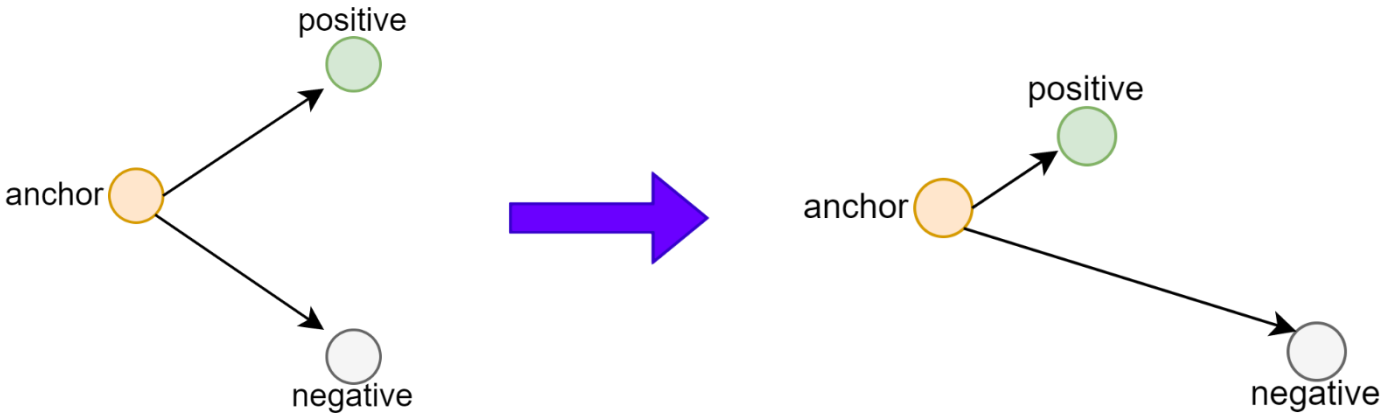
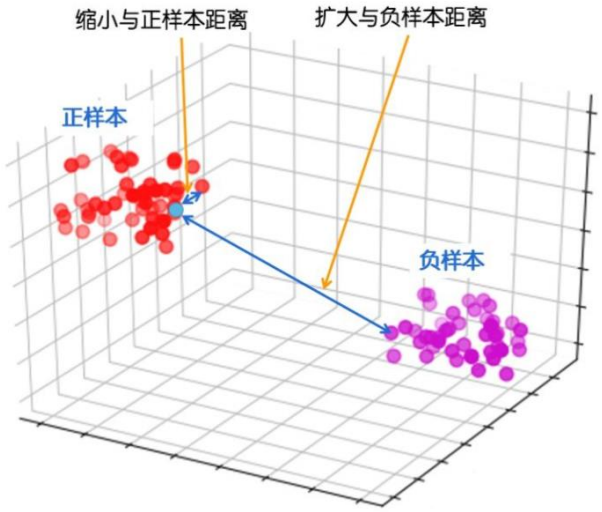
SimGRACE: A Simple Framework for Graph Contrastive Learning without Data Augmentation

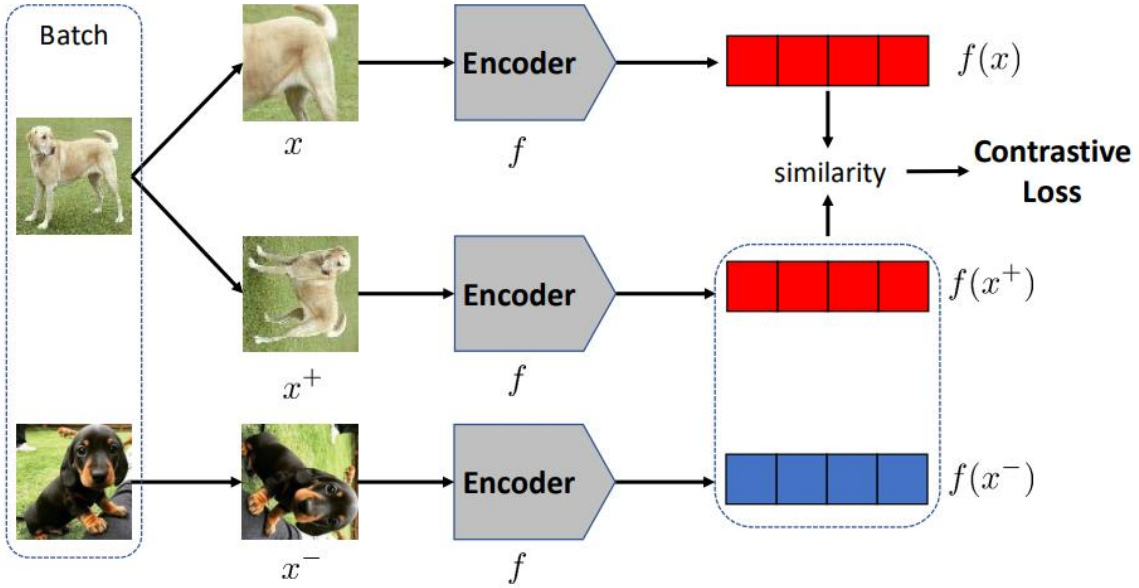
(WWW, 2022)

paper, code

Contrastive Learning

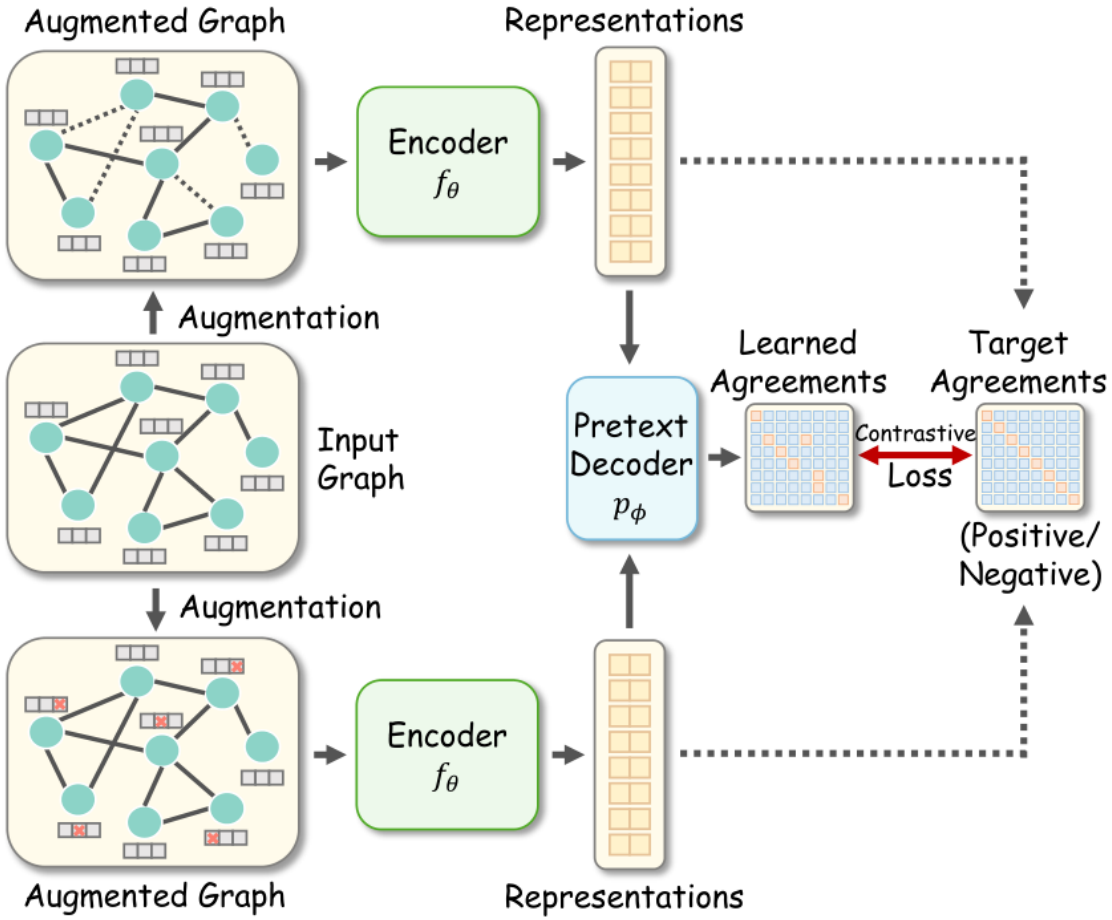




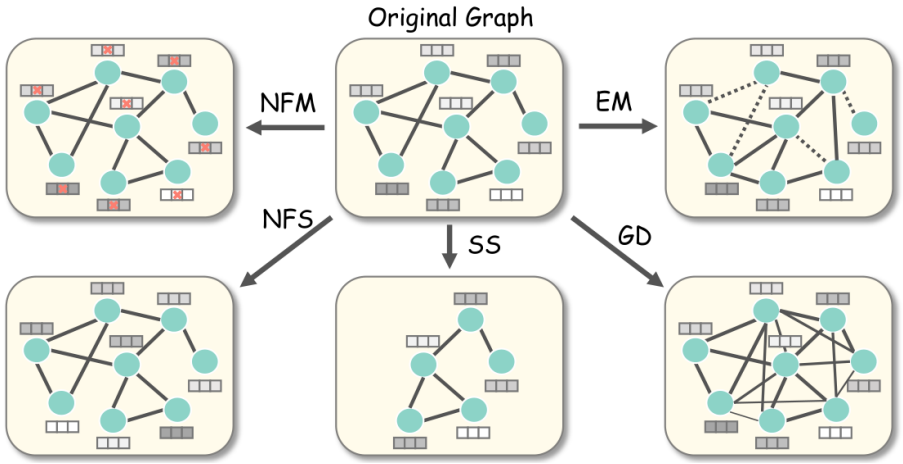


Graph Contrastive Learning

◆ General Framework



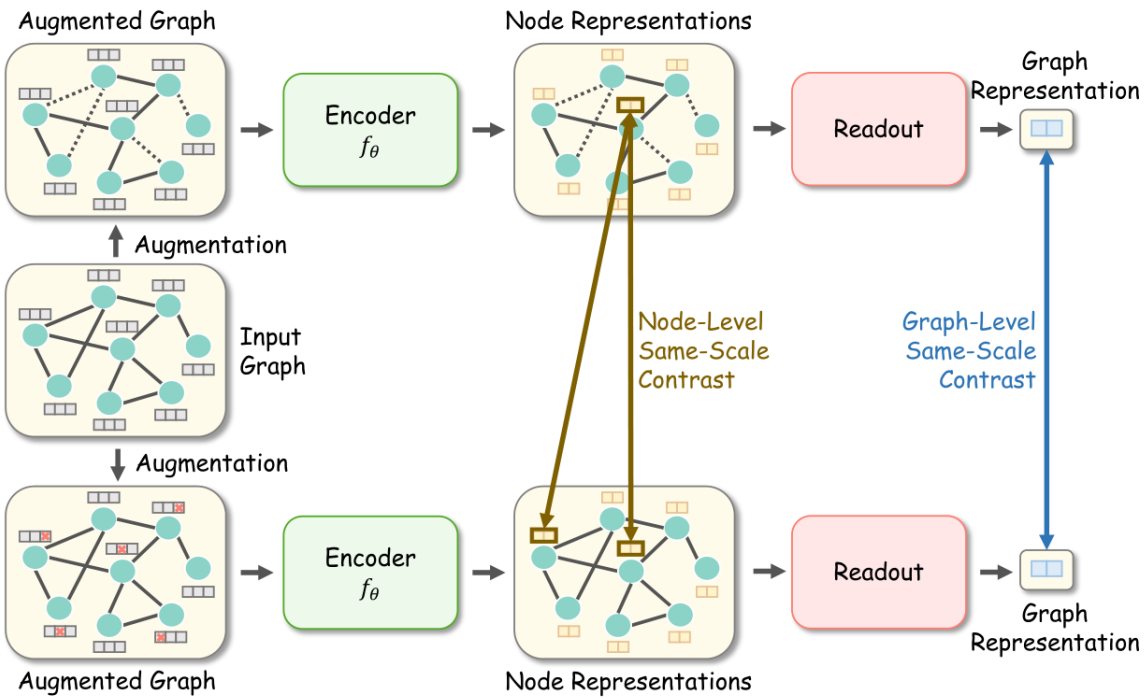
◆ Manual augmentation for CL



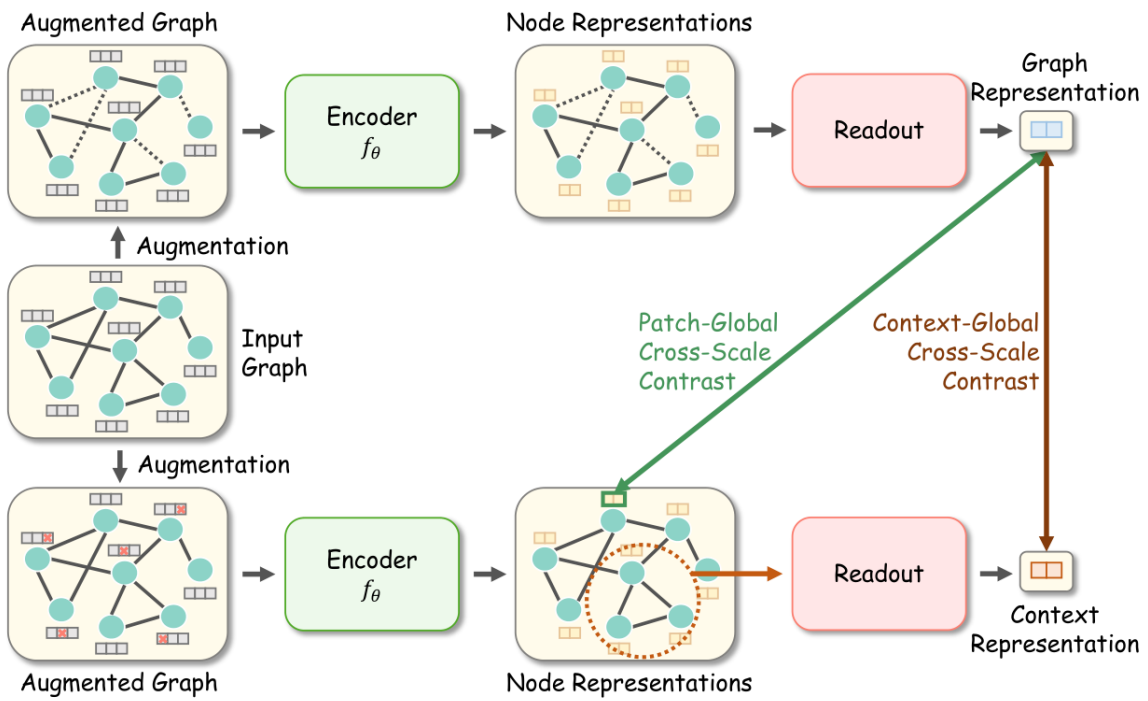
Node Feature Masking (NFM), Node Feature Shuffle (NFS), Edge Modification (EM), Graph Diffusion (GD), and Subgraph Sampling (SS).

Graph Contrastive Learning

◆ Contrast mode



same-scale contrast



cross-scale contrast

Motivation

Depends on data Augmentation

- Difficult to preserve semantics well during augmentations in view of the diverse nature of graph data

- Costly & not generative
  - manually picked per dataset by trial-and-errors.
  - selected via cumbersome search.
  - obtained by introducing expensive domain-specific knowledge as guidance.

No Augmentation!

Framework

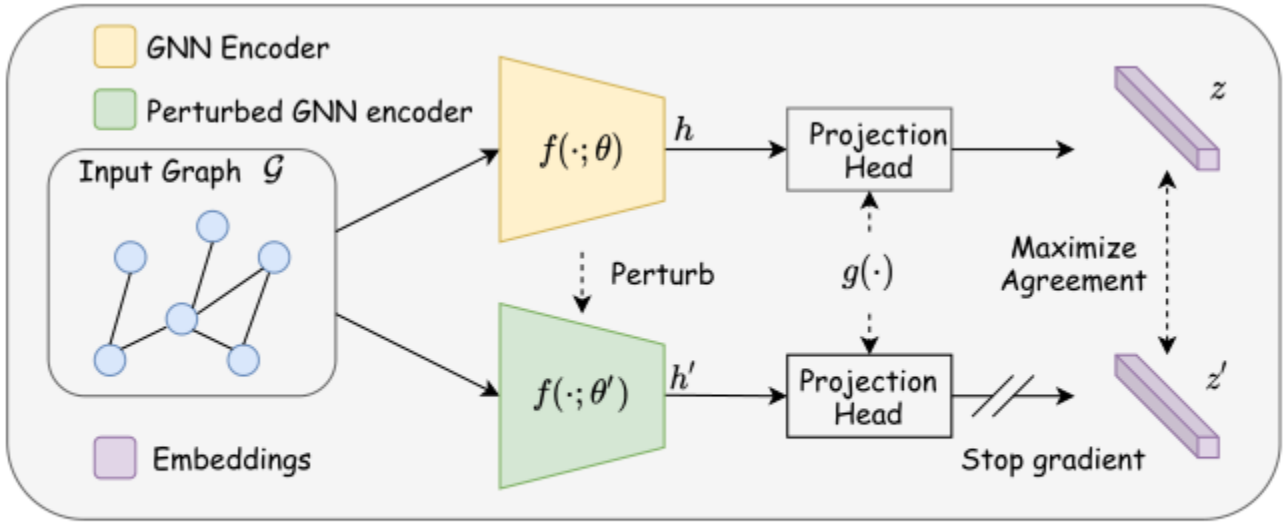


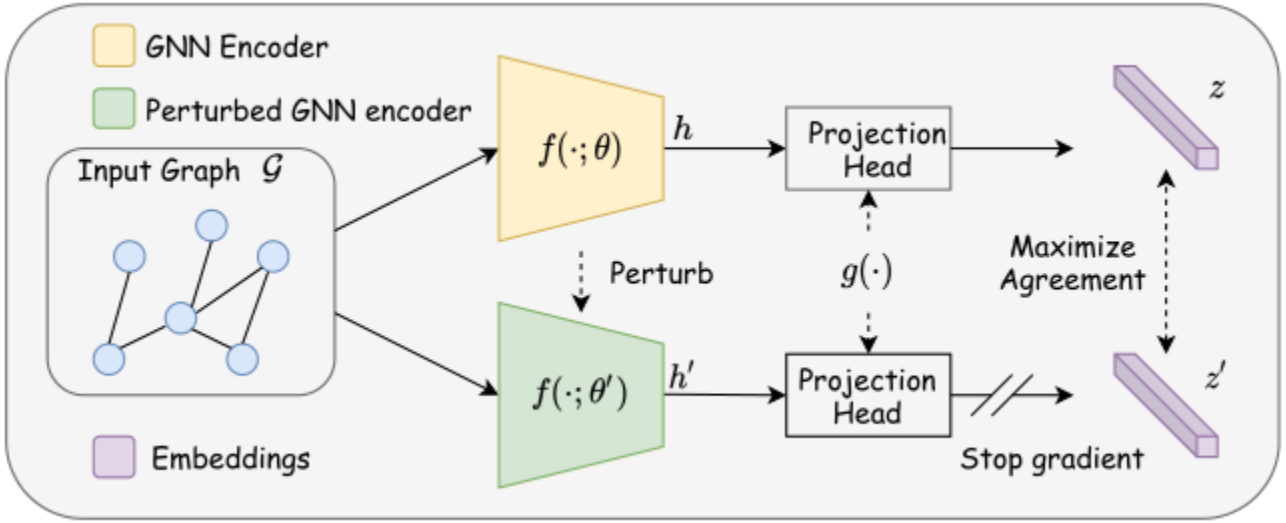
Figure 2: Illustration of SimGRACE, a simple framework of graph contrastive learning. Instead of augmenting the graph data, we feed the original graph $\mathcal{G}$ into a GNN encoder $f(\cdot;\theta)$ and its perturbed version $f(\cdot;\theta')$. After passing a shared projection head $g(\cdot)$, we maximize the agreement between representations z_i and z_j via a contrastive loss.

Table 1: Comparison between state-of-the-art GCL methods (graph-level representation learning) and SimGRACE.

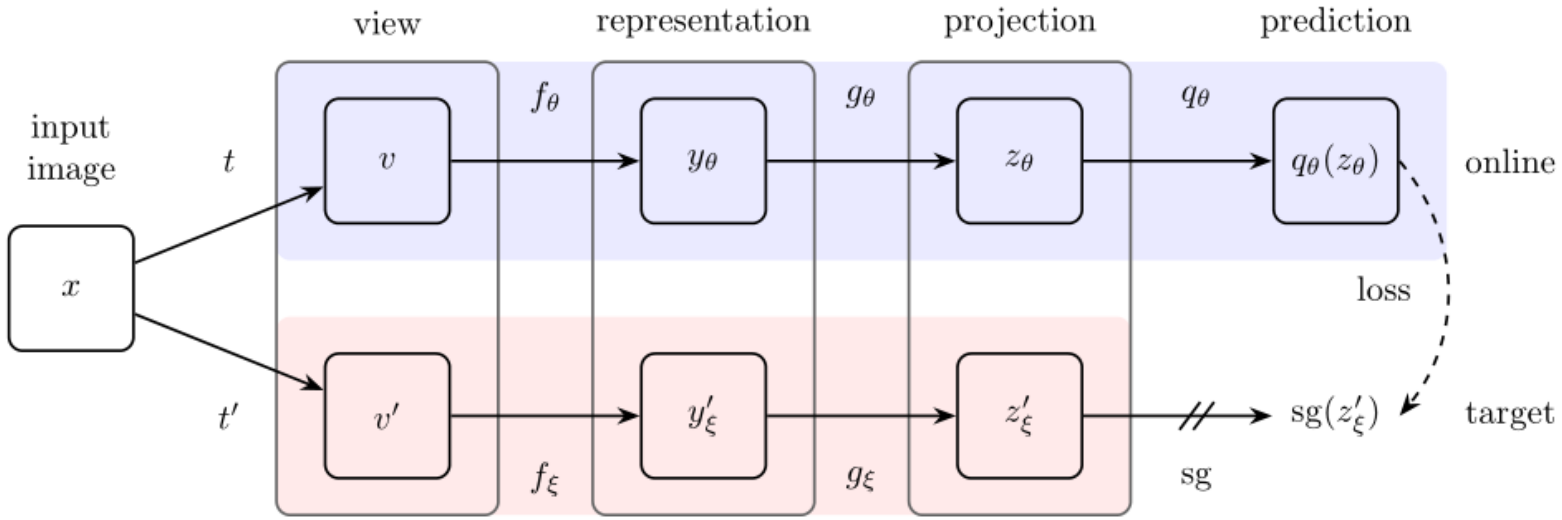
	No manual trial-and-errors	No domain knowledge	Preserving semantics	No cumbersome search	Generality
GraphCL [54]	✗	✓	✗	✓	✗
MoCL [40]	✓	✗	✓	✓	✗
JOAO(v2) [53]	✓	✓	✗	✗	✓
SimGRACE	✓	✓	✓	✓	✓

SimGRACE vs BYOL(BGRL)

SimGRACE

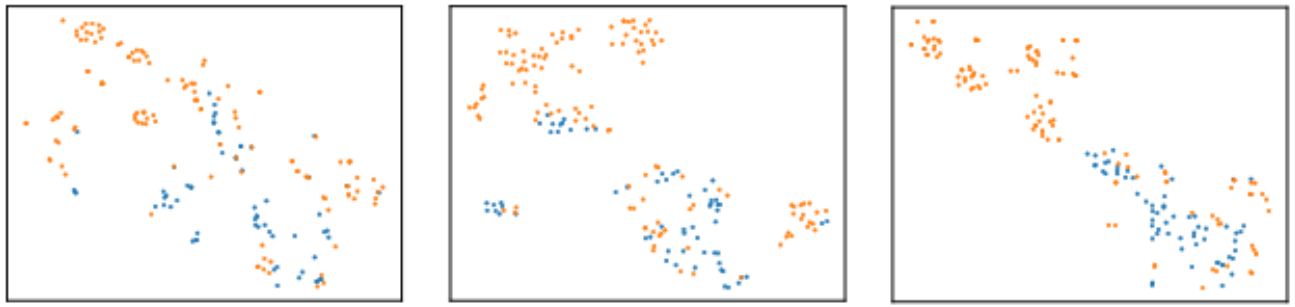


BYOL (BGRL)

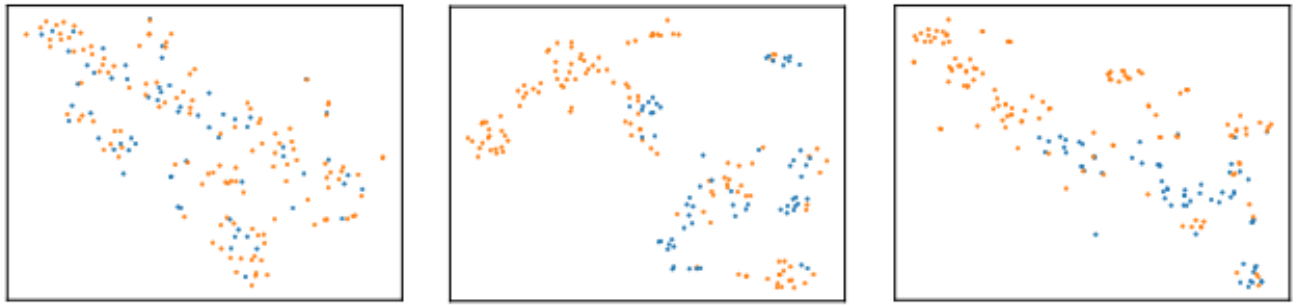


Observation Inspired

- Graph data can preserve their semantics well during encoder perturbations



Original graph/encoder



Perturb graph/encoder

GraphCL MoCL SimGRACE

Comparison on MUTAG dataset. Unlike GraphCL, SimGRACE and MoCL can preserve the class identity semantics well after perturbations.

SimGRACE

(1) Encoder perturbation

A GNN encoder $f(\cdot;\theta)$ and its its perturbed version $f(\cdot;\theta')$

$$\mathbf{h} = f(\mathcal{G};\theta), \mathbf{h}' = f(\mathcal{G};\theta').$$

$$\theta_l' = \theta_l + \eta \cdot \Delta\theta_l; \quad \Delta\theta_l \sim \mathcal{N}\left(0, \sigma_l^2\right),$$

(2) Projection head.

$$z = g(\mathbf{h}), z' = g(\mathbf{h}').$$

(3) Contrastive loss.

we utilize the normalized temperature-scaled cross entropy loss (NT-Xent)

$$\ell_n = -\log \frac{\exp\left(\text{sim}\left(z_n, z_n'\right)/\tau\right)}{\sum_{n'=1,n'\neq n}^{N} \exp\left(\text{sim}\left(z_n, z_{n'}\right)/\tau\right)}, \qquad (4) \qquad \text{sim}(z, z') = z^\top z'/\|z\|\,\|z'\|.$$

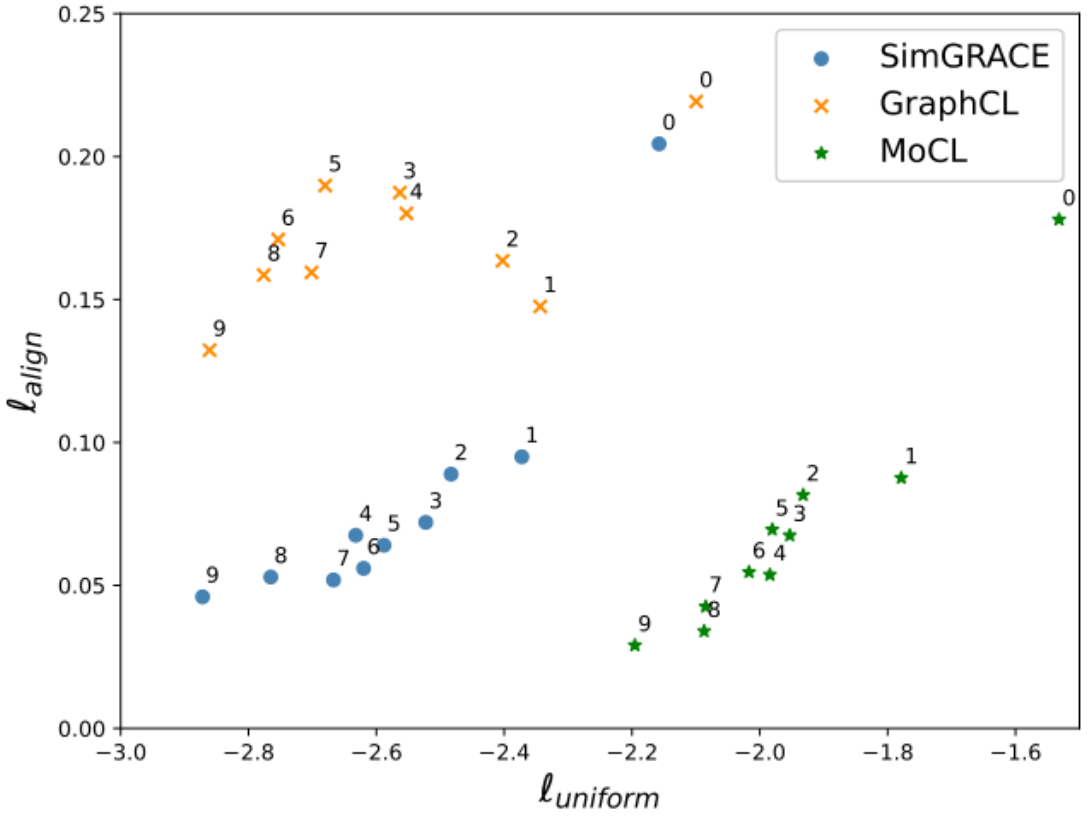
Why can SimGRACE work well?

- Using key properties related to contrastive learning: ***alignment*** and ***uniformity***

$$\ell_{\text{align}}(f;\alpha) \triangleq \mathop{\mathbb{E}}_{(x,y)\sim p_{\text{pos}}} \left[\|f(x)-f(y)\|_2^{\alpha}\right], \quad \alpha>0 \qquad (5)$$

$$\ell_{\text{align}}(f;\alpha) \triangleq \mathop{\mathbb{E}}_{x\sim p_{\text{data}}} \left[\|f(x;\boldsymbol{\theta})-f(x;\boldsymbol{\theta}')\|_2^{\alpha}\right], \quad \alpha>0 \qquad (6)$$

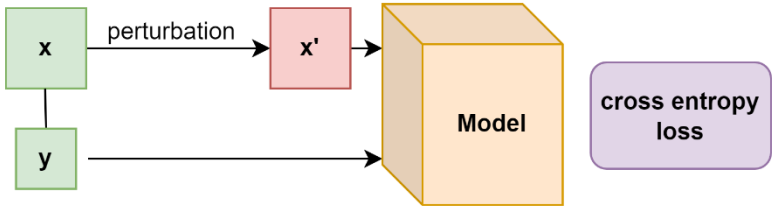
$$\ell_{\text{uniform}}(f;\alpha) \triangleq \log \mathop{\mathbb{E}}_{x,y \overset{i.i.d.}{\sim} p_{\text{data}}} \left[e^{-t\|f(x;\boldsymbol{\theta})-f(y;\boldsymbol{\theta})\|_2^2}\right], \quad t>0 \qquad (7)$$



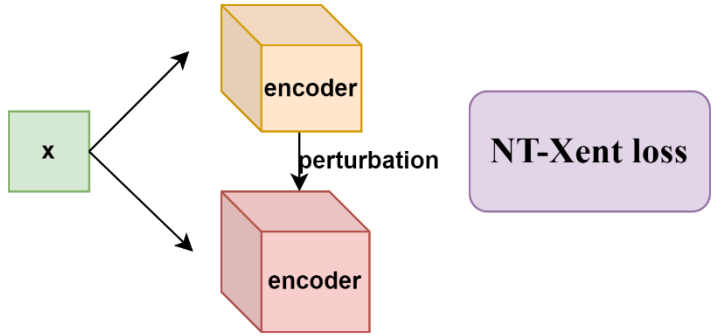
AT-SimGRACE

- We aim to utilize **Adversarial Training** (AT) to improve the adversarial robustness of SimGRACE in a principled way, directly incorporates adversarial examples into the training process to solve the following optimization problem:

$$\min_{\theta} \mathcal{L}'(\theta), \quad \text{where} \quad \mathcal{L}'(\theta) = \frac{1}{n} \sum_{i=1}^{n} \max_{\|\mathbf{x}_i' - \mathbf{x}_i\|_p \leq \epsilon} \ell_i' \left(f\left(\mathbf{x}_i'; \theta\right), y_i \right), \tag{8}$$



$$\mathbf{R}(\mathbf{w}; \epsilon) := \{\boldsymbol{\theta} \in \boldsymbol{\Theta} : \|\boldsymbol{\theta} - \mathbf{w}\| \leq \epsilon\}, \tag{9}$$



$$\min_{\theta} \quad \mathcal{L}(\theta + \Delta), \quad \text{where} \quad \mathcal{L}(\theta + \Delta) = \frac{1}{M} \sum_{i=1}^{M} \max_{\Delta \in \mathbf{R}(\mathbf{0}; \epsilon)} \ell_i \left(f\left(\mathcal{G}_i; \theta + \Delta\right), f\left(\mathcal{G}_i; \theta\right) \right) \tag{10}$$

AT-SimGRACE

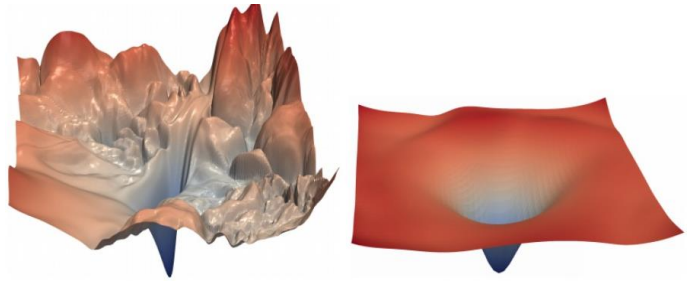
Algorithm 1: Encoder perturbation of AT-SimGRACE

Data: Graph dataset $\mathcal{D} = \{\mathcal{G}_1, \mathcal{G}_2, ..., \mathcal{G}_M\}$, contrastive loss ℓ, batch size N, initial encoder weights $\boldsymbol{\theta}$, inner iterations I, inner learning rate ζ, outer learning rate γ, norm ball radius ϵ.

for *each mini-batch* **do**
  Sample $\mathcal{D}_B = \{\mathcal{G}_i\}_{i=1}^{N}$ from $\mathcal{D}$;
  Initialize perturbation: $\boldsymbol{\Delta} \leftarrow 0$;
  for $t = 0, 1, 2, ..., I-1$ **do**
    Update perturbation:
    $\boldsymbol{\Delta} \leftarrow \boldsymbol{\Delta} + \zeta \sum_{i=1}^{N} \nabla_{\boldsymbol{\theta}} \ell_i \left(f\left(\mathcal{G}_i; \boldsymbol{\theta} + \boldsymbol{\Delta}\right), f\left(\mathcal{G}_i; \boldsymbol{\theta}\right)\right) / N$;
    if $\|\boldsymbol{\Delta}\|_2 > \epsilon$ **then**
      Normalize perturbation: $\boldsymbol{\Delta} \leftarrow \epsilon \boldsymbol{\Delta} / \|\boldsymbol{\Delta}\|_2$;
    end
  end
  Update weights:
  $\boldsymbol{\theta}' \leftarrow \boldsymbol{\theta} - \gamma \sum_{i=1}^{N} \nabla_{\boldsymbol{\theta}} \ell_i \left(f\left(\mathcal{G}_i; \boldsymbol{\theta} + \boldsymbol{\Delta}\right), f\left(\mathcal{G}_i; \boldsymbol{\theta}\right)\right) / N$.
end

AT-SimGRACE Theoretical Justification

Suppose: *it is widely accepted that flatter loss landscape can bring robustness.*



Based on $\boldsymbol{PAC-Bayes\ theory}$, Assuming that the prior distribution P over weights is $N(0,\sigma^2)$, with probability at least $1-\delta$ over the draw of M graphs, the expected error of the encoder can be bounded as:

$$\mathbb{E}_{\{\mathcal{G}_i\}_{i=1}^{M},\Delta}[\mathcal{L}(\theta+\Delta)] \leq \mathbb{E}_{\Delta}[\mathcal{L}(\theta+\Delta)]+4\sqrt{\frac{KL(\theta+\Delta\|P)+\ln\frac{2M}{\delta}}{M}}. \quad (11)$$

Δ as a $N(0,\sigma^2)$ perturbationin every direction, $\sigma = \alpha \| \theta \|$

$$\mathbb{E}_{\{\mathcal{G}_i\}_{i=1}^{M},\Delta}[\mathcal{L}(\theta+\Delta)] \leq \mathcal{L}(\theta) + \underbrace{\{\mathbb{E}_{\Delta}[\mathcal{L}(\theta+\Delta)]-\mathcal{L}(\theta)\}}_{\text{Expected sharpness}} + 4\sqrt{\frac{1}{M}\left(\frac{1}{2\alpha}+\ln\frac{2M}{\delta}\right)}. \quad (12)$$

As $\mathbb{E}_{\Delta}[\mathcal{L}(\theta+\Delta)] \leq \max_{\Delta}[\mathcal{L}(\theta+\Delta)]$, Thus, AT-SimGRACE optimizes the **worst-case of sharpness of loss landscape** $\max_{\Delta}[\mathcal{L}(\theta+\Delta)]-\mathcal{L}(\theta)$ to the bound of the expected error, which explains why AT-SimGRACE can enhance the robustness.

Experiments—Unsupervised and semi-supervised learning (RQ1)

Table 2: Comparing classification accuracy with baselines under the same experiment setting. The top three accuracy or rank for each dataset are emphasized in bold. A.R. denotes average rank. - indicates that results are not available in published papers.

Methods	NCI1	PROTEINS	DD	MUTAG	COLLAB	RDT-B	RDT-M5K	IMDB-B	A.R. ↓
GL	–	–	–	81.66 ± 2.11	–	77.34 ± 0.18	41.01 ± 0.17	65.87 ± 0.98	8.3
WL	**80.01** ± 0.50	72.92 ± 0.56	–	80.72 ± 3.00	–	68.82 ± 0.41	46.06 ± 0.21	**72.30 ± 3.44**	6.2
DGK	**80.31** ± 0.46	73.30 ± 0.82	–	87.44 ± 2.72	–	78.04 ± 0.39	41.27 ± 0.18	66.96 ± 0.56	5.5
node2vec	54.89 ± 1.61	57.49 ± 3.57	–	72.63 ± 10.20	–	–	–	–	9.0
sub2vec	52.84 ± 1.47	53.03 ± 5.55	–	61.05 ± 15.80	–	71.48 ± 0.41	36.68 ± 0.42	55.26 ± 1.54	10.2
graph2vec	73.22 ± 1.81	73.30 ± 2.05	–	83.15 ± 9.25	–	75.78 ± 1.03	47.86 ± 0.26	71.10 ± 0.54	6.7
MVGRL	–	–	–	75.40 ± 7.80	–	82.00 ± 1.10	–	63.60 ± 4.20	8.3
InfoGraph	76.20 ± 1.06	**74.44** ± 0.31	72.85 ± 1.78	**89.01** ± 1.13	**70.65** ± 1.13	82.50 ± 1.42	53.46 ± 1.03	**73.03** ± 0.87	3.8
GraphCL	77.87 ± 0.41	74.39 ± 0.45	**78.62** ± 0.40	86.80 ± 1.34	**71.36** ± 1.15	**89.53** ± 0.84	**55.99** ± 0.28	**71.14** ± 0.44	**3.1**
JOAO	78.07 ± 0.47	**74.55** ± 0.41	77.32 ± 0.54	87.35 ± 1.02	69.50 ± 0.36	85.29 ± 1.35	55.74 ± 0.63	70.21 ± 3.08	4.3
JOAOv2	78.36 ± 0.53	74.07 ± 1.10	**77.40** ± 1.15	**87.67** ± 0.79	69.33 ± 0.34	**86.42** ± 1.45	**56.03** ± 0.27	70.83 ± 0.25	**3.6**
SimGRACE	**79.12** ± 0.44	**75.35** ± 0.09	**77.44** ± 1.11	**89.01** ± 1.31	**71.72** ± 0.82	**89.51** ± 0.89	**55.91** ± 0.34	**71.30** ± 0.77	**2.0**

Experiments——Unsupervised and semi-supervised learning (RQ1)

Table 4: Comparing classification accuracy with baselines under the same semi-supervised setting. The top three accuracy or rank are emphasized in bold. – indicates that label rate is too low for a given dataset size. L.R. and A.R. are short for label rate and average rank, respectively.

L.R.	Methods	NCI1	PROTEINS	DD	COLLAB	RDT-B	RDT-M5K	A.R. ↓
1%	No pre-train.	60.72 ± 0.45	–	–	57.46 ± 0.25	–	–	8.5
	Augmentations	60.49 ± 0.46	–	–	58.40 ± 0.97	–	–	8.0
	GAE	61.63 ± 0.84	–	–	63.20 ± 0.67	–	–	5.5
	Infomax	**62.72** ± 0.65	–	–	61.70 ± 0.77	–	–	4.0
	ContextPred	61.21 ± 0.77	–	–	57.60 ± 2.07	–	–	7.5
	GraphCL	**62.55** ± 0.86	–	–	**64.57** ± 1.15	–	–	**2.0**
	JOAO	61.97 ± 0.72	–	–	63.71 ± 0.84	–	–	4.5
	JOAOv2	62.52 ± 1.16	–	–	**64.51** ± 2.21	–	–	**3.0**
	SimGRACE	**64.21** ± 0.65	–	–	**64.28** ± 0.98	–	–	**2.0**
10%	No pre-train.	73.72 ± 0.24	70.40 ± 1.54	73.56 ± 0.41	73.71 ± 0.27	86.63 ± 0.27	51.33 ± 0.44	7.7
	Augmentations	73.59 ± 0.32	70.29 ± 0.64	74.30 ± 0.81	74.19 ± 0.13	87.74 ± 0.39	52.01 ± 0.20	7.0
	GAE	74.36 ± 0.24	70.51 ± 0.17	74.54 ± 0.68	**75.09** ± 0.19	87.69 ± 0.40	33.58 ± 0.13	6.3
	Infomax	**74.86**± 0.26	72.27 ± 0.40	75.78 ± 0.34	73.76 ± 0.29	88.66 ± 0.95	**53.61** ± 0.31	3.7
	ContextPred	73.00 ± 0.30	70.23 ± 0.63	74.66 ± 0.51	73.69 ± 0.37	84.76 ± 0.52	51.23 ± 0.84	8.3
	GraphCL	**74.63**± 0.25	**74.17**± 0.34	**76.17**± 1.37	74.23 ± 0.21	**89.11**± 0.19	52.55 ± 0.45	**2.8**
	JOAO	74.48 ± 0.27	72.13 ± 0.92	75.69 ± 0.67	**75.30** ± 0.32	88.14 ± 0.25	**52.83**± 0.54	4.2
	JOAOv2	**74.86**± 0.39	**73.31**± 0.48	**75.81**± 0.73	**75.53**± 0.18	**88.79**± 0.65	52.71 ± 0.28	**2.5**
	SimGRACE	74.60 ± 0.41	**74.03** ± 0.51	**76.48** ± 0.52	74.74 ± 0.28	**88.86** ± 0.62	**53.97** ± 0.64	**2.3**

Experiments——Transferability (RQ2) & Adversarial robustness (RQ3)

Pre-Train dataset	PPI-306K	ZINC 2M		
Fine-Tune dataset	PPI	BBBP	ToxCast	SIDER
No Pre-Train	64.8 ± 1.0	65.8 ± 4.5	63.4 ± 0.6	57.3 ± 1.6
EdgePred	**65.7** ± 1.3	68.8 ± 0.8	62.7 ± 0.4	58.4 ± 0.8
AttrMasking	65.2 ± 1.6	67.3 ± 2.4	**64.1** ± 0.6	60.4 ± 0.7
ContextPred	64.4 ± 1.3	64.3 ± 2.8	**64.2** ± 0.5	**61.0**± 0.7
GraphCL	**67.88** ± 0.85	68.0 ± 2.0	**63.9** ± 0.6	**60.9** ± 0.6
JOAO	64.43 ± 1.38	**69.68** ± 0.67	62.40 ± 0.57	60.53 ± 0.88
JOAOv2	63.94 ± 1.59	**70.22**± 0.98	62.94 ± 0.48	59.97 ± 0.79
SimGRACE	**70.25** ± 1.22	**71.25** ± 0.86	63.36 ± 0.52	**60.59** ± 0.96

Table 5: Performance under three adversarial attacks for GNN with different depth following the protocols in [7].

Methods	Two-Layer			Three-Layer			Four-Layer		
	No Pre-Train	GraphCL	**AT-SimGRACE**	No Pre-Train	GraphCL	**AT-SimGRACE**	No Pre-Train	GraphCL	**AT-SimGRACE**
Unattack	93.20	**94.73**	94.24	98.20	98.33	**99.32**	98.87	99.00	**99.13**
RandSampling	78.73	80.68	**81.73**	92.27	92.60	**94.27**	95.13	97.40	**97.67**
GradArgmax	69.47	69.26	**75.13**	64.60	89.33	**93.00**	95.80	**97.00**	96.60
RL-S2V	42.93	42.20	**44.86**	41.93	61.66	**66.00**	70.20	84.86	**85.29**

Experiments——Efficiency (Training time and memory cost) (RQ4)

Dataset	Algorithm	Training Time	Memory
PROTEINS	GraphCL	111*s*	1231*MB*
	JOAOv2	4088*s*	1403*MB*
	SimGRACE	**46 s**	**1175 MB**
COLLAB	GraphCL	1033*s*	10199*MB*
	JOAOv2	10742*s*	7303*MB*
	SimGRACE	**378** s	**6547 MB**
RDT-B	GraphCL	917*s*	4135*MB*
	JOAOv2	10278*s*	3935*MB*
	SimGRACE	**280** s	**2729 MB**

Ablation——Hyper-parameters sensitivity analysis(RQ5)

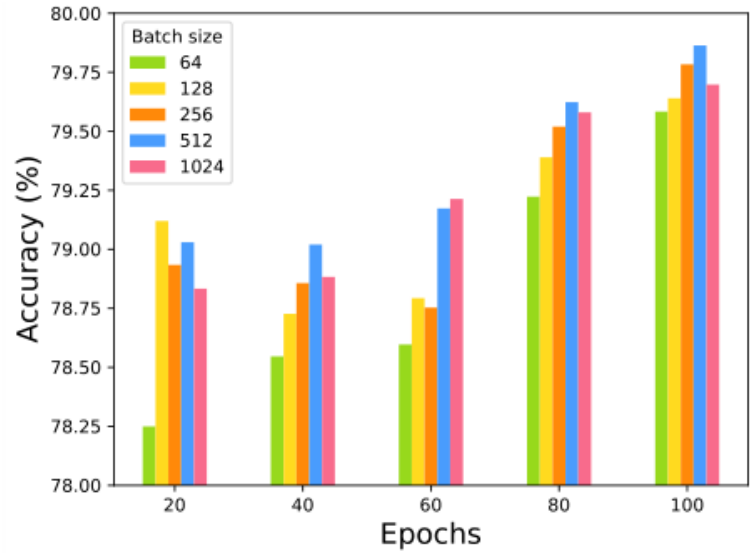
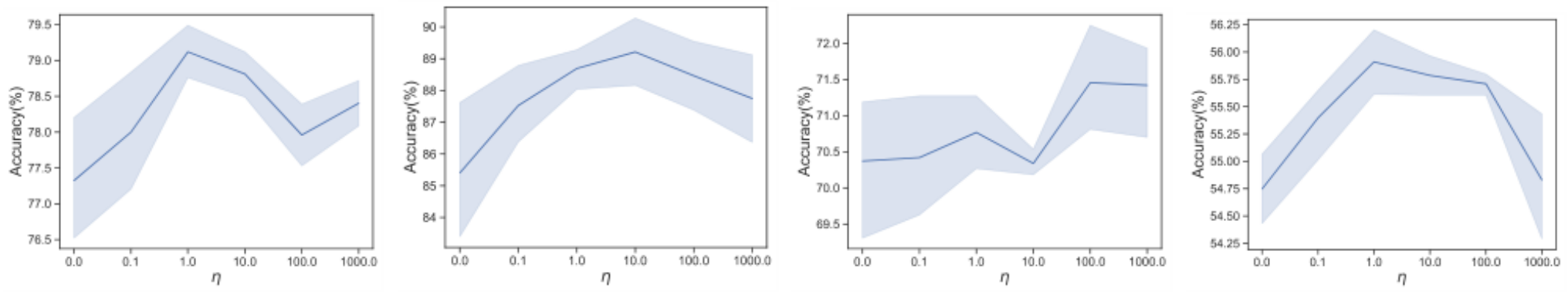


Figure 5: Performance of SimGRACE trained with different batch size and epochs on NCI1 dataset.



(a) NCI1 (b) MUTAG (c) COLLAB (d) RDT-5K

Figure 4: Performance versus magnitude of the perturbation (η) in unsupervised representation learning task.

Thanks