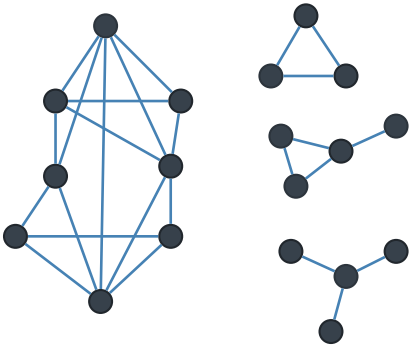
Local Fragments, Global Gains: Subgraph Counting using Graph Neural Networks

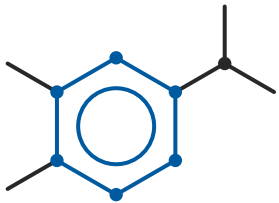
Shubhajit Roy, Shrutimoy Das, Binita Maity, Anant Kumar, Anirban Dasgupta

Indian Institute of Technology Gandhinagar



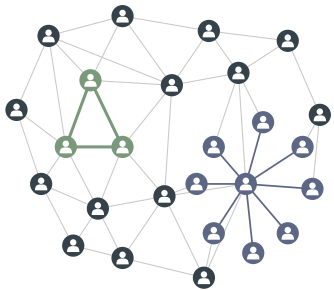
Subgraph Counting Reveals Functional Properties

Computational Biology



Identifying protein binding motifs and rings.

Social Network Analysis

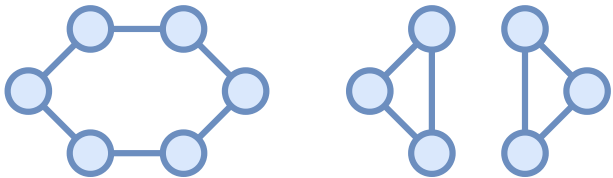


Understanding community formation (triangles) and information propagation (stars).

Why it matters?

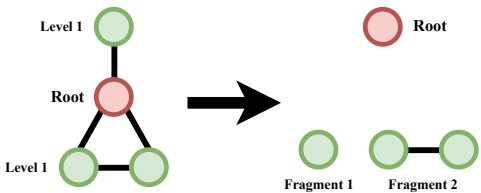
Accurate counting is a critical measure of GNN expressive power. Current methods struggle to distinguish similar graphs with different subgraph counts.

The Expressivity Bottleneck: Why Standard MPNNs Fail



- Standard Message Passing Neural Networks (MPNNs) are equivalent to the 1-dimensional Weisfeiler-Leman test.
- They cannot distinguish between distinct structures if node information is similar.

Core Insight: Decomposition via Fragmentation

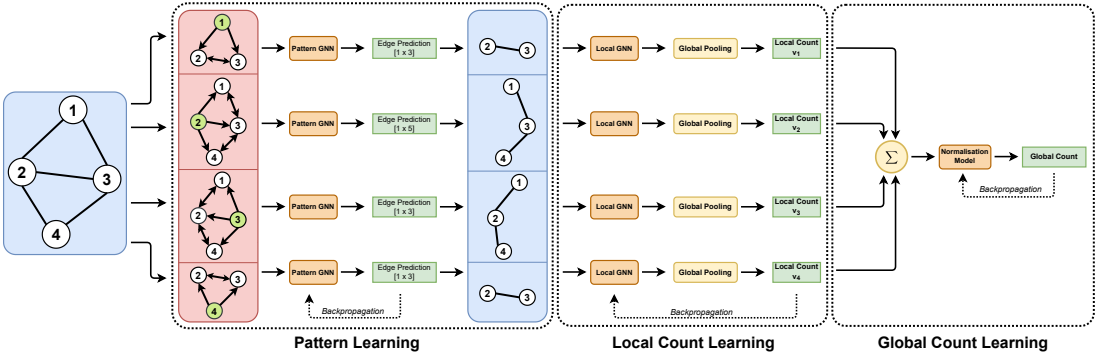


- **Concept:** Decompose complex subgraphs into simple "fragments" that a standard local 1-WL based GNN can count exactly.

Algorithm Fragmentation Algorithm

Require: G; $\mathbb{P}$: List of patterns; M_i^{pattern}: learned model for generating pattern corresponding to $P_i \in \mathbb{P}$; M_i^{count}: learned model for counting $P_i \in \mathbb{P}$;
Initialize $a \leftarrow []$
for each node $v \in V(G)$ **do**
  $H_v = G_v^r \setminus \{v\}$;
  Initialize $b \leftarrow []$
  for each pattern $P_i \in \mathbb{P}$ **do**
    Initialize $c \leftarrow []$
    for each node $u \in H_v$ **do**
      $\mathcal{P}_u = M_i^{\text{pattern}}(H_v, u)$
      c.append$(M_i^{\text{count}}(\mathcal{P}_u))$
    end for
    b.append$(\alpha(c))$ {Learnable function}
  end for
  a.append$(\beta(b))$ {Learnable function}
end for
$Count = \gamma(a)$ {Learnable function}
return $Count$

The InSig Methodology: A Three-stage Pipeline



Dataset Statistics

Table: Dataset statistics. The total number of graphs in the dataset is 5000. We used 4000 graphs for training, 500 for validation, and 500 for testing.

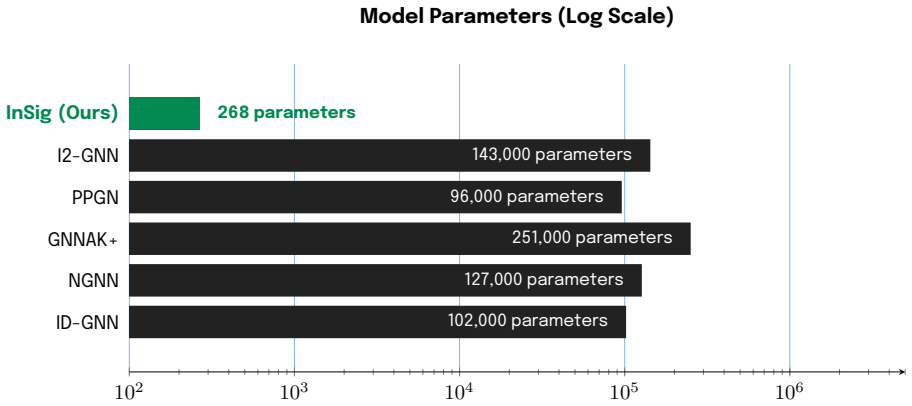
Task	Total pattern count	Zero Count graphs	Standard Deviation of Count	Average number of Nodes	Average number of Edges	Number of graphs
Triangle	25209	195	3.072			
2-Star	429463	0	18.015			
3-Star	309525	0	17.777			
Chordal	19088	1786	4.742	18.7976	62.678	5000
K4	643	4447	0.387			
C4	53002	16	6.938			
Tailed Triangle	177968	195	25.943			

Performance

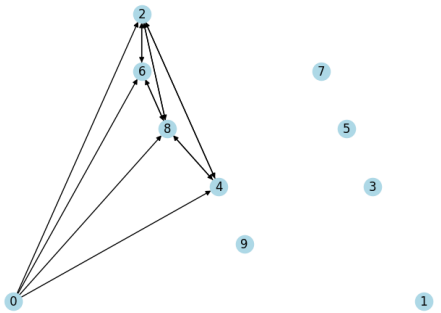
Table: MAE for the subgraph count of different patterns. Some results, such as 2-*star*, 3-*star*, are not conducted by the given baselines; therefore, it is mentioned NA.

Models	Without Fragmentation				Fragmentation		
	Triangle	*3-Star*	*2-Star*	*Chorcal C4*	K_4	C_4	*Tailed Triangle*
ID-GNN (You et al., 2021)	6.00E-04	NA	NA	4.52E-02	2.60E-03	2.20E-03	1.05E-01
NGNN (Zhang and Li, 2021)	3.00E-04	NA	NA	3.92E-02	4.50E-03	1.30E-03	1.04E-01
GNNAK+ Zhao et al. (2022)	4.00E-04	1.50E-02	NA	1.12E-02	4.90E-03	4.10E-03	4.30E-03
PPGN Maron et al. (2019)	3.00E-04	NA	NA	1.50E-03	1.65E-01	9.00E-04	2.60E-03
I2-GNN Huang et al. (2023)	4.00E-04	NA	NA	1.00E-03	3.00E-04	1.60E-03	1.10E-03
InSig	**0.00E-00**	**0.00E-00**	**0.00E-00**	**0.00E-00**	**0.00E-00**	**0.00E-00**	**0.00E-00**

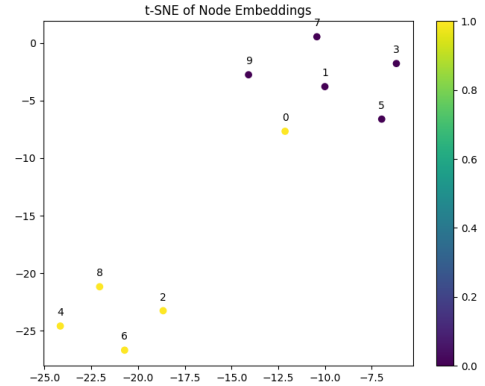
Efficiency: Zero Error with Minimal Parameters



Visualizing Structural Learning



(a) Figure shows an example of G_0^1 for a random graph with 10 nodes



(b) t-SNE plot of the updated node embeddings from the Pattern learning component

Figure: Figure showing the input and updated node embedding from the pattern learning component

Conclusion & Future Work

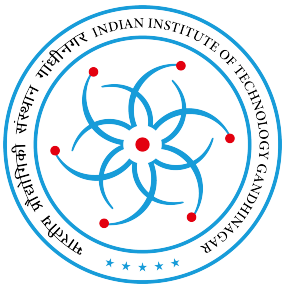
Conclusion

- **Problem:** Standard GNNs are limited by 1-WL expressivity.
- **Solution:** InSig uses Fragmentation to decompose complex patterns into learnable simple components.
- **Result: Zero MAE** (Exact Counting) with Drastically reduced model size (268 parameters)

Future Work

- Extend fragmentation to patterns > 2-hop neighborhoods.
- Theoretical analysis of fragmentation bounds.
- Utilize the methodology in downstream tasks like molecule property prediction.

Acknowledgment



References

Huang, Y., Peng, X., Ma, J., and Zhang, M. (2023). Boosting the cycle counting power of graph neural networks with i^2-GNNs. In *The Eleventh International Conference on Learning Representations*.

Maron, H., Ben-Hamu, H., Serviansky, H., and Lipman, Y. (2019). *Provably Powerful Graph Networks*. Curran Associates Inc.

You, J., Gomes-Selman, J. M., Ying, R., and Leskovec, J. (2021). Identity-aware graph neural networks. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35:10737–10745.

Zhang, M. and Li, P. (2021). Nested graph neural networks. *Advances in Neural Information Processing Systems*, 34:15734–15747.

Zhao, L., Jin, W., Akoglu, L., and Shah, N. (2022). From stars to subgraphs: Uplifting any GNN with local structure awareness. In *International Conference on Learning Representations*.

Thank You