Graph Condensation for Continual Graph Learning

Ruihong Qiu

The University of Queensland

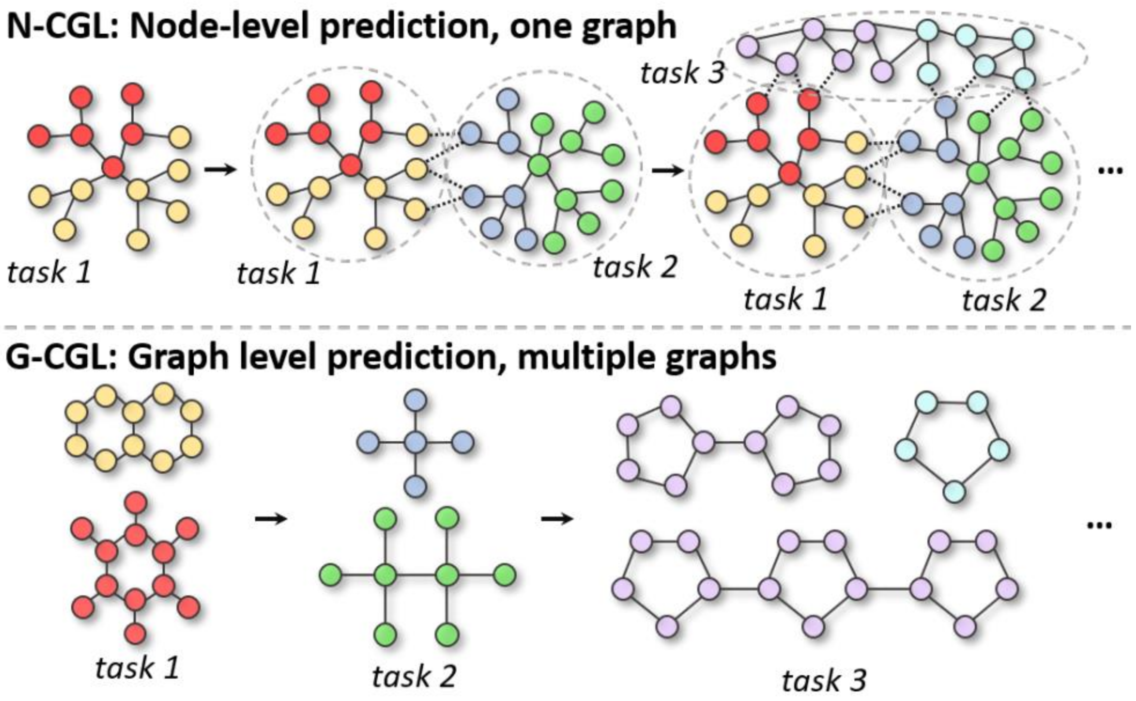
r.qiu@uq.edu.au

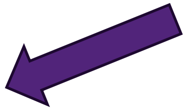
Contents

Contents

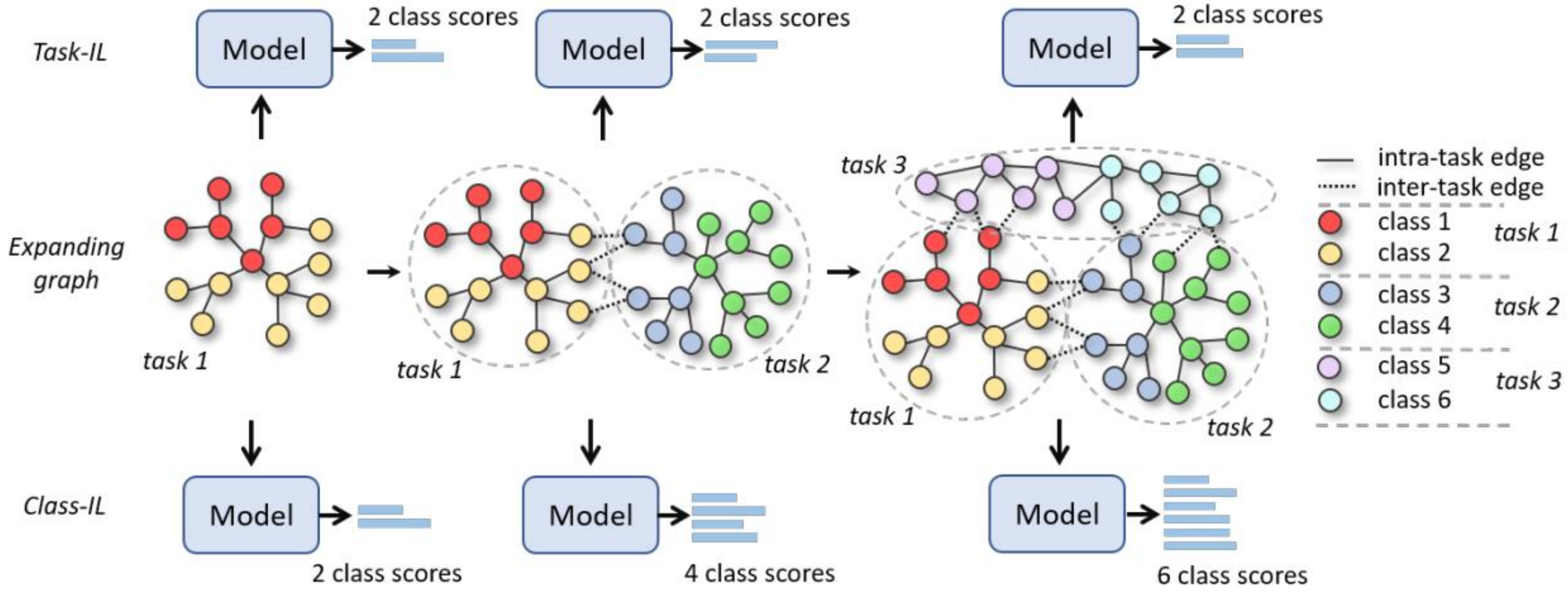
Continual Graph Learning (CGL)

- Traditional Graph Neural Networks (GNNs) are not good at **streaming** inputs.
- **New** nodes can appear **dynamically**





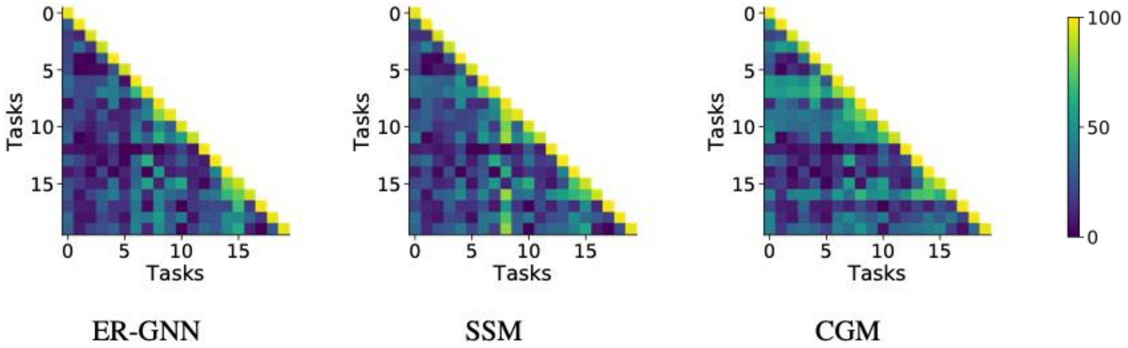
Class/Task Incremental Learning (IL)



- **Class-IL** is much **harder** than Task-IL

Catastrophic Forgetting (CF)

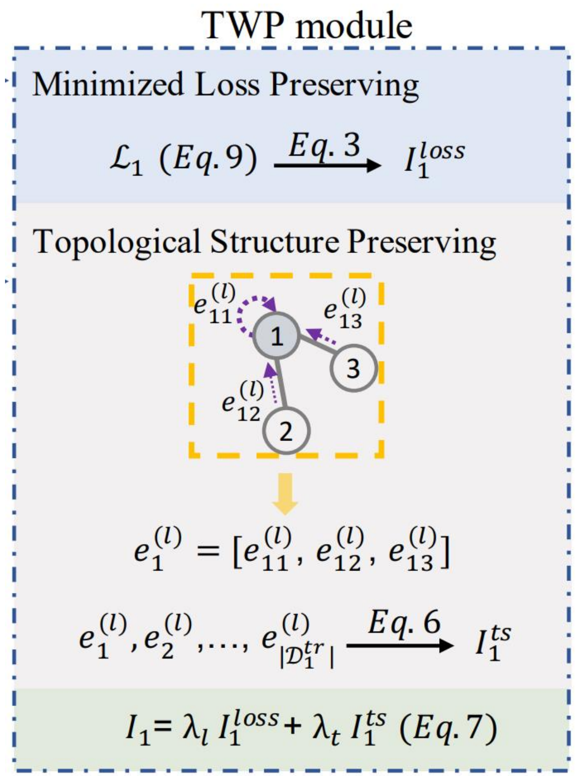
- CF is a **general** challenge in CGL
- **Old** knowledge covered by **new** ones.



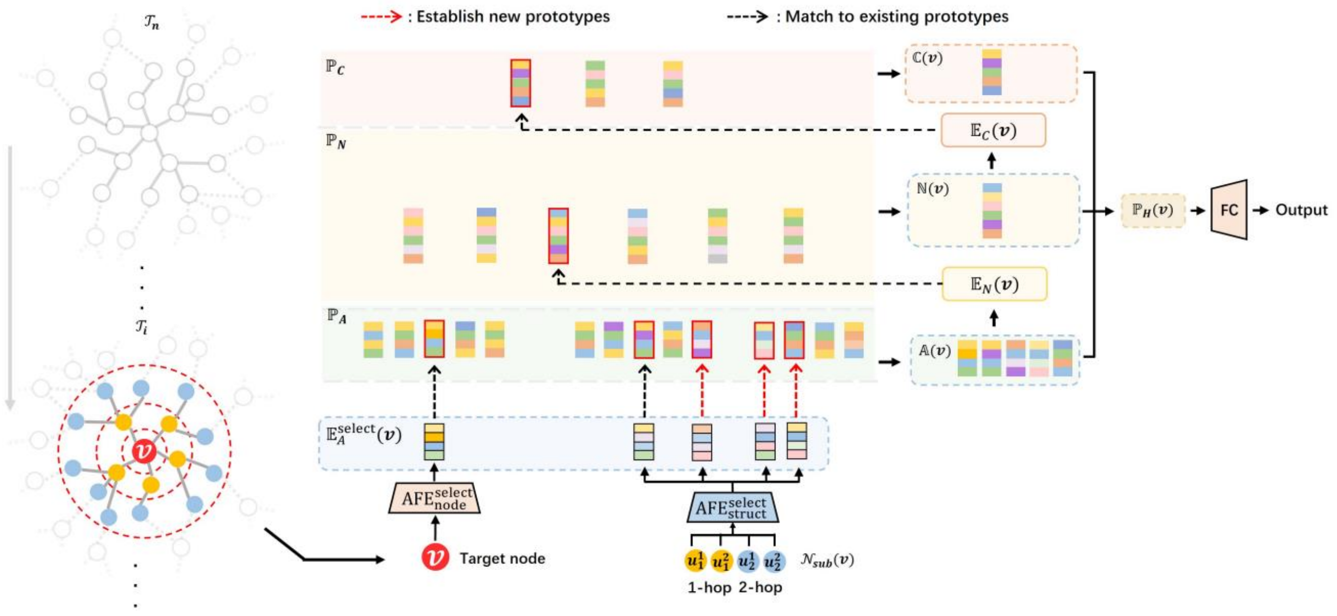
Contents

Related Work

- Regularisation-based: TWP



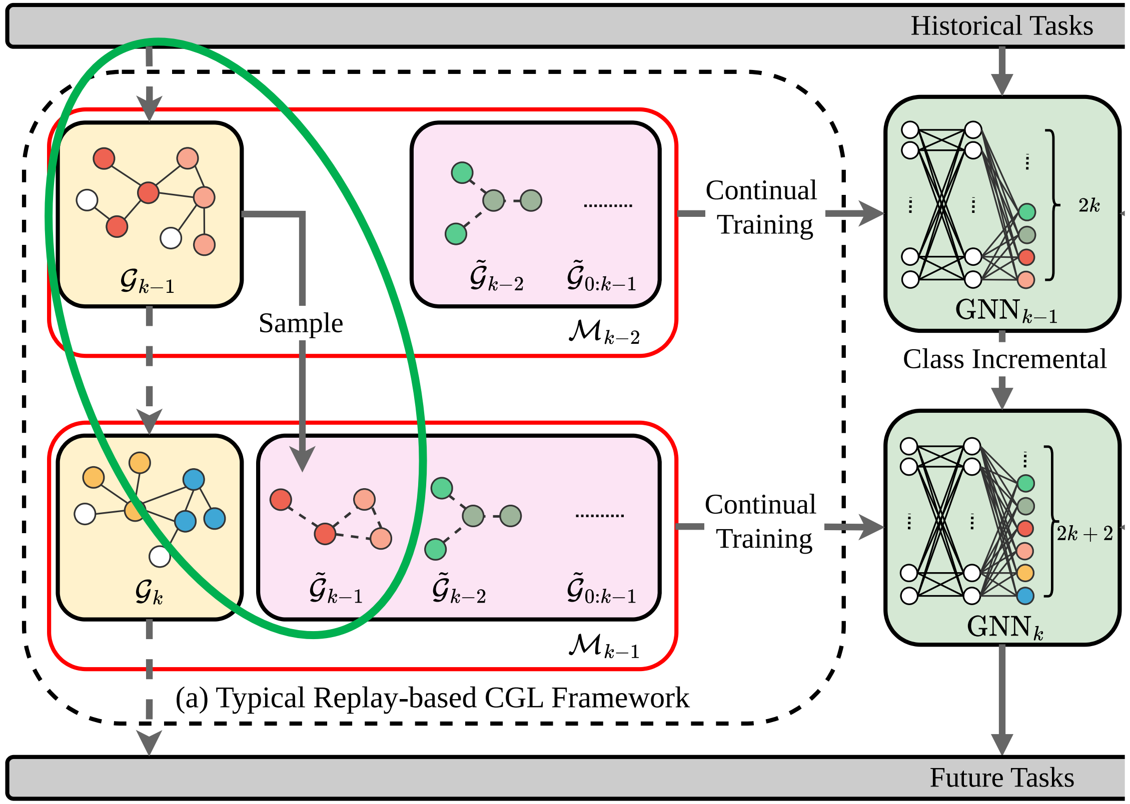
- Architecture-based: HPNs



Overcoming Catastrophic Forgetting in Graph Neural Networks, AAAI 2021
Hierarchical Prototype Networks for Continual Graph Representation Learning, TPAMI 2022

Related Work

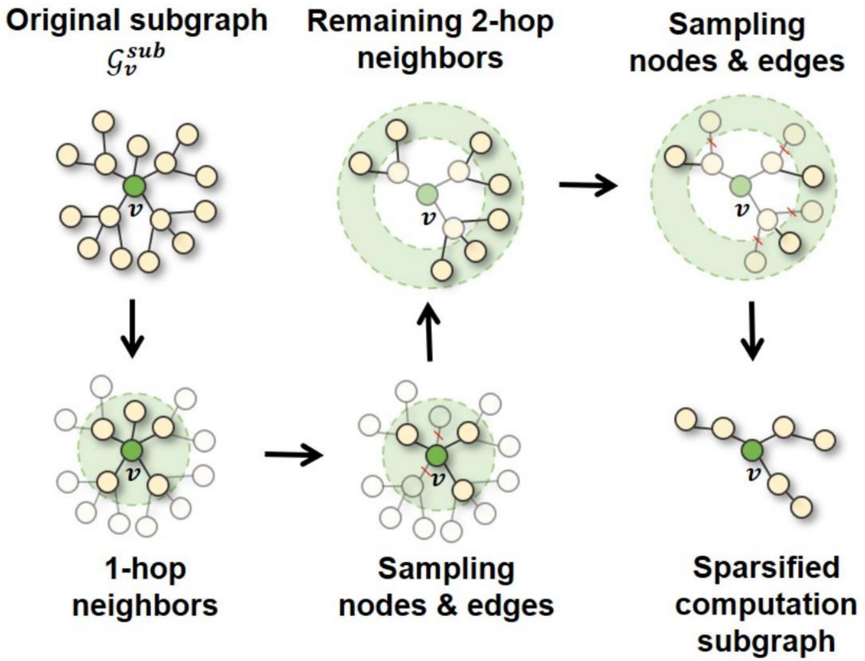
- Replay-based



Related Work

- Replay-based: ER-GNN
  - Mean feature
  - Coverage Maximization
  - Influence Maximization

- SSM



Overcoming Catastrophic Forgetting in Graph Neural Networks with Experience Replay, AAAI 2021
Hierarchical Prototype Networks for Continual Graph Representation Learning, TPAMI 2022

Contents

- Continual Graph Learning
- Related Work
- CaT
- PUMA

CaT 🐈: Balanced Continual Graph Learning with Graph Condensation

Yilun Liu, Ruihong Qiu, Zi Huang

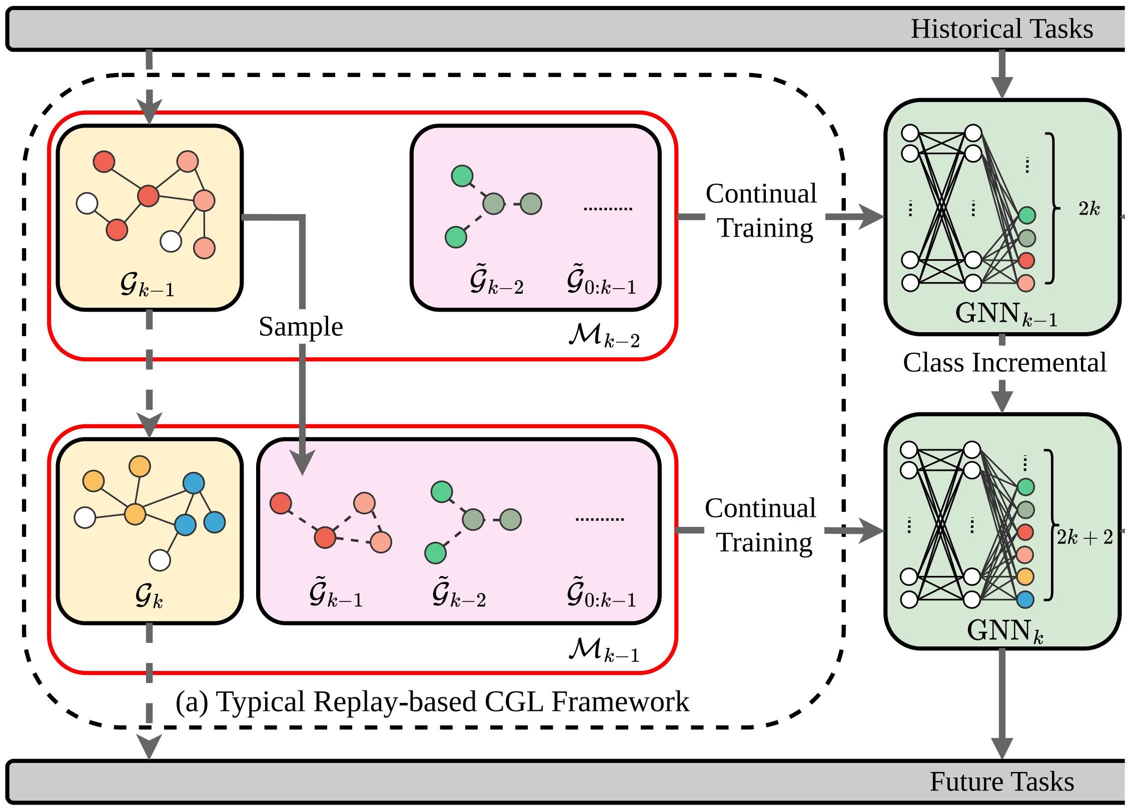
The University of Queensland

yilun.liu@uq.edu.au

https://github.com/superallen13/CaT-CGL

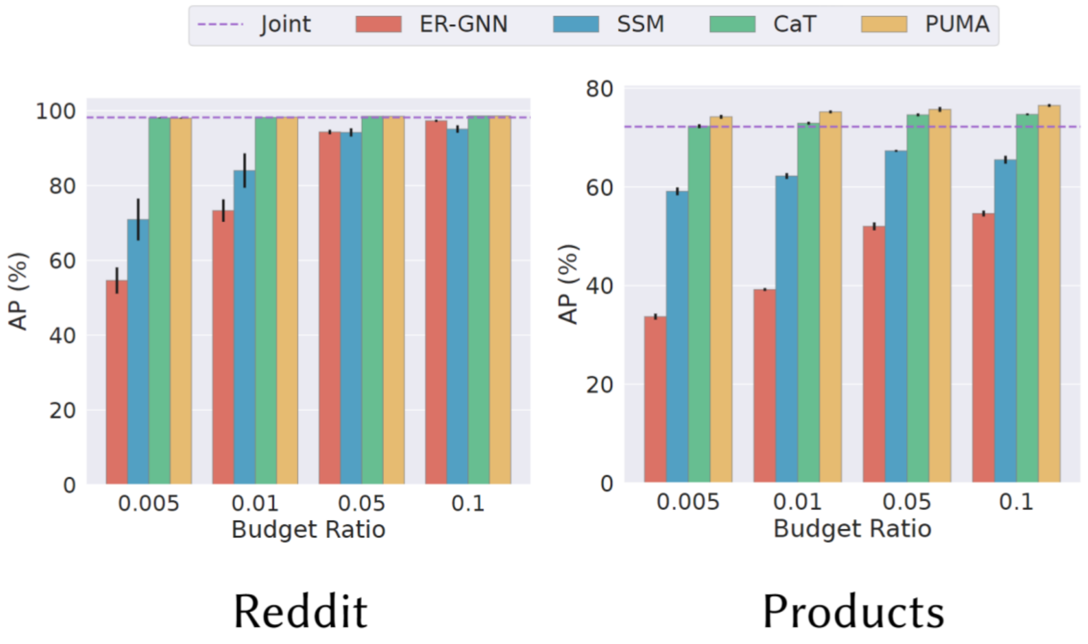
Batch Training in Replay-based CGL

- Sample and combine

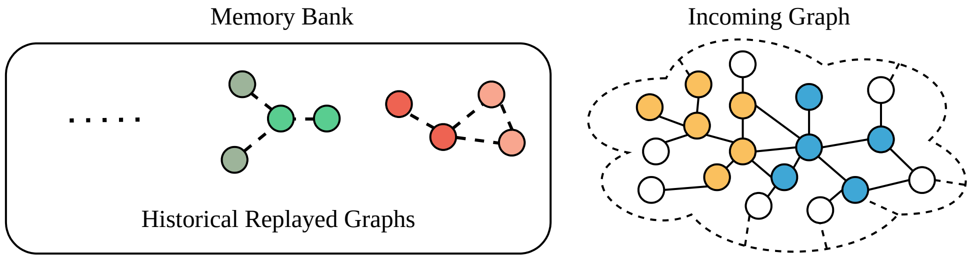


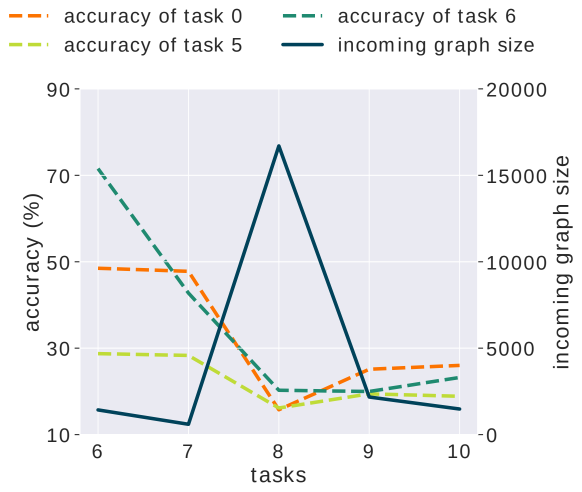
Challenges in Replay

- **Large** storage requirement

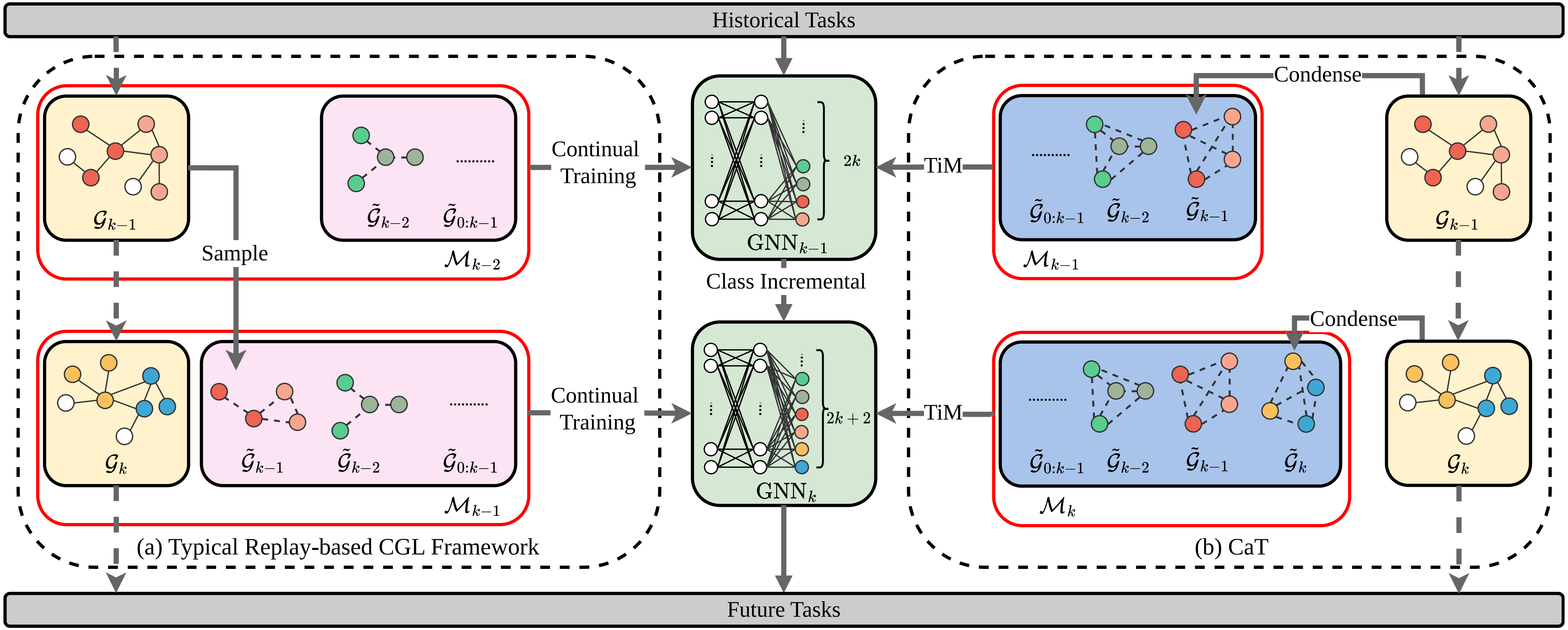


- **Imbalanced** training graph size





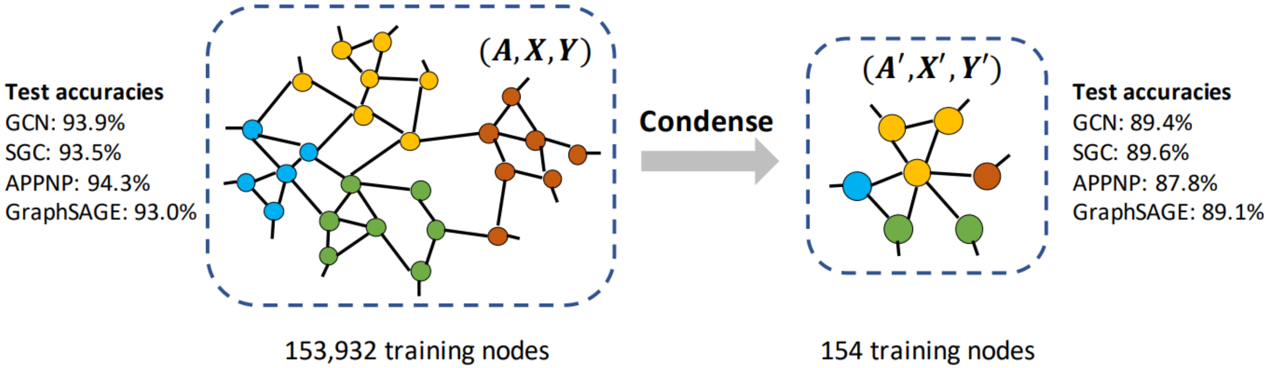
Condense and Train (CaT) Framework



Graph Condensation

- Smaller yet effective
- Model **performance** trained **on condensed** graph matches on **original** graph
- **Bi-level** optimisation problem

$$\min_{\tilde{G}} \mathcal{L}(G; \tilde{\theta}), s.t. \tilde{\theta} = \underset{\theta}{\operatorname{argmin}} \mathcal{L}(\tilde{G}; \theta)$$



Condensed Graph Memory (CGM)

- **Distribution matching** method.
- **Random** GNN encoder to obtain latent features.
- Minimise **MMD** losses

$$\ell_{\text{MMD}} = \sum_{c \in \mathcal{C}_k} r_c \cdot ||\text{Mean}(\boldsymbol{E}_{k,c}) - \text{Mean}(\tilde{\boldsymbol{E}}_{k,c})||^2$$

Train in Memory (TiM)

- Condense **incoming** graph
- **Balanced** replayed graphs.

Algorithm 2: Overall procedure of CaT

Input: A streaming of tasks $\{\mathcal{T}_1, \mathcal{T}_2, ..., \mathcal{T}_K\}$
Output: GNN_K

Initialise a CGL model GNN_0;
Initialise an empty memory bank $\mathcal{M}_0$;
for $k \leftarrow 1$ **to** K **do**
  Extract incoming graph $\mathcal{G}_k$ from $\mathcal{T}_k$;
  Obtain $\tilde{\mathcal{G}}_k$ by CGM; /* Algorithm 1 */
  $\mathcal{M}_k = \mathcal{M}_{k-1} \cup \tilde{\mathcal{G}}_k$; /* Eq. 11 */
  Update GNN_{k-1} to GNN_k ; /* Eq. 12 */
end

Implementation

- Baselines
  - Joint: **full-size**.
  - ER-GNN: informative **nodes**.
  - SSM: **subgraphs**.
- Metrics
  - Average performance (AP): $\frac{1}{k}\sum_{i=1}^{k} m_{k,i}$
  - Backward transfer (BWT): $\frac{1}{k-1}\sum_{i=1}^{k-1} m_{k,i} - m_{i,i}$

- Experiment settings
  - Dataset splitting [CGLB, NeurIPS 2022]
    - Each task contains two classes.
  - Task incremental learning (task-IL)
    - Classification heads are growing.

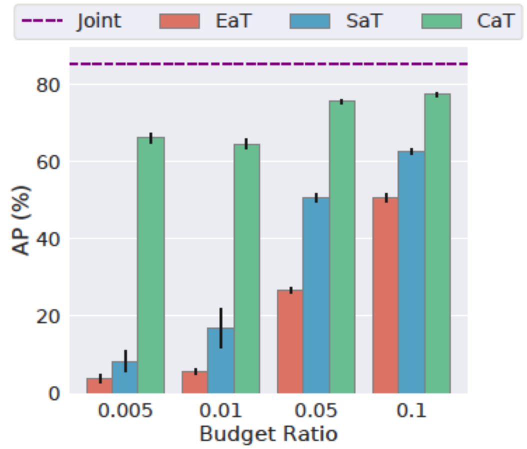
Dataset	Nodes	Edges	Features	Classes	Tasks
CoraFull	19,793	130,622	8,710	70	35
Arxiv	169,343	1,166,243	128	40	20
Reddit	227,853	114,615,892	602	40	20
Products	2,449,028	61,859,036	100	46	23

Overall Results

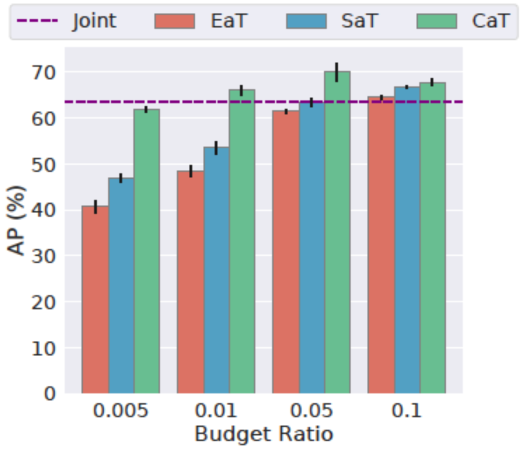
Category	Methods	CoraFull		Arxiv		Reddit		Products	
		AP (%) ↑	BWT (%) ↑	AP (%) ↑	BWT (%) ↑	AP (%) ↑	BWT (%) ↑	AP (%) ↑	BWT (%) ↑
Lower bound	Finetuning	2.2±0.0	-96.6±0.1	5.0±0.0	-96.7±0.1	5.0±0.0	-99.6±0.0	4.3±0.0	-97.2±0.1
Regularisation	EWC	2.9±0.2	-96.1±0.3	5.0±0.0	-96.8±0.1	5.3±0.6	-99.2±0.7	7.6±1.1	-91.7±1.4
	MAS	2.2±0.0	-94.1±0.6	4.9±0.0	-95.0±0.7	10.7±1.4	-92.7±1.5	10.1±0.6	-89.0±0.5
	GEM	2.5±0.1	-96.6±0.1	5.0±0.0	-96.8±0.1	5.3±0.5	-99.3±0.5	4.3±0.1	-96.8±0.1
	TWP	<u>21.2±3.2</u>	<u>-67.4±1.6</u>	4.3±1.1	-93.0±8.3	9.5±2.0	-35.5±5.5	6.8±3.5	-64.3±12.8
Distillation	LWF	2.2±0.0	-96.6±0.1	5.0±0.0	-96.8±0.1	5.0±0.0	-99.5±0.0	4.3±0.0	-96.8±0.2
Replay	ER-GNN	4.0±0.7	-94.3±0.9	30.8±0.6	-68.3±0.7	31.8±4.0	-71.2±4.2	39.5±1.3	-48.2±1.4
	SSM	16.2±2.8	-82.1±2.9	<u>35.1±1.8</u>	<u>-63.7±1.9</u>	<u>51.6±6.4</u>	<u>-50.3±6.7</u>	<u>62.7±0.5</u>	<u>-22.1±0.5</u>
Full dataset	Joint	85.3±0.1	-2.7±0.0	63.5±0.3	-15.7±0.4	98.2±0.0	-0.5±0.0	72.2±0.4	-5.3±0.5
Ours	CaT	**64.5±1.4**	**-3.3±2.6**	**66.0±1.1**	**-13.1±1.0**	**97.6±0.1**	**-0.2±0.2**	**71.0±0.2**	**-4.8±0.4**

- **Replayed**-based methods are overall **better**.
- **CaT** is the **best**, sometimes can **match** the ideal **Joint** scenario.

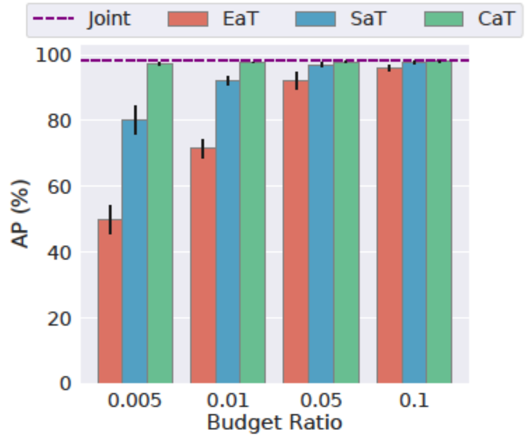
Effectiveness of CGM



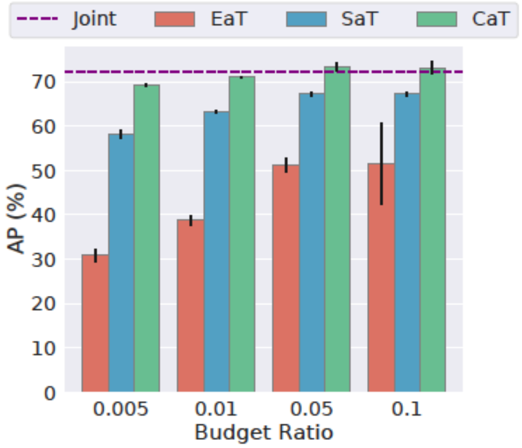
(a) CoraFull



(b) Arxiv



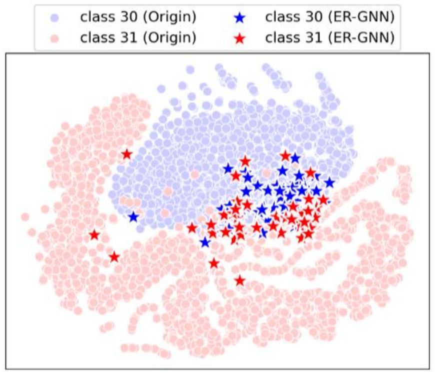
(c) Reddit



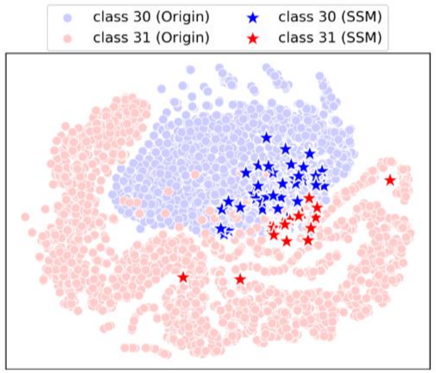
(d) Products

- Use TiM for ER-GNN and SSM
- **CGM** is **more effective** than other memory banks

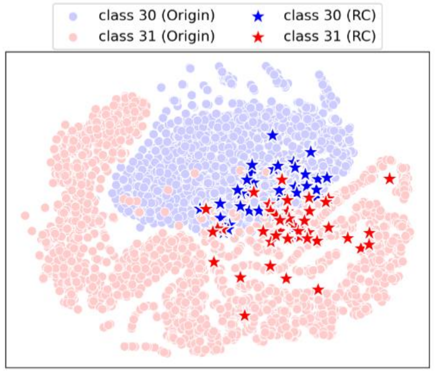
Visualisation of CGM



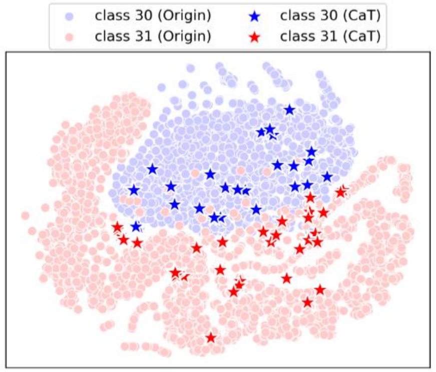
(a) ER-GNN



(b) SSM



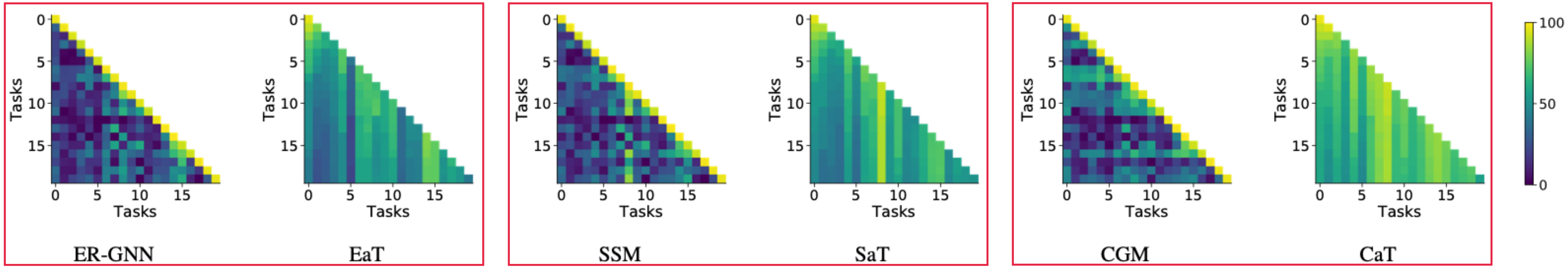
(c) Random Choice



(d) CGM

- Good **coverage** of the **original** distribution

TiM for Catastrophic Forgetting



- TiM has a **balanced** learning to **solve CF**.

Conclusion

1. CGM: Graph **condensation** gives a **small** yet **effective** memory bank

2. TiM: A training scheme for **balanced** continual learning

https://github.com/superallen13/CaT-CGL

Contents

- Continual Graph Learning
- Related Work
- CaT
- PUMA

THE UNIVERSITY OF QUEENSLAND
AUSTRALIA

PUMA 🐆: Efficient Continual Graph Learning via Retraining with Pseudo-label Guided Graph Condensation

Yilun Liu, Ruihong Qiu, Yanran Tang, Zi Huang

The University of Queensland

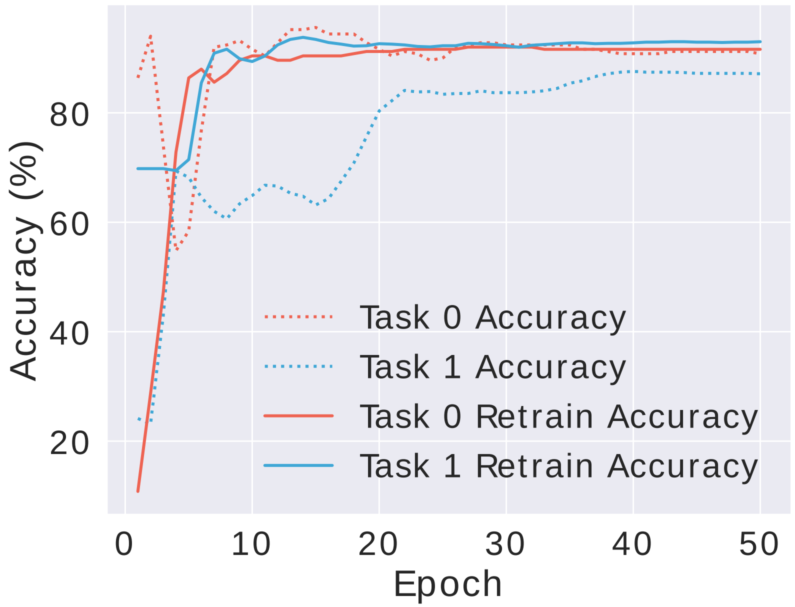
yilun.liu@uq.edu.au

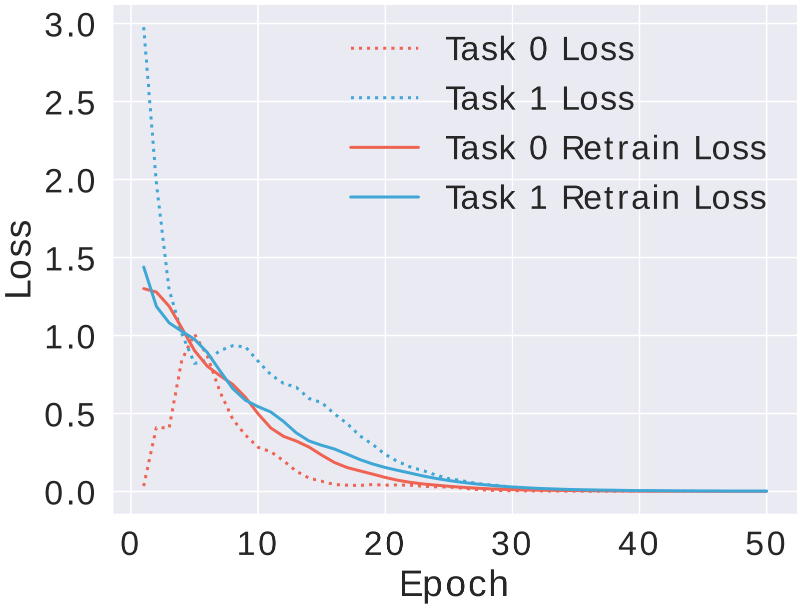
https://github.com/superallen13/PUMA

Problems of CaT

1. Unlabelled nodes in streaming graph data

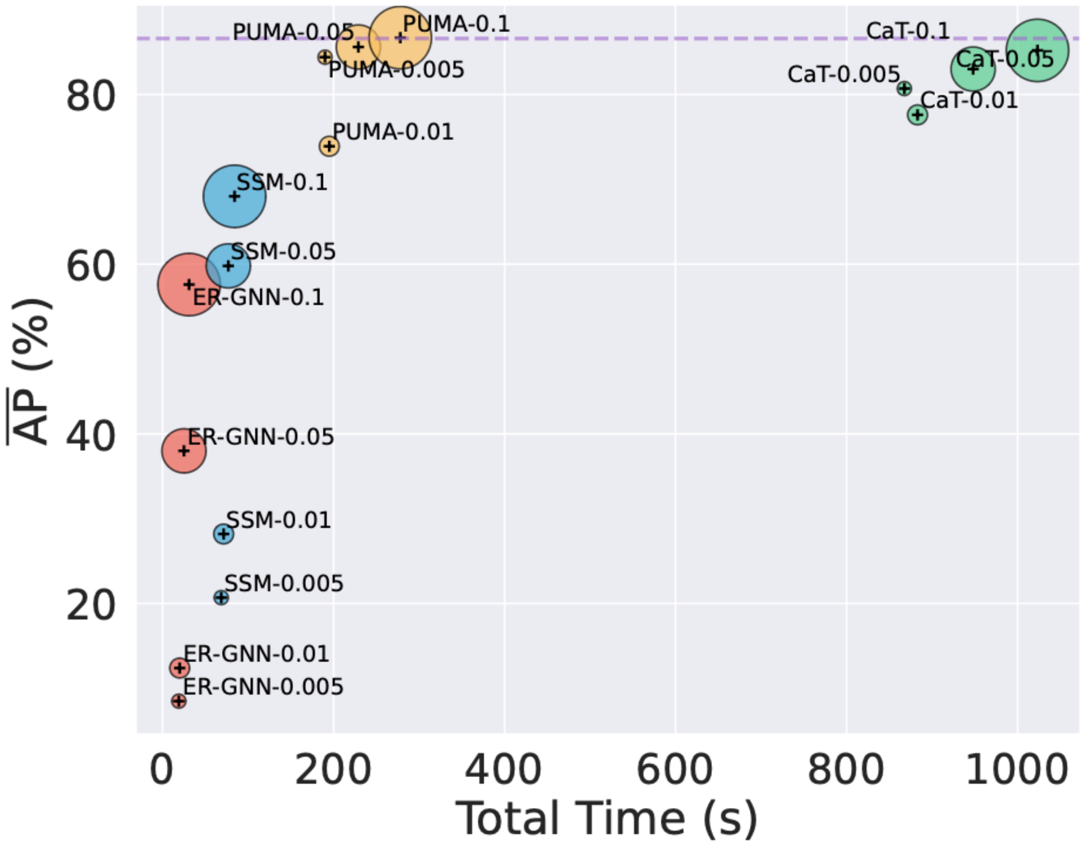
2. Still imbalance historical knowledge



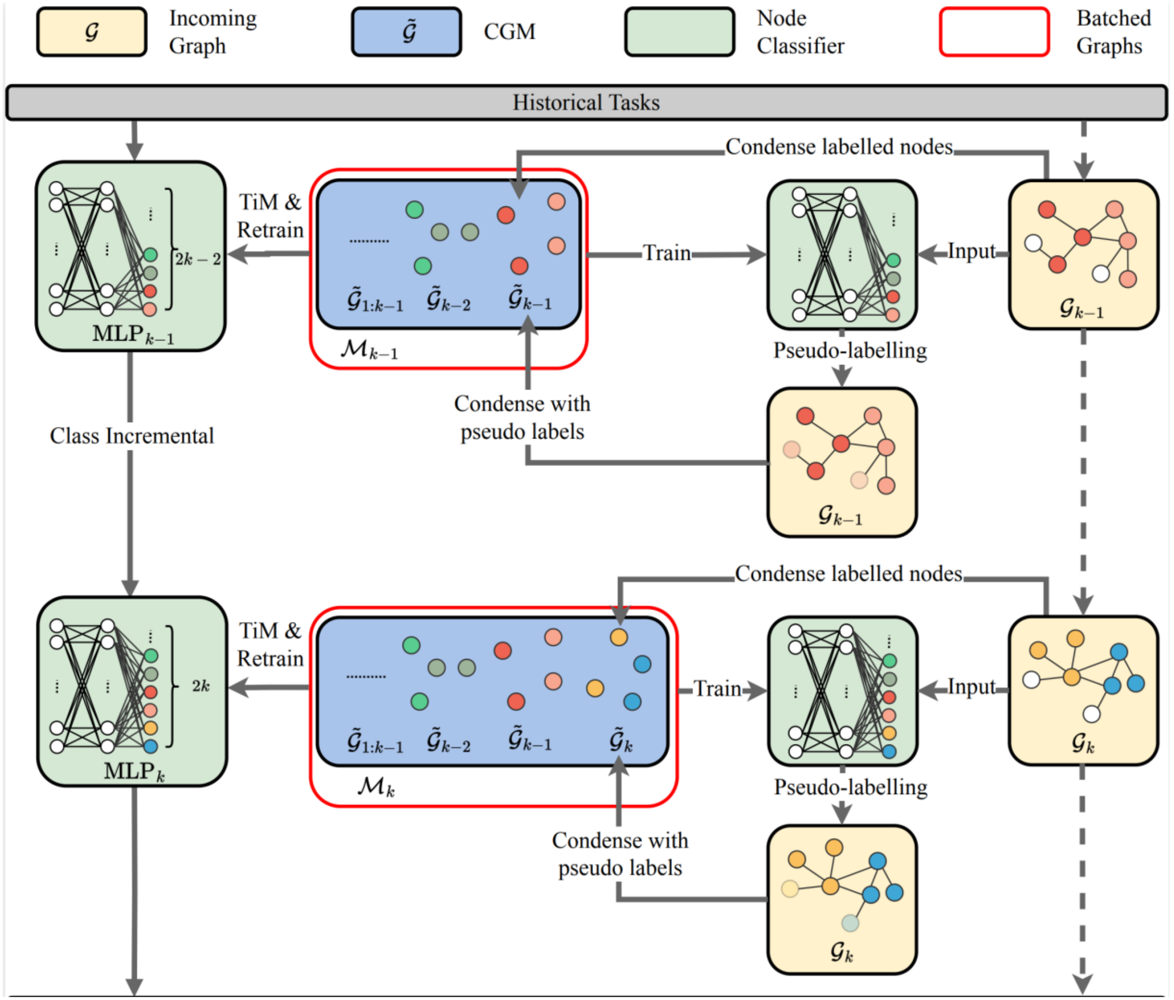


Problems of CaT

3. **Slow** condensation and training



PUMA Framework



Pseudo Labelling-Guided CGM

- An extra classifier for **pseudo labels**
- Select unlabelled nodes with a **high** confidence score
- **Condense both labelled** and confidently **pseudo labelled** nodes

Train from Scratch

- **No** more **continual** training, but retraining **from scratch**
- **Balanced** historical knowledge and incoming knowledge.

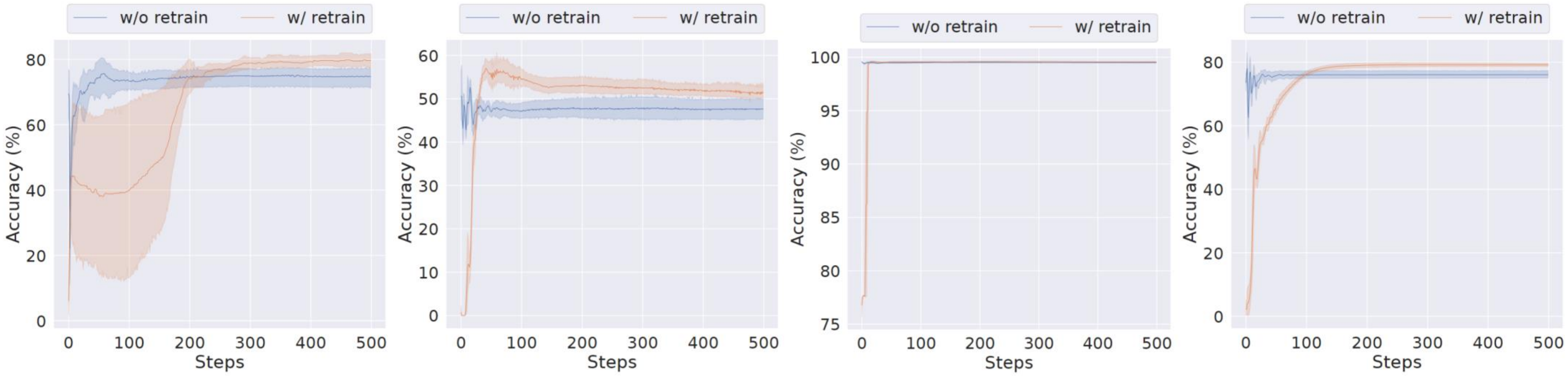
Fast Condensation and Training

- One-time propagation
- Wide graph encoders
- They are fast in calculation and not sacrificing performance. Details in paper

Overall Performance

Category	Methods	CoraFull		Arxiv		Reddit		Products	
		AP (%) ↑	BWT (%) ↑	AP (%) ↑	BWT (%) ↑	AP (%) ↑	BWT (%) ↑	AP (%) ↑	BWT (%) ↑
Lower bound	Finetuning	2.2±0.0	-96.6±0.1	5.0±0.0	-96.7±0.1	5.0±0.0	-99.6±0.0	4.3±0.0	-97.2±0.1
Regularisation	EWC	2.9±0.2	-96.1±0.3	5.0±0.0	-96.8±0.1	5.3±0.6	-99.2±0.7	7.6±1.1	-91.7±1.4
	MAS	2.2±0.0	-94.1±0.6	4.9±0.0	-95.0±0.7	10.7±1.4	-92.7±1.5	10.1±0.6	-89.0±0.5
	GEM	2.5±0.1	-96.6±0.1	5.0±0.0	-96.8±0.1	5.3±0.5	-99.3±0.5	4.3±0.1	-96.8±0.1
	TWP	21.2±3.2	-67.4±1.6	4.3±1.1	-93.0±8.3	9.5±2.0	-35.5±5.5	6.8±3.5	-64.3±12.8
Distillation	LWF	2.2±0.0	-96.6±0.1	5.0±0.0	-96.8±0.1	5.0±0.0	-99.5±0.0	4.3±0.0	-96.8±0.2
Replay	ER-GNN	3.1±0.2	-94.6±0.2	23.2±0.5	-77.1±0.5	20.0±3.0	-83.7±3.1	34.0±1.0	-55.7±0.8
	SSM	3.5±0.5	-94.7±0.5	26.4±0.8	-73.7±0.9	41.8±3.2	-60.8±3.4	58.1±0.4	-29.3±0.5
Full dataset	Joint	85.3±0.1	-2.7±0.0	63.5±0.3	-15.7±0.4	98.2±0.0	-0.5±0.0	72.2±0.4	-5.3±0.5
Condensation	CaT (ours)	68.5±0.9	-5.7±1.3	64.9±0.3	-12.5±0.8	97.7±0.1	-0.4±0.1	71.1±0.3	-5.4±0.3
	PUMA (ours)	**77.9±0.2**	**-4.2±0.9**	**67.0±0.1**	**-12.2±0.3**	**98.0±0.1**	**-0.3±0.1**	**74.2±0.4**	**-4.1±0.5**

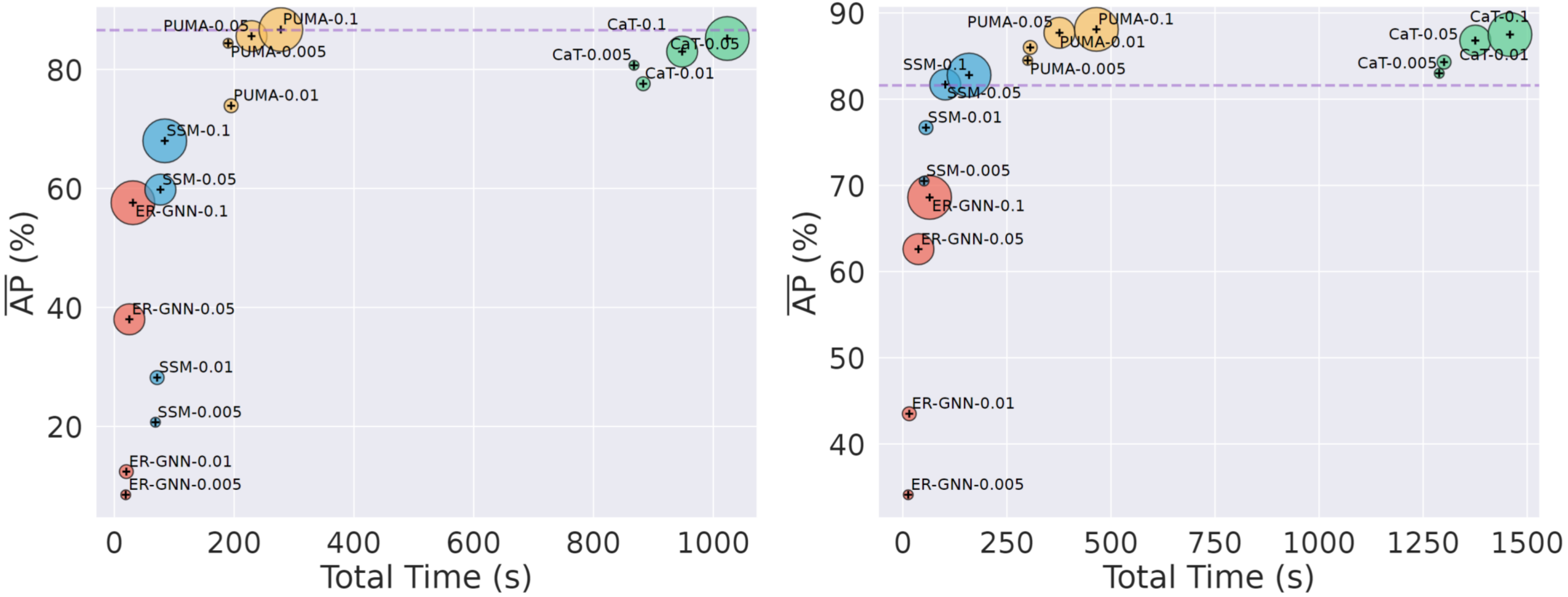
Effectiveness of Retraining



(a) CoraFull (b) Arxiv (c) Reddit (d) Products

- **Converge higher**

Time Efficiency



(a) CoraFull

(b) Products

- **PUMA is fast and performs well**

Conclusion

1. **Unlabelled** nodes help with condensation
2. **Retraining** improve the performance
3. Careful designs to **accelerate**

Q&A

Ruihong Qiu

The University of Queensland

r.qiu@uq.edu.au