Contrastive Learning on Graphs and Some Interpretations

Shengchao Liu
MILA & UdeM

- Backgrounds for Contrastive Learning

- Applications for Contrastive Learning on Images and Graphs

- Some Deeper Insights

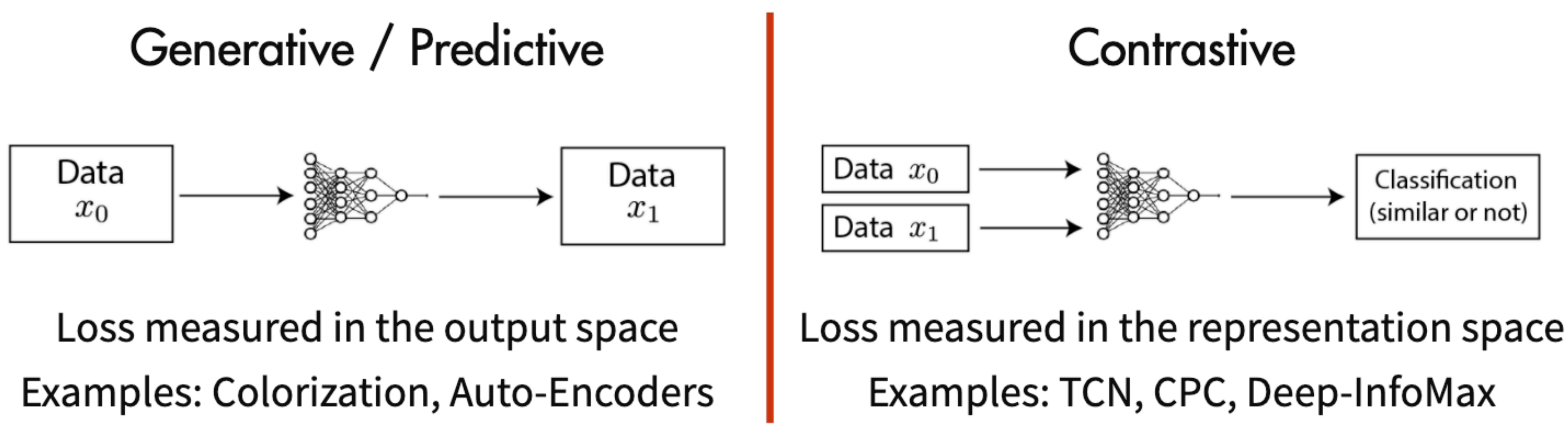
- Comments

Most are about contrastive learning, yet some belong to more general self-supervised learning.
Put them here since they are closely connected.

- Backgrounds for Contrastive Learning
- Applications for Contrastive Learning on Images and Graphs
- Some Deeper Insights
- Comments

Backgrounds for Contrastive Learning

- How to better utilize the unlabeled data?
  - One solution is by pertaining on the self-supervised tasks.
  - Such self-supervised learning can be roughly categorized as 2 types



Credit to Ankesh Anand's blog.

Backgrounds for Contrastive Learning

- Intuitional Motivation: $\underline{\text{score}(f(x), f(x^+))} >> \underline{\text{score}(f(x), f(x^-))}$
  - Anchor point
  - Positive/negative sample
  - Score on the representation
- Objective (InfoNCE): $$\mathcal{L}_N = -\mathbb{E}_X \left[\log \frac{\overline{\exp\left(f(x)^T f\left(x^+\right)\right)}}{\underline{\exp\left(f(x)^T f\left(x^+\right)\right)} + \sum_{j=1}^{N-1} \underline{\exp\left(f(x)^T f\left(x_j\right)\right)}} \right]$$
- Theoretical Motivation: $$I(X;Y) \geq \mathbb{E}\left[\frac{1}{K} \sum_{i=1}^{K} \log \frac{p(y_i|x_i)}{\frac{1}{K}\sum_{j=1}^{K} p(y_i|x_j)} \right],$$

Mutual Information (MI) is bounded by the InfoNCE. There also exist other bounds on MI, check On Variational Bounds of Mutual Information, ICML'19.

Backgrounds for Contrastive Learning

- The Most Common Objective for Contrastive Learning (InfoNCE):

$$\mathcal{L}_N = -\mathbb{E}_X \left[\log \frac{\overline{\exp\left(f(x)^T f\left(x^+\right)\right)}}{\underline{\exp\left(f(x)^T f\left(x^+\right)\right)} + \sum_{j=1}^{N-1} \underline{\exp\left(f(x)^T f\left(x_j\right)\right)}} \right]$$

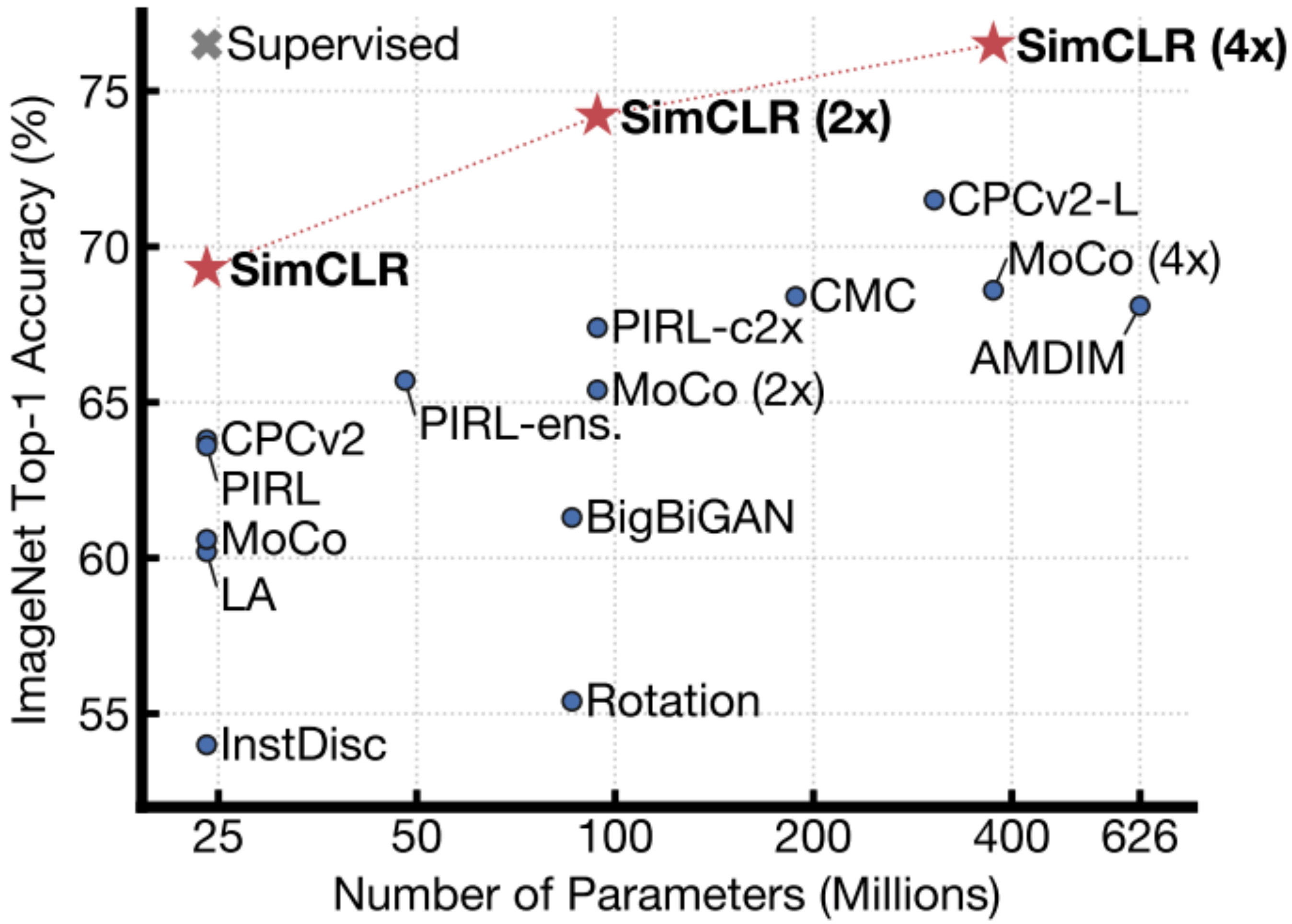
- Two Key Challenges:

1. How to choose/design good positive and negative pairs for different applications?
2. Why does contrastive learning work? And how this can contribute to the design of positive/negative pairs?

- Backgrounds for Contrastive Learning
- Applications for Contrastive Learning on Images and Graphs
- Some Deeper Insights
- Comments

Application: Contrastive Learning on Images

- IR, CVPR'18
- LA, ICCV'19
- CPC, ArXiv'18
- Deep InfoMax (DIM), ICLR'19
- CMC, ArXiv'19
- SimCLR, ICML'20
- SimCLRv2, ArXiv'20



More details can be found in group slack or https://chao1224.github.io/material/slides/202006.pdf

Application: Contrastive Learning on Images

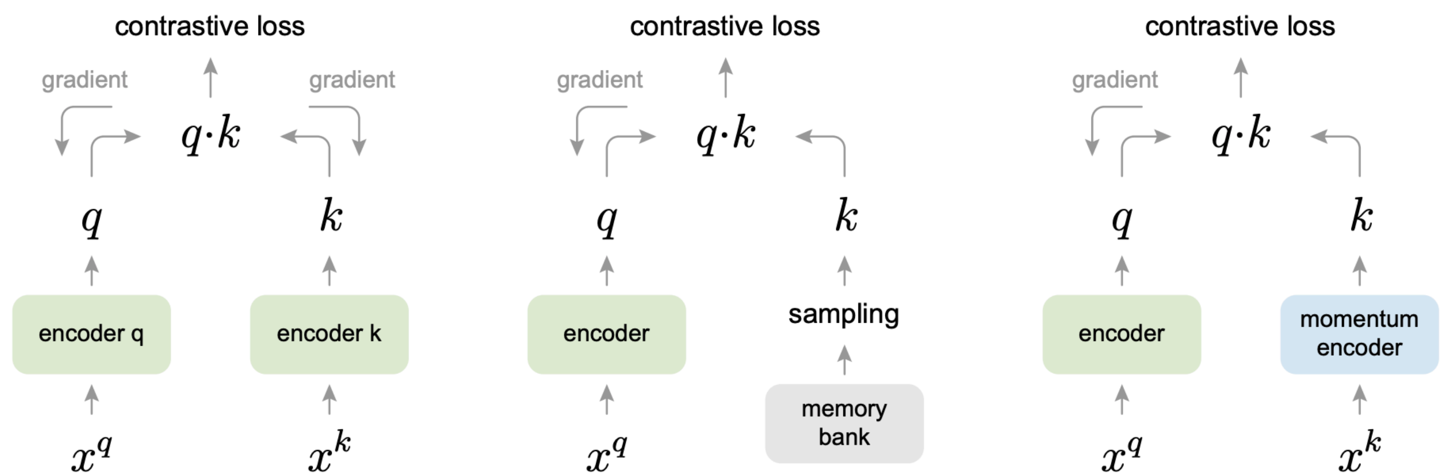
- Positive/negative pairs are from two views of same/different images.

$$\mathcal{L}_N = -\mathbb{E}_X \left[\log \frac{\exp\left(f(x)^T f\left(x^+\right)\right)}{\exp\left(f(x)^T f\left(x^+\right)\right) + \sum_{j=1}^{N-1} \exp\left(f(x)^T f\left(x_j\right)\right)} \right]$$

- Potential views include:
  1. Clustering view: Images belong to the same cluster are positive (LA)
  2. Data augmentation view: Images (Augmented) from the same image are positive (IR, CMC, SimCLR, SimCLRv2, etc.)
  3. Local and global view: Local patch and global image representation from the same image are positive (DIM)

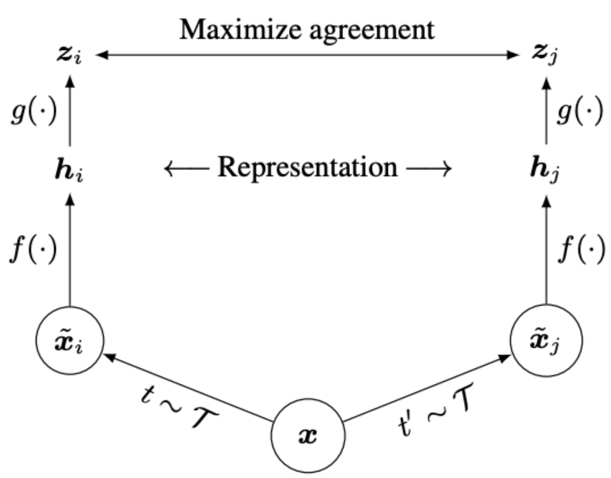
Application: Contrastive Learning on Images

- Some useful tips:
  - Memory Bank, MoCo $\theta_k \leftarrow m\theta_k + (1-m)\theta_q.$



(a) end-to-end (b) memory bank (c) MoCo

  - Projection Head (SimCLR, SimCLR-V2)



Application: Contrastive Learning on Graphs

Method	Local Embedding	Global Embedding
EdgePred	nearby and disparate nodes discrimination	—
DGI	(node-graph) discrimination	
InfoGraph	(node-graph) discrimination & supervised and unsupervised discrimination	
Contrastive Multi-View Graph	(node-graph) discrimination	
Pre-Training	context prediction & attribute masking	property prediction on large datasets[1]
GCC	substructure/neighborhood discrimination	—
GROVER*	contextual prediction on nodes & edges	motif prediction
ASGN	node and distance prediction	molecular graph clustering

For more details, please feel free to check our survey paper.

Application: Contrastive Learning on Graphs

- **[1] Edge Prediction (GraphSAGE), NIPS'17:**
  - Nearby nodes are positive, otherwise negative.

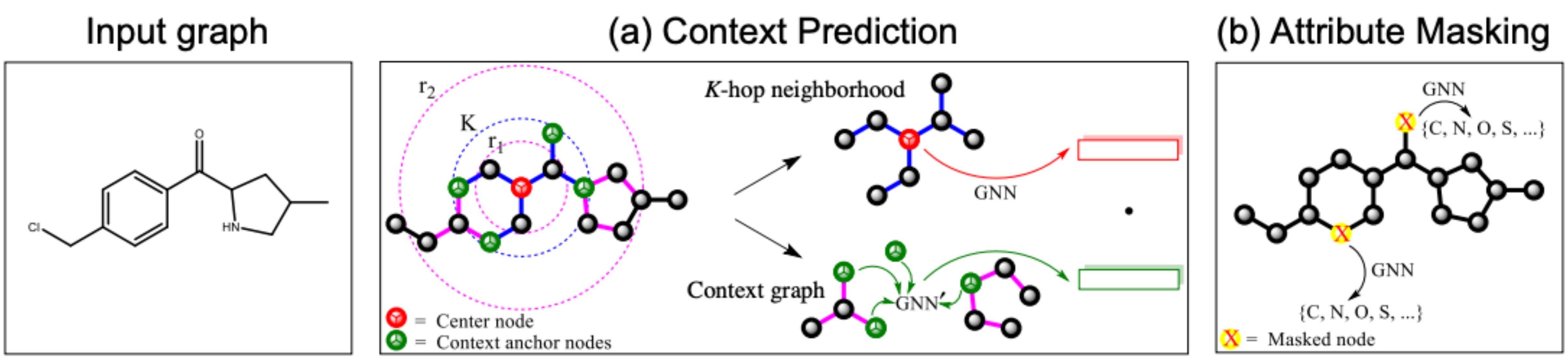
$$J_{\mathcal{G}}(\mathbf{z}_u) = -\log\left(\sigma(\mathbf{z}_u^\top \mathbf{z}_v)\right) - Q \cdot \mathbb{E}_{v_n \sim P_n(v)} \log\left(\sigma(-\mathbf{z}_u^\top \mathbf{z}_{v_n})\right),$$

- **[2] Deep Graph Infomax (DGI), ICLR'19 / InfoGraph, NIPS'19**
  - Contrast local (node) and global (graph) representation.
  - Local and global pairs from the same/different graphs are positives/ negatives.

$$I_{\phi,\psi}(h_\phi^i(G); H_\phi(G)) :=$$
$$\mathbb{E}_{\mathbb{P}}[-\mathrm{sp}(-T_{\phi,\psi}(\vec{h}_\phi^i(x), H_\phi(x)))] - \mathbb{E}_{\mathbb{P}\times\tilde{\mathbb{P}}}[\mathrm{sp}(T_{\phi,\psi}(\vec{h}_\phi^i(x'), H_\phi(x)))]$$

[3] Strategies for Pre-training Graph Neural Networks, ICLR’19

- 2 node-level pre-training methods:
  - Masking Node/Edge Attribute
  - Context Prediction
    - **Subgraph**: K-hop neighborhood
    - Context graph: a region between r_1-hop and r_2-hop
    - **Context anchor nodes**: between r_1-hop and K-hop
    - Use context anchor nodes to predict subgraph
      - Subgraph-Context pairs with the same/different center nodes are positive/negative



[3] Strategies for Pre-training Graph Neural Networks, ICLR'19

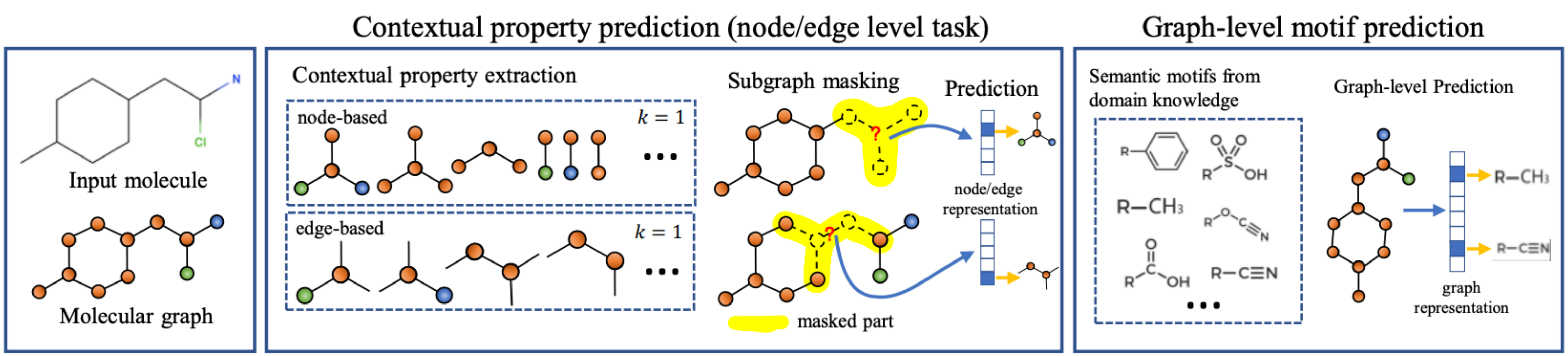
- Graph-Level
  - Supervised training on ChEMBL datasets
  - 450k chemicals and 1.3k tasks

- Experiments
  - Transfer from more common scaffolds to less common ones.

Dataset		BBBP	Tox21	ToxCast	SIDER	ClinTox	MUV	HIV	BACE	**Average**
# Molecules		2039	7831	8575	1427	1478	93087	41127	1513	/
# Binary prediction tasks		1	12	617	27	2	17	1	1	/
Pre-training strategy		Out-of-distribution prediction (scaffold split)								
Graph-level	Node-level									
–	–	65.8 ±4.5	74.0 ±0.8	63.4 ±0.6	57.3 ±1.6	58.0 ±4.4	71.8 ±2.5	75.3 ±1.9	70.1 ±5.4	67.0
–	Infomax	**68.8 ±0.8**	75.3 ±0.5	62.7 ±0.4	58.4 ±0.8	69.9 ±3.0	75.3 ±2.5	76.0 ±0.7	75.9 ±1.6	70.3
–	EdgePred	67.3 ±2.4	76.0 ±0.6	64.1 ±0.6	60.4 ±0.7	64.1 ±3.7	74.1 ±2.1	76.3 ±1.0	79.9 ±0.9	70.3
–	AttrMasking	64.3 ±2.8	76.7 ±0.4	64.2 ±0.5	61.0 ±0.7	71.8 ±4.1	74.7 ±1.4	77.2 ±1.1	79.3 ±1.6	71.1
–	ContextPred	68.0 ±2.0	75.7 ±0.7	63.9 ±0.6	60.9 ±0.6	65.9 ±3.8	75.8 ±1.7	77.3 ±1.0	79.6 ±1.2	70.9
Supervised	–	68.3 ±0.7	77.0 ±0.3	64.4 ±0.4	62.1 ±0.5	57.2 ±2.5	79.4 ±1.3	74.4 ±1.2	76.9 ±1.0	70.0
Supervised	Infomax	68.0 ±1.8	77.8 ±0.3	64.9 ±0.7	60.9 ±0.6	**71.2 ±2.8**	**81.3 ±1.4**	77.8 ±0.9	80.1 ±0.9	72.8
Supervised	EdgePred	66.6 ±2.2	**78.3 ±0.3**	**66.5 ±0.3**	**63.3 ±0.9**	70.9 ±4.6	78.5 ±2.4	77.5 ±0.8	79.1 ±3.7	72.6
Supervised	AttrMasking	66.5 ±2.5	77.9 ±0.4	65.1 ±0.3	**63.9 ±0.9**	**73.7 ±2.8**	**81.2 ±1.9**	77.1 ±1.2	80.3 ±0.9	73.2
Supervised	ContextPred	**68.7 ±1.3**	**78.1 ±0.6**	65.7 ±0.6	62.7 ±0.8	**72.6 ±1.5**	**81.3 ±2.1**	**79.9 ±0.7**	**84.5 ±0.7**	**74.2**

[4] GROVER: Self-supervised Message Passing Transformer on Large-scale Molecule Data, NIPS'20 In Submission

- Node/edge level task: subgraph masking
- Graph-level motif prediction

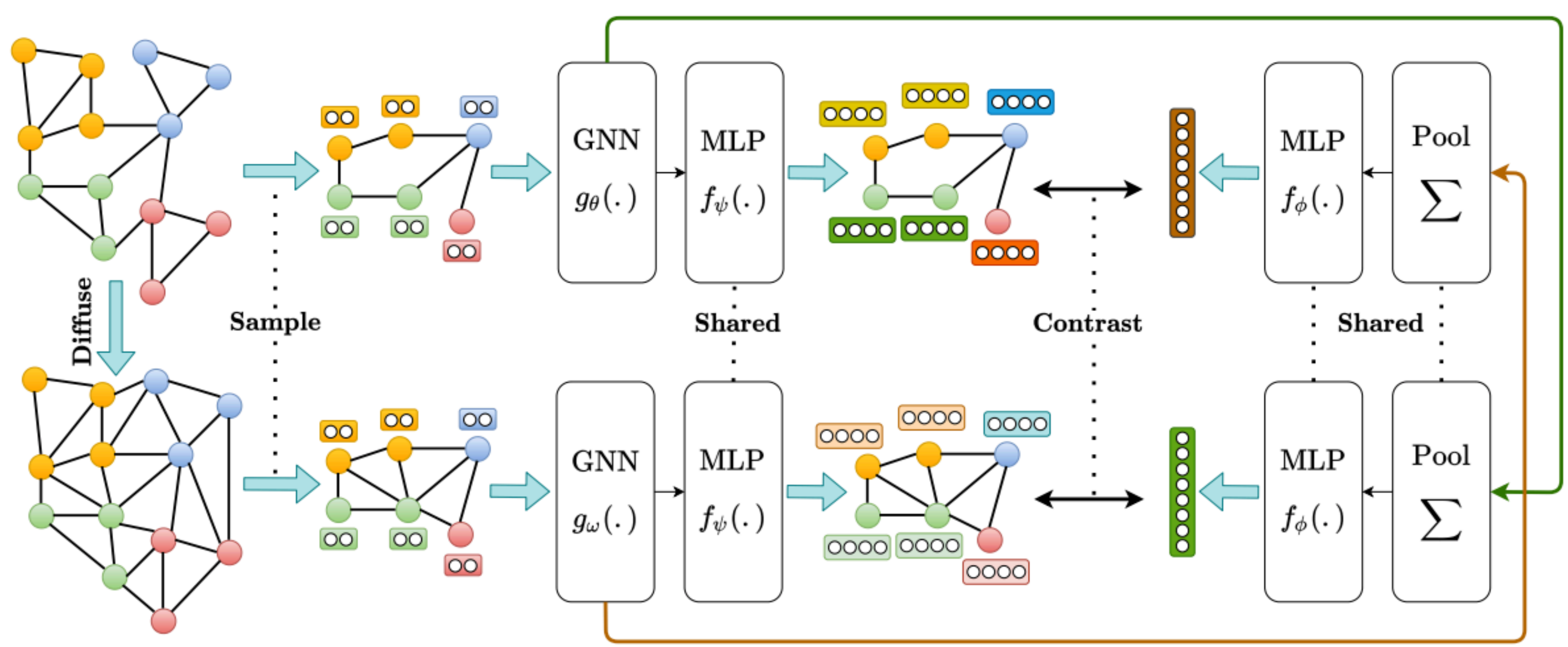


- Other details:
  - A novel base GNN model: dynamic GNN (dyMPN)
  - GNN Transformer
  - …

[5] Contrastive Multi-View Representation Learning on Graphs, ICML'20

- Graph Diffusion as data/graph augmentation
  - Transform the adjacency matrix to a diffusion matrix
  - Take the two matrices as congruent views of the same graph.

$$\max_{\theta,\omega,\phi,\psi} \frac{1}{|\mathcal{G}|} \sum_{g \in \mathcal{G}} \left[\frac{1}{|g|} \sum_{i=1}^{|g|} \left[\mathbf{MI}\left(\vec{h}_i^{\alpha}, \vec{h}_g^{\beta}\right) + \mathbf{MI}\left(\vec{h}_i^{\beta}, \vec{h}_g^{\alpha}\right) \right] \right]$$



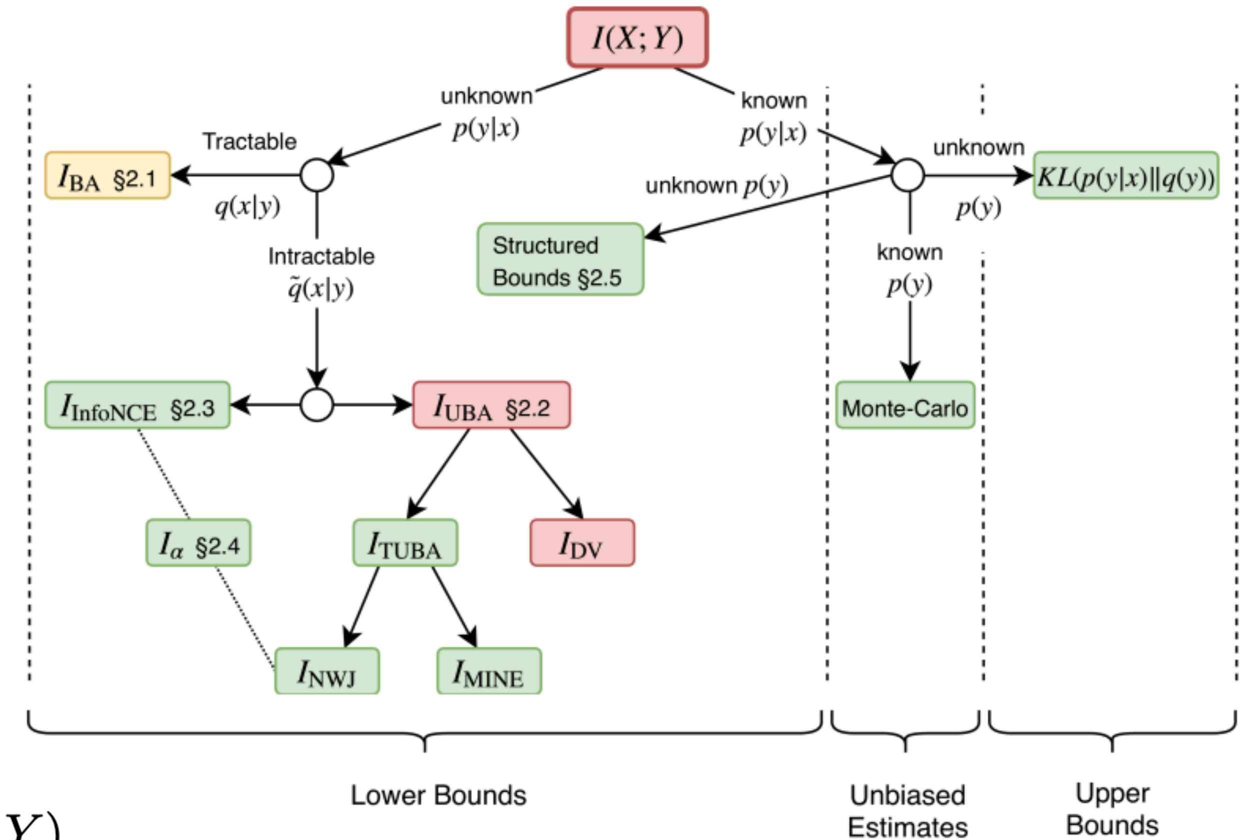
[5] <u>Contrastive Multi-View Representation Learning on Graphs</u>, ICML'20

- Other interesting observations/conclusions:
  - Increasing the number of views doesn't help (for graph).
  - A simple readout is better than complicated pooling functions like DiffPool.
  - Applying regularization or normalization has a negative effect. (?)

- Backgrounds for Contrastive Learning
- Applications for Contrastive Learning on Images and Graphs
- Some Deeper Insights
- Comments

- [1] On Mutual Information Maximization For Representation Learning
- [2] Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere
- [3] Bootstrap Your Own Latent A New Approach to Self-Supervised Learning
- [4] When Does Self-Supervision Help Graph Convolutional Networks?

[1] On Mutual Information Maximization For Representation Learning, ICLR'20



- Various bounds on mutual information

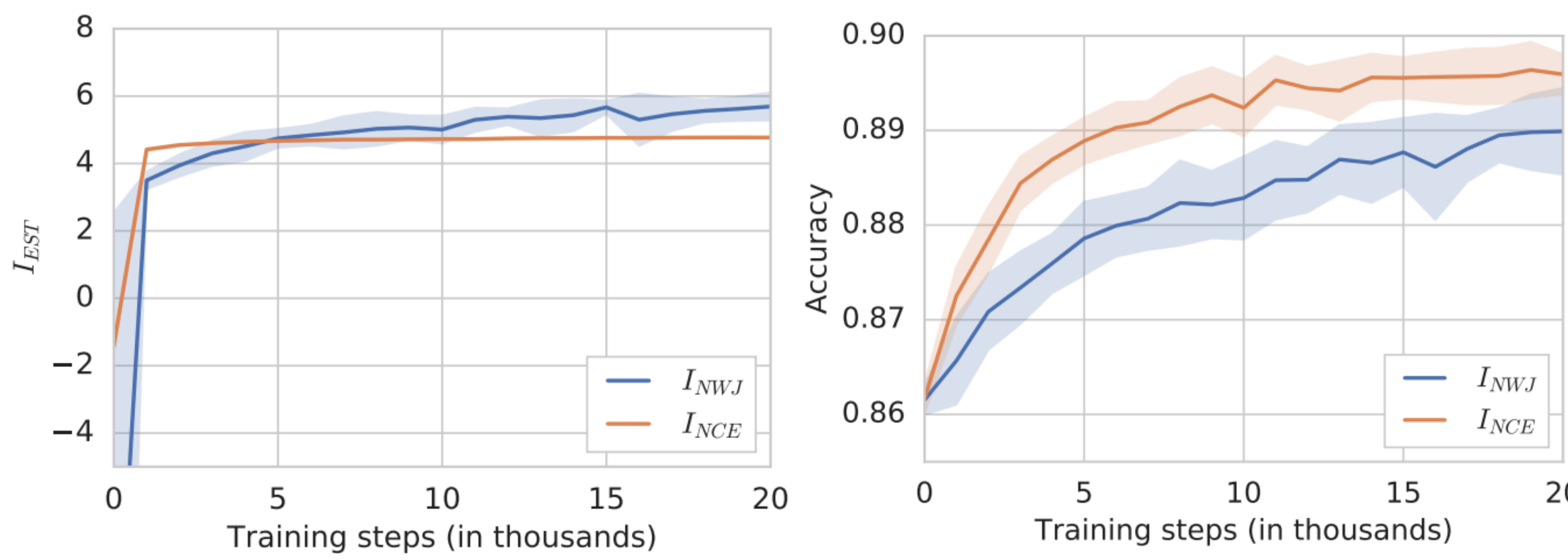
- InfoNCE

$$I(X;Y) \geq \mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K}\log\frac{e^{f(x_i,y_i)}}{\frac{1}{K}\sum_{j=1}^{K}e^{f(x_i,y_j)}}\right] \triangleq I_{\text{NCE}}(X;Y),$$

- Variational form of KL-D (NWJ)

$$I(X;Y) \geq \mathbb{E}_{p(x,y)}[f(x,y)] - e^{-1}\mathbb{E}_{p(x)}[\mathbb{E}_{p(y)}e^{f(x,y)}] \triangleq I_{\text{NWJ}}(X;Y).$$

- Observation:



(a) (b)

[1] On Mutual Information Maximization For Representation Learning, ICLR'20

- Connection to Deep Metric Learning
  - InfoNCE

$$I_{\text{NCE}} = \mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K}\log\frac{e^{f(x_i,y_i)}}{\frac{1}{K}\sum_{j=1}^{K}e^{f(x_i,y_j)}}\right] = \log K - \mathbb{E}\left[\frac{1}{K}\sum_{i=1}^{K}\log\left(1+\sum_{j\neq i}e^{f(x_i,y_j)-f(x_i,y_i)}\right)\right].$$

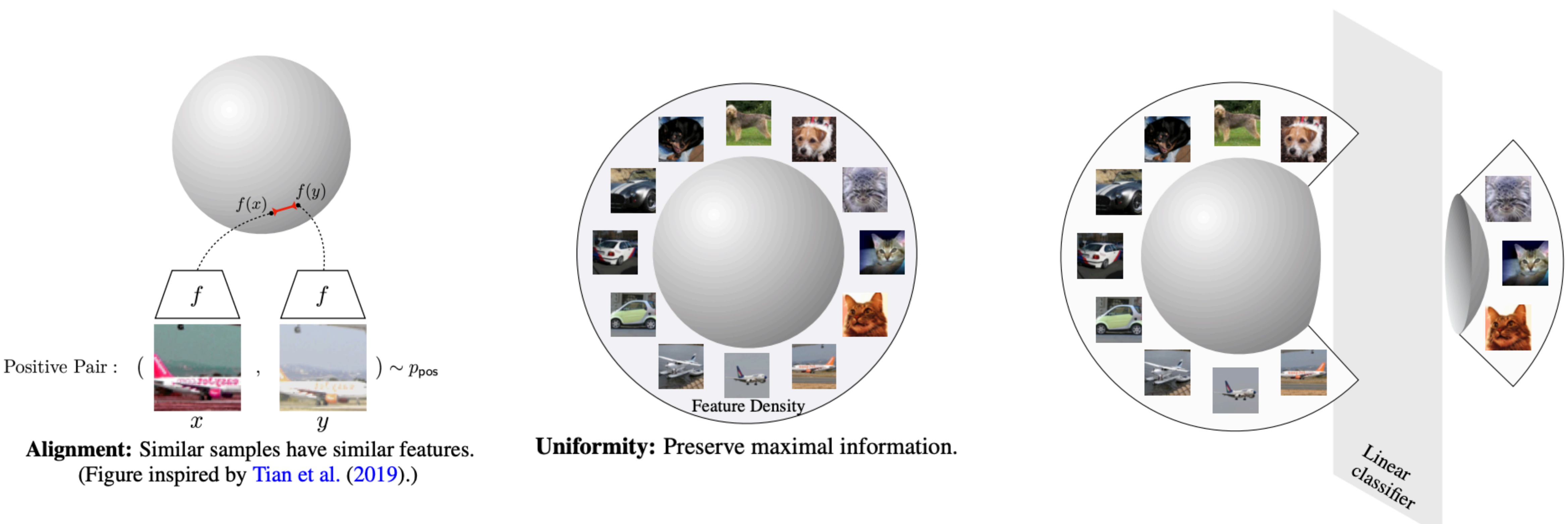
  - Multi-class K-pair loss (?)

$$L_{\text{K-pair-mc}}\left(\{(x_i,y_i)\}_{i=1}^{K},\phi\right) = \frac{1}{K}\sum_{i=1}^{K}\log\left(1+\sum_{j\neq i}e^{\phi(x_i)^{\top}\phi(y_j)-\phi(x_i)^{\top}\phi(y_i)}\right).$$

  - Maximizing InfoNCE by using a critic $f(x, y) = \phi(x)^T\phi(y)$, thus is equivalent to metric learning.

[2] Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere, ICML'20

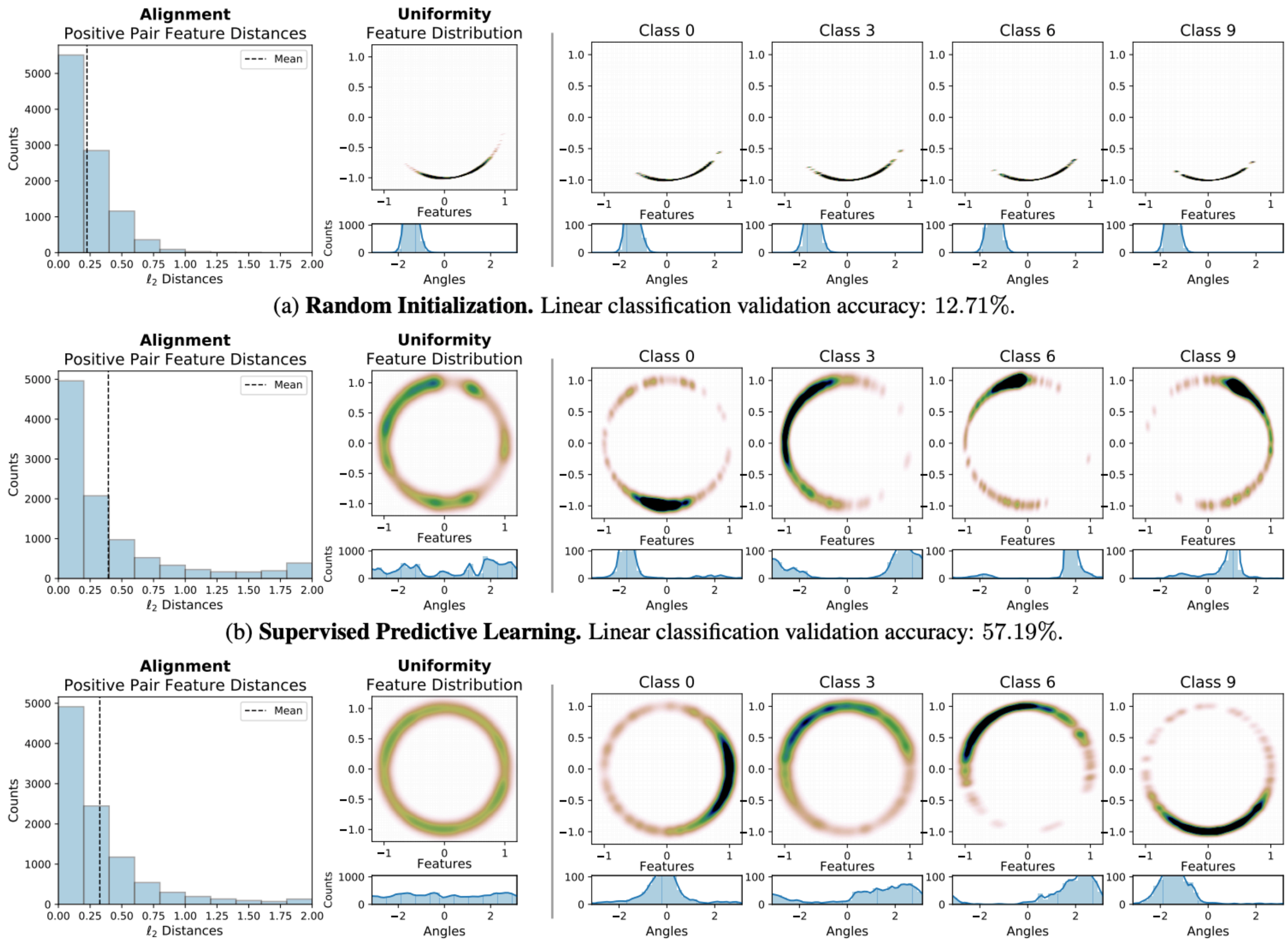
- $L_{contrastive} = \mathbb{E}_{(x,y)\sim p_{pos}}[-f(x)^T f(y)/\tau] + \mathbb{E}_{(x,y)\sim p_{pos}, x\sim p_{data}}[\log(\exp(f(x)^T f(y)/\tau) + \sum_i \exp(f(x)^T f(x_i)/\tau))]$
- Two key properties of contrastive loss, with metric to quantify each property
  - Alignment/closeness: Learned pos pairs should be similar, thus invariant to noise factors. $L_{align}(f) = -\mathbb{E}_{(x,y)\sim p_{pos}}[\|f(x) - f(y)\|_2^\alpha], \alpha > 0$
  - Uniformity: features should be roughly uniformly distributed on the unit hypersphere. $L_{uniform} = \log \mathbb{E}_{(x,y)\sim p_{data}}[\exp(-t\|f(x) - f(y)\|_2^2)], t > 0$



Alignment: Similar samples have similar features. (Figure inspired by Tian et al. (2019).)

Uniformity: Preserve maximal information.

[2] Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere, ICML'20



(a) **Random Initialization.** Linear classification validation accuracy: 12.71%.

(b) **Supervised Predictive Learning.** Linear classification validation accuracy: 57.19%.

(c) **Unsupervised Contrastive Learning.** Linear classification validation accuracy: 28.60%.

[2] Understanding Contrastive Representation Learning through Alignment and Uniformity on the Hypersphere, ICML'20

- InfoMax principle: maximizing the mutual information $\max I(f(x), f(y)), \forall (x, y) \sim p_{pos}$.

$$
\begin{aligned}
&\lim_{M \to \infty} \mathcal{L}_{\mathsf{contrastive}}(f; \tau, M) - \log M = \\
&\qquad - \frac{1}{\tau} \mathop{\mathbb{E}}_{(x,y) \sim p_{\mathsf{pos}}} \left[f(x)^\mathsf{T} f(y) \right] \\
&\qquad + \mathop{\mathbb{E}}_{x \sim p_{\mathsf{data}}} \left[\log \mathop{\mathbb{E}}_{x^- \sim p_{\mathsf{data}}} \left[e^{f(x^-)^\mathsf{T} f(x) / \tau} \right] \right].
\end{aligned}
$$

- Theorem 1: Perfectly alignment and perfectly uniform are solutions to the first and second term.
- This paper concludes: Instead of interpreted with InfoMAX, what contrastive loss doing is to learn an aligned and information-preserving encoder.

[3] Bootstrap Your Own Latent A New Approach to Self-Supervised Learning, In Submission NeurIPS'20

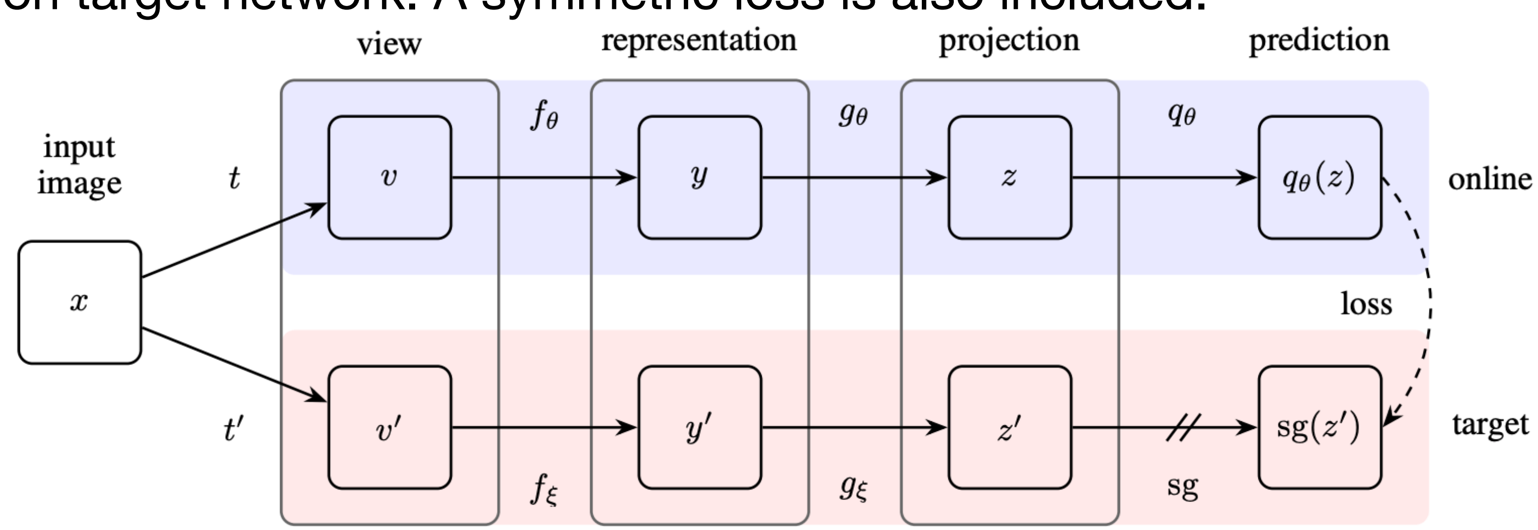
- Comparison between BYOL and contrastive learning
  - No negative sampling
  - More robust to the choice of image augmentation
  - Iteratively refine its representation
- Two networks and two views.
  1. Online network: $v_1 \rightarrow f_\theta, g_\theta \rightarrow z_1$
  2. Target network: $v_2 \rightarrow f_\xi, g_\xi \rightarrow z_2$
  3. Use online network (representation) to predict target network (representation)

$$\|\bar{q}_\theta(z_1) - \bar{z}_2\|^2 = 2 - 2 \cdot \frac{q_\theta(z_1)^T, z_2}{\|q_\theta(z_1)\|_2 \|z_2\|_2}$$

- Above is v_1 on online network and v_2 on target network. A symmetric loss is also included.

Note: This is Self-Training, No Contrasting

Note: SimCLR suggests adding projection

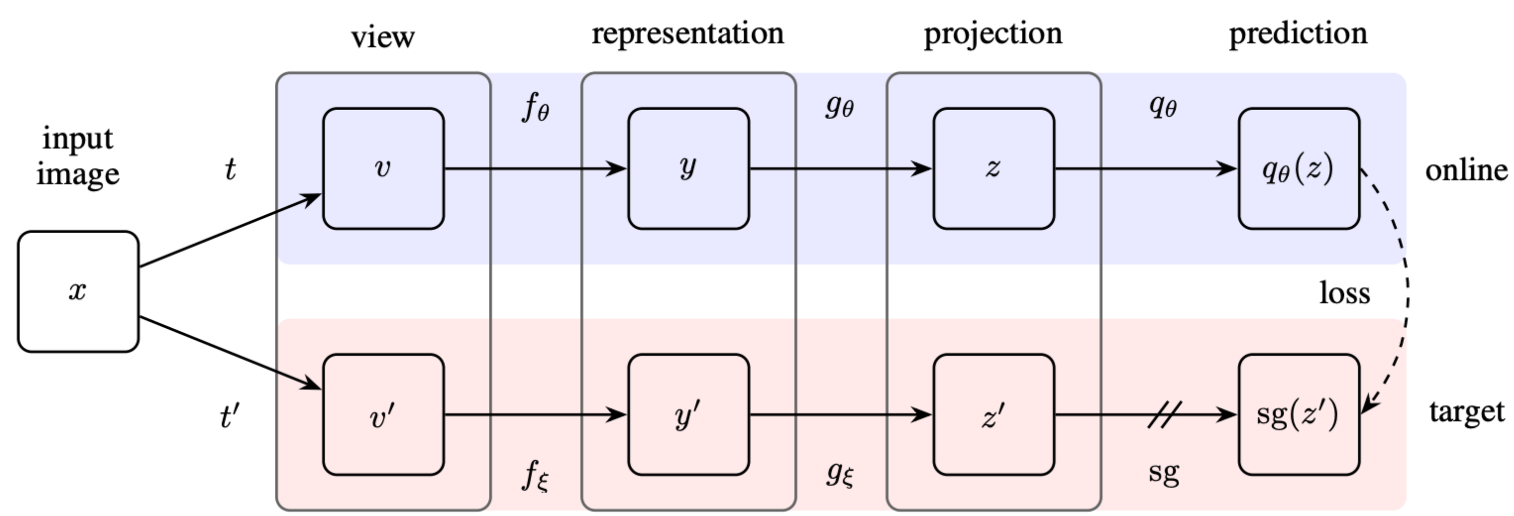


[3] Bootstrap Your Own Latent A New Approach to Self-Supervised Learning, In Submission NeurIPS'20

- Update target weights: $\xi \leftarrow \tau\xi + (1-\tau)\theta.$
- Goal: y as the final representation
- An ablation study:
  - Randomly initialized network is uniform, but not well aligned. (?)
  - Applying BYOL with $\tau = 1$ does learn a useful representation.

Target	τ_{base}	Top-1
Constant random network	1	18.8±0.7
Moving average of online	0.999	69.8
Moving average of online	0.99	**72.5**
Moving average of online	0.9	68.4
Stop gradient of online[†]	0	0.3

  - Explanation, follow the idea from [2]:
    - BYOL is explicitly doing alignment, no uniformity.
    - (Conjecture): Moving average is scattering features, implicitly doing uniformity.



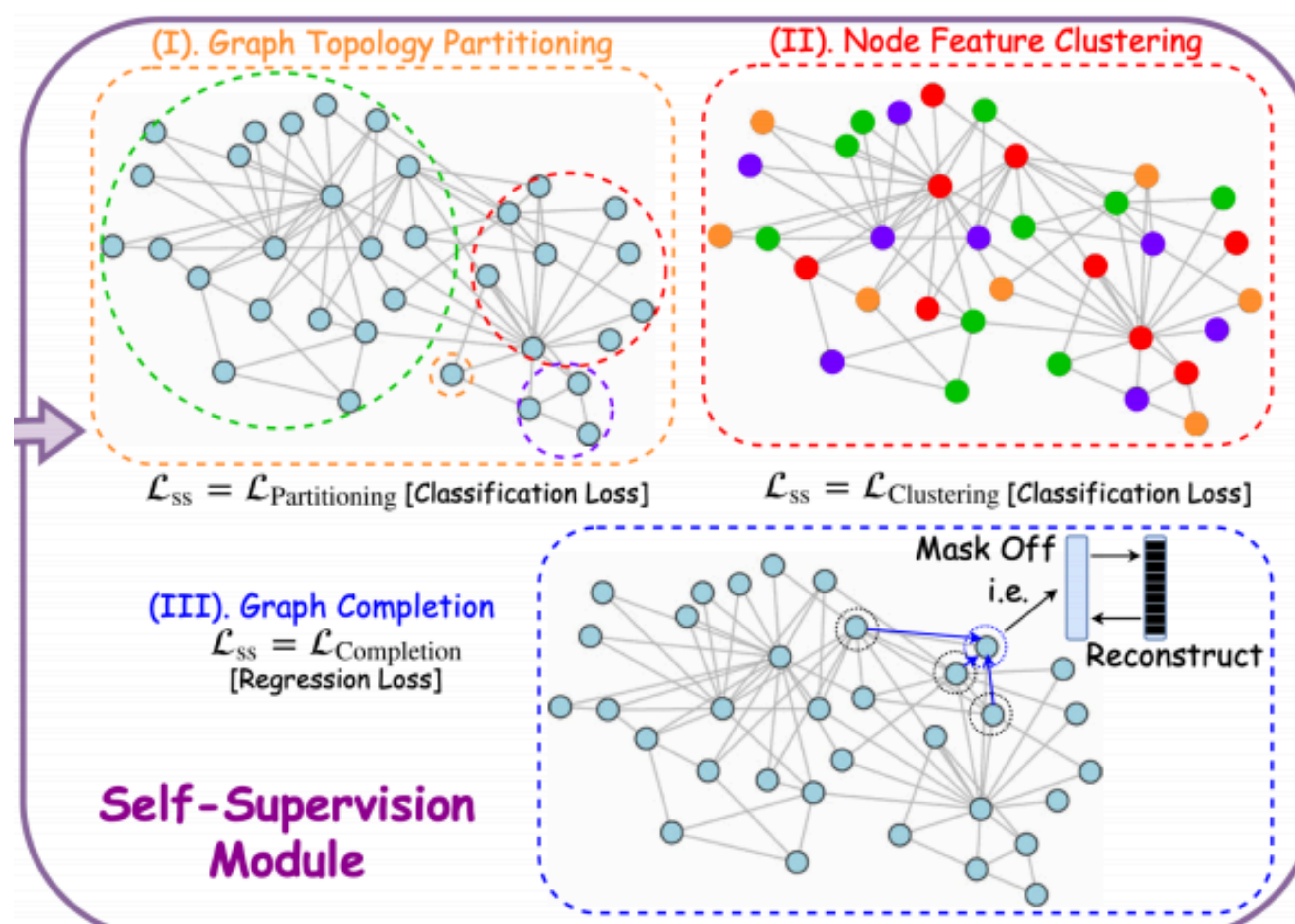
[4] <u>When Does Self-Supervision Help Graph Convolutional Networks?</u>, ICML'20

- Motivation:
  - Consider transductive semi-supervised setting for GCN: which makes predictions on unlabeled data (nodes/edges)
  - Self-supervised learning is better at utilizing the unlabeled data

- Three schemes to combine self-supervision and our target task
  - Pre-training & fine-tuning: sequentially transfer
  - Self-training: incrementally transfer
  - Multi-task learning: simultaneously transfer

[4] When Does Self-Supervision Help Graph Convolutional Networks?, ICML'20

- Three pretext tasks:
  - Node clustering: nodes with similar features tend to be similar
  - Graph partitioning: nodes with more connections tend to be similar (similar to node clustering, while the objective is to minimize edgecut)
  - Graph completion: masking and reconstruction



- (Adversarial Defense)

[4] When Does Self-Supervision Help Graph Convolutional Networks?, ICML'20

- Experiment 1
  - 3 schemes: pre-training & fine-tuning (**P&F**), self-training (**M3S**), multi-task (**MTL**)
  - 3 pretext tasks: Node **Clu**stering, Graph **Par**titioning, Graph **Comp**letion
  - Conclusion:
    - P&F does help, but not on larger datasets like Citeseer and PubMed.
    - Conjecture: though pre-training can learn graph structure, such info will be lost during fine-tuning; GCN is too shallow
    - MTL is more general

	Cora	Citeseer	PubMed
GCN	81.00 ± 0.67 81.5	70.85 ± 0.70 70.3	79.10 ± 0.21 79.0
P&F-Clu	81.83 ± 0.53	71.06 ± 0.59	79.20 ± 0.22
P&F-Par	81.42 ± 0.51	70.68 ± 0.81	79.19 ± 0.21
P&F-Comp	81.25 ± 0.65	71.06 ± 0.55	79.19 ± 0.39
M3S	81.60 ± 0.51	71.94 ± 0.83	79.28 ± 0.30
MTL-Clu	81.57 ± 0.59	70.73 ± 0.84	78.79 ± 0.36
MTL-Par	81.83 ± 0.65	71.34 ± 0.69	80.00 ± 0.74
MTL-Comp	81.03 ± 0.68	71.66 ± 0.48	79.14 ± 0.28

[4] When Does Self-Supervision Help Graph Convolutional Networks?, ICML'20

- Experiment 2: MTL on SOTA
- **Par** is generally beneficial to all SOTAs
  1. **Clu** is not working because feature dim is low while dataset is large
  2. Topology-based **Par** has a general assumption
  3. The potential benefits of Comp can benefit other tasks (adversarial robustness)

Datasets	Cora	Citeseer	PubMed
GCN	81.00 ± 0.67	70.85 ± 0.70	79.10 ± 0.21
GCN+Clu	81.57 ± 0.59	70.73 ± 0.84	78.79 ± 0.36
GCN+Par	81.83 ± 0.65	71.34 ± 0.69	80.00 ± 0.74
GCN+Comp	81.03 ± 0.68	71.66 ± 0.48	79.14 ± 0.28
GAT	77.66 ± 1.08	68.90 ± 1.07	78.05 ± 0.46
GAT+Clu	79.40 ± 0.73	69.88 ± 1.13	77.80 ± 0.28
GAT+Par	80.11 ± 0.84	69.76 ± 0.81	80.11 ± 0.34
GAT+Comp	80.47 ± 1.22	70.62 ± 1.26	77.10 ± 0.67
GIN	77.27 ± 0.52	68.83 ± 0.40	77.38 ± 0.59
GIN+Clu	78.43 ± 0.80	68.86 ± 0.91	76.71 ± 0.36
GIN+Par	81.83 ± 0.58	71.50 ± 0.44	80.28 ± 1.34
GIN+Comp	76.62 ± 1.17	68.71 ± 1.01	78.70 ± 0.69
GMNN	83.28 ± 0.81	72.83 ± 0.72	81.34 ± 0.59
GMNN+Clu	83.49 ± 0.65	73.13 ± 0.72	79.45 ± 0.76
GMNN+Par	83.51 ± 0.50	73.62 ± 0.65	80.92 ± 0.77
GMNN+Comp	83.31 ± 0.81	72.93 ± 0.79	81.33 ± 0.59
GraphMix	83.91 ± 0.63	74.33 ± 0.65	80.68 ± 0.57
GraphMix+Clu	83.87 ± 0.56	75.16 ± 0.52	79.99 ± 0.82
GraphMix+Par	84.04 ± 0.57	74.93 ± 0.43	81.36 ± 0.33
GraphMix+Comp	83.76 ± 0.64	74.43 ± 0.72	80.82 ± 0.54

- Backgrounds for Contrastive Learning
- Applications for Contrastive Learning on Images and Graphs
- Some Deeper Insights
- Comments

Conclusions and some thoughts:

- Understanding the role of contrastive learning is important, yet still an open question.
- More domain knowledge on molecular graph. (scaffold in GROVER)

- Other details:
    - Pre-training & multi-task learning.
    - Negative sampling. (MoCo, BYOL)
    - The connection between base model (GNN model) and contrastive methods, and how they are combined to affect the performance.

More details can be found in group slack or https://chao1224.github.io/material/slides/202006.pdf