SMU Classification: Restricted

Learning to Count Isomorphisms with Graph Neural Networks

Xingtong Yu[1*], Zemin Liu[2*], Yuan Fang[3†], Xinming Zhang [1†]

[1] University of Science and Technology of China, China
[2] National University of Singapore, Singapore
[3] Singapore Management University, Singapore

In Proceeding of 37th AAAI Conference on Artificial Intelligence, February 7-14, 2023

* Co-first author
† Corresponding author

Outline

1 .Introduction

2 .Proposed Model: Count-GNN

3 .Experiments

4 .Conclusions

SMU Classification: Restricted

PART 01

Introduction

Introduction

01

Isomorphism Counting

02

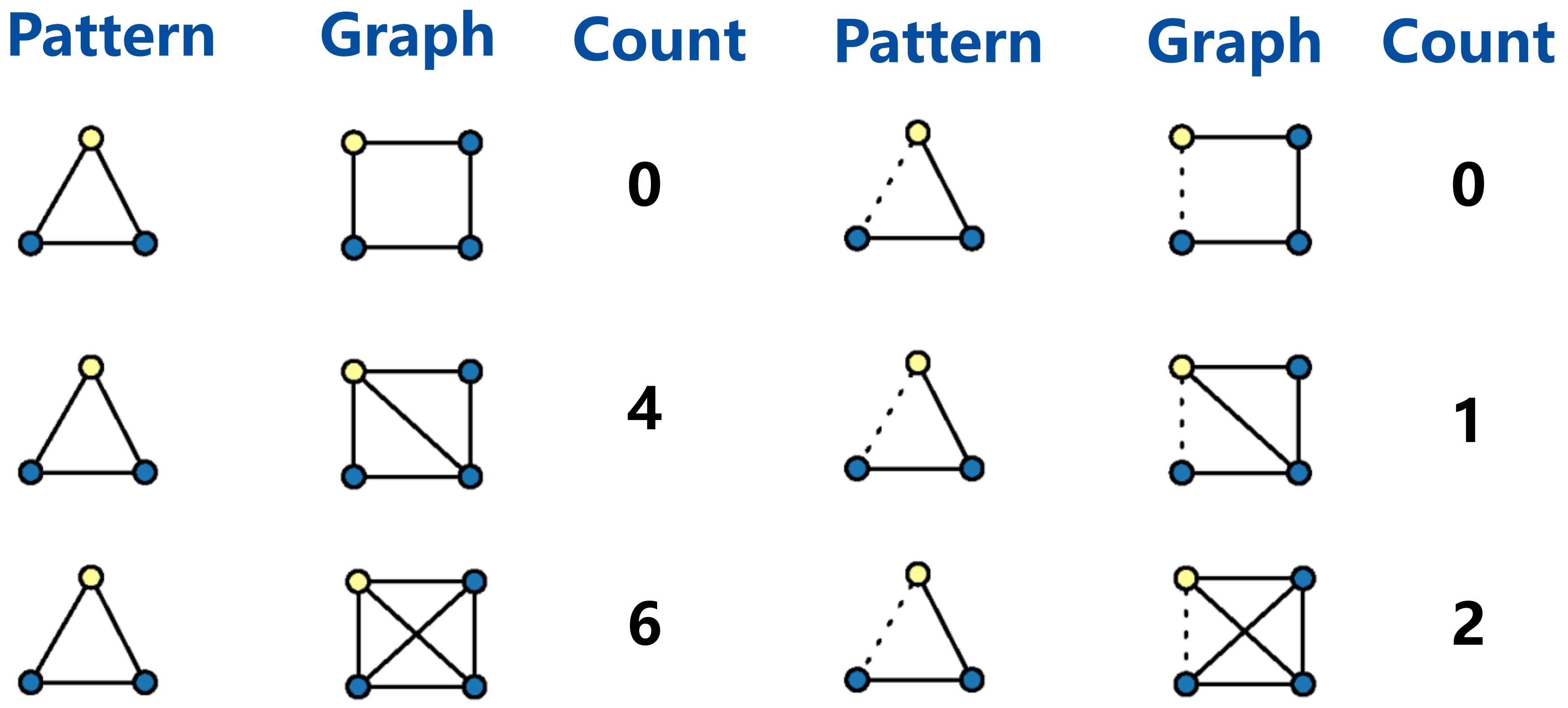
Related Work

03

Contribution

01 Isomorphism Counting

Subgraph Isomorphisms Counting



02 Related Work

Exact Database Methods[1][2][3]	Approximate Database Methods[4]	GNN Methods[5][6]
Search-Based	**Sampling-Based**	**GNN+Counter**
Exact Results	**Approximate Results**	**Approximate Results**
Excessive Computation Cost	**Large Computation Cost**	**Little Computation Cost**

[1] Ullmann, J. R. 1976. An algorithm for subgraph isomorphism. JACM.
[2] Cordella, L. P. et al. 2004. A (sub) graph isomorphism algorithm for matching large graphs. PAMI.
[3] Carletti, V. et al.2017. Challenging the time complexity of exact subgraph isomorphism for huge and dense graphs with VF3. PAMI.
[4] Bressan, M. et al. 2021. Faster motif counting via succinct color coding and adaptive sampling. ACM Trans Knowl Discov Data.
[5] Liu, X. et al. 2020. Neural subgraph isomorphism counting. KDD.
[6] Zhengdao, C. et al. 2020. Can Graph Neural Networks Count Substructures? NeurIPS.

SMU Classification: Restricted

02 Related Work

Shortage of General GNN-based Isomorphism Counting Models

Node Centric		Fixed Graph Representation	
Isomorphism Counting focus on topology information	Hard to capture interaction among nodes	Distinct structures of queries	Fixed graph representation

SMU Classification: Restricted

03 Contribution

Count-GNN

Edge-Centric Aggregating GNN

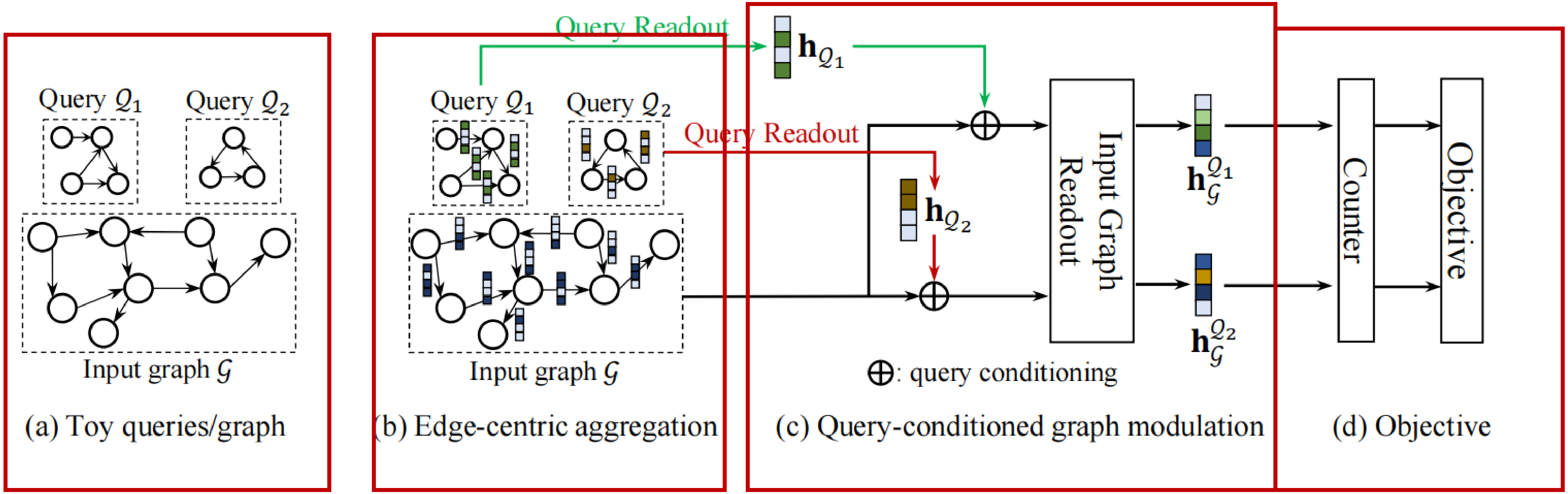
Query-Conditioned Graph Modulation

Experiments on Four Benchmark Datasets
8x~26x speedups over exact methods
3.1x~6.5x speedups over GNN methods

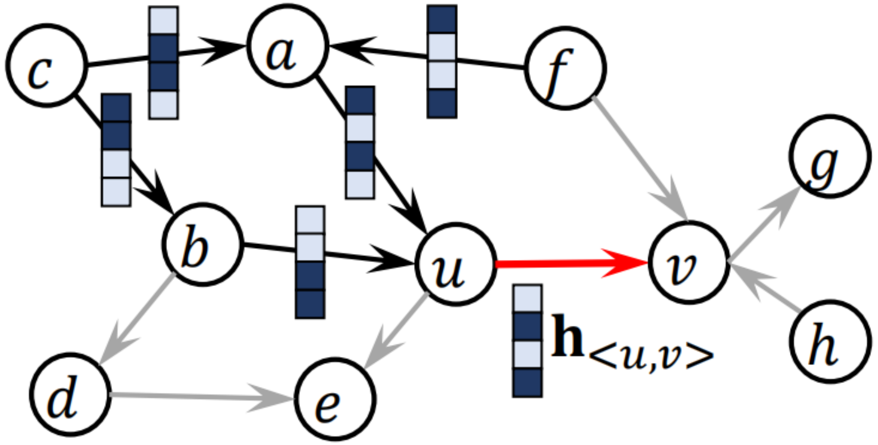
PART 02

Proposed Model: Count-GNN

Overall-Framework



Edge-Centric Aggregation



(a) Edge-centric aggregation

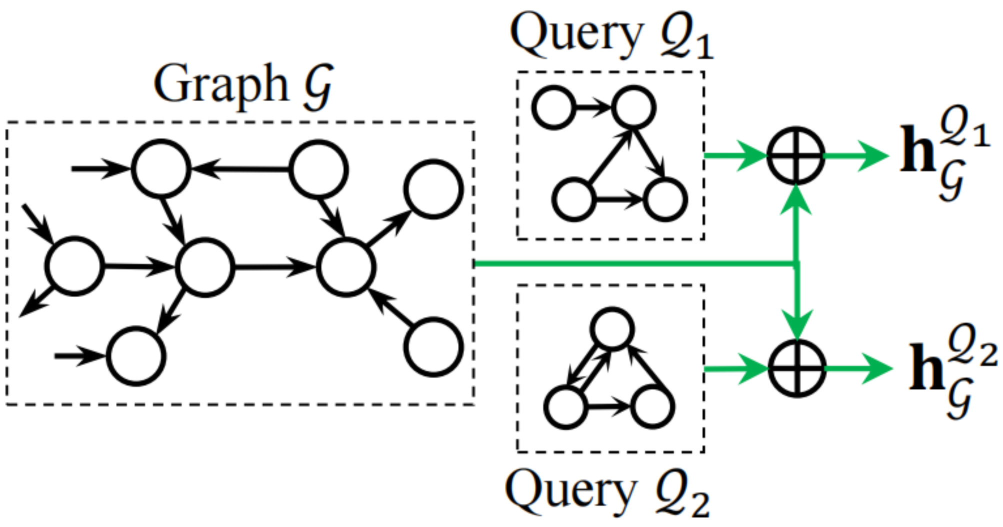
$$\mathbf{h}^0_{\langle u,v\rangle} = \mathbf{x}_u \parallel \mathbf{x}_{\langle u,v\rangle} \parallel \mathbf{x}_v \in \mathbb{R}^{d_0}$$

$$\mathbf{h}^l_{\langle u,v\rangle} = \sigma(\mathbf{W}^l\mathbf{h}^{l-1}_{\langle u,v\rangle} + \mathbf{U}^l\mathbf{h}^{l-1}_{\langle \cdot,u\rangle} + \mathbf{b}^l),$$

$$\mathbf{h}^{l-1}_{\langle \cdot,u\rangle} = \mathrm{AGGR}(\{\mathbf{h}^{l-1}_{\langle i,u\rangle} | \langle i,u\rangle \in E\})$$

- $\mathbf{x}_*$: encoded nodes or edges into input features
- $||$: concatenation operator
- $\mathbf{h}^l_{\langle u,v\rangle}$: message on edge $\langle u,v\rangle$ in the l-th layer
- $\mathbf{h}^{l-1}_{\langle \cdot,u\rangle}$: message aggregated from the preceding edges of $\langle u,v\rangle$
- **AGGR**: aggregation operator
- $\mathbf{W}^l$,$\mathbf{U}^l$,$\mathbf{b}^l$: learnable parameter

Query graph representation



(b) Modulation conditioned on queries

$$\mathbf{h}_{\mathcal{Q}} = \sigma(\mathbf{Q} \cdot \text{AGGR}(\{\mathbf{h}_{\langle u,v\rangle} | \langle u,v\rangle \in E_{\mathcal{Q}}\}))$$

$$\tilde{\mathbf{h}}_{\langle u,v\rangle} = (\gamma_{\langle u,v\rangle} + \mathbf{1}) \odot \mathbf{h}_{\langle u,v\rangle} + \beta_{\langle u,v\rangle}$$

$$\gamma_{\langle u,v\rangle} = \sigma(\mathbf{W}_{\gamma}\mathbf{h}_{\langle u,v\rangle} + \mathbf{U}_{\gamma}\mathbf{h}_{\mathcal{Q}} + \mathbf{b}_{\gamma})$$

$$\beta_{\langle u,v\rangle} = \sigma(\mathbf{W}_{\beta}\mathbf{h}_{\langle u,v\rangle} + \mathbf{U}_{\beta}\mathbf{h}_{\mathcal{Q}} + \mathbf{b}_{\beta})$$

$$\mathbf{h}_{\mathcal{G}}^{\mathcal{Q}} = \sigma(\mathbf{G} \cdot \text{AGGR}(\{\tilde{\mathbf{h}}_{\langle u,v\rangle} | \langle u,v\rangle \in E_{\mathcal{G}}\}))$$

- **Q**: encoded nodes or edges into input features
- $\boldsymbol{\gamma}_{\langle \mathbf{u},\mathbf{v}\rangle}$, $\boldsymbol{\beta}_{\langle \mathbf{u},\mathbf{v}\rangle}$: FiLM factors for scaling and shifting
- $\odot$: Hadamard product
- $\mathbf{W}_{\gamma}$, $\mathbf{U}_{\gamma}$, $\mathbf{W}_{\beta}$, $\mathbf{U}_{\beta}$: learnable weight metrices
- $\mathbf{b}_{\gamma}$, $\mathbf{b}_{\beta}$: learnable bias vectors

Counter Module and Overall Objective

Counter Module

$$\hat{n}(\mathcal{Q},\mathcal{G}) = \textsc{ReLU}(\mathbf{w}^\top \textsc{Match}(\mathbf{h}_\mathcal{Q}, \mathbf{h}_\mathcal{G}^\mathcal{Q}) + b)$$

$$\textsc{Match}(\mathbf{x},\mathbf{y}) = \text{FCL}(\mathbf{x} \parallel \mathbf{y} \parallel \mathbf{x}-\mathbf{y} \parallel \mathbf{x} \odot \mathbf{y})$$

- **MATCH(·, ·)**: outputs the matchability between its arguments
- **w, b**: learnable weight vector and bias
- **FCL**: full connected layer

Overall Objective

$$\frac{1}{|\mathcal{T}|} \sum_{(\mathcal{Q}_i,\mathcal{G}_i,n_i)\in\mathcal{T}} |\hat{n}(\mathcal{Q}_i,\mathcal{G}_i) - n_i| + \lambda \mathcal{L}_{\text{FiLM}} + \mu \|\Theta\|_2^2$$

$$\mathcal{L}_{\text{FiLM}} = \sum_{(\mathcal{Q}_i,\mathcal{G}_i,n_i)\in\mathcal{T}} \sum_{\langle u,v\rangle \in E_{\mathcal{G}_i}} \|\gamma_{\langle u,v\rangle}\|_2^2 + \|\beta_{\langle u,v\rangle}\|_2^2$$

- n_i: ground truth
- L_{FiLM}: regularizer on the FiLM factors
- $\| \Theta \|_2^2$: learnable weight vector and bias

SMU Classification: Restricted

Theoretical Analysis of Count-GNN

Lemma 1 (Generalization). *Count-GNN can be reduced to a node-centric GNN, i.e., Count-GNN can be regarded as a generalization of the latter.* □

Theorem 1 (Expressiveness). *Count-GNN is more powerful than node-centric GNNs, which means (i) for any two non-isomorphic graphs that can be distinguished by a node-centric GNN, they can also be distinguished by Count-GNN; and (ii) there exists two non-isomorphic graphs that can be distinguished by Count-GNN but not by a node-centric GNN.* □

PART 03

Experiments

SMU Classification: Restricted

Datasets & Baselines

- **Datasets**

	SMALL	LARGE	MUTAG	OGB-PPA
# Queries	75	122	24	12
# Graphs	6,790	3,240	188	6,000
# Triples	448,140	395,280	4,512	57,940
Avg($\|V_{\mathcal{Q}}\|$)	5.20	8.43	3.50	4.50
Avg($\|E_{\mathcal{Q}}\|$)	6.80	12.23	2.50	4.75
Avg($\|V_{\mathcal{G}}\|$)	32.62	239.94	17.93	152.75
Avg($\|E_{\mathcal{G}}\|$)	76.34	559.68	39.58	1968.29
Avg(Counts)	14.83	34.42	17.76	13.83
Max($\|L\|$)	16	64	7	8
Max($\|L'\|$)	16	64	4	1

- **Baselines**

Conventional GNNs:

GCN , GAT, DPGCNN, GIN, DiffPool

GNN-based isomorphism counting models:

RGCN-DN, RGCN-Sum, RGIN-DN, RGIN-Sum[2], LRP, DMPNN-LRP[3]

Exact Apporaches:

VF2, Peregrine[1]

[1] Jamshidi, K. et al. 2020. Peregrine: a pattern-aware graph mining system. EuroSys.
[2] Liu, X. et al. 2020. Neural subgraph isomorphism counting. KDD.
[3] Liu, X.; and Song, Y. 2022. Graph convolutional networks with dual message passing for subgraph isomorphism counting and matching. AAAI.

Empricial Results

Methods	SMALL MAE ↓	SMALL Q-error ↓	SMALL Time/s ↓	LARGE MAE ↓	LARGE Q-error ↓	LARGE Time/s ↓	MUTAG MAE ↓	MUTAG Q-error ↓	MUTAG Time/s ↓	OGB-PPA MAE ↓	OGB-PPA Q-error ↓	OGB-PPA Time/s ↓
GCN	14.8 ± 0.5	2.1 ± 0.1	7.9 ± 0.2	33.0 ± 0.4	3.5 ± 1.0	29.8 ± 0.7	19.9 ± 9.7	4.2 ± 1.5	0.88 ± 0.02	36.8 ± 1.4	2.1 ± 0.4	12.5 ± 0.3
GraphSAGE	14.0 ± 2.7	2.5 ± 0.8	**7.0** ± 0.1	33.8 ± 1.6	3.1 ± 0.4	**27.5** ± 1.3	13.9 ± 2.8	4.7 ± 0.8	0.88 ± 0.02	32.5 ± 4.5	2.5 ± 0.5	**11.1** ± 0.1
GAT	12.2 ± 0.7	2.0 ± 0.5	14.3 ± 0.3	37.3 ± 5.2	6.0 ± 1.2	59.4 ± 0.7	30.8 ± 6.7	6.0 ± 0.3	0.91 ± 0.01	35.8 ± 2.4	2.2 ± 0.6	30.4 ± 0.8
DPGCNN	16.8 ± 0.7	2.9 ± 0.2	21.7 ± 0.4	39.8 ± 3.7	5.4 ± 1.6	64.8 ± 0.9	27.5 ± 2.5	4.9 ± 0.6	1.54 ± 0.01	38.4 ± 1.2	2.3 ± 0.3	19.4 ± 0.7
DiffPool	14.8 ± 2.6	2.1 ± 0.4	**7.0** ± 0.1	34.9 ± 1.4	3.8 ± 0.7	32.5 ± 0.7	6.4 ± 0.3	2.5 ± 0.2	0.86 ± 0.00	35.9 ± 4.7	2.7 ± 0.3	15.4 ± 2.2
GIN	12.6 ± 0.5	2.1 ± 0.1	7.1 ± 0.0	35.9 ± 0.6	4.8 ± 0.2	33.5 ± 0.6	21.3 ± 1.0	5.6 ± 0.7	0.41 ± 0.01	34.6 ± 1.4	2.5 ± 0.5	12.3 ± 0.4
RGCN-Sum	24.2 ± 6.1	3.7 ± 1.2	13.2 ± 0.1	80.9 ± 26.3	6.3 ± 1.3	61.8 ± 0.2	8.0 ± 0.8	**1.5** ± 0.1	0.89 ± 0.01	34.5 ± 13.6	4.7 ± 0.8	33.0 ± 0.2
RGCN-DN	16.6 ± 2.3	3.2 ± 1.3	48.1 ± 0.2	73.7 ± 29.2	9.1 ± 4.2	105.0 ± 0.4	7.3 ± 0.8	2.6 ± 0.2	1.19 ± 0.04	57.1 ± 15.7	5.0 ± 1.3	31.2 ± 0.1
RGIN-Sum	10.7 ± 0.3	2.0 ± 0.2	12.2 ± 0.0	33.2 ± 2.2	4.2 ± 1.3	61.4 ± 1.0	10.8 ± 0.9	1.9 ± 0.1	0.45 ± 0.02	29.1 ± 1.7	1.2 ± 0.6	21.0 ± 1.2
RGIN-DN	11.6 ± 0.2	2.4 ± 0.0	49.7 ± 1.8	32.5 ± 1.9	4.3 ± 2.0	104.0 ± 1.5	8.6 ± 1.9	3.3 ± 0.8	0.73 ± 0.03	35.8 ± 6.4	4.4 ± 1.1	28.8 ± 0.3
DMPNN-LRP	9.1 ± 0.2	1.5 ± 0.1	32.4 ± 1.4	**28.1** ± 1.3	3.4 ± 1.5	184.2 ± 1.8	5.4 ± 1.8	1.8 ± 1.0	0.13 ± 0.05	**25.6** ± 4.9	1.1 ± 1.3	63.0 ± 0.6
Count-GNN	**8.5** ± 0.0	**1.4** ± 0.1	7.9 ± 0.3	30.9 ± 4.3	**2.5** ± 0.5	59.2 ± 1.7	**4.2** ± 0.1	1.8 ± 0.0	**0.02** ± 0.00	28.7 ± 3.9	**1.0** ± 0.2	18.1 ± 0.6
VF2	0	1	1049.2 ± 2.7	0	1	9270.5 ± 5.9	0	1	1.30 ± 0.04	0	1	5836.3 ± 4.8
Peregrine	-	-	72.4 ± 2.0	-	-	904.2 ± 4.5	-	-	0.2 ± 0.03	-	-	450.1 ± 3.9

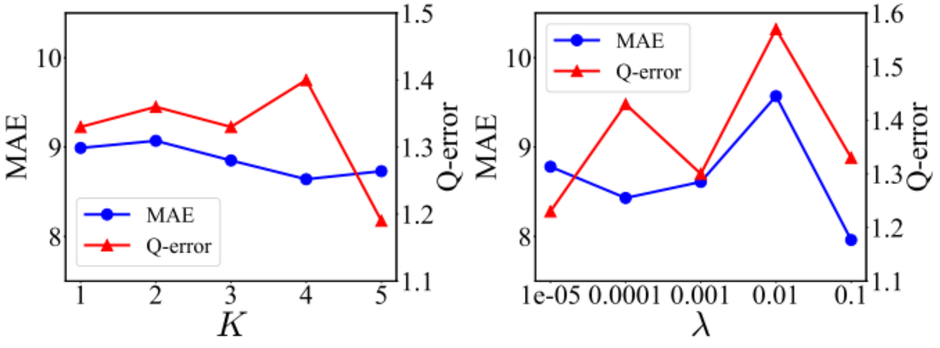
- **Observations**
  - Count-GNN achieves 65x ~ 324x speedups over the classical VF2, 8x ~ 26x speedups over Peregrine
  - Count-GNN is more efficient than other GNN-based isomorphism counting models
  - Count-GNN is more accurate than conviential GNN models by at least 30% improvements in most cases.

SMU Classification: Restricted

Ablation Study & Parameter Sensitivity

Table 5: Ablation study on Count-GNN.

Methods	SMALL		LARGE		MUTAG	
	MAE	Q-error	MAE	Q-error	MAE	Q-error
Count-GNN\E	11.3	2.07	33.58	4.96	18.63	5.92
Count-GNN\M	8.66	1.46	**29.65**	3.34	4.41	1.82
Count-GNN	**8.54**	**1.41**	30.91	**2.46**	**4.22**	**1.76**



(a) Sensitivity of K ↓ (b) Sensitivity of λ ↓

Figure II: Parameters sensitivity on dataset SMALL.

- **Ablation study**
  - **Count-GNN\E:** replaces the edge-centric aggregation with the node-centric GIN
  - **Count-GNN\N:** replaces the query-conditioned modulation with a simple sumpooling as the readout for the input graph
  - Count-GNN\E has the lowest accuracy
  - Count-GNN\M performs better than Count-GNN\E
  - Full model Count-GNN achieves the best performance

- **Parameters sensitivity**
  - As K increases, the performance in terms of MAE and Q-error generally become better, only with one exception on Q-error when K = 4
  - λ = 0.01 may result in an inferior performance. Interval [1e-5, 1e-3] might be a good range for superior performance of subgraph isomorphism counting.

PART 04

Couclusions

Conclusions

- **Problem**
  - Subgraph isomorphic counting
- **Proposed-Model: Count-GNN**
  - Edge-centric message passing
  - Query-conditioned graph modulation
- **Experiment**
  - Count-GNN achieves significant speedups over exact methods
  - Count-GNN is more efficient than other GNN-based isomorphism counting models
  - Count-GNN is more accurate than conviential GNN models by at least 30% improvements in most cases.

Thanks!

Paper, data & code available at https://xingtongyu.netlify.app/

Xingtong Yu*, Zemin Liu*, Yuan Fang†, Xinming Zhang†

Learning to Count Isomorphisms with Graph Neural Networks

In Proceeding of 37th AAAI Conference on Artificial Intelligence, February 7-14, 2023