One Size Fits All for Semantic Shifts: Adaptive Prompt Tuning for Continual Learning

Doyoung Kim, Susik Yoon, Dongmin Park, Youngjun Lee, Hwanjun Song, Jihwan Bang, Jae-Gil Lee

Korea Advanced Institute of Science and Technology

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

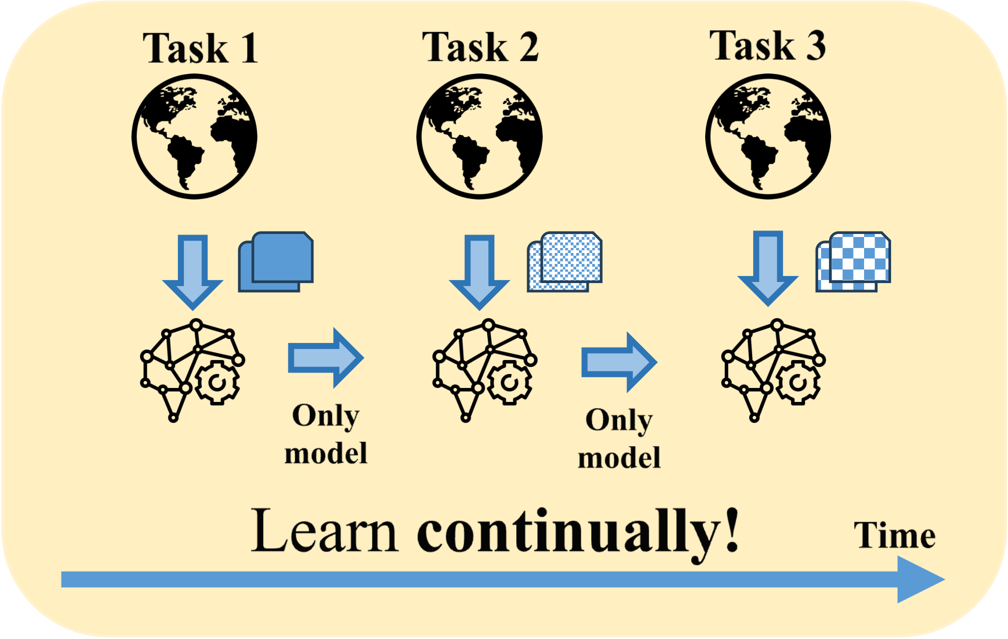
INDEX

01 | Introduction

Continual Learning (CL)

- Practical DNN learning scenario in real-world applications
  - ✓ Data distributions (“tasks”) arbitrarily change over time

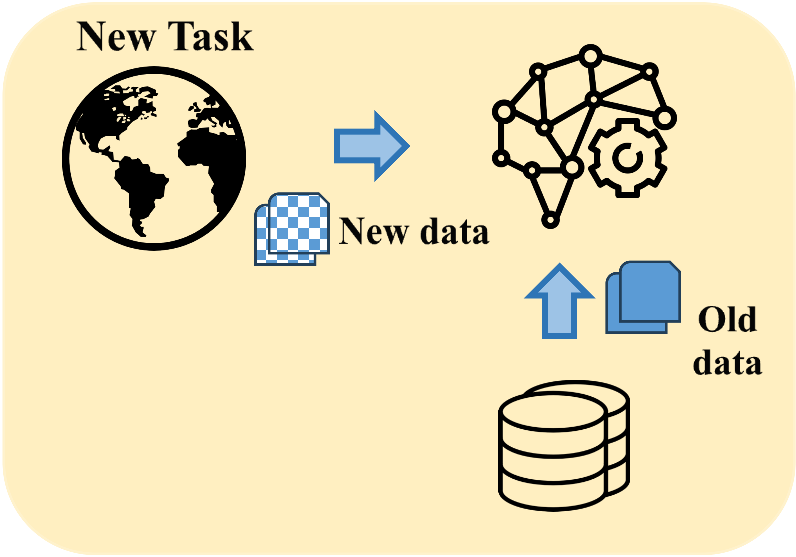
Continual learning



Traditional CL Methods

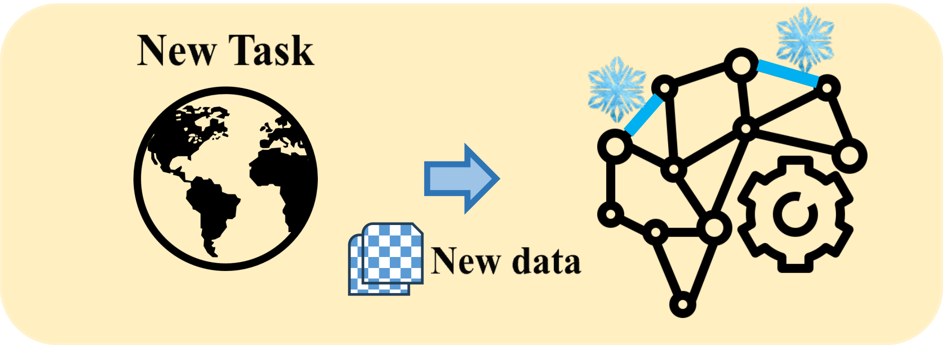
- CL methods mitigates **stability-plasticity** dilemma

Rehearsal-based OCL



Retain subsets of past data

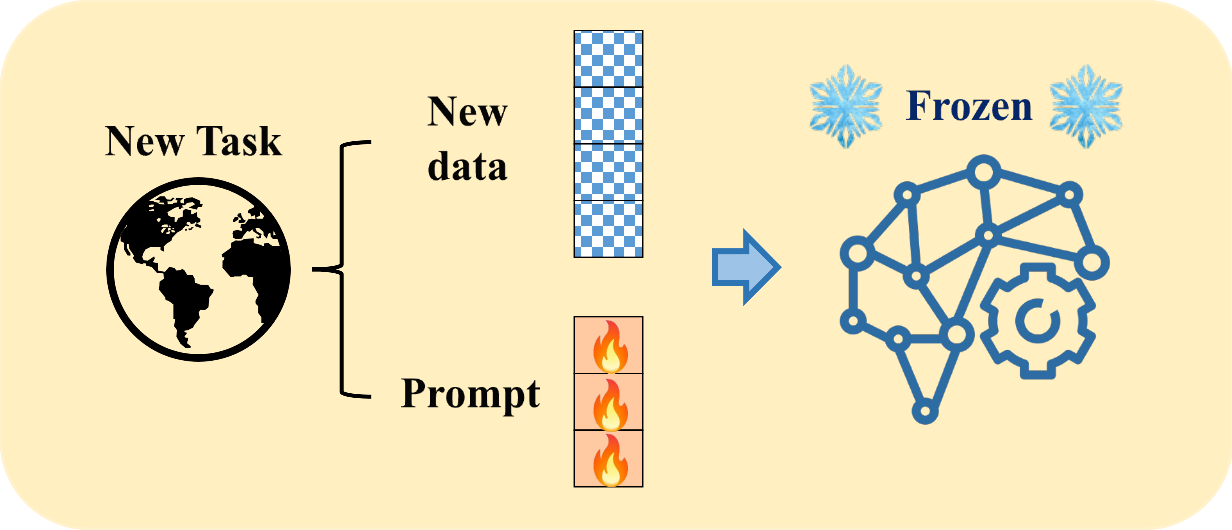
Regularization-based OCL



Constrain updates of past knowledge

Rehearsal-Free CL Methods

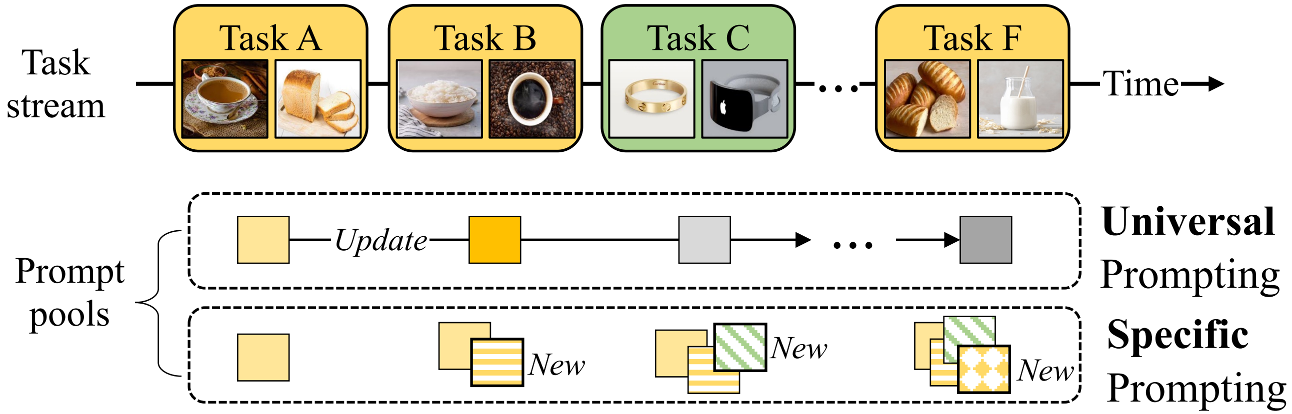
- Gained interest due to **superior** performance without using past data
- Use small learnable parameters, known as **prompts**, to refine a **pretrained** model for various tasks



Train **prompts** for new tasks

Categorization of Rehearsal-Free CL Methods

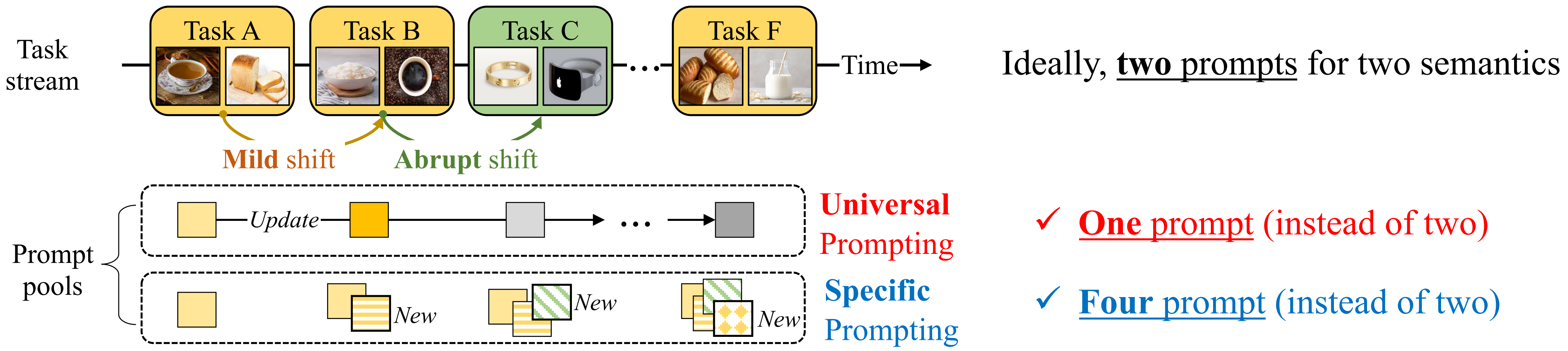
- **Universal** prompting methods [1-3]
  - ✓ Train fixed set of prompts for **all** tasks
- **Specific** prompting methods [4]
  - ✓ Train specific prompt for **each** task



Comparison of existing prompting CL methods

Limitations of Existing Prompting Methods

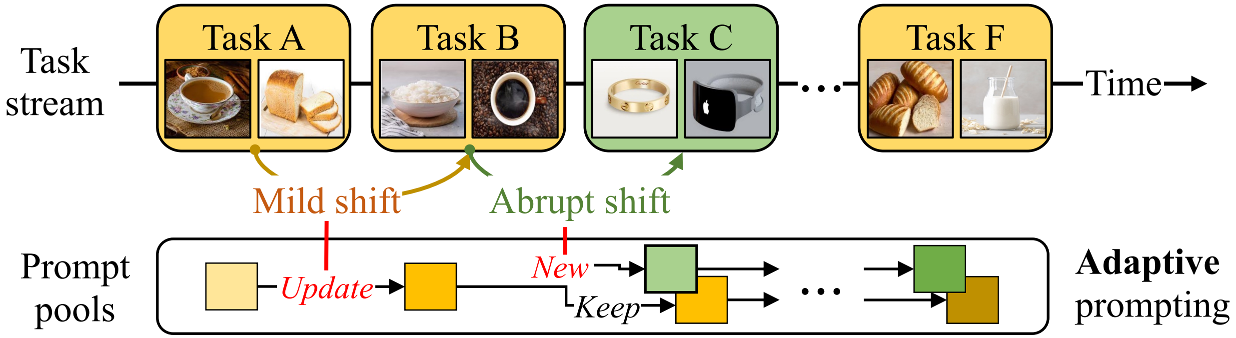
- Real-world CL scenarios: **Varying** degrees of semantic shifts
  - ✓ Universal prompting methods: **Insufficient** prompts
  - ✓ Specific prompting methods: **Redundant** prompts



Limitations of existing prompting CL methods

Research Goal: Accommodating Varying Semantic Shifts

- **Adaptive** prompting method
  - ✓ Handle **varying** degrees of semantic shifts by based on **task semantics**
- Key idea: Adaptive prompting via **semantic grouping**
  - ✓ **Update existing** prompts for **mild** shifts
  - ✓ Introduce **new** prompts for **abrupt** shifts

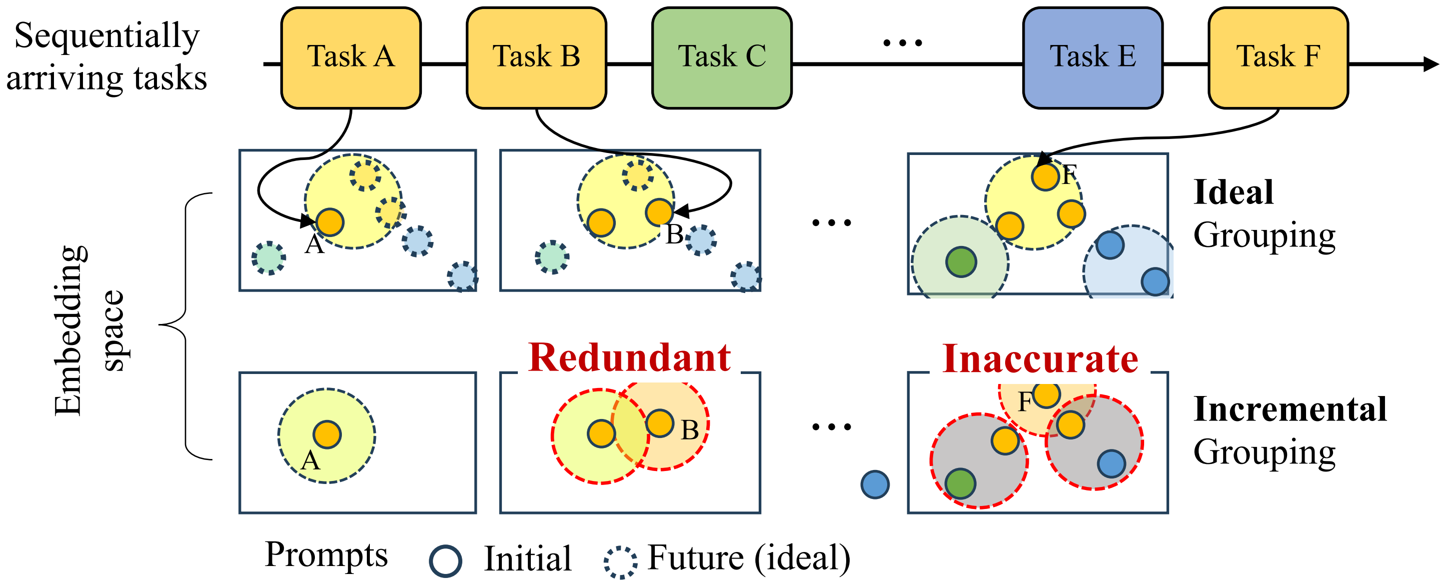


Adaptive prompting via semantic grouping

2 | Methodology: AdaPromptCL via Semantic Grouping

Challenge in Semantic Grouping under CL Environment

- **Inaccuracies** in grouping due to **continual** task emergence
  - ✓ Online clustering quality is affected by **data insertion orders** [5]



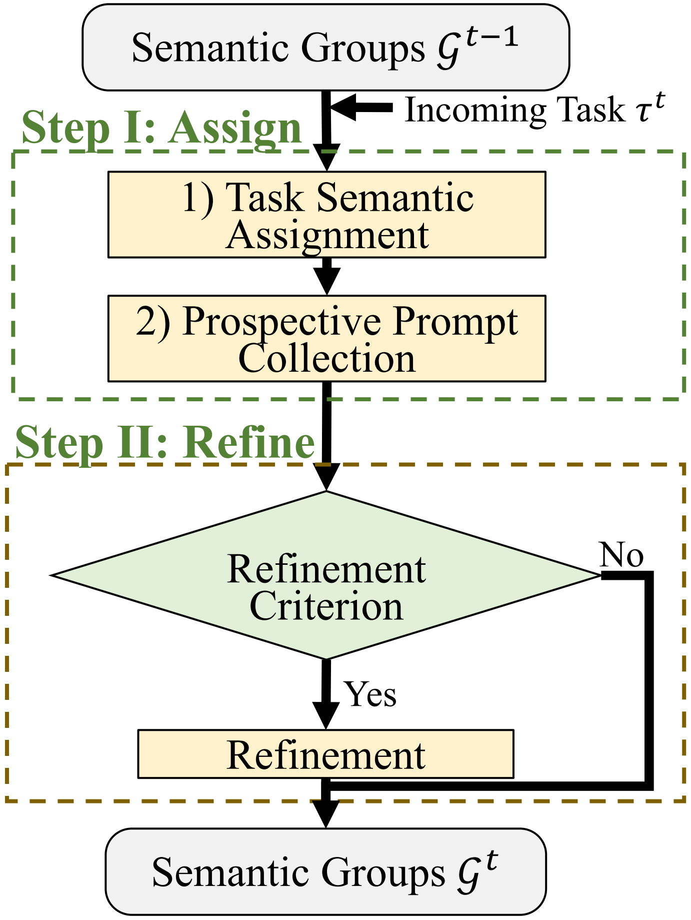
Inaccuracies in semantic grouping in CL

Proposed Method: AdaPromptCL

- AdaPromptCL realizes *accurate* semantic grouping through *assign-and-refine semantic grouping*

✓ Step I: Assignment step
1) Add tasks to groups based on similarity
2) **Store potential groups** for future refinements

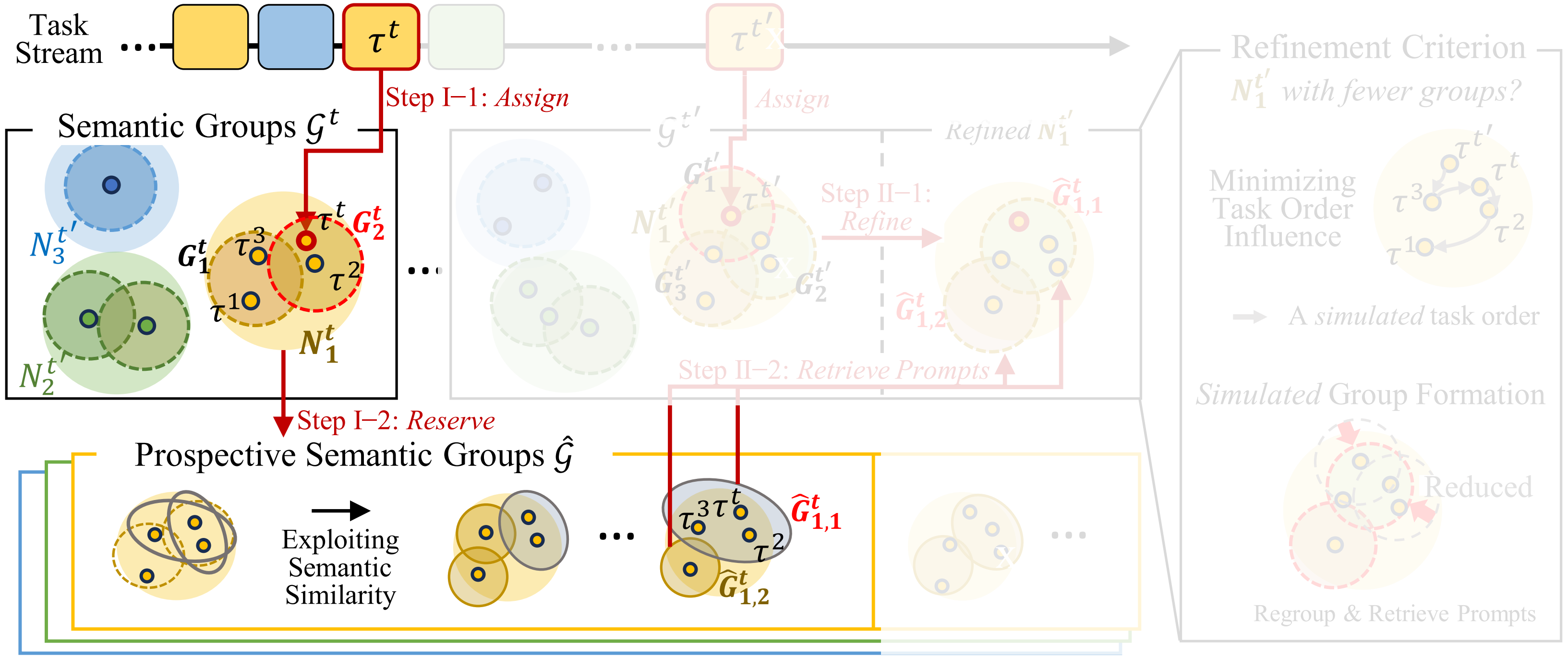
✓ Step II: Refinement step
Refine groups if better grouping is found



Overall flow

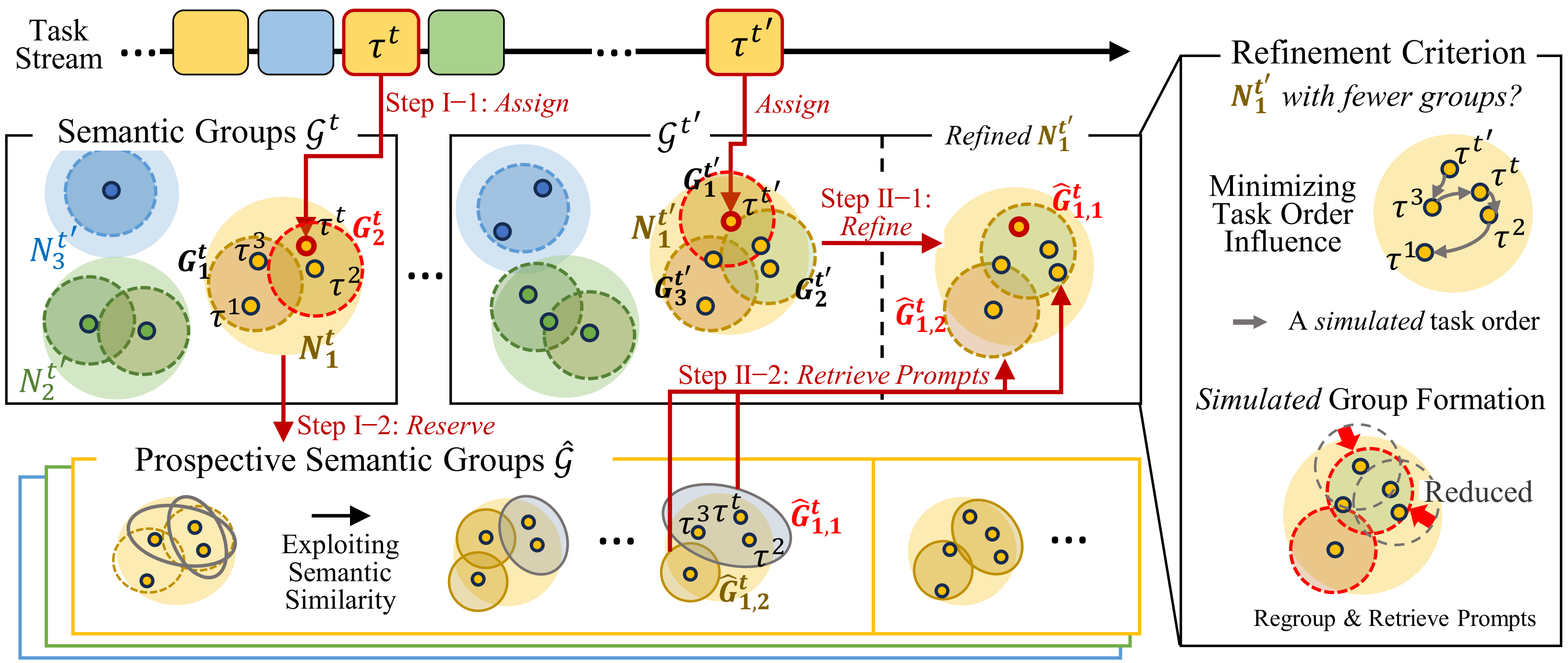
(Detailed view) Step I: Assignment Step

- Step I-1: Add the task τ^t to the semantic group G_2^t
- Step I-2: **Reserve prospective groups** as $\hat{G}_{1,1}^t$ with their prompts



(Detailed view) Step II: Refinement Step

- Step II-1: If a refinement is needed when the task $\tau^{t'}$ is received, three groups $G_1^{t'}$, $G_2^{t'}$, $G_3^{t'}$ are **reduced** to fewer prospective groups $\hat{G}_{1,1}^{t}$, $\hat{G}_{1,2}^{t}$
- Step II-2: **Retrieve** the prompts of the updated semantic groups



Semantic Group-based Training and Inference

- Training stage
  - ✓ **Proactively** train prompts for new semantic groups in **prospective semantic groups** to prepare for forthcoming refinements

- Inference stage
  - ✓ Find the **prompt most similar** to the given test instance from the current semantic groups.

03 | Evaluation

Experiment Setup

- Efficacy is evaluated across a **spectrum of task semantic shifts**
  - ✓ **Uniformly mild** scenario
    - ✓ High semantic similarities across tasks
  - ✓ **Uniformly abrupt** scenario
    - ✓ Low semantic similarities across tasks
  - ✓ **Varying** scenario: Real-world CL scenario
    - ✓ Mixed task similarities, either semantically similar or dissimilar

- Five recent prompting baselines are compared with AdaPromptCL

Overall Performance Comparison

- AdaPromptCL consistently shows high performances in **all** scenarios
- Baselines shows significant **performance gaps** according to the extent of semantic shifts

Shift Scenarios	CL Datasets	Prompt Tuning CL Algorithms: L2P	VPT	LAE	DP	S-Prompts	AdaPromptCL: AdaPromptCL	*Avg. Improv.*	*# Semantics*
Varying	VTAB-Sim25	35.77 (±0.40)	36.38 (±0.30)	35.29 (±0.42)	35.77 (±0.34)	<u>36.61</u> (±0.42)	**38.29** (±0.36)	*6.49%*	16.0
	VTAB-Sim50	36.06 (±0.39)	36.15 (±0.32)	35.23 (±0.61)	35.57 (±0.40)	<u>36.25</u> (±0.45)	**37.92** (±0.47)	*5.78%*	13.4
	VTAB-Sim75	35.66 (±0.44)	36.06 (±0.42)	35.66 (±0.60)	<u>36.35</u> (±0.46)	35.76 (±0.53)	**37.59** (±0.37)	*4.72%*	12.8
	VTAB-Rec2	52.18 (±0.13)	51.84 (±0.12)	52.34 (±0.37)	51.66 (±0.15)	<u>53.28</u> (±0.16)	**56.40** (±0.25)	*7.93%*	5.0
	VTAB-Rec5	52.68 (±0.24)	52.76 (±0.11)	<u>54.06</u> (±0.26)	51.66 (±0.16)	52.82 (±0.36)	**56.86** (±0.33)	*7.72%*	5.6
	VTAB-Rec10	52.20 (±0.81)	51.70 (±0.44)	<u>52.34</u> (±0.35)	50.80 (±0.58)	47.72 (±0.20)	**54.22** (±0.52)	*6.54%*	6.2
Uniformly Mild	ImageNet-R	68.05 (±0.11)	69.31 (±0.09)	**70.11** (±0.12)	67.81 (±0.19)	65.89 (±0.10)	<u>69.45</u> (±0.14)	*1.83%*	1.0
	CIFAR100	84.55 (±0.08)	**85.34** (±0.15)	85.16 (±0.11)	84.79 (±0.14)	83.73 (±0.10)	<u>85.31</u> (±0.11)	*0.71%*	1.0
Uniformly Abrupt	VTAB-19T	28.23 (±0.14)	28.11 (±0.20)	26.71 (±0.29)	28.48 (±0.07)	<u>30.83</u> (±0.08)	**32.39** (±0.32)	*14.00%*	18.0
	VTAB-5T	34.31 (±0.04)	34.03 (±0.05)	35.30 (±0.41)	34.17 (±0.04)	<u>38.27</u> (±0.18)	**38.67** (±0.15)	*10.02%*	5.0

In-depth Analysis on Semantic Refinement (Step II)

- Ablation study on semantic refinement
  - ✓ **“No Refine”** and **“Avg Merge” variants** underperform AdaPromptCL
    - ✓ “No Refine” omits semantic refinement and “Avg Merge” uses simple average of prompts for the refined groups without proactive tuning

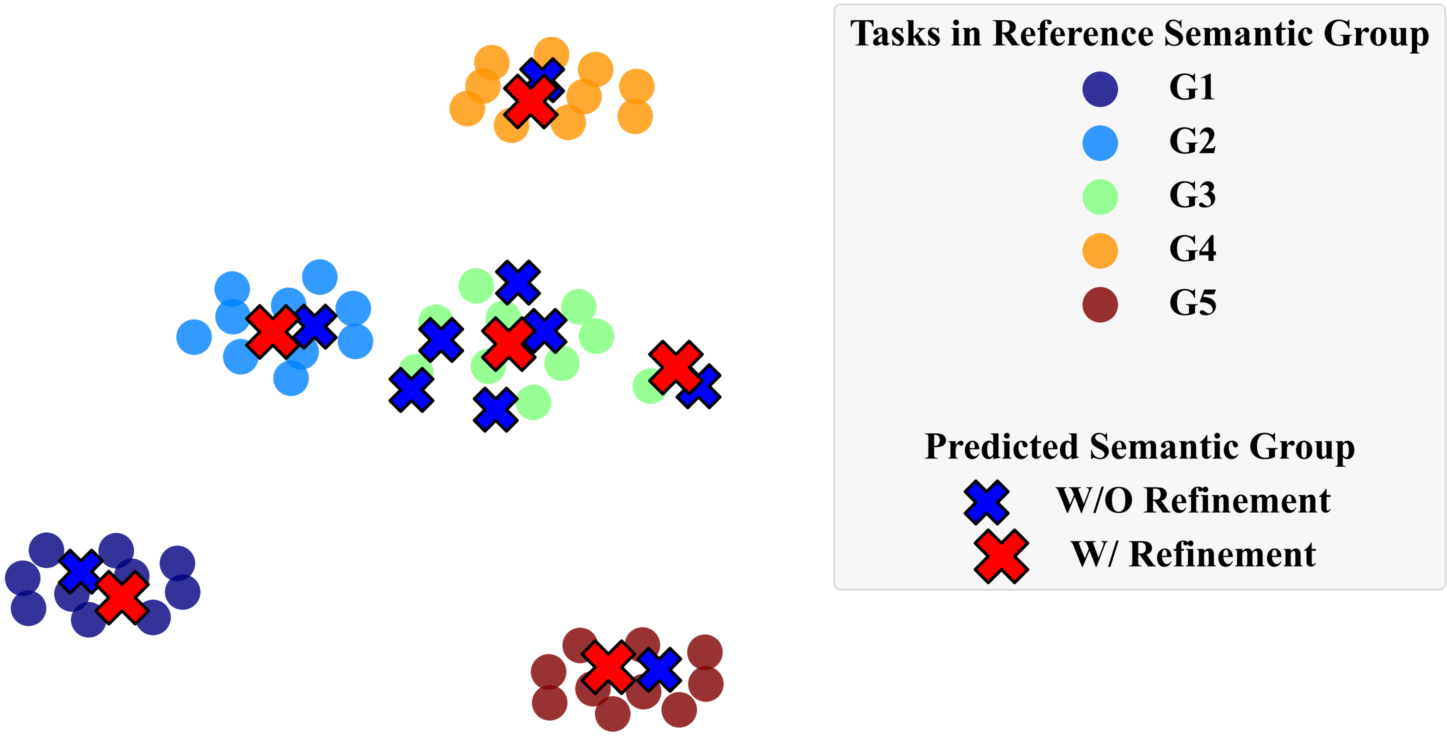
Shift Scenarios	No Refine	Avg Merge	AdaPromptCL
Uniformly Mild (ImageNet-R)	85.31 (±0.11)	85.31 (±0.11)	85.31 (±0.11)
Uniformly Abrupt (VTAB-19T)	31.25 (±0.20)	30.08 (±0.26)	**32.39** (±0.32)
Varying (VTAB-Rec10)	51.80 (±0.62)	52.92 (±1.06)	**54.22** (±0.52)
Average Degradation.	2.8%	3.4%	-

- Correctness of semantic refinement

	No Refine	AdaPromptCL	*Reference*
# Semantic Groups	9.60 (±0.540)	**6.20** (±0.090)	5
Adj. Rand Index	0.89 (±0.031)	**0.97** (±0.005)	1
Norm. Mutual Information	0.90 (±0.027)	**0.98** (±0.004)	1

Visualization of Semantic Refinement

- AdaPromptCL refines overgeneration of semantic groups by correctly merging similar tasks



04 | Conclusion

Conclusion

- AdaPromptCL realizes **adaptive prompting** to effectively handle diverse and unpredictable semantic shifts
- **Assign-and-refine grouping** ensures precise prompting tailored to evolving task semantics
- AdaPromptCL demonstrates its superior efficacy, highlighting its adapta bility in real-world CL scenarios

Reference

[1] Jia, M., Tang, L., Chen, B.-C., Cardie, C., Belongie, S., Hariharan, B., and Lim, S.-N. Visual Prompt Tuning. In ECCV, pp. 709–727, 2022.

[2] Wang, Z., Zhang, Z., Lee, C.-Y., Zhang, H., Sun, R., Ren, X., Su, G., Perot, V., Dy, J., and Pfister, T. Learning to Prompt for Continual Learning. In CVPR, pp. 139–149, 2022.

[3] Gao, Q., Zhao, C., Sun, Y., Xi, T., Zhang, G., Ghanem, B., and Zhang, J. A Unified Continual Learning Framewor k with General Parameter-Efficient Tuning. In ICCV, 2023.

[4] Wang, Y., Huang, Z., and Hong, X. S-prompts Learning with Pre-trained Transformers: An Occam's Razor for Domain Incremental Learning. In NeurIPS, volume 35, pp. 5682–5695, 2022.

[5] Zhang, T., Ramakrishnan, R., and Livny, M. Birch: An Efficient Data Clustering Method for Very Large Databases. SIGMOD, 25(2):103–114, 1996.

THANK YOU!