Appendix

0.1 Formal Definition of Precision

To evaluate the accuracy of pattern summarization, we use **precision**(P) to evaluate how accurately the summarization process groups nodes according to their true labels and encodings. It is defined based on two key criteria:

- **Label Precision (P_L)**: The proportion of nodes correctly grouped under the same summarization label as their ground truth label.
- **Encoding Precision (P_E)**: The proportion of nodes correctly assigned to a summarization structure matching their ground truth encoding.
- **Overall Precision (P)**: The proportion of nodes that are simultaneously correctly grouped and encoded.

Formally, given a graph $G = (V, E)$, let:

- y_i^L be the ground truth label of node i
- $\hat{y}_i^L$ be the predicted label after summarization
- y_i^E be the ground truth encoding of node i
- $\hat{y}_i^E$ be the predicted encoding after summarization
- $\mathbb{1}(\cdot)$ be the indicator function, returning 1 if the condition is true and 0 otherwise

Then, the precision metrics are:

$$P_L = \frac{1}{|V|} \sum_{i \in V} \mathbb{1}(\hat{y}_i^L = y_i^L)$$

$$P_E = \frac{1}{|V|} \sum_{i \in V} \mathbb{1}(\hat{y}_i^E = y_i^E)$$

$$P = \frac{1}{|V|} \sum_{i \in V} \mathbb{1}(\hat{y}_i^L = y_i^L \wedge \hat{y}_i^E = y_i^E)$$

where P_L measures how well the summarization maintains node groupings, P_E measures the accuracy of encoding assignments, and P evaluates the joint correctness of both aspects.

A higher precision indicates that the summarization more accurately preserves the original graph structure.

0.2 Runtime Analysis

Figure 1 illustrates the running time of all techniques in Sec. 5.2 on different-sized datasets (number of nodes). They can be roughly divided into two groups. One contains our two techniques (GRD and RDM) and two simulation algorithms (Evolutionary Reordering and MinLA). The else contains the other algorithms. In comparison to two simulation methods, the running time of our two techniques is reasonable. However, compared to others, our running time is two orders of magnitude longer. It is because the pattern-aware summarization requires merging all 2-hops node pairs. It is time-consuming ($O(d_{av}^3(d_{av} + logn + logd_{av}))$), where d_{av} is the average node degree). Using the Randomized summarization reduces the time complexity ($O(d_{av}^3)$), but it is still time-consuming.

We leave it as a future work that may be solved by – 1) using locality sensitive hashing (LSH) to accelerate the neighborhood searching; 2) using parallel computing to calculate cost reductions of different node pairs simultaneously; and 3) using a progressive pattern cost computation, which avoids recalculating the cost of a super-node in each merge.

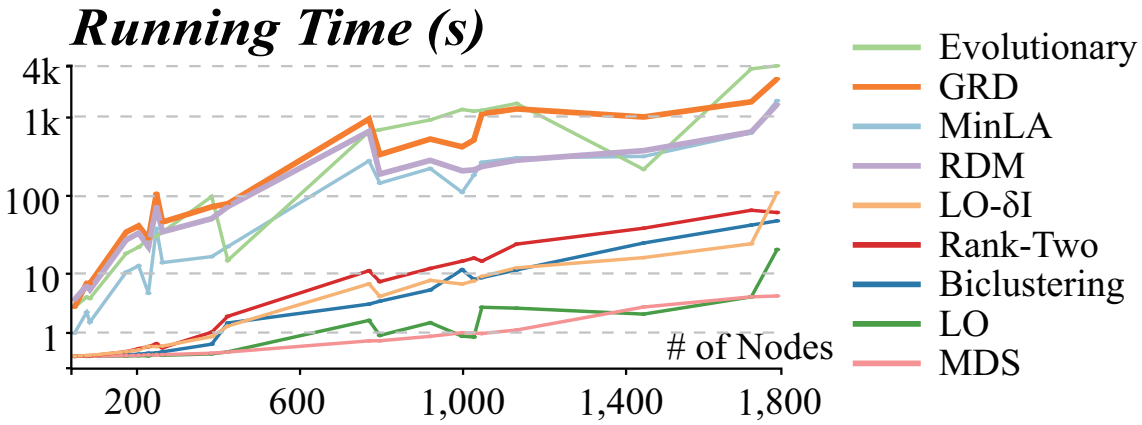

Figure 1: The running time of nine algorithms on twenty datasets. A bi-symmetric log scale is employed on the y-axis.

0.3 Dataset Sources

The datasets used in our experiments originate from publicly available sources:

- **chesapeake, bio-grid-mouse, bio-grid-plant**: Retrieved from the Network Repository [1].
- **everglades**: Ecological network data from [2, 3].
- **lesmis, jazz, visbrazil, netscience, dwt_419, price_1000**: Available from [4].
- **radoslaw**: Email communication network from [5].
- **sch**: Student interaction network from [6, 7].

- **econ-wm2**: Economic network data from [8].
- **Caltech36**: Social network dataset from [9].
- **asoiaf, petster-hamster, wiki_edit_eu, wiki_talk_br**: Available from the KONECT dataset repository [10], with additional data on Wikipedia messages from [11].
- **bn-mouse, bn-fly**: Brain network datasets from [12].

For further details, these datasets can be accessed through their respective repositories and original publications.

0.4 Qualitative Results for Sec. 5.2

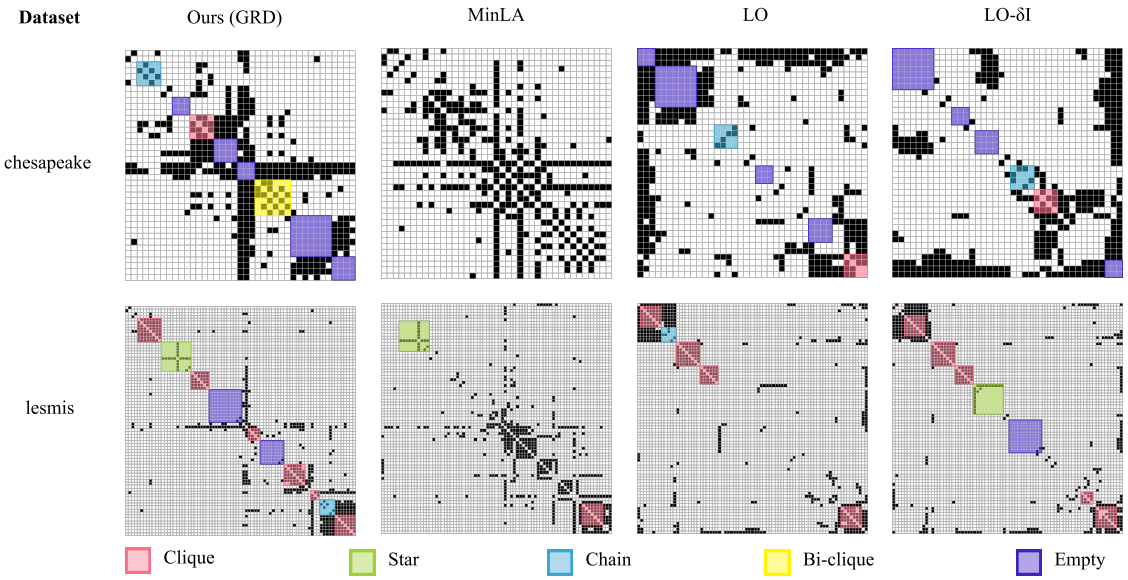

Figure 2: Matrices of chesapeake graph (39 nodes and 170 edges) and lesmis graph (77 nodes and 254 edges) reordered by our approach (GRD) and the other three reordering algorithms (MinLA [13], LO [14], and LO-δ_I [15, 16]). In each matrix, we highlight 5 types of salient patterns identified by the graph summarization used by GRD.

Bibliography

[1] Rossi R A, Ahmed N K. The network data repository with interactive graph analytics and visualization. In *AAAI*, 2015, p. 4292–4293.

[2] Melián C J, Bascompte J. Food web cohesion. *Ecology*, 2004, 85(2):352–358.

[3] Ulanowicz R E, DeAngelis D L. Network analysis of trophic dynamics in south florida ecosystems. *FY97: The Florida Bay Ecosystem*, 1998, pp. 20688–20038.

[4] Davis T A, Hu Y. The university of Florida sparse matrix collection. *ACM Transactions on Mathematical Software*, dec 2011, 38(1):25.

[5] Michalski R, Palus S, Kazienko P. Matching organizational structure and social network extracted from email communication. In *Lecture Notes in Business Information Processing*, 2011, pp. 197–206.

[6] Stehlé J, Voirin N, Barrat A, Cattuto C, Isella L, Pinton J, Quaggiotto M, Van den Broeck W, Régis C, Lina B, Vanhems P. High-resolution measurements of face-to-face contact patterns in a primary school. *PLOS ONE*, Aug 2011, 6(8):e23176.

[7] Gemmetto V, Barrat A, Cattuto C. Mitigation of infectious disease at school: targeted class closure vs school closure. *BMC Infectious Diseases*, December 2014, 14(1):695.

[8] Duff I S, Grimes R G, Lewis J G. Users' guide for the Harwell-Boeing sparse matrix collection (Release I). *Rutherford Appleton Laboratory Technical Reports*, 12 1992.

[9] Traud A L, Mucha P J, Porter M A. Social structure of Facebook networks. *Physica A*, Aug 2012, 391(16):4165–4180.

[10] Kunegis J. Konect: the koblenz network collection. In *Proceedings of International Conference on World Wide Web*, 2013, pp. 1343–1350.

[11] Sun J, Kunegis J, Staab S. Predicting user roles in social networks using transfer learning with feature transformation. In *Proceedings of IEEE ICDMW*, 2016, pp. 128–135.

[12] Amunts K, Lepage C, Borgeat L, Mohlberg H, Dickscheid T, Rousseau M É, Bludau S, Bazin P L, Lewis L B, Oros-Peusquens A M, Shah N J, Lippert T, Zilles K, Evans A C. BigBrain: An ultrahigh-resolution 3D human brain model. *Science*, 2013, 340(6139):1472–1475.

[13] Rodriguez-Tello E, Hao J, Torres-Jimenez J. An effective two-stage simulated annealing algorithm for the minimum linear arrangement problem. *Computers and Operations Research*, 2008, 35(10):3331–3346.

[14] Bar-Joseph Z, Gifford D K, Jaakkola T S. Fast optimal leaf ordering for hierarchical clustering. In *Proceedings of International Conference on Intelligent Systems for Molecular Biology*, 2001, pp. 22–29.

[15] Beusekom N, Meulemans W, Speckmann B. Simultaneous matrix orderings for graph collections. *IEEE Transactions on Visualization and Computer Graphics*, 2022, 28(1):1–10.

[16] Reorder.js. https://github.com/jdfekete/reorder.js/. Accessed: 2022-08-19.