OpenGSL: A Comprehensive Benchmark for Graph Structure Learning

Reporter: Zhiyao Zhou

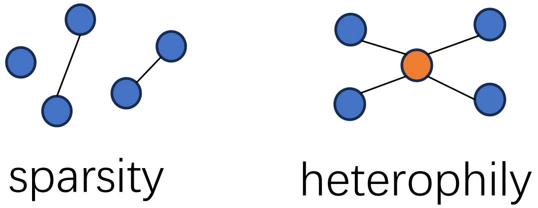
Graph Structure Learning: A Data-centric Perspective

- Model-centric Research：

  Researchers have proposed a series of new models to address issues such as over-smoothing, over-squashing, and expressivity.

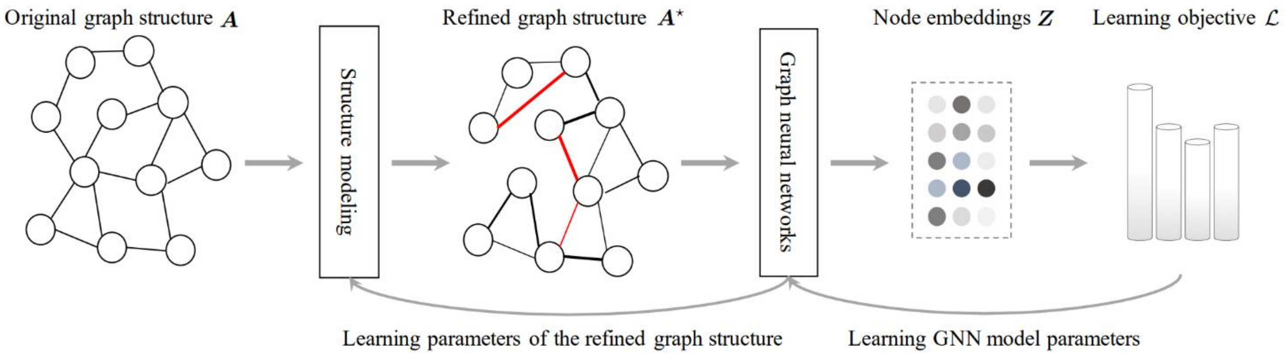
  However, these model-centric approaches overlook the inherent flaws in the graph structure, and may lead to suboptimal results.

- Flaws of Graph Structure：

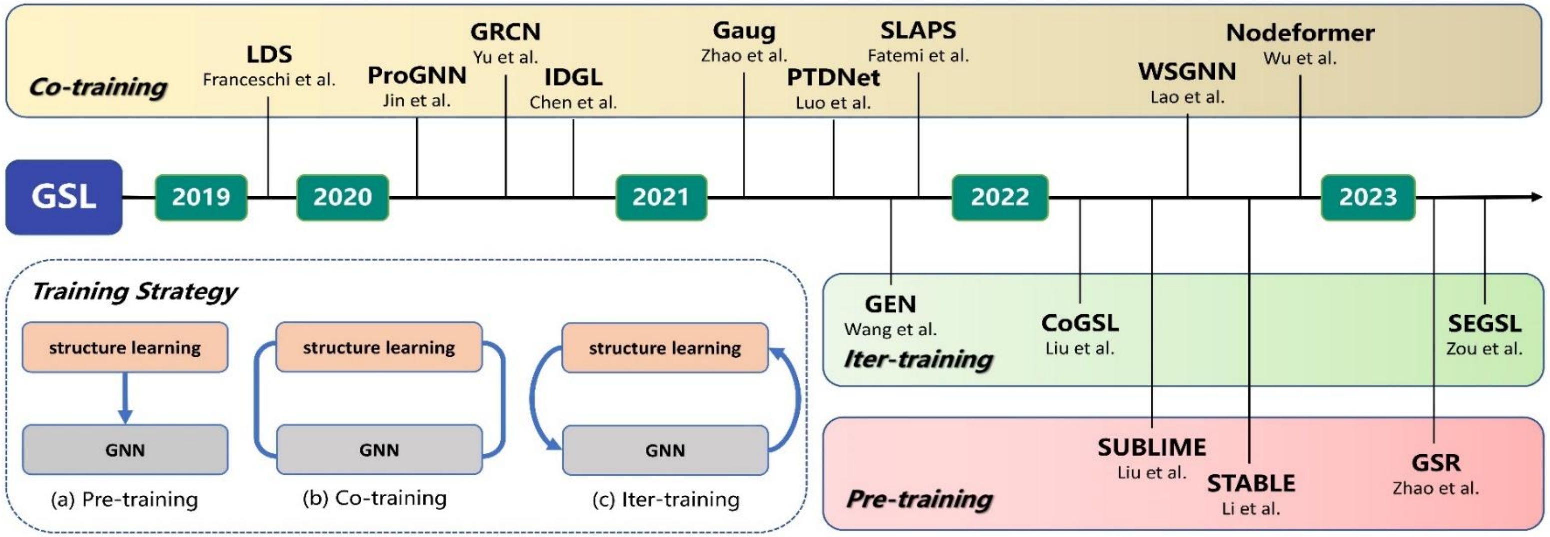


- Graph Structure Learning：

  Graph Structure Learning (GSL) jointly optimizes the graph structure and GNN to learn enhanced graph representations from refined graph structure.



Why OpenGSL?

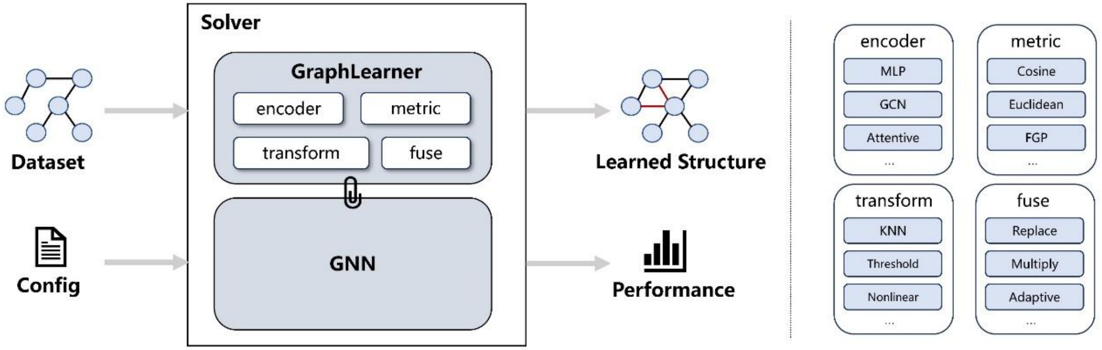


There lacks a comprehensive benchmark for GSL, which significantly impedes the understanding and progress of GSL in several aspects:

- Different experimental settings.
- Lack of understanding of the learned structure.
- Efficiency is overlooked.

Why OpenGSL?

We introduce OpenGSL, the first comprehensive benchmark for GSL



Algorithms

LDS, ProGNN, GRCN, IDGL, GAug, GEN, SLAPS, CoGSL, SUBLIME, WSGNN, STABLE, Nodeformer, SEGSL

Datasets

Cora, Citeseer, Pubmed, Questions, Minesweeper
Blogcatalog, Flickr, Amazon-ratings, Roman-empire, Wiki-cooc

Comparison	Efficiency	Robustness
Homophily	Generalizability	Backbones

- Open-sourced library with good usability and reproducibility.
- Fair comparisons through careful reimplementations and unified experimental settings.
- Multi-dimensional analysis thourgh well-designed experiments.

OpenGSL

Performance Comparison

- Observation 1: For homophilous graphs, many GSL methods work well in datasets with balanced classes, while they cannot handle highly imbalanced situations.

- Observation 2: For heterophilous graphs, GSL methods can be effective on specific datasets.

Node classification results on homophilous datasets

Model	Cora	Citeseer	Pubmed	Questions	Minesweeper
GCN	81.95 ± 0.62	71.34 ± 0.48	78.98 ± 0.35	**75.80 ± 0.51**	**78.28 ± 0.44**
LDS	**84.13 ± 0.52**	**75.16 ± 0.43**	–	–	–
ProGNN	80.27 ± 0.48	71.35 ± 0.42	79.39 ± 0.29	–	51.43 ± 2.22
IDGL	**84.19 ± 0.61**	**73.26 ± 0.53**	**82.78 ± 0.44**	50.00 ± 0.00	50.00 ± 0.00
GRCN	**84.61 ± 0.34**	72.34 ± 0.73	79.30 ± 0.34	**74.50 ± 0.84**	72.57 ± 0.49
GAug	83.43 ± 0.53	72.79 ± 0.86	78.73 ± 0.77	–	**77.93 ± 0.64**
SLAPS	72.29 ± 1.01	70.00 ± 1.29	70.96 ± 0.99	–	50.89 ± 1.72
WSGNN	83.66 ± 0.30	71.15 ± 1.01	79.78 ± 0.35	–	67.91 ± 3.11
Nodeformer	78.81 ± 1.21	70.39 ± 2.04	78.38 ± 1.94	**72.61 ± 2.29**	77.29 ± 1.71
GEN	81.66 ± 0.91	**73.21 ± 0.62**	78.49 ± 3.98	–	**79.56 ± 1.09**
CoGSL	81.46 ± 0.88	72.94 ± 0.71	78.38 ± 0.41	–	–
SEGSL	81.04 ± 1.07	71.57 ± 0.40	79.26 ± 0.67	–	–
SUBLIME	83.33 ± 0.73	72.44 ± 0.89	**80.56 ± 1.32**	67.21 ± 0.99	49.93 ± 1.36
STABLE	83.25 ± 0.86	70.99 ± 1.19	**81.46 ± 0.78**	–	70.78 ± 0.27

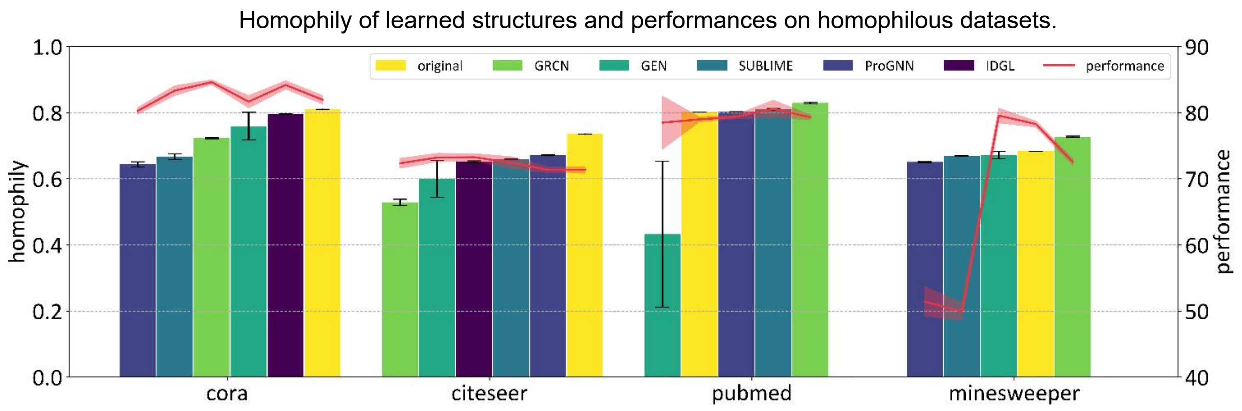
Node classification results on heterophilous datasets

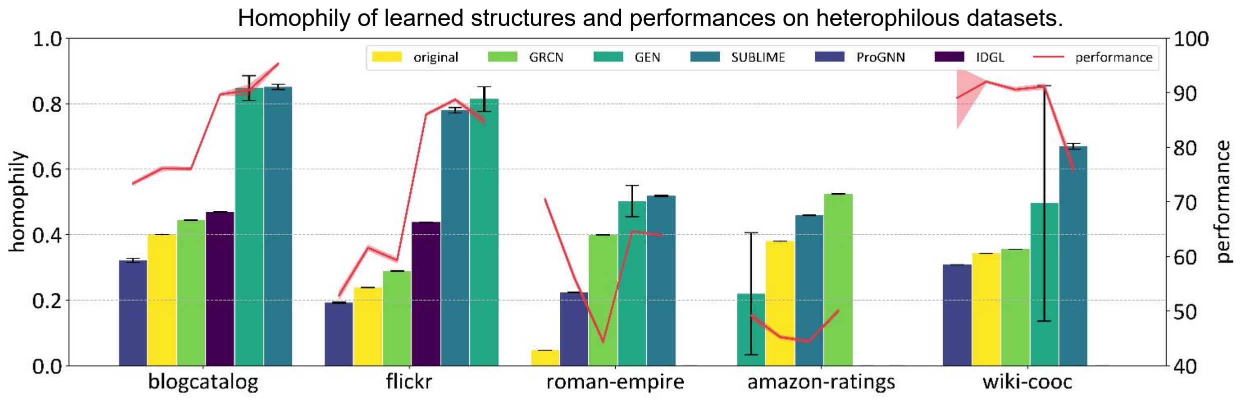
Model	BlogCatalog	Flickr	Amazon-ratings	Roman-empire	Wiki-cooc
GCN	76.12 ± 0.42	61.60 ± 0.49	45.24 ± 0.29	**70.41 ± 0.47**	**92.03 ± 0.19**
LDS	77.10 ± 0.27	–	–	–	–
ProGNN	73.38 ± 0.30	52.88 ± 0.76	–	56.21 ± 0.58	89.07 ± 5.59
IDGL	89.68 ± 0.24	**86.03 ± 0.25**	45.87 ± 0.58	47.10 ± 0.65	90.18 ± 0.27
GRCN	76.08 ± 0.27	59.31 ± 0.46	**50.06 ± 0.38**	44.41 ± 0.41	90.59 ± 0.37
GAug	76.92 ± 0.34	61.98 ± 0.67	**48.42 ± 0.39**	52.74 ± 0.48	**91.30 ± 0.23**
SLAPS	**91.73 ± 0.40**	83.92 ± 0.63	40.97 ± 0.45	**65.35 ± 0.45**	89.09 ± 0.54
WSGNN	**92.30 ± 0.32**	**89.90 ± 0.19**	42.36 ± 1.03	57.33 ± 0.69	90.10 ± 0.28
Nodeformer	44.53 ± 22.62	67.14 ± 6.77	41.33 ± 1.25	56.54 ± 3.73	54.83 ± 4.43
GEN	90.48 ± 0.99	84.84 ± 0.81	**49.17 ± 0.68**	–	**91.15 ± 0.49**
CoGSL	83.96 ± 0.54	75.10 ± 0.47	40.82 ± 0.13	46.52 ± 0.48	–
SeGSL	75.03 ± 0.28	60.59 ± 0.54	–	–	–
SUBLIME	**95.29 ± 0.26**	**88.74 ± 0.29**	44.49 ± 0.30	**63.93 ± 0.27**	76.10 ± 1.12
STABLE	71.84 ± 0.56	51.36 ± 1.24	48.36 ± 0.21	41.00 ± 1.18	80.46 ± 2.44

Exploring Homophily

- Observation 3: The homophily of the learned structures varies on homophilous and heterophilous datasets—nearly unchanged on homophilous datasets while significantly improved on hetrophilous datasets.

- Observation 4: Homophily is not always a proper guidance for structure learning. In most cases, we do not observe positive correlation between the performance and the homophily

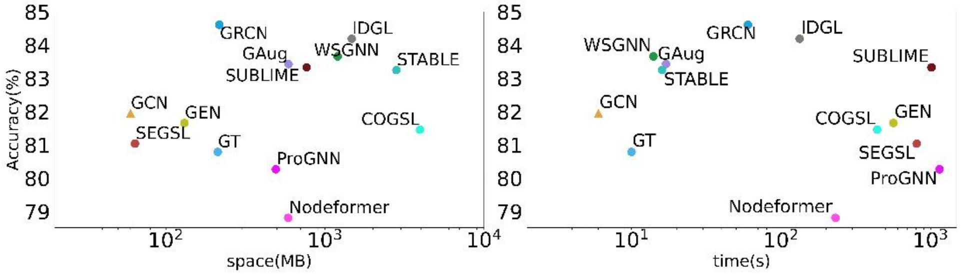




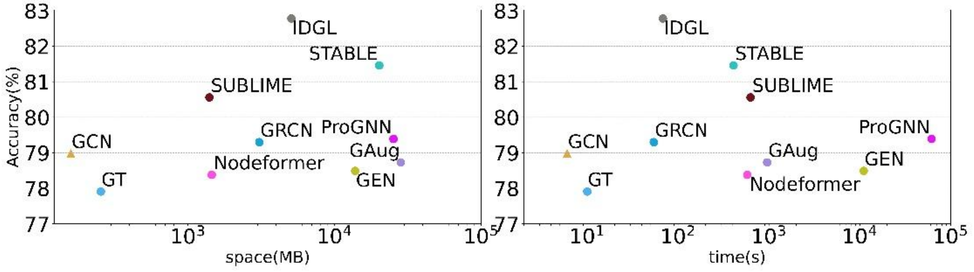
Datasets	Cora	Citeseer	Pubmed	Minesweeper	BlogCatalog	Flickr	Roman-empire	Amazon-ratings	Wiki-cooc
Pearson	0.29	-0.50	0.62	0.50	0.86	0.87	-0.25	-0.11	-0.84

Efficiency

Time and space consumption of different methods on Cora



Time and space consumption of different methods on Cora



- Observation 5: Most GSL methods have large time and space consumptions.

Future Directions

- **Rethinking the necessity of homophily in GSL.** Experiments suggest that the improvements achieved do not necessarily originate from increased homophily.

- **Designing adaptive GSL methods for diverse datasets.** Current GSL method do not universally work well across diverse datasets.

- **Developing task-agnostic GSL methods.** Existing works are mainly task-dependent. However, real-world scenarios sometimes require the refinement of a graph structure without accessing the downstream task.

- **Improving the efficiency of GSL methods.** Although some attempts have been made to improve the efficiency, they usually compromise the expressiveness.

Conclusion

We introduce a comprehensive benchmark for graph structure learning (GSL), OpenGSL.

The fair comparison and comprehensive analysis unearth several key findings on this promising research topic.

We believe that this benchmark will have a positive impact on this emerging research domain. We have made our code publicly available and welcome any contributions.

OpenGSL

Paper :

Library :

Thank you

Reporter: Zhiyao Zhou