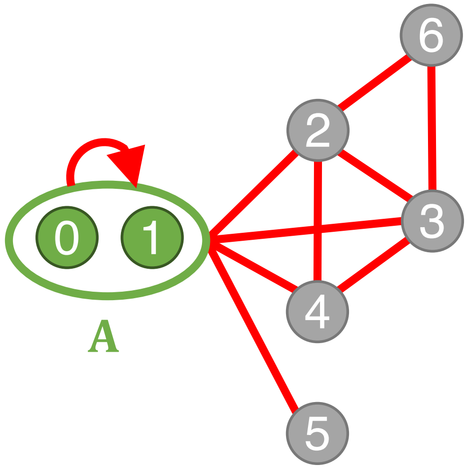
Hierarchy Trees of Supernodes



Hierarchical Graph Summarization Model

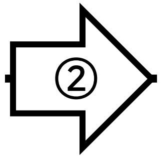
- Example: An undirected graph with 7 nodes and 14 edges

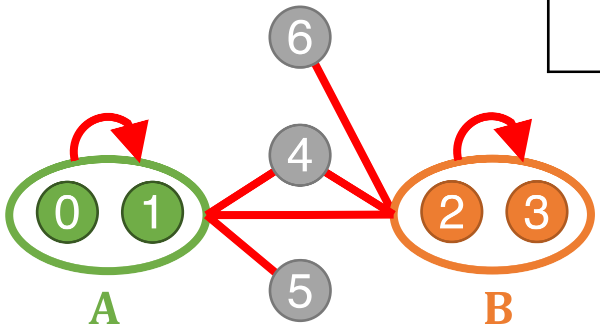
p-edges and n-edges



$\lvert H \rvert$	$\lvert P^+ \rvert$	$\lvert P^- \rvert$
2	10	0

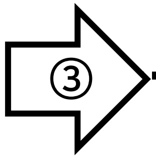
Merge 2 and 3



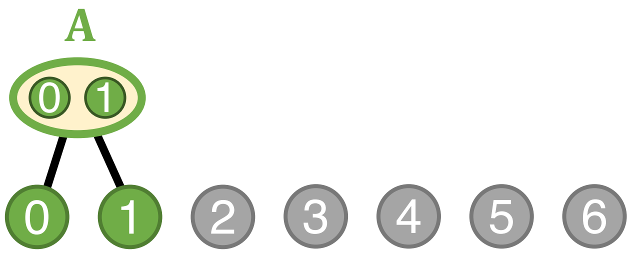


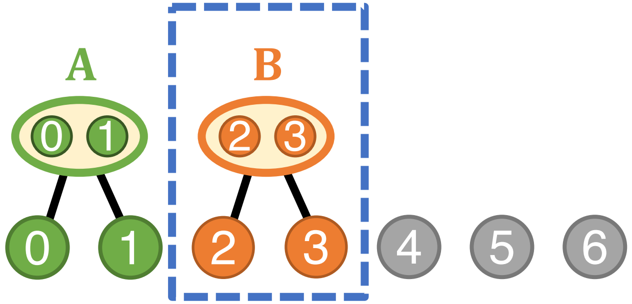
$\lvert H \rvert$	$\lvert P^+ \rvert$	$\lvert P^- \rvert$
4	7	0

Merge 0 1 and 2 3



Hierarchy Trees of Supernodes



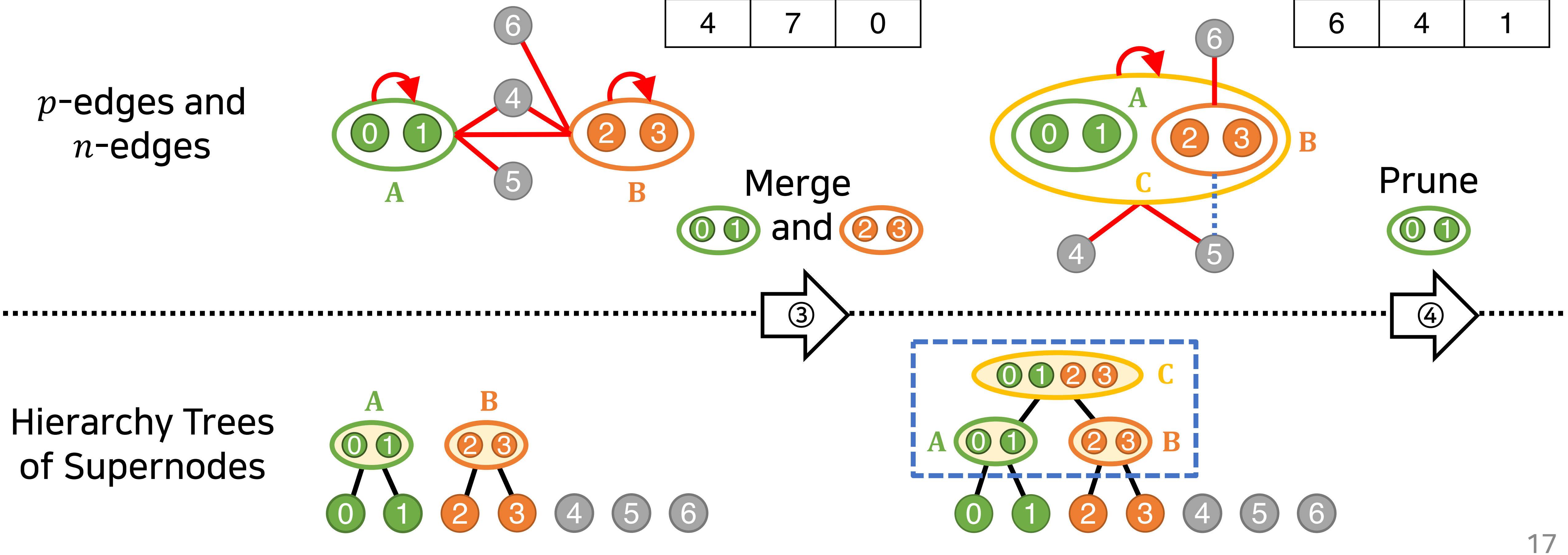


Hierarchical Graph Summarization Model

- Example: An undirected graph with 7 nodes and 14 edges

$\lvert H \rvert$	$\lvert P^{+} \rvert$	$\lvert P^{-} \rvert$
4	7	0

$\lvert H \rvert$	$\lvert P^{+} \rvert$	$\lvert P^{-} \rvert$
6	4	1

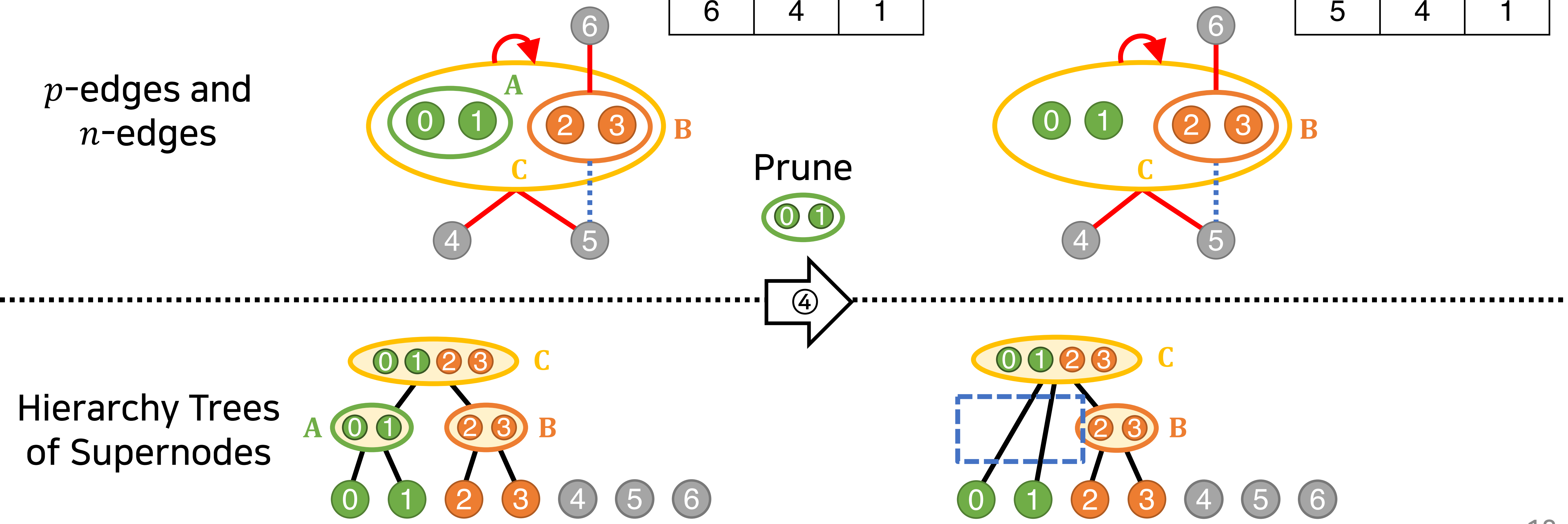


Hierarchical Graph Summarization Model

- Example: An undirected graph with 7 nodes and 14 edges

$\lvert H\rvert$	$\lvert P^{+}\rvert$	$\lvert P^{-}\rvert$
6	4	1

$\lvert H\rvert$	$\lvert P^{+}\rvert$	$\lvert P^{-}\rvert$
5	4	1



Problem Formulation

- **Given** an undirected graph G
- **Find** a summary graph $(\mathrm{S}, \mathrm{P}, \mathrm{C}^+, \mathrm{C}^-)$
- **To Minimize** the total count of edges $(|\mathrm{P}| + |\mathrm{C}^+| + |\mathrm{C}^-|)$

Lossless Graph Summarization

Problem Formulation

- **Given** an undirected graph G
- **Find** a summary graph $(\mathrm{S}, \mathrm{P}, \mathrm{C}^{+}, \mathrm{C}^{-})$
- **To Minimize** the total count of edges $(|\mathrm{P}| + |\mathrm{C}^{+}| + |\mathrm{C}^{-}|)$

Lossless Graph Summarization

Lossless Hierarchical Graph Summarization

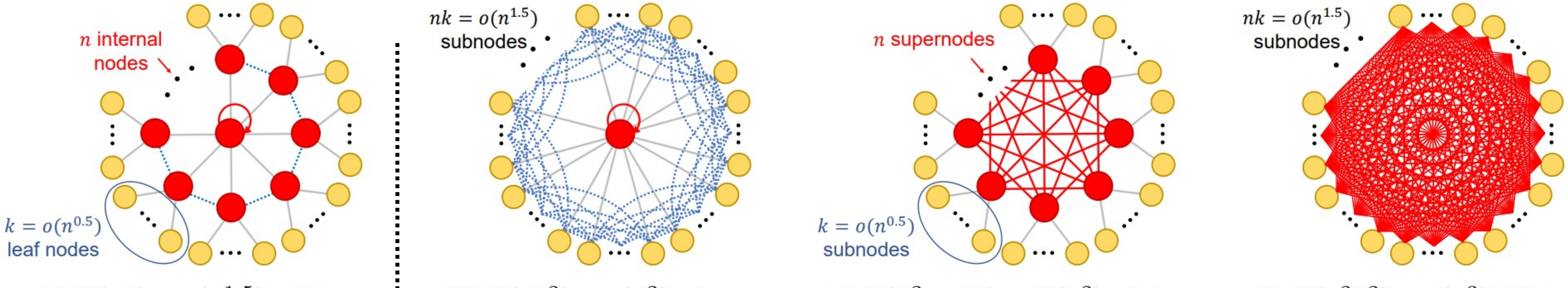
- **Given** an undirected graph G
- **Find** a hierarchical summary graph $(\mathrm{S}, \mathrm{P}^{+}, \mathrm{P}^{-}, \mathrm{H})$
- **To Minimize** the total count of edges $(|\mathrm{P}^{+}| + |\mathrm{P}^{-}| + |\mathrm{H}|)$

Merits of the Proposed Model

- Generalization of the previous model
  - Superedges in P -> p-edges between root nodes
  - Subedges in C^+/C^- -> p-edges and n-edges between singleton supernodes
- Strictly more concise than the previous model

Output Representation with Our Model

Possible Output Representations with the Previous Model



(a) $\Theta(nk) = o(n^{1.5})$ edges

(b) $\Theta(nk^2) = o(n^2)$ edges

(c) $\Theta(n^2 + nk) = \Theta(n^2)$ edges

(d) $\Theta(n^2k^2) = o(n^3)$ edges

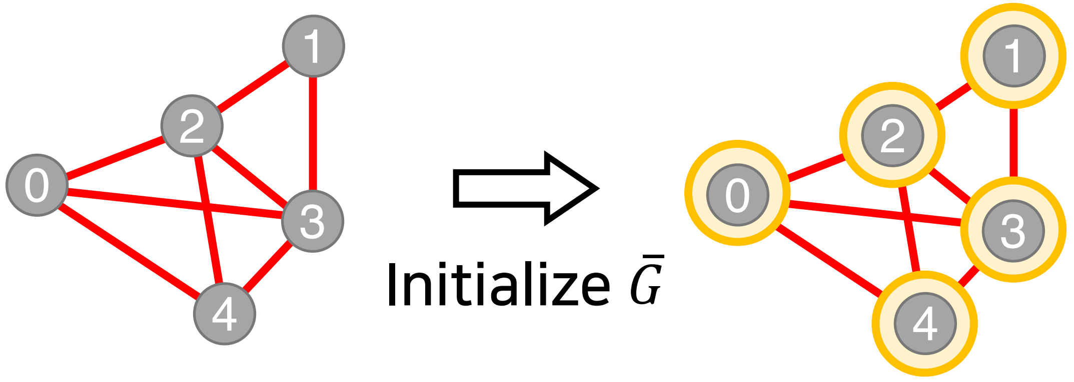
Outline

- Proposed Model: Hierarchical Graph Summarization Model
- Proposed Algorithm: SLUGGER
- Experimental Results
- Conclusions

Overview of SLUGGER

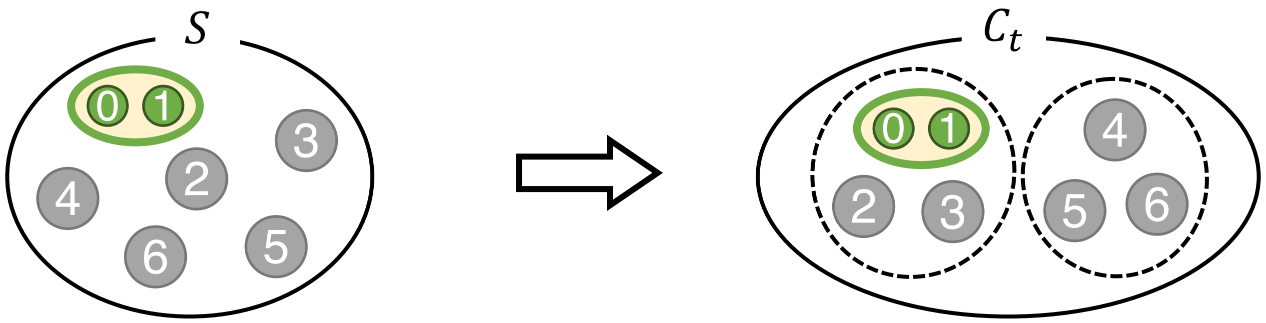
- **Input**
  - An input graph $G = (V, E)$
  - The number of iterations T
- **Output**: A hierarchical graph $\bar{G} = (S, P^+, P^-, H)$
- **Objective**: Finding $\bar{G}$ that minimizes the encoding cost $|P^+| + |P^-| + |H|$

- **Initialization**
  - S as set of singleton supernodes
  - P^+ as set of edges between the singleton supernodes
  - P^- and H as empty sets

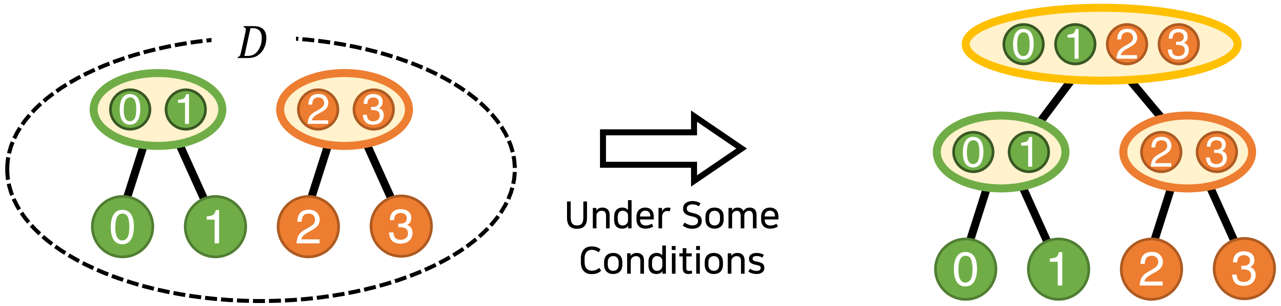


Overview of SLUGGER

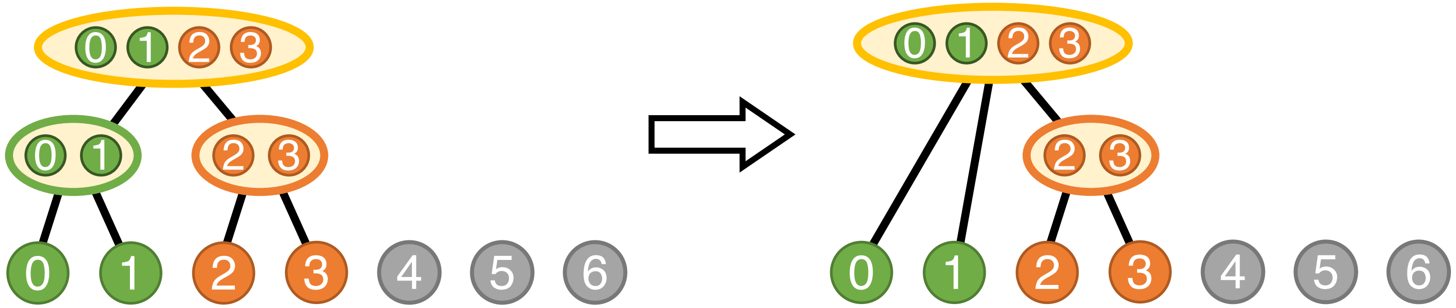
- Candidate Generation Step

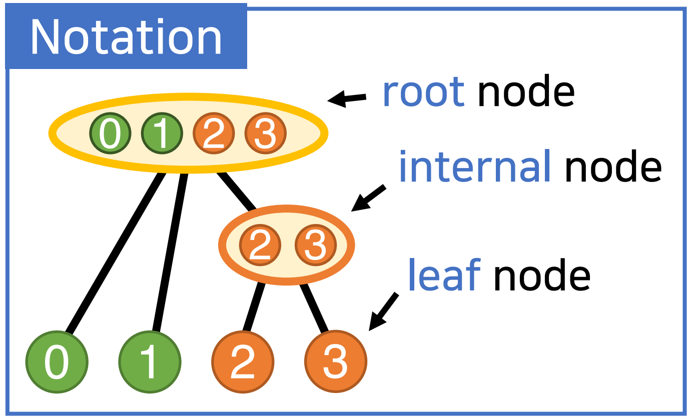


- Merging Step



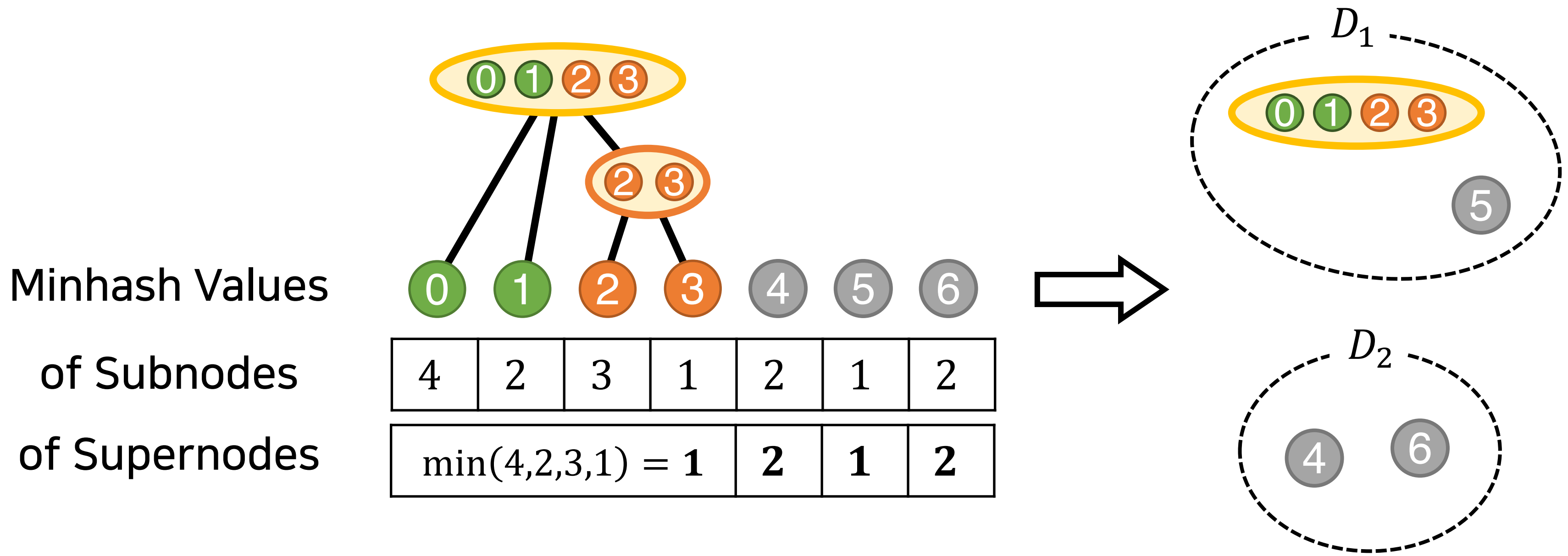
- Pruning Step





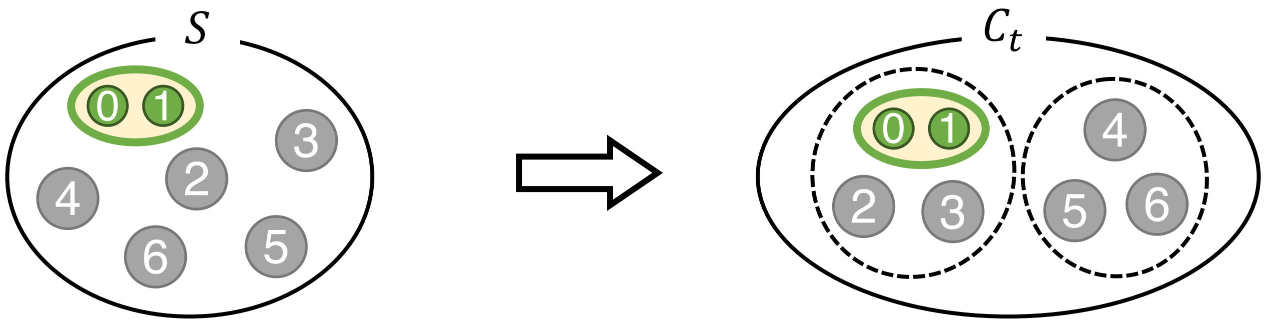
Candidate Generation Step

- SLUGGER divides root nodes into candidate sets
- For rapid and effective search, candidate sets should
  - be small
  - contain nodes with similar connectivity
- Our strategy: group root nodes as a candidate set using min-hashing

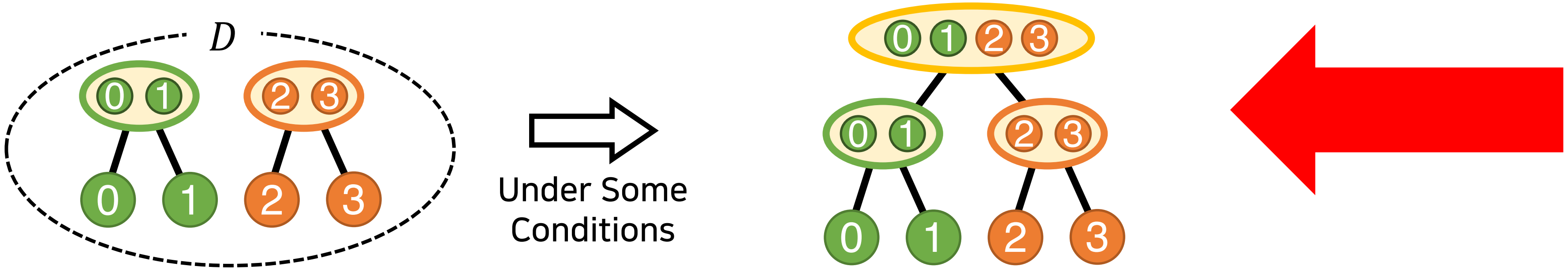


Overview of SLUGGER

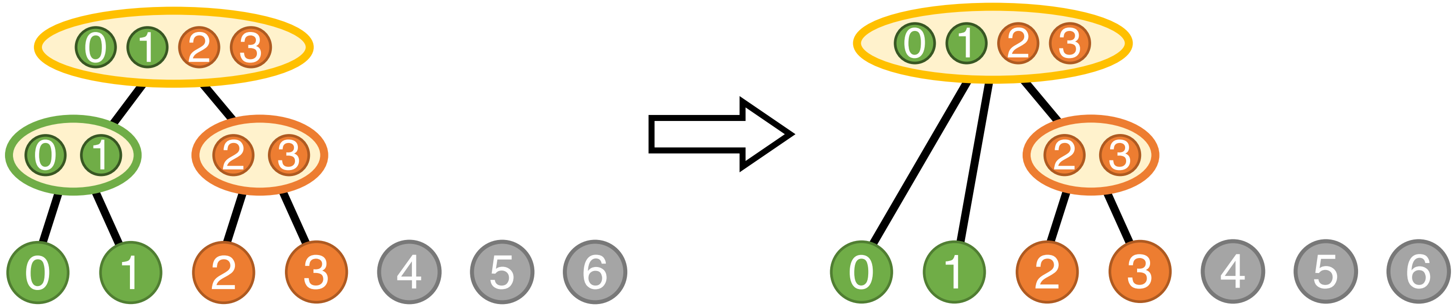
- Candidate Generation Step



- Merging Step



- Pruning Step



Merging Step

- SLUGGER greedily repeats merging root nodes and updating the encoding
- In each candidate set D determined at the previous step,
  - Repeat
    - Select a random root node A
    - Choose B that maximizes the saving of the encoding cost
    - If saving > $\theta(t)$, merge A and B and update the encoding

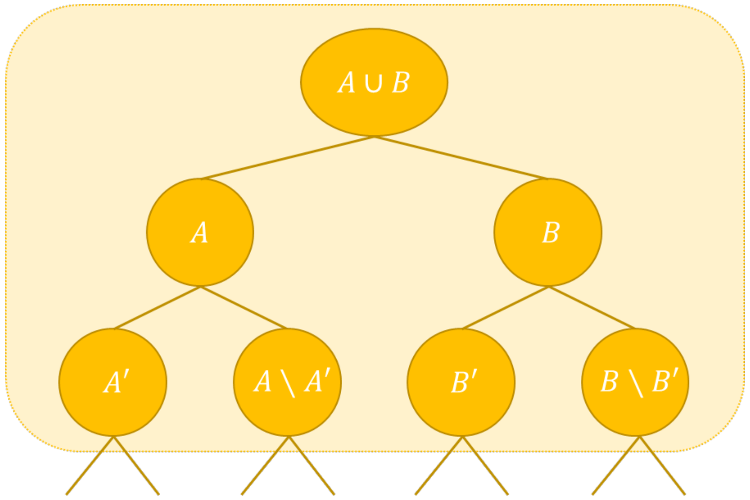
Details

$$Saving(A, B) \coloneqq 1 - \frac{(\text{encoding cost for } A \cup B \text{ after merging})}{(\text{encoding cost for } A \text{ and } B \text{ before merging})}$$

$$\theta(t) \coloneqq \begin{cases} (1+t)^{-1} & \text{if } t < T \\ 0 & \text{if } t = T \end{cases}$$

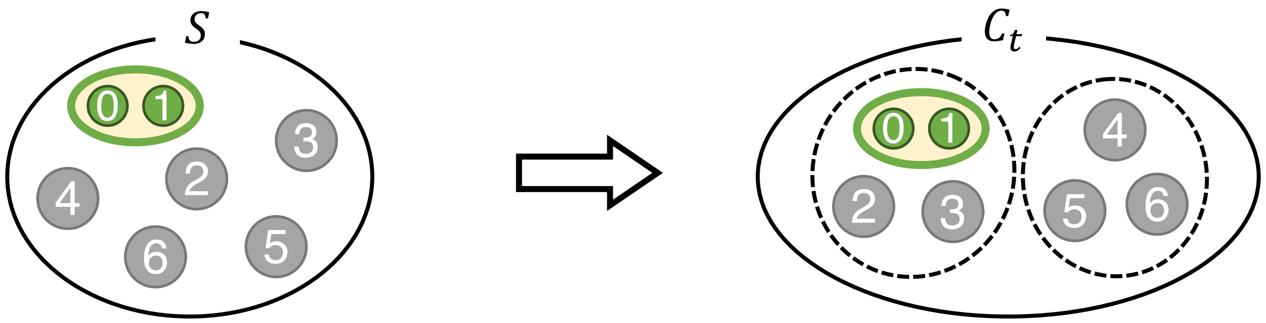
How does SLUGGER merge two root nodes

- Problem: Exactly minimizing the encoding cost is computationally expensive
- Idea: To focus only on a small number of supernodes
  - (a) merged nodes
  - (b) neighbors
  - (c) direct children of (a) and (b) in the hierachy
- Only a constant number of possibilities exist and they can be searched exhaustively

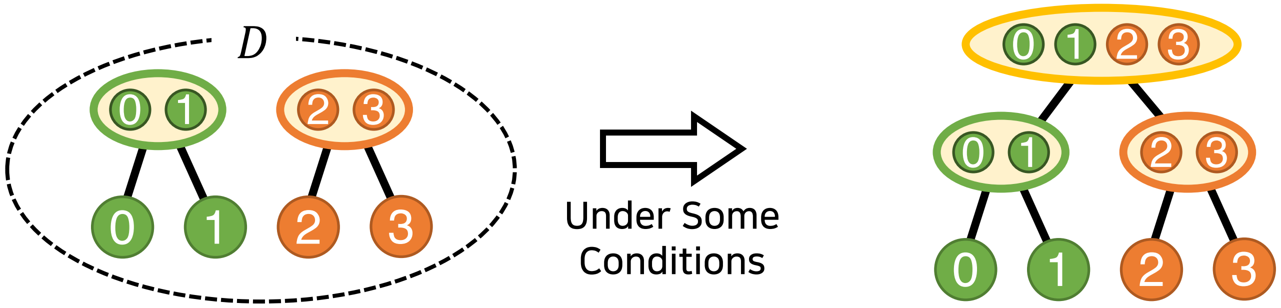


Overview of SLUGGER

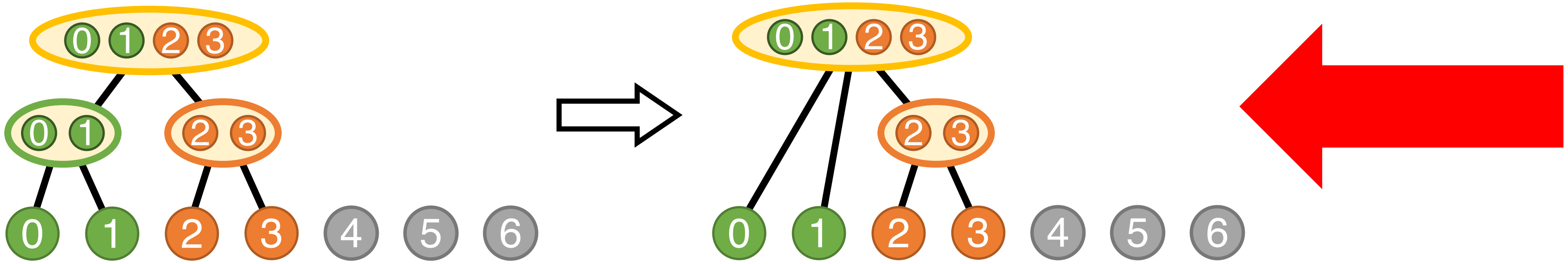
- Candidate Generation Step



- Merging Step



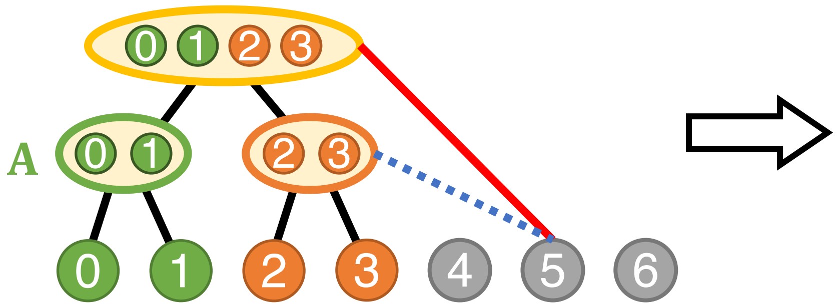
- Pruning Step

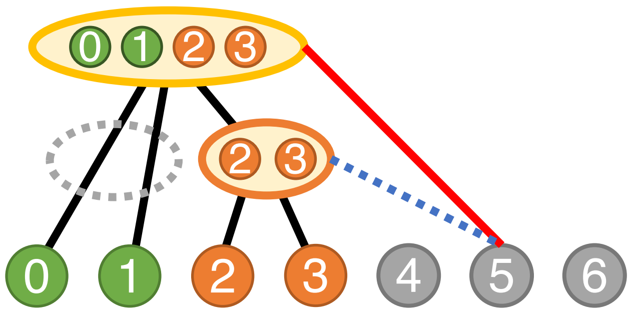


Details # Pruning Step

- SLUGGER further reduces the encoding cost by removing unnecessary supernodes

- **(Step 1)** Remove a non-leaf node that is not incident to any p or n-edge
  - $|H|$ and the total encoding cost decrease by 1



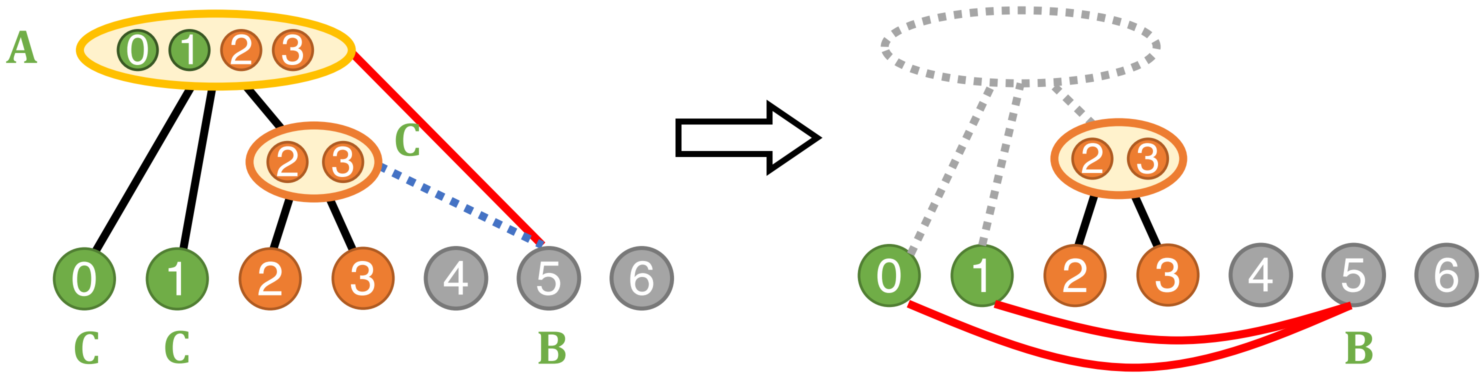


$\|H\|$	$\|P^+\|$	$\|P^-\|$	Total
6 -> 5	1	1	8 -> 7

Details Pruning Step

- SLUGGER further reduces the encoding cost by removing unnecessary supernodes

- **(Step 2)** Remove a root node A with only one incident non-loop p or n-edge (A, B)
  - Add edges of the same type or remove edges of different types or between B and all direct children of A

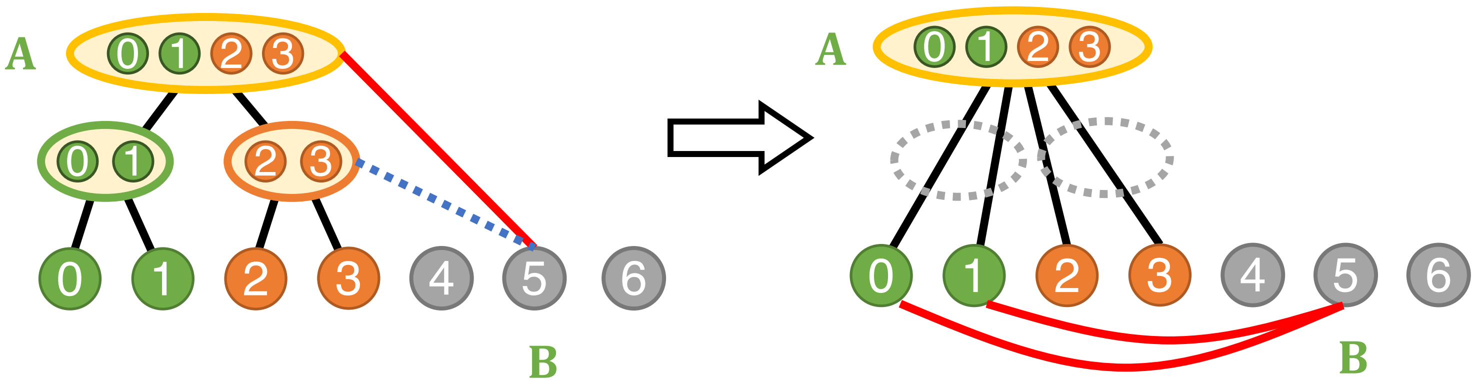


$\|H\|$	$\|P^+\|$	$\|P^-\|$	Total
5 -> 2	1 -> 2	1 -> 0	6 -> 4

Details Pruning Step

- SLUGGER further reduces the encoding cost by removing unnecessary supernodes

- **(Step 3)** Partially use the encoding of SWeG
  - SWeG does not allow p-edges and n-edges incident to internal nodes
  - So, it may make more supernodes be pruned



$\vert H\vert$	$\vert P^{+}\vert$	$\vert P^{-}\vert$	Total
6 -> 4	1 -> 2	1 -> 0	8 -> 6

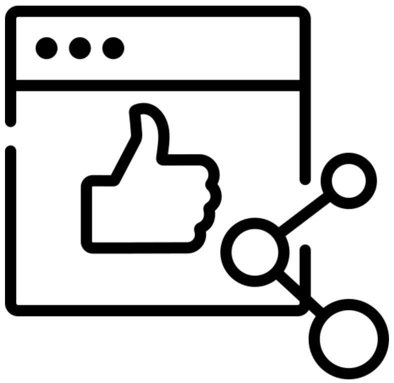
Outline

- Proposed Model: Hierarchical Graph Summarization Model
- Proposed Algorithm: SLUGGER
- Experimental Results
- Conclusions

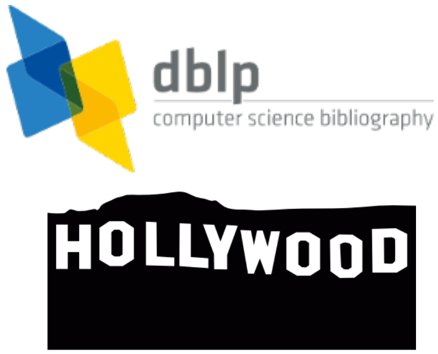
Experimental Settings

- **Datasets:** 16 Real-world Graphs (up to 0.8B edges)

Hyperlinks

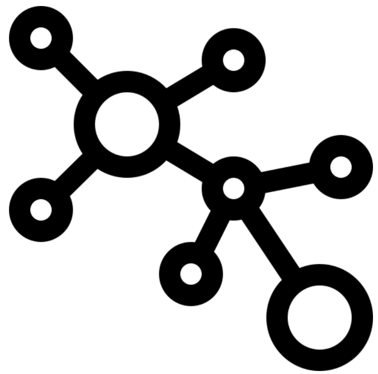


Social



Collaboration

Email

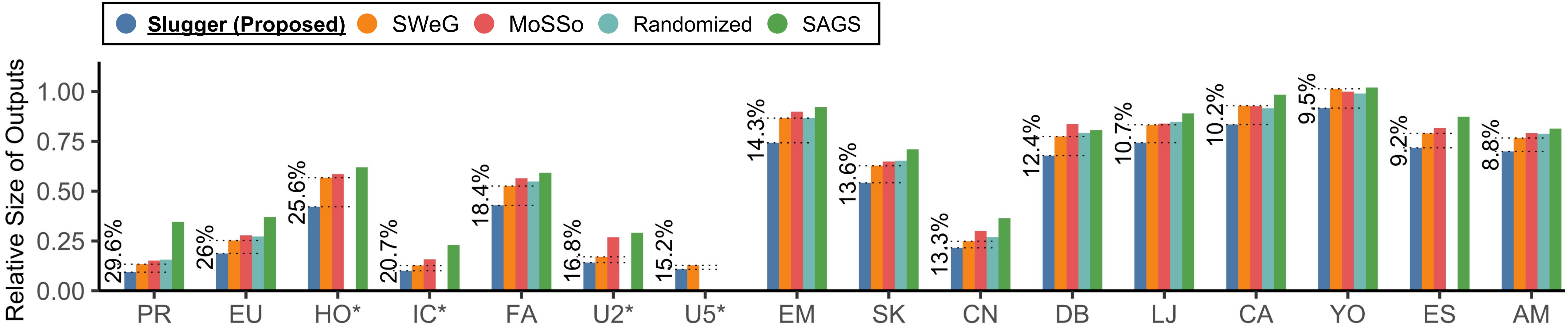


And others!

- **Competitors:** Lossless graph summarization algorithms
  - Randomized [NSR08], SAGS [KNL15], SWeG [SGKR19], MoSSo [KKS20]

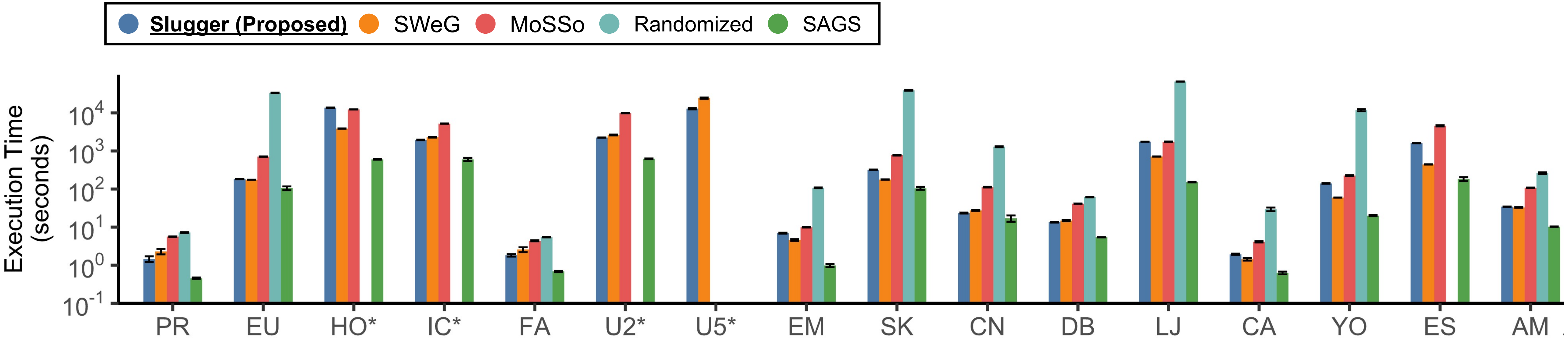
Results: Compactness of SLUGGER

- SLUGGER gave most concise outputs in all 16 datasets
  - up to 29.6% and on average 13.5%



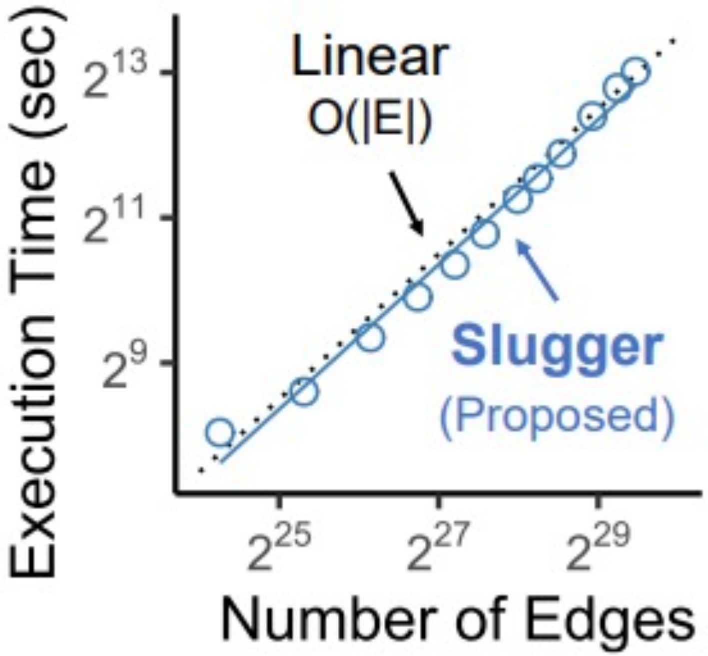
Results: Speed of SLUGGER

- SLUGGER was as fast as SWeG (strongest competitor)
  - SAGS was fastest, but its output was least concise



Results: Scalability of SLUGGER

- SLUGGER scaled linearly with the size of the graph
- SLUGGER successfully summarized the largest real-world graph with about 0.8B edges



Details

Additional Experiments

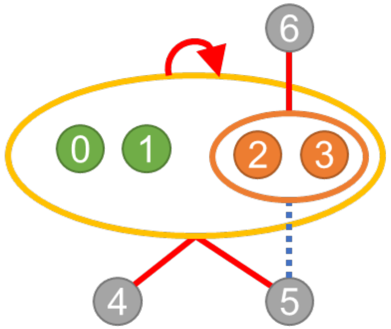
- Effects of Iterations
  - About 40 iterations are enough
- Effects of Pruning
  - Each substep is effective
- Effects of Bounding the Height of Hierarchical Trees
  - Height can be upper bounded for rapid query processing at the expense of conciseness

Outline

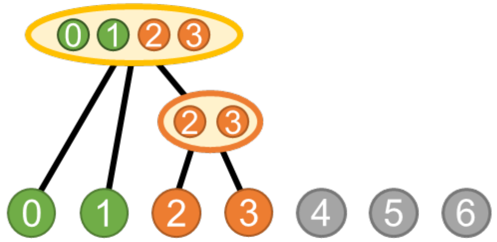
- Proposed Model: Hierarchical Graph Summarization Model
- Proposed Algorithm: SLUGGER
- Experimental Results
- Conclusions

Conclusions

- Novel Graph Representation Model

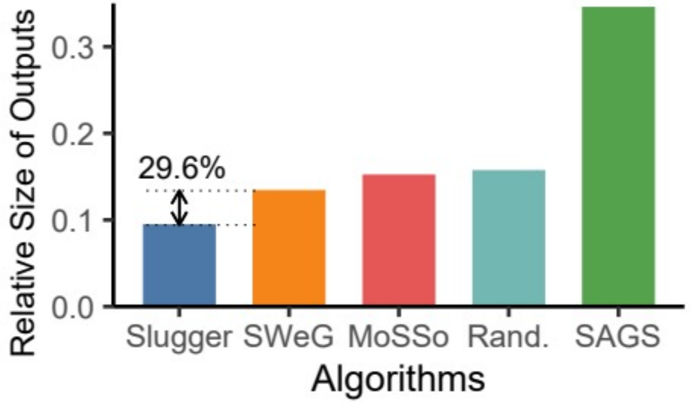


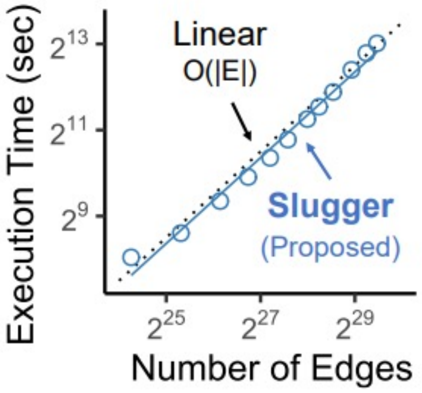
Positive and Negative Edges



Hierarchical Trees of Supernodes

- Fast and Effective Algorithm





The code and datasets used in the paper are available at https://github.com/KyuhanLee/slugger

References

[BV04] Boldi and Vigna, “The webgraph framework i: compression techniques,” in WWW, 2004

[NRS08] Navlakha et al., “Graph summarization with bounded error,” in SIGMOD, 2008

[LT10] LeFevre and Terzi, “Grass: Graph structure summarization,” in SDM, 2010.

[KKVF14] Koutra et al., “Vog: Summarizing and understanding large graphs,” in SDM, 2014

[SGKR19] Shin et al., “Sweg: Lossless and lossy summarization of web-scale graphs,” in WWW, 2019

[KKS20] Ko et al., “Incremental Lossless Graph Summarization”, in KDD, 2020

[CB97] Crovella and Bestavros, “Self-similarity in World Wide Web traffic: evidence and possible causes”, IEEE /ACM Transactions on Networking, 5(6):835–846, 1997

[RB03] Ravasz and Barabasi, “Hierarchical organization in complex networks”, Physical Review E, 67(2):026112, 2003

[GN02] Girvan and Newman, “Community structure in social and biological networks,” PNAS, vol. 99, no. 12, pp. 7821–7826, 2002

[SGMA07] Sales-Pardo et al., “Extracting the hierarchical organization of complex systems,” PNAS, vol. 104, no. 39, pp. 15 224–15 229, 2007

[LCKFG10] Leskovec et al., “Kronecker graphs: an approach to modeling networks.” JMLR, vol. 11, no. 2, 2010

KAIST AI
Kim Jaechul Graduate School

ICDE 2022
Kuala Lumpur | Malaysia

SLUGGER: Lossless Hierarchical Summarization of Massive Graphs


Kyuhan Lee*


Jihoon Ko*


Kijung Shin