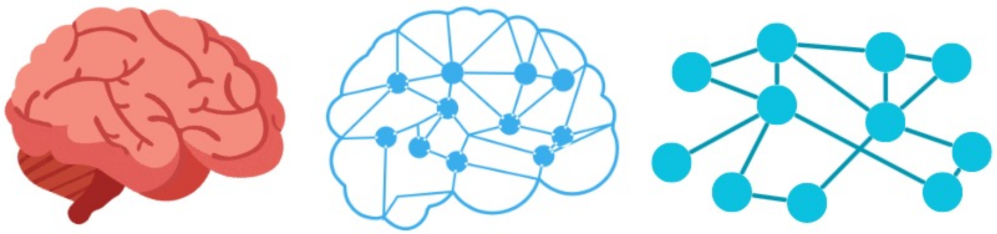
KDD 2025 – Research Track

Oldie but Goodie: Re-illuminating Label Propagation on Graphs with Partially Observed Features

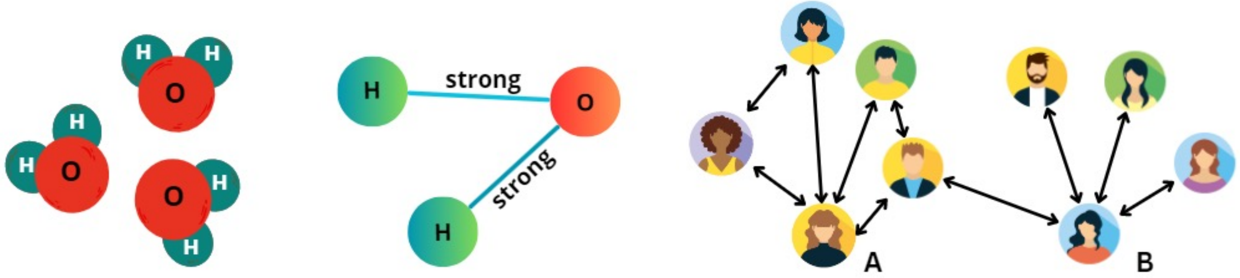
Sukwon Yun, Xin Liu, Yunhak Oh, Junseok Lee, Sungwon Kim, Tianlong Chen, Tsuyoshi Murata, **Chanyoung Park**

MOTIVATION: MISSING FEATURES IN GRAPH

Various Types of Graph-Structured Data



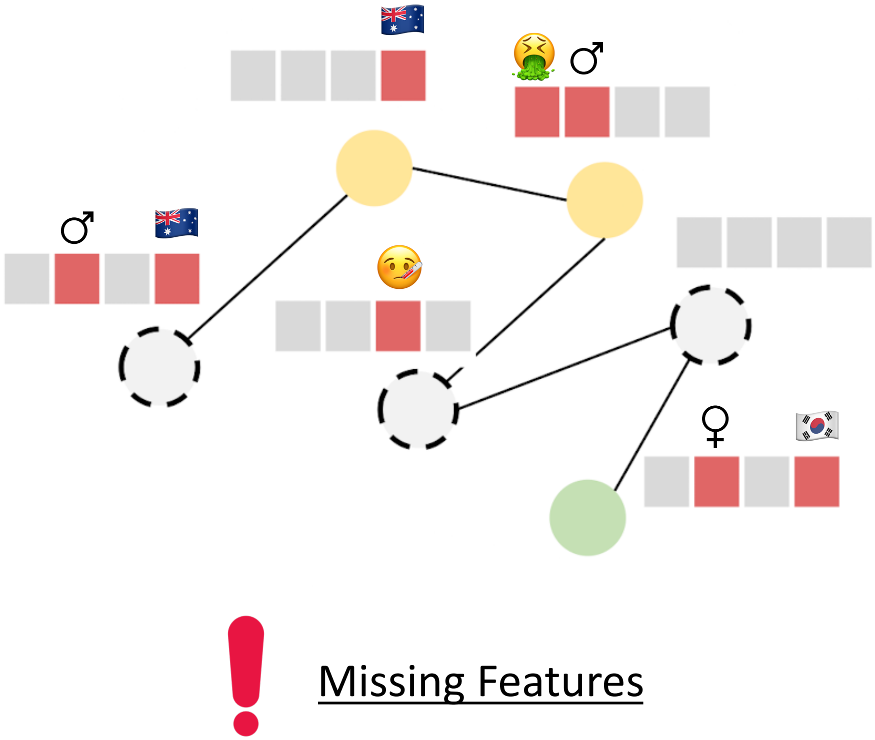
Brain Networks



Chemical Compounds

Social Networks

Real-world Scenarios

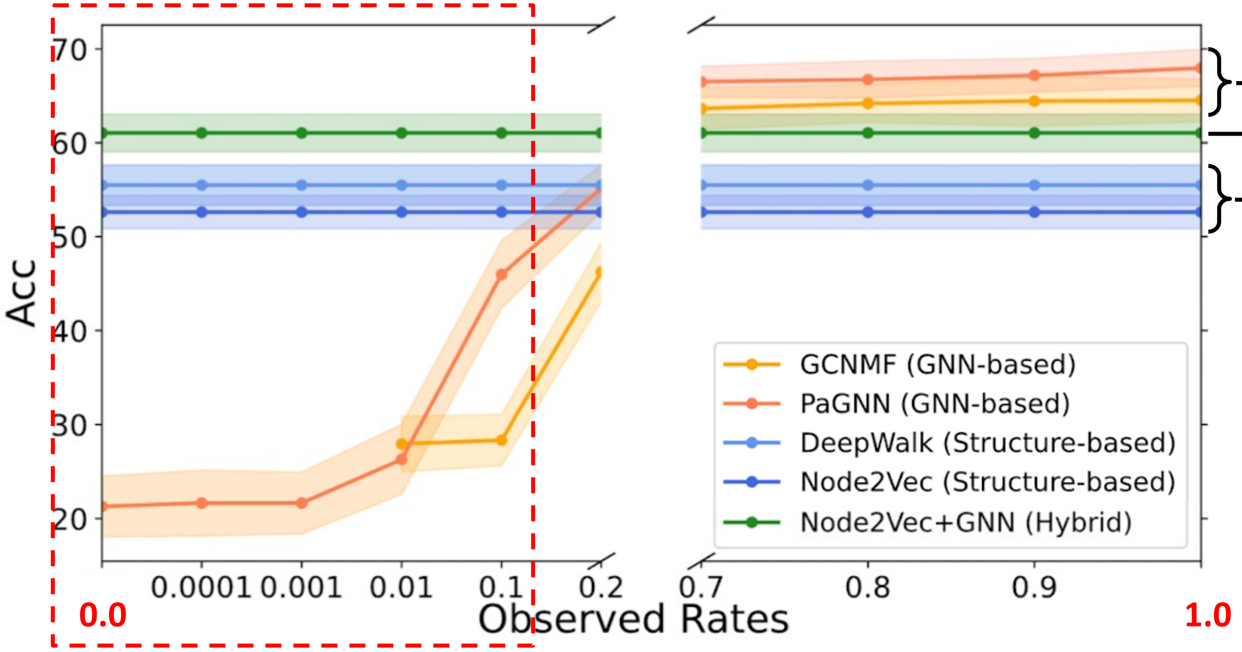


Missing Features

In real-world scenarios, **features are not always available** for graph data.

EMPIRICAL MOTIVATION

CiteSeer graph with missing features



GNN-based

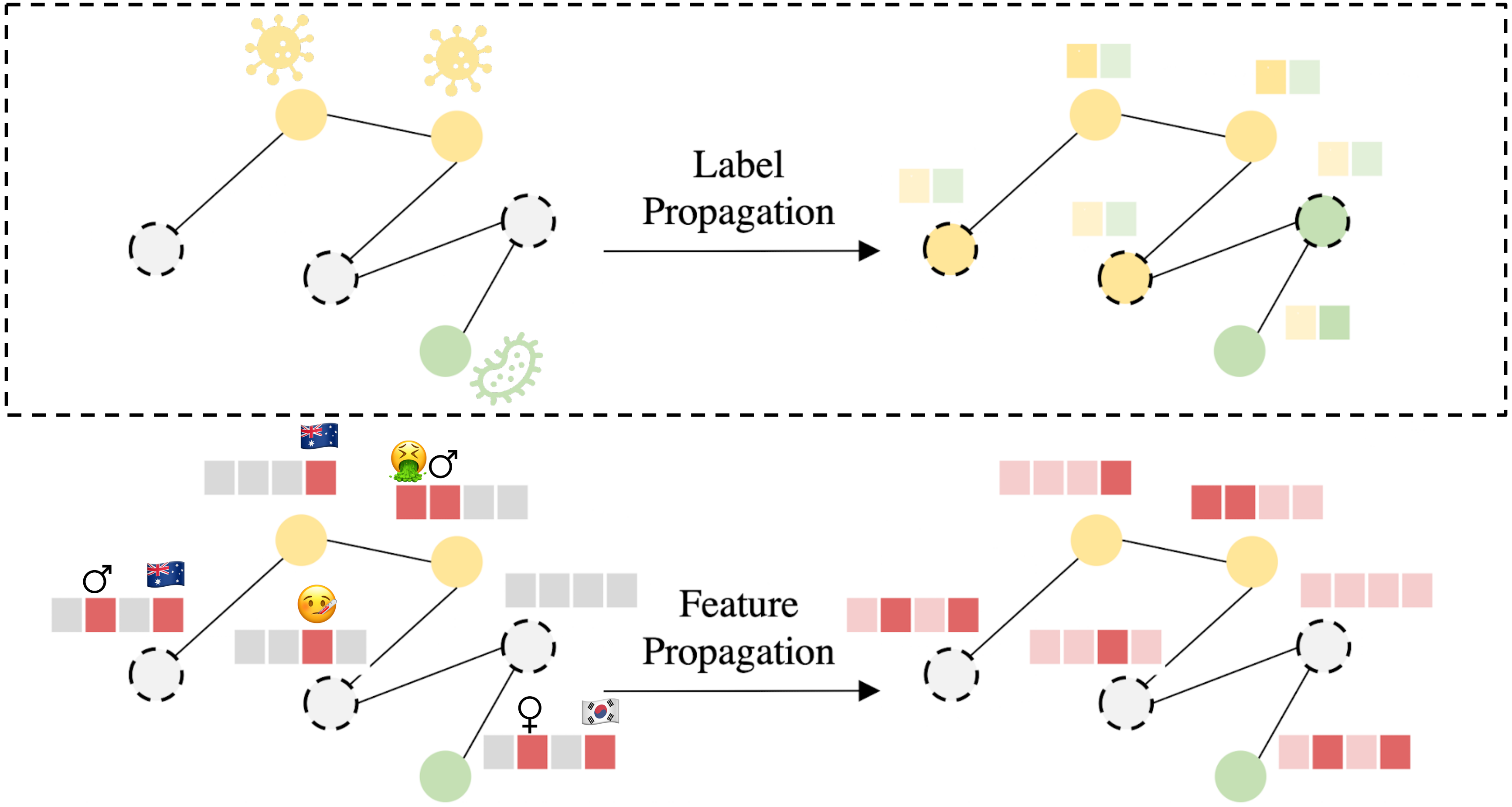
Structure-based + GNN (Hybrid)

[Node features are initialized with node2vec embedding]

Structure-based (no features used)

For graphs with **severely missing features** (observed rate ≤ 0.1),
structure-based models outperform GNN-based models.

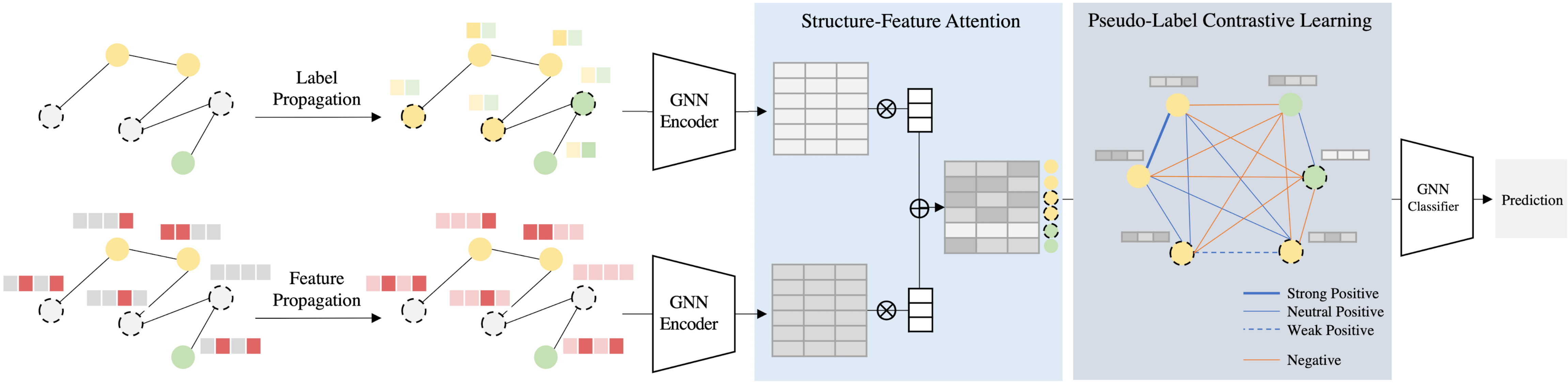
LABEL PROPAGATION (OLDIE)



We revisit classical **Label Propagation (Oldie)** as an effective remedy for missing features, thanks to its ability to exploit both structure and label information.

METHODOLOGY (GOODIE)

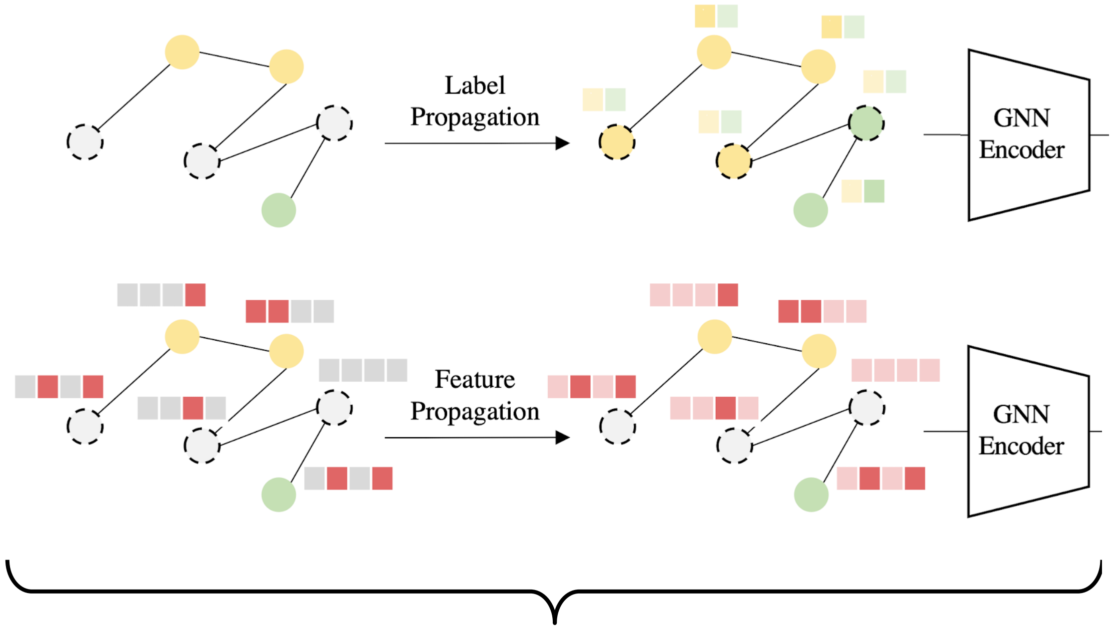
We propose a hybrid approach that **bridges the gap between the traditional structure-based model and the recent GNN-based model, especially when only a few features are available.**



1. Label/Feature Propagation | **2. Structure-Feature Attention** | **3. Pseudo-Label Contrastive Learning** | **Prediction**

METHODOLOGY

- **Step 1. LP & FP branch**



Label/Feature Propagation

- Label Propagation

$$\mathbf{Y}^{(k+1)} = \hat{\mathbf{A}}_{sym}\mathbf{Y}^{(k)},$$

$$\mathbf{y}_i^{(k+1)} = \mathbf{y}_i^{(0)}, \forall i \leq m$$ ⋯ replace with the known label if it exists

$$\mathbf{H}^{\mathrm{LP}} = \sigma(\hat{\mathbf{A}}_{sym}\hat{\mathbf{Y}}\mathbf{W}_{\mathrm{LP}}), \quad \hat{\mathbf{Y}} = \mathbf{Y}^{(K)}$$

- Feature Propagation

$$\mathbf{X}^{(k+1)} = \hat{\mathbf{A}}_{sym}\mathbf{X}^{(k)},$$

$$\mathbf{x}_{i,d}^{(k+1)} = \mathbf{x}_{i,d}^{(0)}, \forall i \in \mathcal{V}_{known,d}, \forall d$$

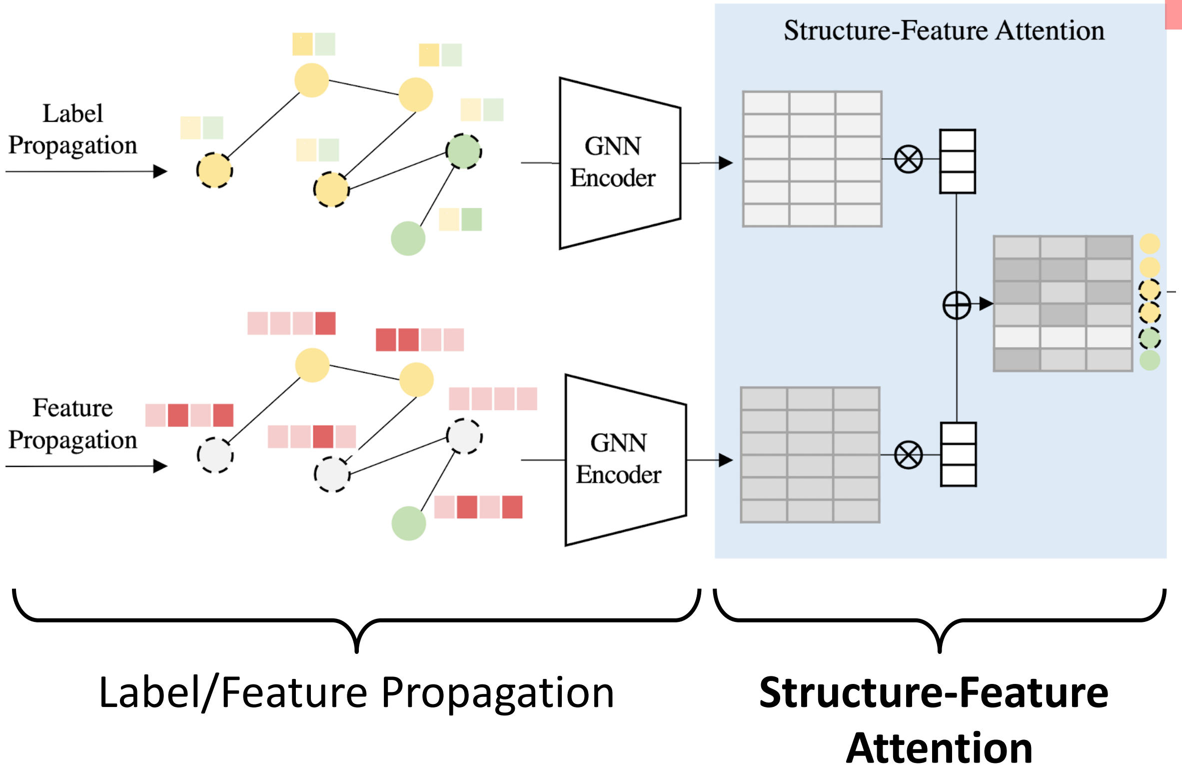
⋯ replace with the known feature if available

$$\mathbf{H}^{\mathrm{FP}} = \sigma(\hat{\mathbf{A}}_{sym}\hat{\mathbf{X}}\mathbf{W}_{\mathrm{FP}}), \quad \hat{\mathbf{X}} = \mathbf{X}^{(K)}$$

We **compensate for missing label or feature values using a propagation** method
until each converges at iteration K, and project both onto an embedding space.

METHODOLOGY

- **Step 2. Structure-Feature Attention**



Low when features are sparse

High when features are abundant

$$\mathbf{z}_i = \alpha_{i,\mathrm{LP}} \mathbf{h}_i^{\mathrm{LP}} + \alpha_{i,\mathrm{FP}} \mathbf{h}_i^{\mathrm{FP}}$$

$$\alpha_{i,LP} = \frac{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{LP}}))}{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{LP}})) + \exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{FP}}))}$$

$$\alpha_{i,FP} = \frac{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{FP}}))}{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{LP}})) + \exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{FP}}))}$$

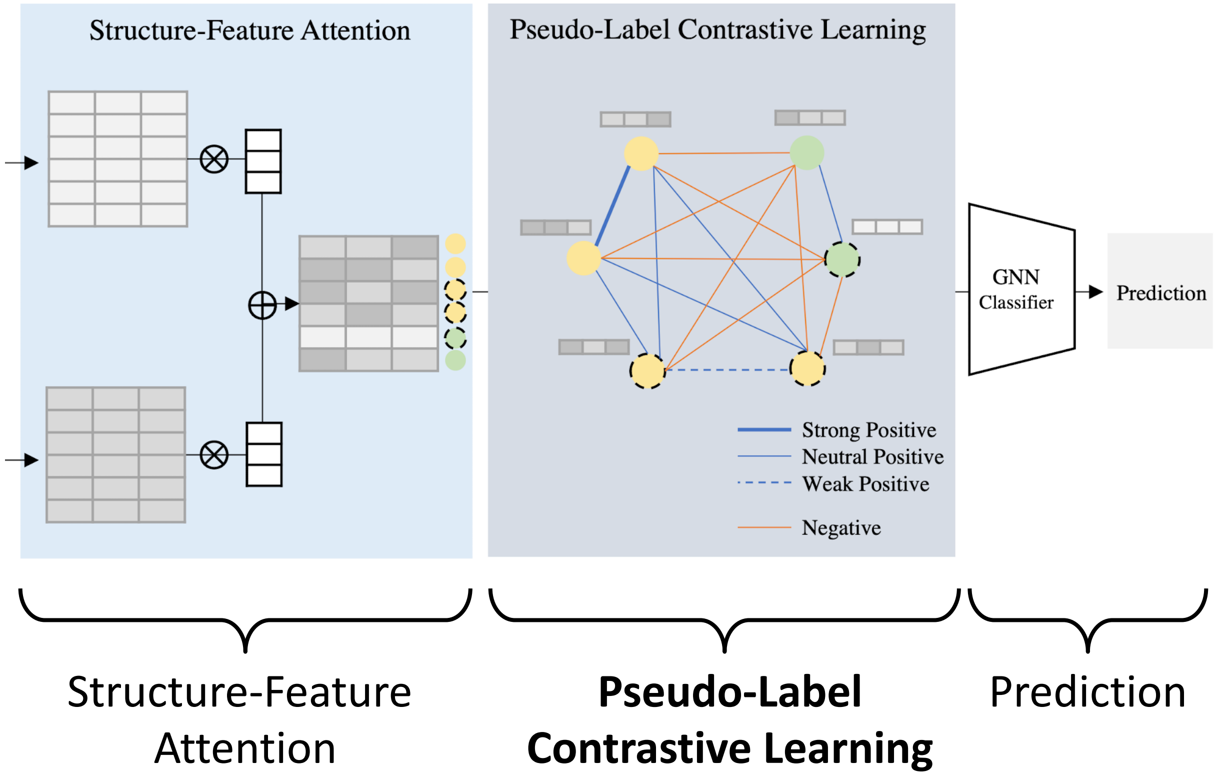
$1 - \alpha_{i,FP}$

$$\mathcal{L}_{\mathrm{ce}} = \sum_{v \in \mathcal{V}_{tr}} \sum_{c \in C} CE(\mathbf{p}_v[c]), \mathbf{P} = \mathrm{softmax}(\sigma(\hat{\mathbf{A}}_{sym}\mathbf{Z}\mathbf{W}_{\mathrm{cls}}))$$

Due to the uncertainty in missing feature rates, we **adaptively capture attention between embeddings from each label and feature branch.**

METHODOLOGY

- **Step 3. Pseudo-Label Contrastive Learning (PseudoCon)**



- Vanilla SupCon [1] loss

$$\mathcal{L}_{\text{sup}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)}$$

- Pseudo-Label Contrastive loss (Ours)

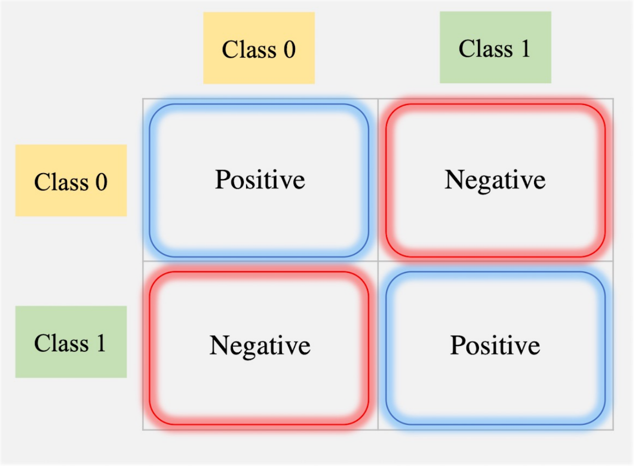
$$\mathcal{L}_{\text{pseudo}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \boxed{w_{ip}} \cdot \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)}$$

[1] Supervised Contrastive Learning (NeurIPS 2020)

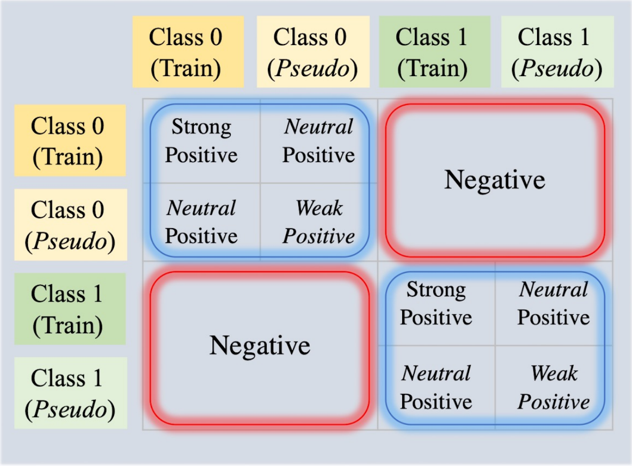
We **further leverage the potential of the LP branch by exploiting additional supervision** it provides. Specifically, we design a contrastive learning objective using the **pseudo-labels** generated by LP.

METHODOLOGY

- **Step 3. Pseudo-Label Contrastive Learning (PseudoCon)**



(a) Supervised Contrastive Learning



(b) Pseudo-Label Contrastive Learning (Ours)

- Pseudo-Label Contrastive loss (Ours)

$$\mathcal{L}_{\text{pseudo}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} \boxed{w_{ip}} \cdot \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)}$$

$$w_{ip} = \begin{cases} 1, & \text{if } i, p \in \hat{Y}_{train} \\ \tilde{y}_p, & \text{if } i \in \hat{Y}_{train} \text{ and } p \in \hat{Y}_{pseudo} \\ \tilde{y}_i, & \text{if } i \in \hat{Y}_{pseudo} \text{ and } p \in \hat{Y}_{train} \\ \tilde{y}_i * \tilde{y}_p, & \text{if } i, p \in \hat{Y}_{pseudo} \end{cases}$$

$$0 \leq \underbrace{\tilde{y}_i * \tilde{y}_p}_{\textbf{Weak Positive}} < \underbrace{\tilde{y}_i, \tilde{y}_p}_{\textbf{Neutral Positive}} \leq \underbrace{1}_{\textbf{Strong Positive}}$$

Final Loss of GOODIE: $\mathcal{L}_{\text{final}} = \mathcal{L}_{\text{ce}} + \lambda \mathcal{L}_{\text{pseudo}}$

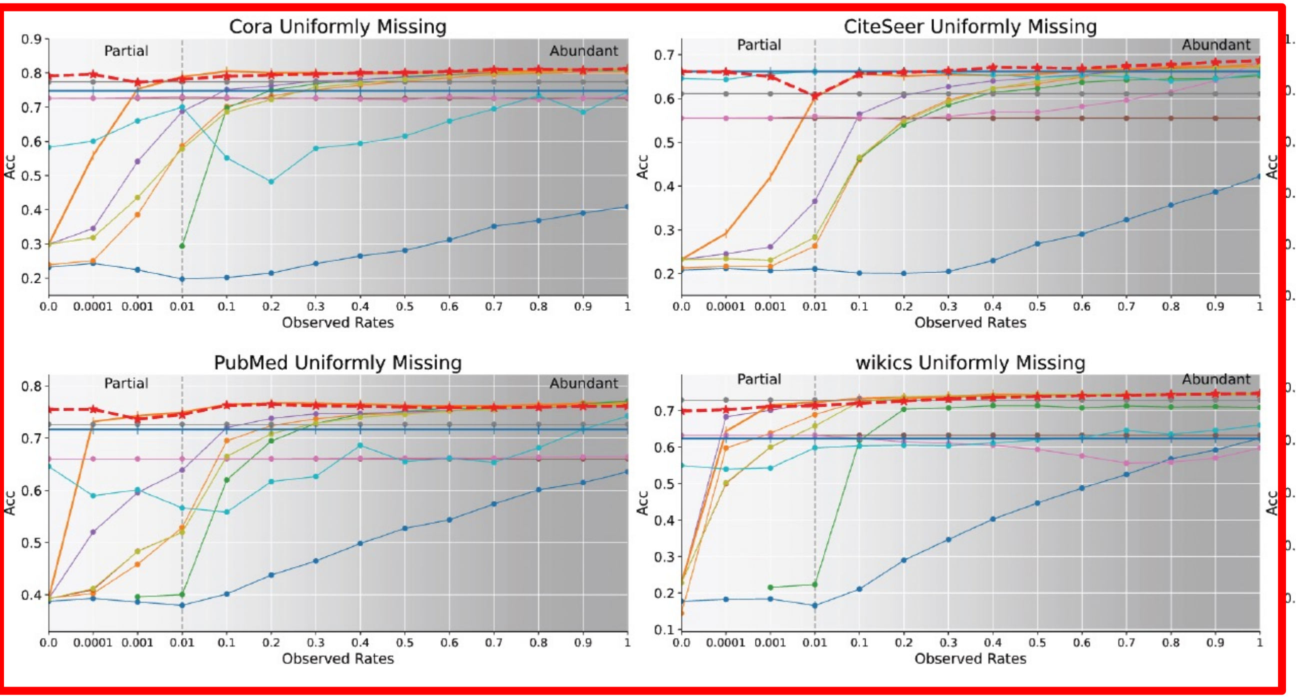
We adjust the **supervision weight based on the reliability of the labels.**

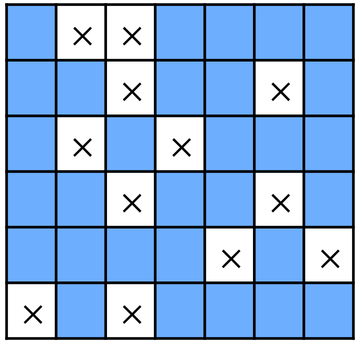
EXPERIMENTS

Dataset
Cora
CiteSeer
PubMed
WikiCS
Coauthor CS
Coauthor Physics
OGBN-Arxiv

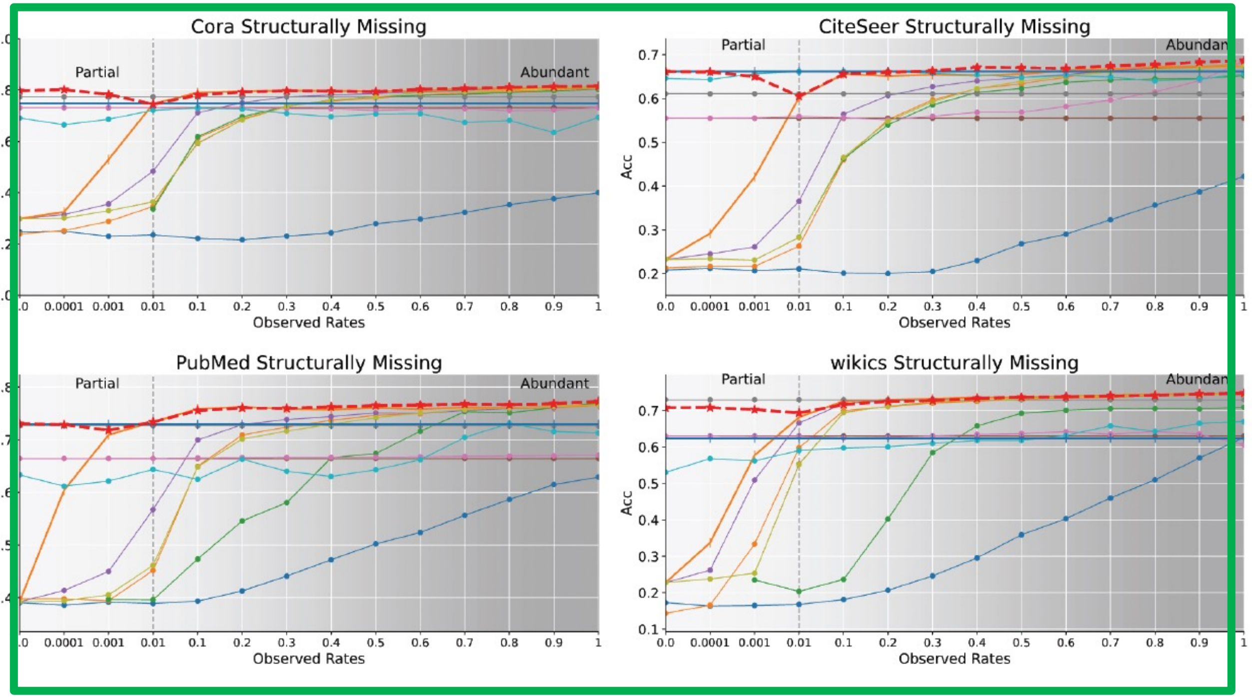
- 1. Performance (%) of GOODIE on **Node Classification** across various observed feature rates

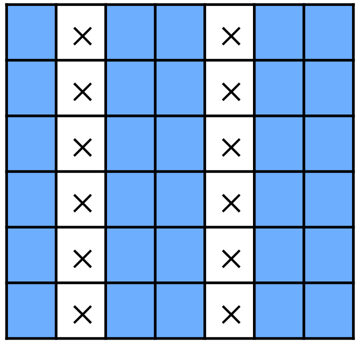
Uniformly Missing

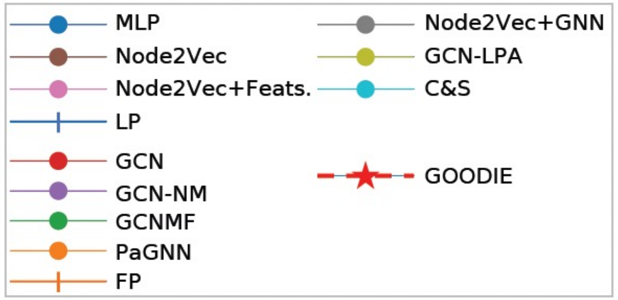




Structurally Missing







EXPERIMENTS

- 2. Performance (%) of GOODIE on Node Classification at missing rate (mr) of 99.99%

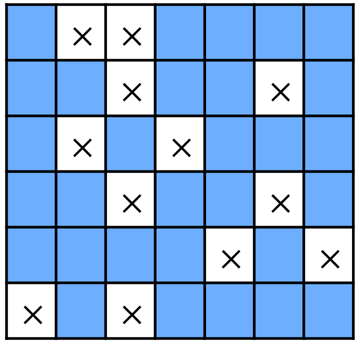
	GNN-based		Hybrid	Label/Feature Propagation		
Dataset	PaGNN	GCN-LPA	C&S	LP	FP	GOODIE
Cora	25.1±4.61	31.88±2.17	60.02±5.06	74.77±1.00	55.89±7.46	**79.58**±1.01
CiteSeer	22.6±1.77	24.24±1.07	65.55±1.61	66.15±1.67	51.06±4.17	**66.44**±1.41
PubMed	40.26±1.22	41.17±1.18	58.98±10.05	72.32±4.35	73.18±1.51	**75.54**±0.65
WikiCS	59.87±2.06	50.23±4.61	53.99±8.16	62.39±3.03	64.36±9.22	**70.28**±2.57
Co. CS	27.79±2.87	36.58±1.53	53.99±8.16	76.54±1.52	**78.54**±0.63	76.38±1.65
Co. Physics	44.16±5.89	53.34±1.44	45.03±23.68	85.86±1.91	**87.92**±1.55	87.86±1.61
OGBN-Arxiv	44.69±0.41	22.15±5.86	67.72±0.14	67.36±0.0	63.72±0.54	**69.04**±0.14
Average	38.78	37.08	57.90	72.20	67.81	**75.02**

EXPERIMENTS

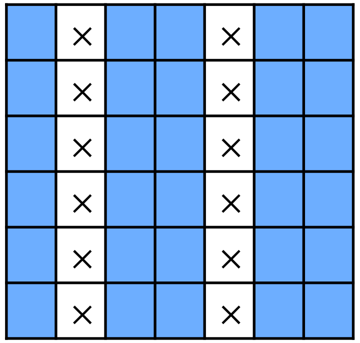
- **3. Performance (%) of GOODIE on Link Prediction at missing rate (mr) 0.9999**

Dataset		Uniformly Missing				Structurally Missing			
		GCN	GCN-NM	FP	GOODIE	GCN	GCN-NM	FP	GOODIE
Cora	AUC	0.50±0.02	0.54±0.04	0.82±0.02	**0.84**±0.02	0.50±0.02	0.50±0.01	0.60±0.12	**0.84**±0.02
	AP	0.51±0.01	0.55±0.03	0.82±0.03	**0.85**±0.01	0.51±0.02	0.51±0.01	0.59±0.11	**0.84**±0.02
CiteSeer	AUC	0.50±0.02	0.50±0.04	0.75±0.09	**0.81**±0.03	0.48±0.02	0.50±0.02	0.60±0.12	**0.81**±0.02
	AP	0.50±0.01	0.53±0.04	0.76±0.10	**0.83**±0.04	0.50±0.01	0.50±0.02	0.59±0.12	**0.84**±0.02
PubMed	AUC	0.50±0.02	0.59±0.03	0.67±0.09	**0.75**±0.04	0.44±0.03	0.50±0.03	0.64±0.12	**0.70**±0.04
	AP	0.52±0.02	0.64±0.03	0.70±0.10	**0.79**±0.04	0.47±0.02	0.50±0.03	0.65±0.11	**0.74**±0.04
Coauthor CS	AUC	0.68±0.02	0.87±0.01	0.90±0.02	**0.93**±0.01	0.53±0.01	0.54±0.03	0.79±0.13	**0.87**±0.04
	AP	0.69±0.02	0.87±0.02	0.88±0.03	**0.93**±0.01	0.53±0.01	0.55±0.03	0.77±0.12	**0.86**±0.05

- Uniformly Missing

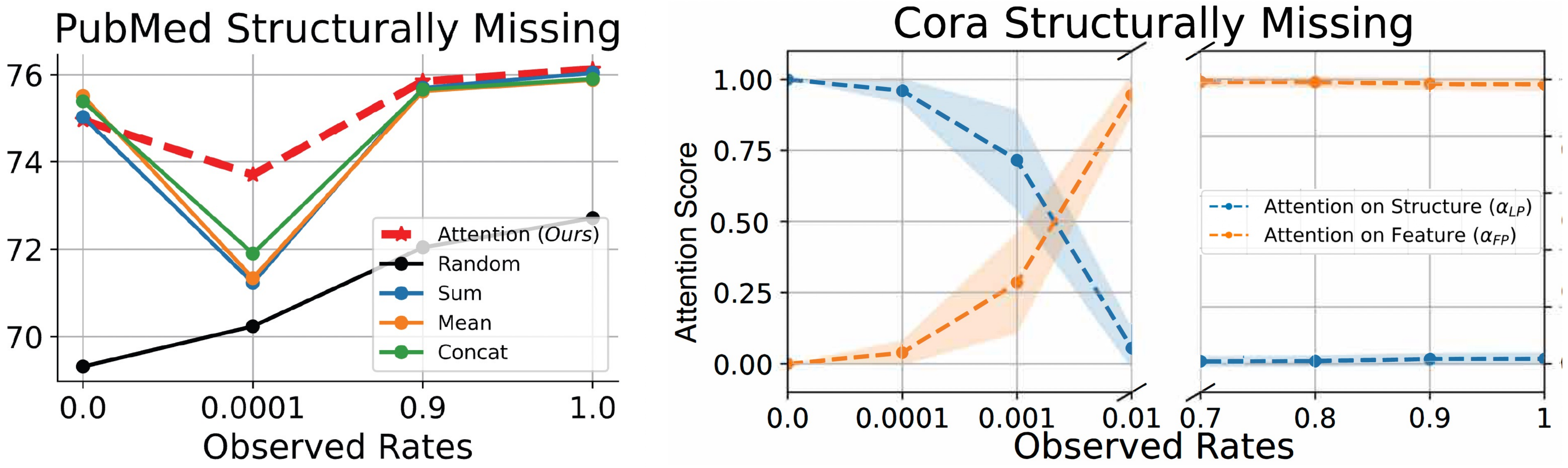


- Structurally Missing



EXPERIMENTS

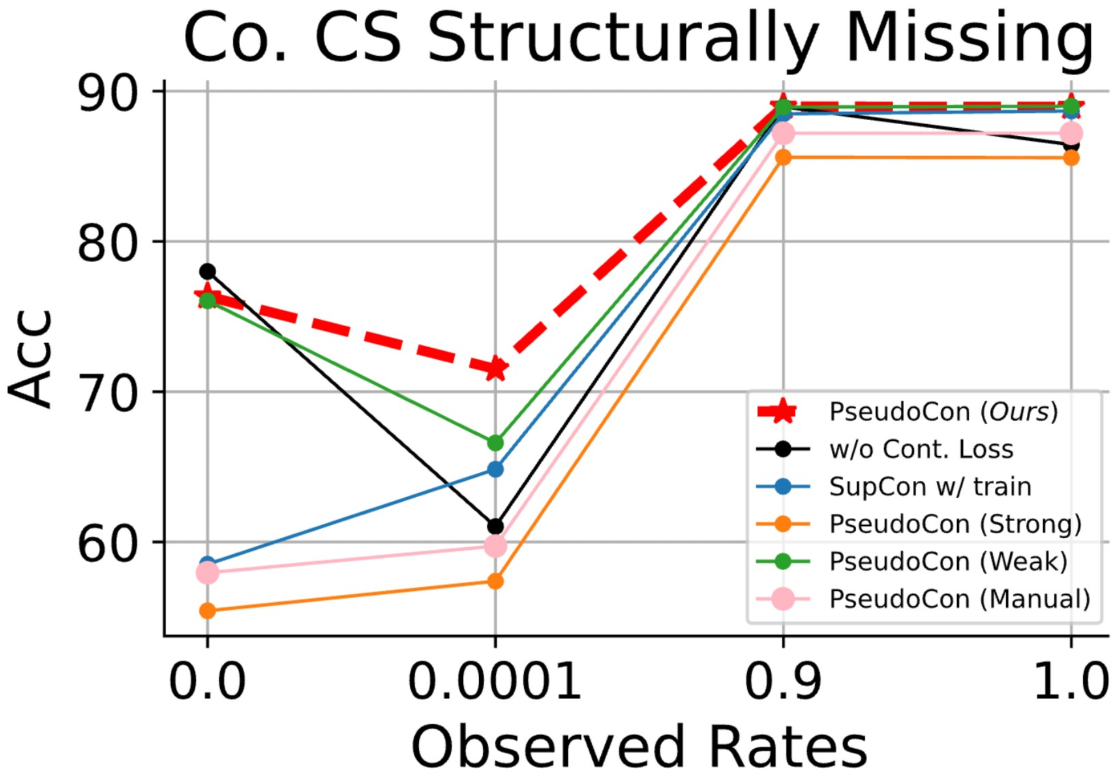
- **4. Effectiveness of Structure-Feature Attention and its Score at Different Observed Rates**

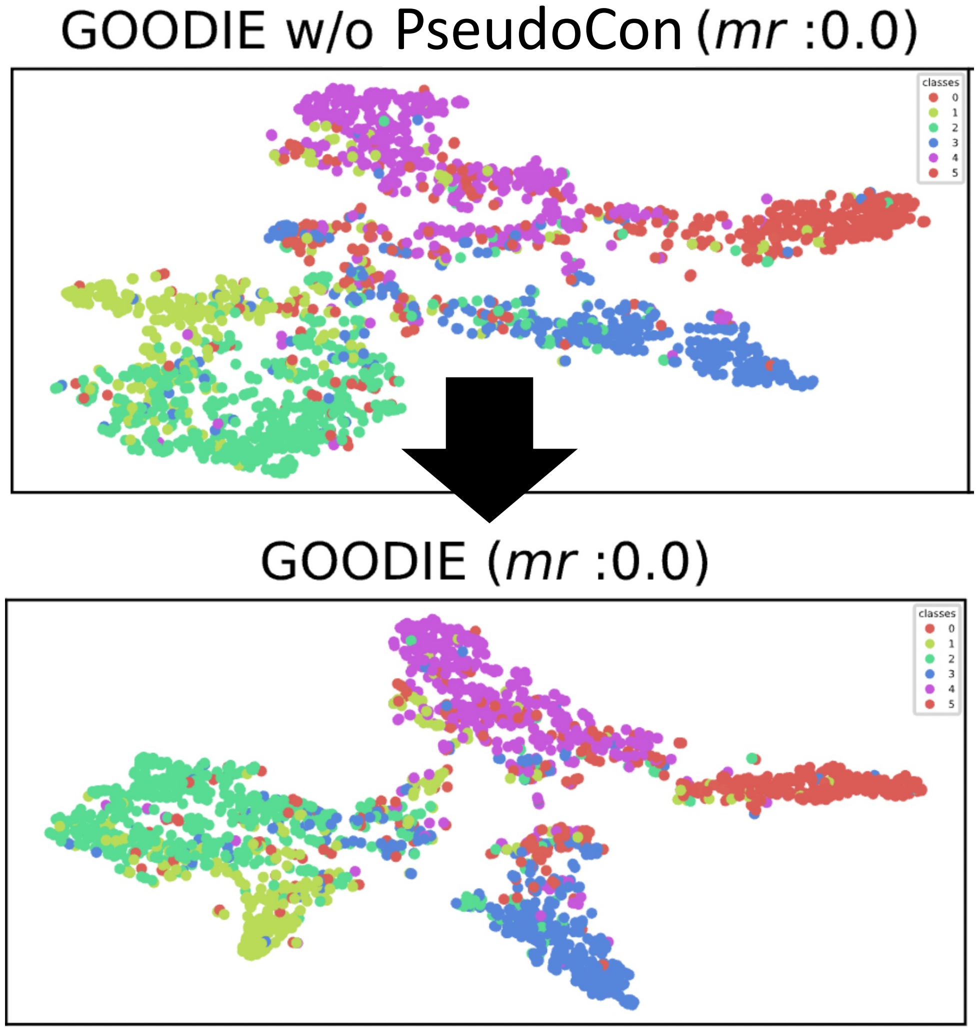


When handling embeddings from the FP and LP branches, **attention appears to be the most effective** strategy, as it naturally **places more weight on the FP branch as the observed rates increase**.

EXPERIMENTS

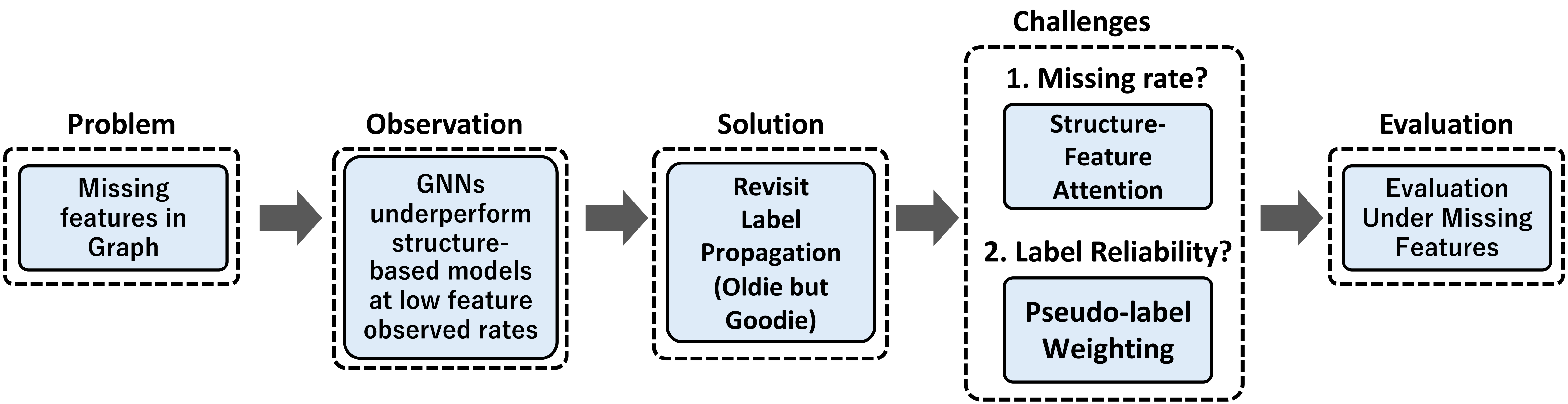
- **5. Effectiveness of PseudoCon Loss**





PseudoCon appears **effective when both strong and weak positive pairs are involved,**
as it **enhances intra-class embedding cohesion** while promoting inter-class separation.

CONCLUSION



Oldie but Goodie: Re-illuminating Label Propagation on Graphs with Partially Observed Features

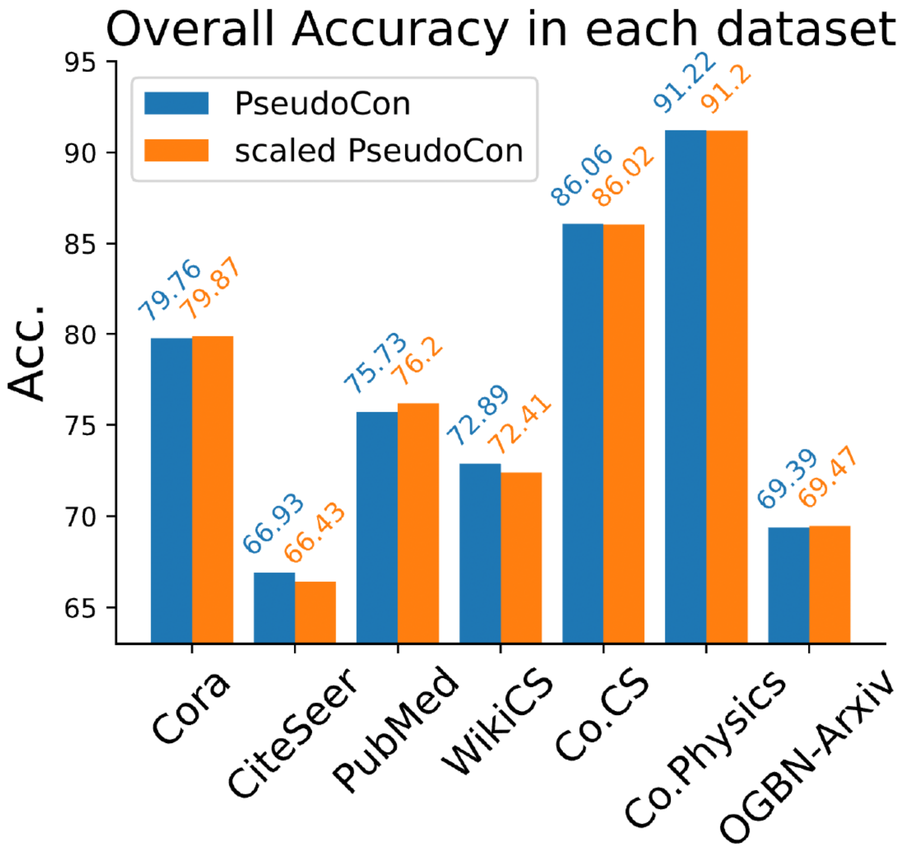
Email: cy.park@kaist.ac.kr

Lab homepage: https://dsail.kaist.ac.kr/

APPENDIX

ENHANCING SCALABILITY OF PSEUDO-LABEL CONTRASTIVE LEARNING

- **Comparison with Scalaed PseudoCon Loss**



(1) **PseudoCon Loss** → $\mathcal{O}(N^2)$, $N = number\ of\ nodes$

$$\mathcal{L}_{\text{pseudo}} = \sum_{i \in I} \frac{-1}{|P(i)|} \sum_{p \in P(i)} w_{ip} \cdot \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p / \tau)}{\sum_{a \in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a / \tau)}$$

(2) **Scaled PseudoCon Loss** → $\mathcal{O}(C^2)$, $C = number\ of\ classes$

$$\mathcal{L}_{\text{pseudo}} = -\sum_{c \in C} \log \frac{1}{\sum_{b \in B(c)} \exp(\mathbf{z}^c \cdot \mathbf{z}^b / \tau)}$$

where, $$\mathbf{z}^c = \frac{1}{|\hat{Y}^c|} \Big(\underbrace{\sum_{i \in \hat{Y}^c_{train}} 1 \cdot \mathbf{z}_i}_{\textbf{Strong}} + \underbrace{\sum_{j \in \hat{Y}^c_{pseudo}} \tilde{y}_j \cdot \mathbf{z}_j}_{\textbf{Neutral \& Weak}} \Big)$$

Class Prototypes

To reduce PseudoCon Loss complexity $\mathcal{O}(N^2)$, we introduce **scaled PseudoCon, a class prototype-based variant** that maintains strong performance with improved efficiency, $\mathcal{O}(C^2)$.

METHODOLOGY

Algorithm

Algorithm 1 Pseudocode for training GOODIE

Require: A graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, Partially observed feature matrix $\mathbf{X}^{(0)}$, and train label matrix $\mathbf{Y}^{(0)}$

Ensure: Final prediction matrix $\mathbf{P} \in \mathrm{R}^{\mathrm{N}\times|\mathrm{C}|}$

$\hat{\mathbf{A}}_{sym} = \tilde{\mathbf{D}}^{-1/2}\tilde{\mathbf{A}}\tilde{\mathbf{D}}^{-1/2}$

$\hat{\mathbf{Y}}, \hat{\mathbf{X}}, \mathcal{W}$ = PROPAGATION($\mathbf{Y}^{(0)}, \mathbf{X}^{(0)}$)

while *Convergence* **do**

  $\mathbf{P}, \mathbf{Z}$ = TRAIN_GOODIE($\hat{\mathbf{Y}}, \hat{\mathbf{X}}$)

  $\mathcal{L}_{\mathrm{ce}} = \sum_{v\in\mathcal{V}_{tr}} \sum_{c\in C} CE(\mathbf{p}_v[c])$

  $\mathcal{L}_{\mathrm{pseudo}} = \sum_{i\in I} \frac{-1}{|P(i)|} \sum_{p\in P(i)} w_{ip} \cdot \log \frac{\exp(\mathbf{z}_i \cdot \mathbf{z}_p/\tau)}{\sum_{a\in A(i)} \exp(\mathbf{z}_i \cdot \mathbf{z}_a/\tau)}$

  $\mathcal{L}_{\mathrm{final}} = \mathcal{L}_{\mathrm{ce}} + \lambda\mathcal{L}_{\mathrm{pseudo}}$

end while

function TRAIN_GOODIE($\hat{\mathbf{Y}}, \hat{\mathbf{X}}$)

  $\mathbf{H}^{\mathrm{LP}} = \sigma(\hat{\mathbf{A}}_{sym}\hat{\mathbf{Y}}\mathbf{W}_{\mathrm{LP}})$

  $\mathbf{H}^{\mathrm{FP}} = \sigma(\hat{\mathbf{A}}_{sym}\hat{\mathbf{X}}\mathbf{W}_{\mathrm{HP}})$

  for $i \in \mathcal{V}$ **do**

    $\alpha_{i,LP} = \frac{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{LP}}))}{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{LP}}))+\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{FP}}))}$

    $\alpha_{i,FP} = \frac{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{FP}}))}{\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{LP}}))+\exp(\mathrm{LeakyReLU}(\mathbf{a}^\top \mathbf{h}_i^{\mathrm{FP}}))}$

    $\mathbf{z}_i = \alpha_{i,\mathrm{LP}}\mathbf{h}_i^{\mathrm{LP}} + \alpha_{i,\mathrm{FP}}\mathbf{h}_i^{\mathrm{FP}}$

  end for

  return $\mathrm{softmax}(\sigma(\hat{\mathbf{A}}_{sym}\mathbf{Z}\mathbf{W}_{\mathrm{cls}})), \mathbf{Z}$

end function

function PROPAGATION($\mathbf{Y}^{(0)}, \mathbf{X}^{(0)}$)

  for $k = 0; k < K; k = k + 1$ **do**

    $\mathbf{Y}^{(k+1)} \leftarrow \hat{\mathbf{A}}_{sym}\mathbf{Y}^{(k)}$

    $\mathbf{y}_i^{(k+1)} \leftarrow \mathbf{y}_i^{(0)}, \forall i \leq m.$

  end for

  $\mathcal{W}$ = WEIGHT CALCULATOR($\mathbf{Y}^{(K)}$)

  for $k = 0; k < K; k = k + 1$ **do**

    $\mathbf{X}^{(k+1)} \leftarrow \hat{\mathbf{A}}_{sym}\mathbf{X}^{(k)}$

    $\mathbf{x}_{i,d}^{(k+1)} = \mathbf{x}_{i,d}^{(0)}, \forall i \exists \mathcal{V}_{known,d}, \forall d.$

  end for

  return $\mathbf{Y}^{(K)}, \mathbf{X}^{(K)}, \mathcal{W}$

end function

function WEIGHT CALCULATOR($\hat{\mathbf{Y}}$)

  $\tilde{\mathbf{y}} = \max(\mathrm{softmax}(\hat{\mathbf{Y}}))$

  $\mathcal{W} = [\mathbf{0}, \cdots, \mathbf{0}]^T$

  for $i = 0;\ i < N;\ i = i + 1$ **do**

    for $j = i;\ j < N;\ j = j + 1$ **do**

      if $argmax(\hat{y}_i) == argmax(\hat{y}_j)$ **then**

        $p \leftarrow j$

$$w_{ip} = \begin{cases} 1, & \text{if } i, p \in \hat{Y}_{train} \\ \tilde{y}_p, & \text{if } i \in \hat{Y}_{train} \text{ and } p \in \hat{Y}_{pseudo} \\ \tilde{y}_i, & \text{if } i \in \hat{Y}_{pseudo} \text{ and } p \in \hat{Y}_{train} \\ \tilde{y}_i * \tilde{y}_p, & \text{if } i, p \in \hat{Y}_{pseudo} \end{cases}$$

      end if

    end for

  end for

  return $\mathcal{W}$

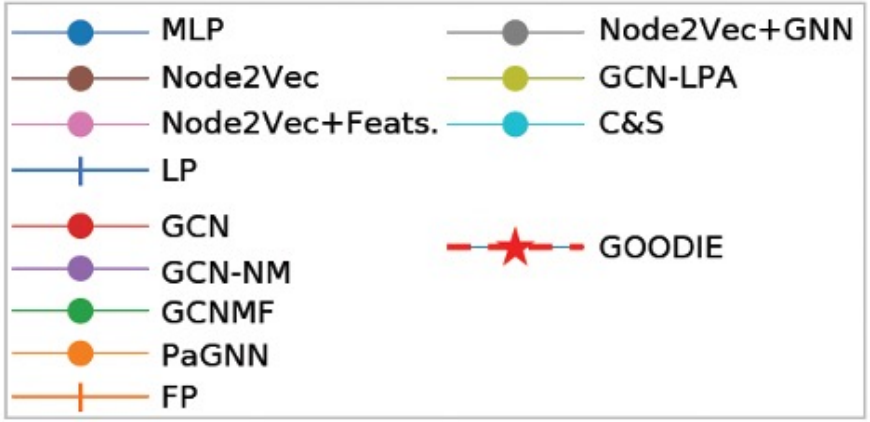
end function

EXPERIMENTS

Datasets and Baselines

Dataset	URL link to the dataset
Cora	https://github.com/shchur/gnn-benchmark
CiteSeer	https://github.com/shchur/gnn-benchmark
PubMed	https://github.com/shchur/gnn-benchmark
WikiCS	https://github.com/pmernyei/wiki-cs-dataset
Coauthor CS	https://github.com/shchur/gnn-benchmark
Coauthor Physics	https://github.com/shchur/gnn-benchmark
OGBN-Arxiv	https://ogb.stanford.edu/docs/nodeprop/#ogbn-arxiv

Baseline	URL link to the code
Node2Vec	https://github.com/aditya-grover/node2vec
FP	https://github.com/twitter-research/feature-propagation
GCNMF	https://github.com/marblet/GCNmf
GCN	https://github.com/tkipf/pygcn
GCN-NM	https://github.com/twitter-research/feature-propagation
PaGNN	https://github.com/twitter-research/feature-propagation
GCN-LPA	https://github.com/hwwang55/GCN-LPA
C&S	https://github.com/CUAI/CorrectAndSmooth



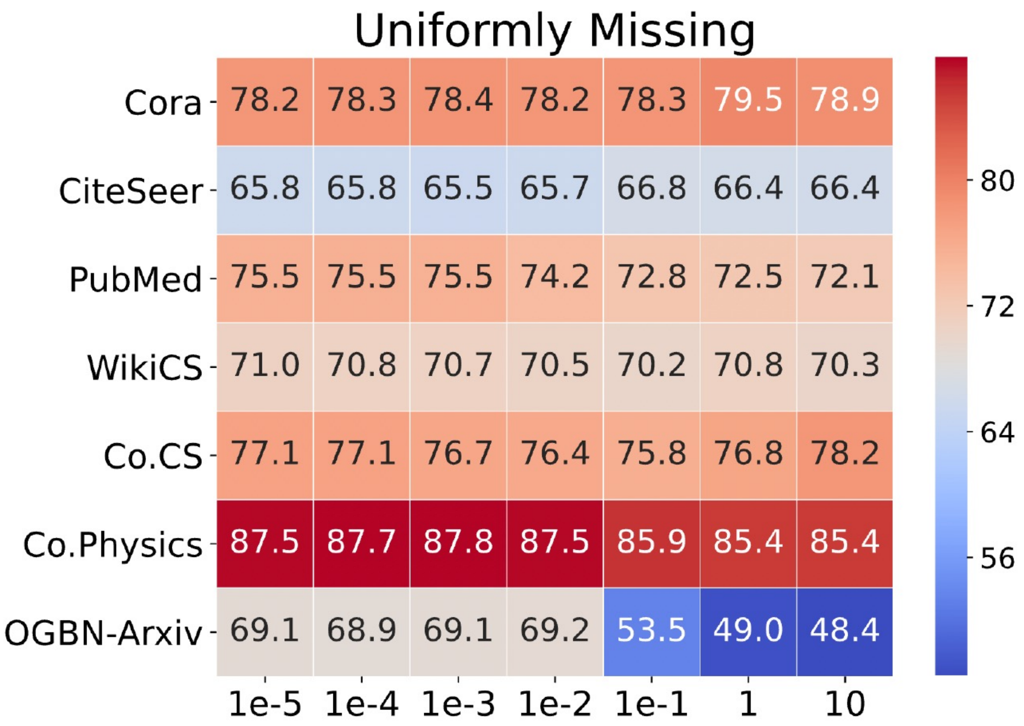
EXPERIMENTS

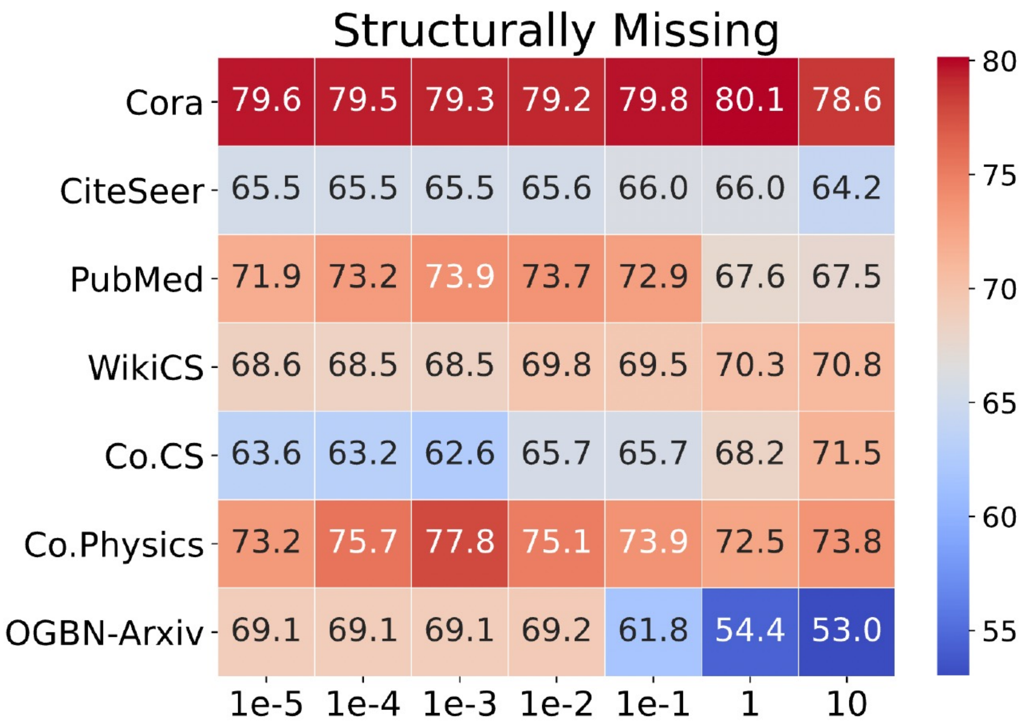
Hyperparameter Setting

Dataset	Common			Uniform	Structural
	α	τ	scaled	λ	λ
Cora	0.99	0.01	False	1	1
CiteSeer	0.999	0.01	False	1	1
PubMed	0.999	0.01	False	0.0001	0.1
WikiCS	0.9	0.01	False	10	10
Coauthor CS	0.99	0.01	False	0.01	10
Coauthor Physics	0.999	0.01	True	0.00001	0.001
OGBN-Arxiv	0.8	0.01	True	0.001	0.001

EXPERIMENTS

- **Sensitivity analysis on pseudo label controlling parameter, λ at missing rate (mr) 0.9999**





Across diverse datasets, λ demonstrates robustness in the range from 1e-5 to 10.