Learning Generalizable Representations from Unlabeled Graphs via Contrastive Learning

Atlas Wang, ECE@UT Austin

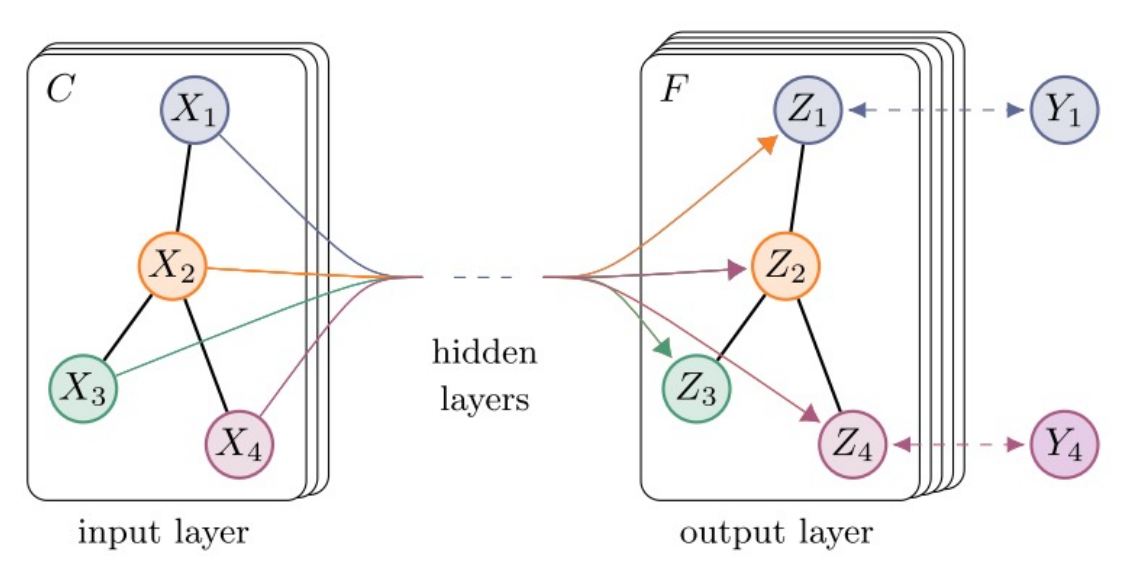
https://vita-group.github.io/

Work done with: Tianlong Chen (UT Austin), Yuning You & Prof. Yang Shen (TAMU)

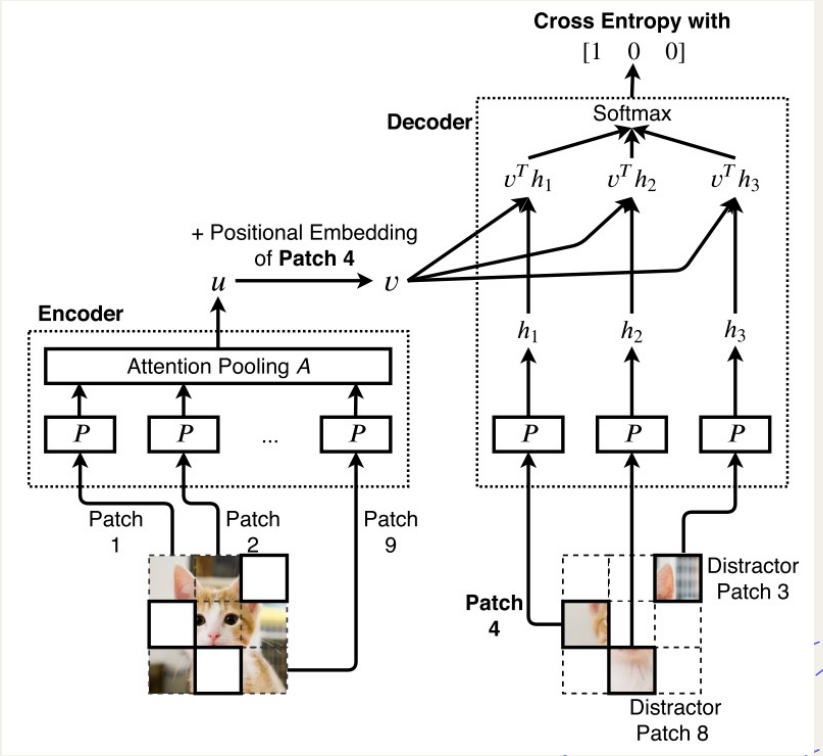
VITA

Background

+ Graph neural networks



+ Self-supervision (SS) in images



Background:
Simple Contrastive Learning (simCLR)

A Simple Framework for Contrastive Learning of Visual Representations

Ting Chen [1] Simon Kornblith [1] Mohammad Norouzi [1] Geoffrey Hinton [1]

- **Pre-text is not the greatest idea due to ad-hoc ….**
- **Simple idea:** maximizing the agreement of representations under data transformation, using a contrastive loss in the latent/feature space
- **Super effective:** 10% relative improvement over previous SOTA (cpc v2), outperforms AlexNet with 100X fewer labels

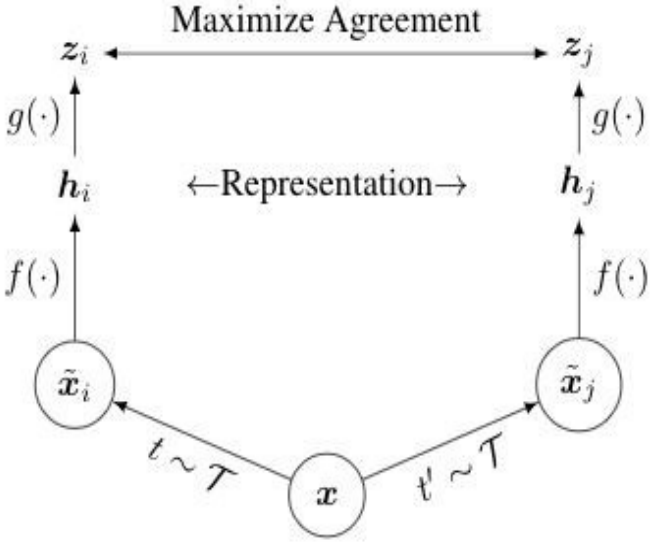


Figure 2. A framework for contrastive representation learning. Two separate stochastic data augmentations $t, t' \sim \mathcal{T}$ are applied to each example to obtain two correlated views. A base encoder network $f(\cdot)$ with a projection head $g(\cdot)$ is trained to maximize agreement in *latent representations* via a contrastive loss.

Background

+ Pre-training graph neural networks (GNNs) is under-explored with some exceptions, while its necessity emerges in recent years;

+ Designing GNN pre-training schemes is challenging due to the dataset diversity;

+ Recent surge of interest in contrastive learning in computer vision provides us with a potential GNN pre-training scheme.

Methods: Data Augmentation for Graphs

+Data augmentation: creating novel and realistically rational data via certain transformation without affecting the semantics label;

+Little exploration on data augmentations on graphs;

+We propose four general data augmentations for graph-structured data and discuss the intuitive priors that they introduce.

Table 1: Overview of data augmentations for graphs.

Data augmentation	Type	Underlying Prior
Node dropping	Nodes, edges	Vertex missing does not alter semantics.
Edge perturbation	Edges	Semantic robustness against connectivity variations.
Attribute masking	Nodes	Semantic robustness against losing partial attributes per node.
Subgraph	Nodes, edges	Local structure can hint the full semantics.

Graph Contrastive Learning (GraphCL)

- GraphCL: maximizing agreement between two augmented views of graph via a contrastive loss in the latent space.
  - "InfoMax": essentially maximizing a lower bound of two views' mutual information

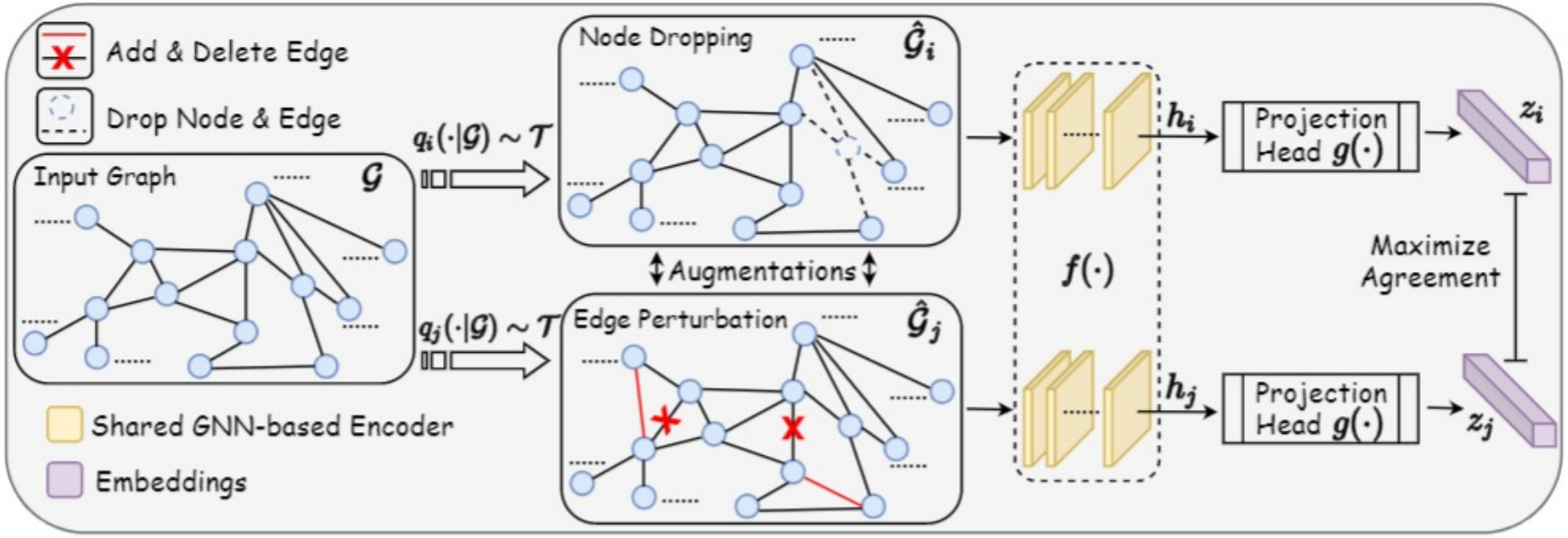


Figure 1: A framework of graph contrastive learning. Two graph augmentations $q_i(\cdot|\mathcal{G})$ and $q_j(\cdot|\mathcal{G})$ are sampled from an augmentation pool $\mathcal{T}$ and applied to input graph $\mathcal{G}$. A shared GNN-based encoder $f(\cdot)$ and a projection head $g(\cdot)$ are trained to maximize the agreement between representations z_i and z_j via a contrastive loss.

The Role of Data Augmentation in GraphCL

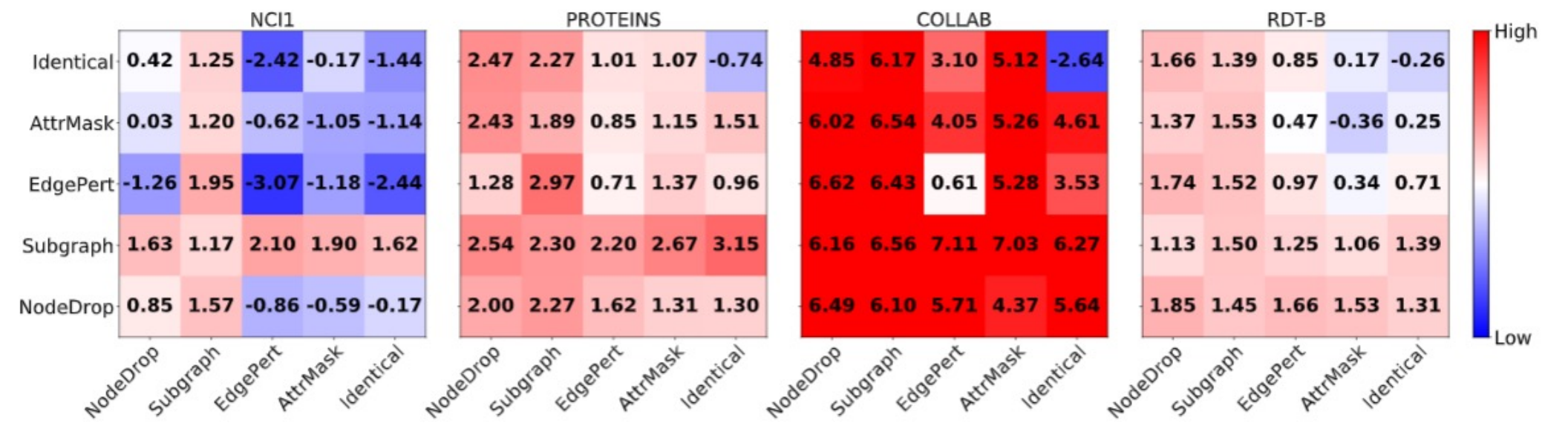


Figure 2: Semi-supervised learning accuracy gain (%) when contrasting different augmentation pairs, compared to training from scratch, under four datasets: NCI1, PROTEINS, COLLAB, and RDT-B. Pairing "Identical" stands for a no-augmentation baseline for contrastive learning, where the positive pair diminishes and the negative pair consists of two non-augmented graphs. Warmer colors indicate better performance gains. The baseline training-from-scratch accuracies are 60.72%, 70.40%, 57.46%, 86.63% for the four datasets respectively.

Table 2: Datasets statistics.

Datasets	Category	Graph Num.	Avg. Node	Avg. Degree
NCI1	Biochemical Molecules	4110	29.87	1.08
PROTEINS	Biochemical Molecules	1113	39.06	1.86
COLLAB	Social Networks	5000	74.49	32.99
RDT-B	Social Networks	2000	429.63	1.15

The Role of Data Augmentation in GraphCL

+ Obs. 1. Data augmentations are crucial in graph contrastive learning;
+ Obs. 2. Composing different augmentations benefits more;
+ Obs. 3. Edge perturbation benefits social networks but hurts some biochemical molecules;
+ Obs. 4. Applying attribute masking achieves better performance in denser graphs;
+ Obs. 5. Node dropping and subgraph are generally beneficial across datasets;
+ Obs. 6. Overly Simple Contrastive Tasks Do Not Help.

Comparison with the State-of-the-arts

+ Semi-supervised learning:

Table 3: Semi-supervised learning with pre-training & finetuning. Red numbers indicate the best performance and the number that overlap with the standard deviation of the best performance (comparable ones). 1% or 10% is label rate; baseline and Aug. represents training from scratch without and with augmentations, respectively.

Dataset	NCI1	PROTEINS	DD	COLLAB	RDT-B	RDT-M5K	GITHUB	MNIST	CIFAR10
1% baseline	60.72±0.45	-	-	57.46±0.25	-	-	54.25±0.22	60.39±1.95	27.36±0.75
1% Aug.	60.49±0.46	-	-	58.40±0.97	-	-	56.36±0.42	67.43±0.36	27.39±0.44
1% GAE	61.63±0.84	-	-	63.20±0.67	-	-	59.44±0.44	57.58±2.07	21.09±0.53
1% Infomax	62.72±0.65	-	-	61.70±0.77	-	-	58.99±0.50	63.24±0.78	27.86±0.43
1% GraphCL	62.55±0.86	-	-	64.57±1.15	-	-	58.56±0.59	83.41±0.33	30.01±0.84
10% baseline	73.72±0.24	70.40±1.54	73.56±0.41	73.71±0.27	86.63±0.27	51.33±0.44	60.87±0.17	79.71±0.65	35.78±0.81
10% Aug.	73.59±0.32	70.29±0.64	74.30±0.81	74.19±0.13	87.74±0.39	52.01±0.20	60.91±0.32	83.99±2.19	34.24±2.62
10% GAE	74.36±0.24	70.51±0.17	74.54±0.68	75.09±0.19	87.69±0.40	53.58±0.13	63.89±0.52	86.67±0.93	36.35±1.04
10% 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	65.21±0.88	83.34±0.24	41.07±0.48
10% 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	65.81±0.79	93.11±0.17	43.87±0.77

+ Unsupervised representation learning:

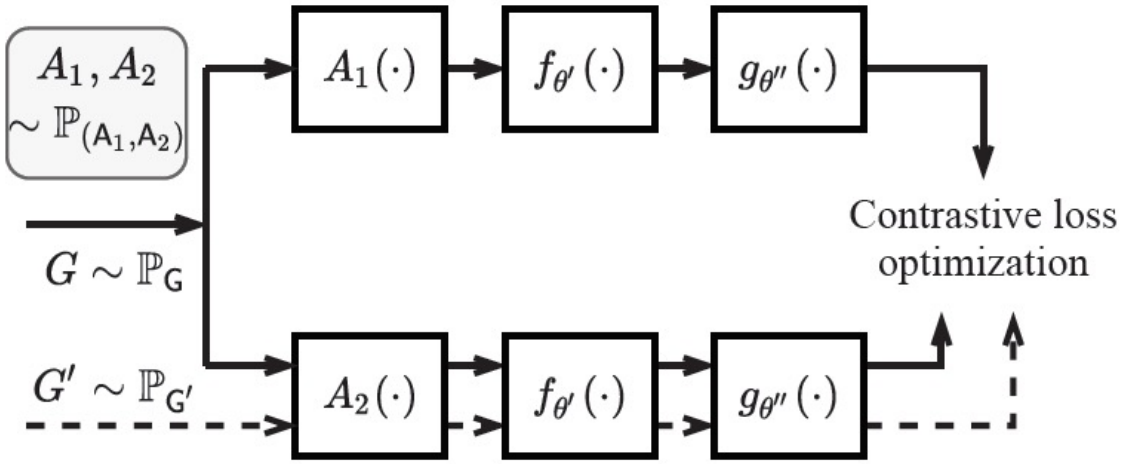
Table 4: Comparing classification accuracy on top of graph representations learned from graph kernels, SOTA representation learning methods, and GIN pre-trained with GraphCL. The compared numbers are from the corresponding papers under the same experiment setting.

Dataset	NCI1	PROTEINS	DD	MUTAG	COLLAB	RDT-B	RDT-M5K	IMDB-B
GL	-	-	-	81.66±2.11	-	77.34±0.18	41.01±0.17	65.87±0.98
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
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
node2vec	54.89±1.61	57.49±3.57	-	72.63±10.20	-	-	-	-
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
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
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
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

GraphCL: The Remaining Gap?

- Unlike images, graph datasets are abstractions of diverse nature (e.g,. pandemics, citation networks, biochemical molecules, or social networks).
- GraphCL constructs specific contrastive views of graph data via hand-picking ad-hoc augmentations for every dataset
- The choice of augmentation follows empirical rules of thumb, typically summarized from many trial-and-error experiments per dataset.
- Can we get more **principled and automated**?

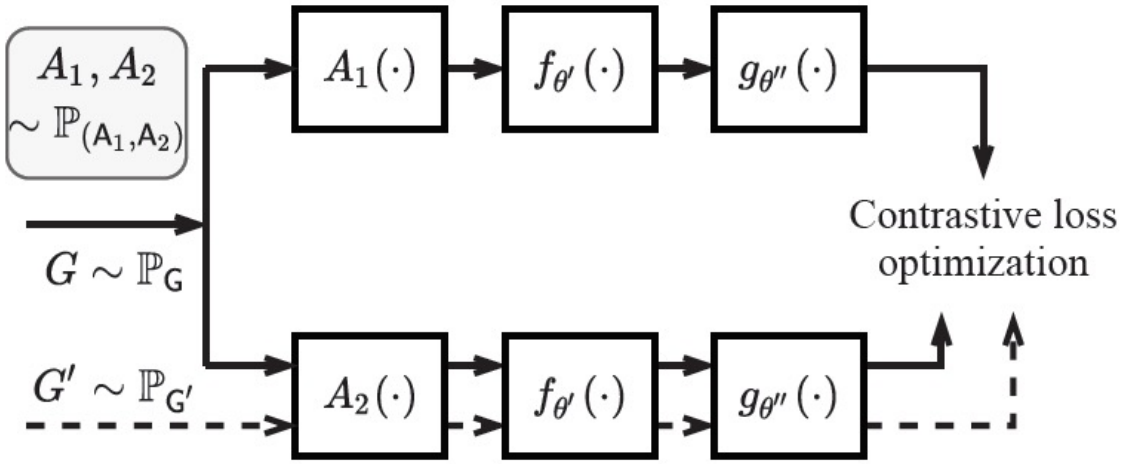
GraphCL Automated: Bi-Level Optimization



$$\min_\theta \quad \mathcal{L}(\mathsf{G}, \mathsf{A}_1, \mathsf{A}_2, \theta),$$
$$\text{s.t.} \quad \mathbb{P}_{(\mathsf{A}_1,\mathsf{A}_2)} \in \arg\min_{\mathbb{P}_{(\mathsf{A}'_1,\mathsf{A}'_2)}} \mathcal{D}(\mathsf{G}, \mathsf{A}'_1, \mathsf{A}'_2, \theta),$$

+ **Upper-level objective L**: the same GraphCL objective
+ **Lower-level objective D**: optimizing the sampling distribution $P(A_1, A_2)$ jointly for augmentation pairs
+ We exploit the signals from the self-supervised training itself, without accessing downstream labeled data

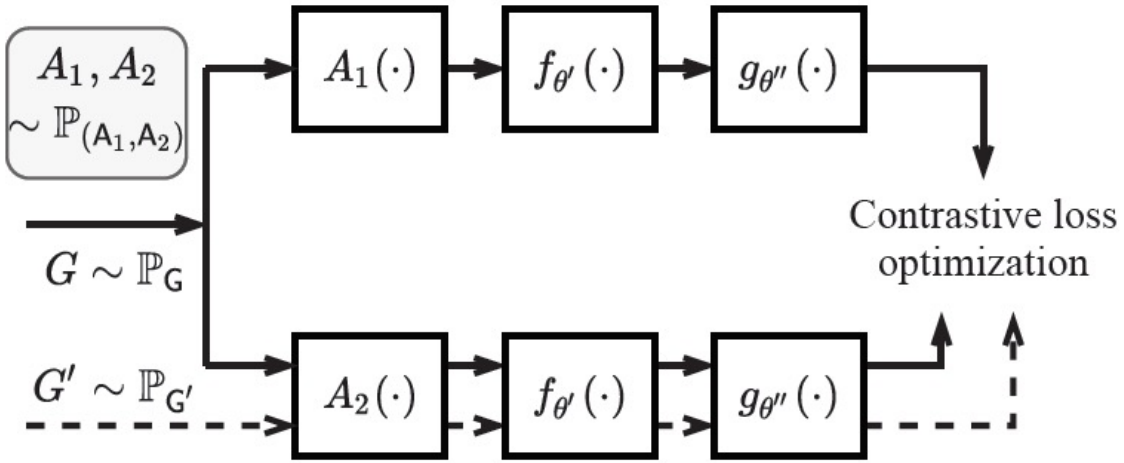
GraphCL Automated: Minimax Principle



$$\min_\theta \quad \mathcal{L}(\mathsf{G}, \mathsf{A}_1, \mathsf{A}_2, \theta),$$
$$\text{s.t.} \quad \mathbb{P}_{(\mathsf{A}_1,\mathsf{A}_2)} \in \arg\max_{\mathbb{P}_{(\mathsf{A}'_1,\mathsf{A}'_2)}} \left\{ \mathcal{L}(\mathsf{G}, \mathsf{A}'_1, \mathsf{A}'_2, \theta) \right\}, \qquad (3)$$

+ Philosophy 1: to always exploit the most challenging augmentations!

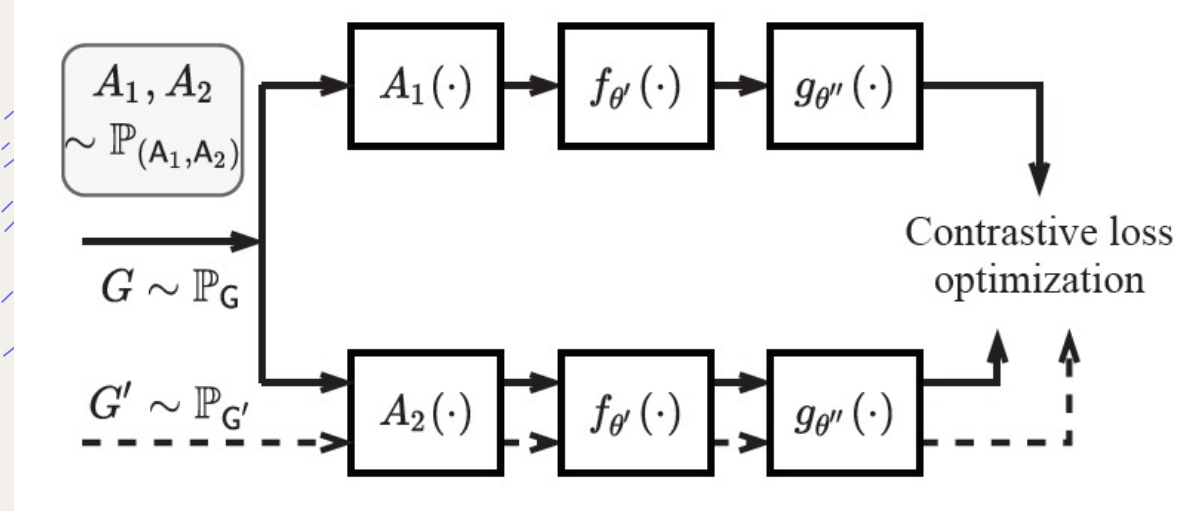
GraphCL Automated: Diversity Principle

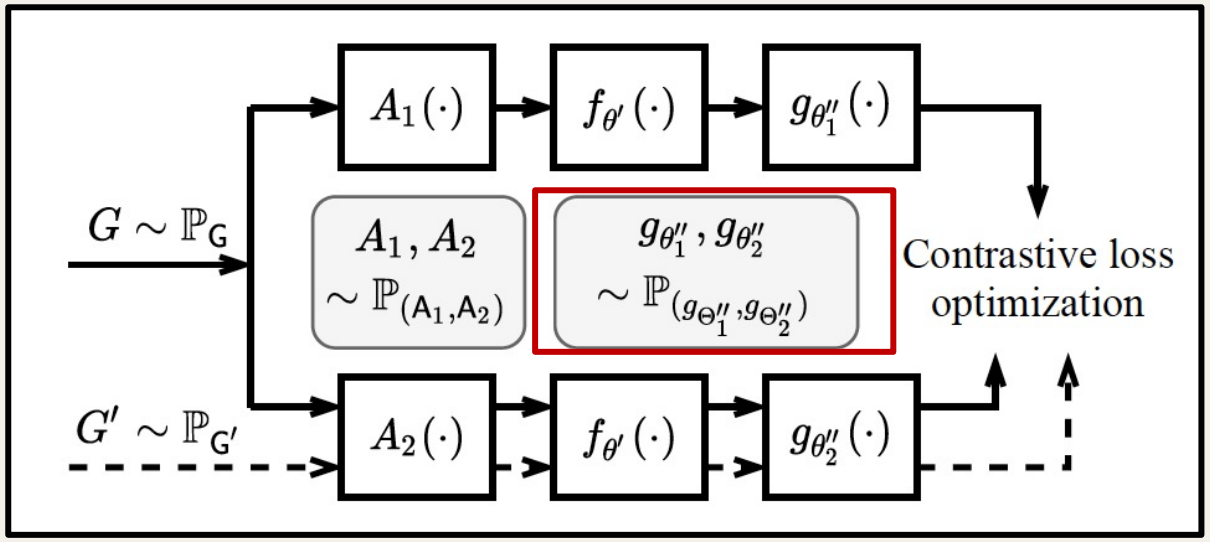


$$\min_\theta \quad \mathcal{L}(\mathsf{G}, \mathsf{A}_1, \mathsf{A}_2, \theta),$$
$$\text{s.t.} \quad \mathbb{P}_{(\mathsf{A}_1,\mathsf{A}_2)} \in \arg\max_{\mathbb{P}_{(\mathsf{A}_1',\mathsf{A}_2')}} \Big\{ \mathcal{L}(\mathsf{G}, \mathsf{A}_1', \mathsf{A}_2', \theta) - \frac{\gamma}{2} \operatorname{dist}(\mathbb{P}_{(\mathsf{A}_1',\mathsf{A}_2')}, \mathbb{P}_{\text{prior}}) \Big\}, \tag{3}$$

+ Philosophy 1: to always exploit the most challenging augmentations!
+ Philosophy 2: avoid selection ”collapse” and choose P_{prior} as the uniform distribution for diversity

GraphCL Automated: Conditional Projection Heads





+ Philosophy 1: to always exploit the most challenging augmentations!
+ Philosophy 2: avoid selection ”collapse” and choose P_{prior} as the uniform distribution for diversity
+ Architecture modification: more augmentations -> more “augmentation-specific” projection heads

Performance Overview: GraphCL Automated

- Obs. 1. Across datasets originated from diverse sources, automated augmentation selection performs comparably to GraphCL with exhaustively hand-tuned augmentation rules
- Obs. 2. Automatically discovered augmentations largely recover the ”best practice” discovered in previous GraphCL (by exhaustive hand tuning)
- Obs. 3. On “unseen” datasets from specific bioinformatics domains, we achieve better performance than GraphCL whose empirical rules were not derived from such data, indicating better generalizability to unseen datasets
- Obs. 4. Our method outperforms heuristic self-supervised methods, with few exceptions. It also scales up well to larger graph datasets, e.g., OGB (ogbg-ppa, ogbg-code).

Uprising field, and still way to go!

+ Heterogeneous is the future key
+ Scaling up, and efficient training
+ More automated “priors”
+ Imbalanced graph and “cold-start”
+ Online and continual learning
+ …..

Reference

Y. You et. al., *“Graph Contrastive Learning Automated”*, ICML 2021 (long oral)

Y. You et. al. *“Graph Contrastive Learning with Augmentations”*, NeurIPS 2020

Y. You et. al. *“When Does Self-Supervision Help Graph Convolutional Networks?”*, ICML 2020

Thank you!

Q&A Please

VITA