TEXAS A&M UNIVERSITY
Engineering

Self-Supervised Representation Learning via Latent Graph Prediction

Yaochen Xie, Zhao Xu, Shuiwang Ji, Texas A&M University

ICML 2022

Self-Supervised Learning on Graphs

- SSL of GNNs is emerging as a promising way of leveraging unlabeled data.
- SSL taxonomies: contrastive v.s. predictive.
- Contrastive methods: current SOTA are mostly contrastive, depend on large sample size, hard to handle large-scale graphs.
- Predictive methods: memory-efficient, not enough theoretical guidance or justifications.

Latent Graphs

- We consider the concept latent data, where any observed graph $G = (A, X)$ is generated from a corresponding latent data that determine its semantic.
- WLOG, we specifically consider latent data $G_\ell = (A, F)$ in graph-structure with the same connectivity and satisfying two assumptions (non-structural and unbiased noise).
- Theorems can be generalized with other distances when considering latent data in different forms.

Latent Graph Prediction

- We adopt the prediction/reconstruction of the latent graph to derive our predictive SSL task.

$$f^* = \arg\min_f \mathbb{E} \left\| f(\boldsymbol{A}, \boldsymbol{X}) - \boldsymbol{F} \right\|^2$$

- We derive a self-supervised upper bound for the above objective to eliminate the need of unknown ***F***

$$\mathbb{E}_{\boldsymbol{A},\boldsymbol{X},\boldsymbol{F}} \left[\left\| f(\boldsymbol{A}, \boldsymbol{X}) - \boldsymbol{F} \right\|^2 + \left\| \boldsymbol{X} - \boldsymbol{F} \right\|^2 \right] \leq \mathbb{E}_{\boldsymbol{A},\boldsymbol{X}} \left\| f(\boldsymbol{A}, \boldsymbol{X}) - \boldsymbol{X} \right\|^2 + 2\sigma |V| \, \mathbb{E}_J \left[\frac{\mathbb{E}_{\boldsymbol{A},\boldsymbol{X}} \left\| f_J(\boldsymbol{A}, \boldsymbol{X}) - f_J(\boldsymbol{A}, \boldsymbol{X}_{J^c}) \right\|^2}{|J|} \right]^{1/2}$$

LaGraph Objectives

TEXAS A&M UNIVERSITY Engineering

Node-level representation learning

Corollary 2.2. *Let $G = (\boldsymbol{A}, \boldsymbol{X})$ be a given graph, $G_{\mathcal{I}} = (\boldsymbol{A}, \boldsymbol{F})$ be its latent graph, $\mathcal{E}$ and $\mathcal{D}$ be a graph encoder and a prediction head (decoder) consisting of fully-connected layers. If the prediction head $\mathcal{D}$ is ℓ-Lipschitz continuous with respect to l_2-norm, we further have the following inequality,*

$$\mathbb{E}\left[\,\|\mathcal{D}(\boldsymbol{H}) - \boldsymbol{F}\|^2 + \|\boldsymbol{X} - \boldsymbol{F}\|^2\,\right] \leq \mathbb{E}\,\|\mathcal{D}(\boldsymbol{H}) - \boldsymbol{X}\|^2 + 2\sigma|V|\ell\,\mathbb{E}_J\left[\frac{\mathbb{E}\,\|\boldsymbol{H}_J - \boldsymbol{H}'_J\|^2}{|J|}\right]^{1/2}, \tag{3}$$

where $\boldsymbol{H} = \mathcal{E}(\boldsymbol{A}, \boldsymbol{X})$ and $\boldsymbol{H}' = \mathcal{E}(\boldsymbol{A}, \boldsymbol{X}_{J^c})$ denote the node embedding of the given graph and the masked graph, respectively, and $\boldsymbol{H}_J := \boldsymbol{H}[J,:]$ selects rows with indices in J.

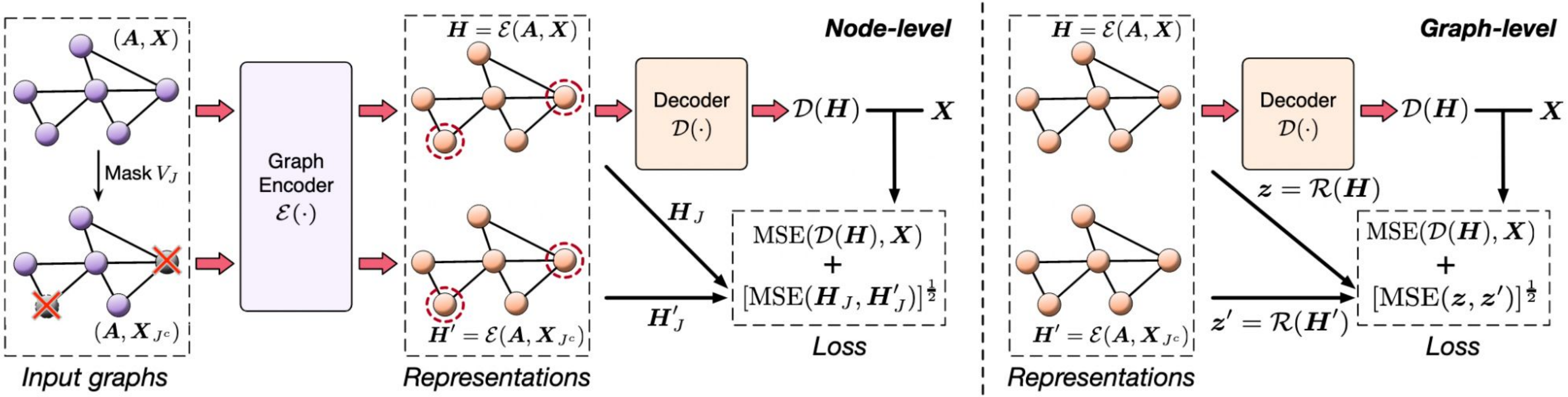
Graph-level representation learning

Corollary 2.3. *Let $G = (\boldsymbol{A}, \boldsymbol{X})$ be a given graph, $G_{\mathcal{I}} = (\boldsymbol{A}, \boldsymbol{F})$ be its hidden latent graph, $\mathcal{E}$ be a graph encoder, $\mathcal{R}$ be a readout function satisfying k-Bilipschitz continuity with respect to l_2-norm, and $\mathcal{D}$ be a prediction head (decoder). If the prediction head $\mathcal{D}$ is ℓ-Lipschitz continuous with respect to l_2-norm, we have the following inequality,*

$$\mathbb{E}\left[\,\|\mathcal{D}(\boldsymbol{H}) - \boldsymbol{F}\|^2 + \|\boldsymbol{X} - \boldsymbol{F}\|^2\,\right] \leq \mathbb{E}\,\|\mathcal{D}(\boldsymbol{H}) - \boldsymbol{X}\|^2 + 2\sigma|V|k\ell\,\mathbb{E}_J\left[\frac{\mathbb{E}\,\|\boldsymbol{z} - \boldsymbol{z}'\|^2}{|J|}\right]^{1/2}, \tag{4}$$

where $\boldsymbol{z} = \mathcal{R}(\boldsymbol{H})$ and $\boldsymbol{z}' = \mathcal{R}(\boldsymbol{H}')$ denote the graph-level representations of the given graph and the masked graph, respectively.

The LaGraph Framework



Please refer to Section 3 in our paper for further discussions and theoretically analysis on the relationship and differences between LaGraph and other theoretically sound methods, including Denoising Autoencoders, the Bottleneck Principle, contrastive methods, and BGRL ...

Results: Node-level Tasks

Transductive	Am.Comp.	Am.Pht.	Co.CS	Co.Phy
Raw features	73.8±0.0	78.5±0.0	90.4±0.0	93.6±0.0
DeepWalk	85.7±0.1	89.4±0.1	84.6±0.2	91.8±0.2
GAE	87.7±0.3	92.7±0.3	92.4±0.2	95.3±0.1
VGAE	88.1±0.3	92.8±0.3	92.5±0.2	95.3±0.1
Supervised	86.5±0.5	92.4±0.2	93.0±0.3	95.7±0.2
DGI	84.0±0.5	91.6±0.2	92.2±0.6	94.5±0.5
GMI	82.2±0.3	90.7±0.2	OOM	OOM
MVGRL	87.5±0.1	91.7±0.1	92.1±0.1	95.3±0.0
GRACE	87.5±0.2	92.2±0.2	92.9±0.0	95.3±0.0
GCA	88.9±0.2	92.5±0.2	93.1±0.0	95.7±0.0
BGRL	**89.7±0.3**	92.9±0.3	93.2±0.2	95.6±0.1
LaGraph	88.0±0.3	**93.5±0.4**	**93.3±0.2**	**95.8±0.1**

Inductive	PPI	Flickr	Reddit
Raw feat.	42.5±0.3	20.3±0.2	58.5±0.1
GAE	75.7±0.0	50.7±0.2	OOM
VGAE	75.8±0.0	50.4±0.2	OOM
Super-GCN	51.5±0.6	48.7±0.3	93.3±0.1
Super-GAT	97.3±0.2	OOM	OOM
GraphSAGE	46.5±0.7	36.5±1.0	90.8±1.1
DGI	63.8±0.2	42.9±0.1	94.0±0.1
GMI	65.0±0.0	44.5±0.2	95.0±0.0
SUBG-CON	66.9±0.2	48.8±0.1	**95.2±0.0**
BGRL-GCN	69.6±0.2	50.0±0.3*	OOM*
BGRL-GAT	70.5±0.1	44.2±0.1*	OOM*
LaGraph	**74.6±0.0**	**51.3±0.1**	**95.2±0.0**

Top: Performance on transductive and inductive node-level datasets.

Right: Model robustness when trained on subset of nodes.

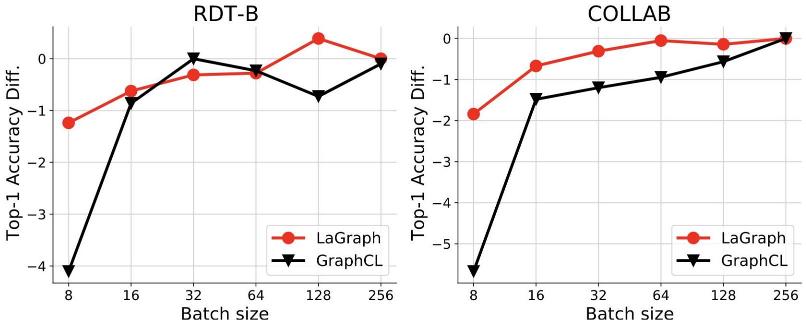
	# nodes sampled	100	1,000	2,500	5,000	10,000	all
Flickr	% nodes sampled	0.22%	2.24%	5.60%	11.20%	22.41%	100.00%
	F1-score - *LaGraph*	6.07	51.12	51.12	51.27	51.29	51.26
	Memory - *LaGraph*	1389MB	1465MB	1553MB	1725MB	2065MB	4211MB
	F1-score - GraphCL	45.27	45.27	45.27	45.38	45.45	45.48
	Memory - GraphCL	1647MB	2599MB	4137MB	6741MB	11905MB	47939MB

Results: Graph-level Tasks

	NCI1	PROTEINS	DD	MUTAG	COLLAB	RDT-B	RDT-M5K	IMDB-B
GL	–	–	–	81.7±2.1	–	77.3±0.2	41.0±0.2	65.9±1.0
WL	80.0±0.5	72.9±0.6	–	80.7±3.0	–	68.8±0.4	46.1±0.2	72.3±3.4
DGK	80.3±0.5	73.3±0.8	–	87.4±2.7	–	78.0±0.4	41.3±0.2	67.0±0.6
Node2Vec	54.9±1.6	57.5±3.6	75.1±0.5	72.6±10.2	55.7±0.2	73.8±0.5	34.1±0.4	50.0±0.8
Sub2Vec	52.8±1.5	53.0±5.6	73.6±1.5	61.1±15.8	62.1±1.4	71.5±0.4	36.7±0.4	55.3±1.5
Graph2Vec	73.2±1.8	73.3±2.1	76.2±0.1	83.2±9.3	59.9±0.0	75.8±1.0	47.9±0.3	71.1±0.5
GAE	73.3±0.6	74.1±0.5	77.9±0.5	84.0±0.6	56.3±0.1	74.8±0.2	37.6±1.6	52.1±0.2
VGAE	73.7±0.3	74.0±0.5	77.6±0.4	84.4±0.6	56.3±0.0	74.8±0.2	39.1±1.6	52.1±0.2
InfoGraph	76.2±1.1	74.4±0.3	72.9±1.8	89.0±1.1	70.7±1.1	82.5±1.4	53.5±1.0	73.0±0.9
GraphCL	77.9±0.4	74.4±0.5	**78.6±0.4**	86.8±1.3	71.4±1.2	89.5±0.8	56.0±0.3	71.1±0.4
MVGRL	75.1±0.5	71.5±0.3	OOM	89.7±1.1	OOM	84.5±0.6	OOM	**74.2±0.7**
LaGraph	**79.9±0.5**	**75.2±0.4**	78.1±0.4	**90.2±1.1**	**77.6±0.2**	**90.4±0.8**	**56.4±0.4**	73.7±0.9

Top: Performance on graph-level classification tasks, scores are averaged over 5 run.

Right: Model robustness to small batch sizes on RDT-B and COLLAB.



TEXAS A&M UNIVERSITY

Engineering

Thank you!

Code available under the DIG library: *https://github.com/divelab/DIG/*

Email: *ethanycx@tamu.edu*