Contrastive Modules with Temporal Attention for Multi-Task Reinforcement Learning

Siming Lan, Rui Zhang, Qi Yi, Jiaming Guo, Shaohui Peng, Yunkai Gao,
Fan Wu, Ruizhi Chen, Zidong Du, Xing Hu, Xishan Zhang, Ling Li, Yunji Chen

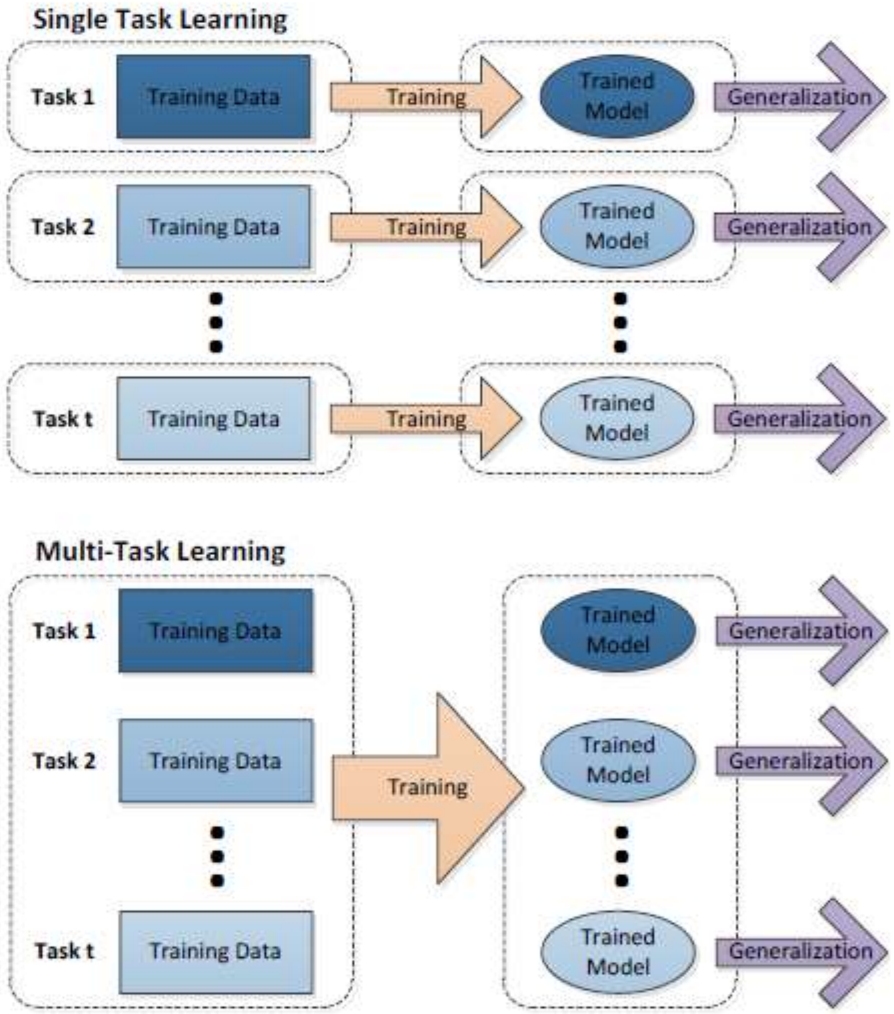
Background

Multi-task RL vs Single-task RL:

- better sample efficiency

  (share knowledge across tasks)
- better performance in theory

  (use additional auxiliary task)
- fewer model parameters



Negative Transfer

- In theory, multi-task RL can achieve better performance.

Negative Transfer

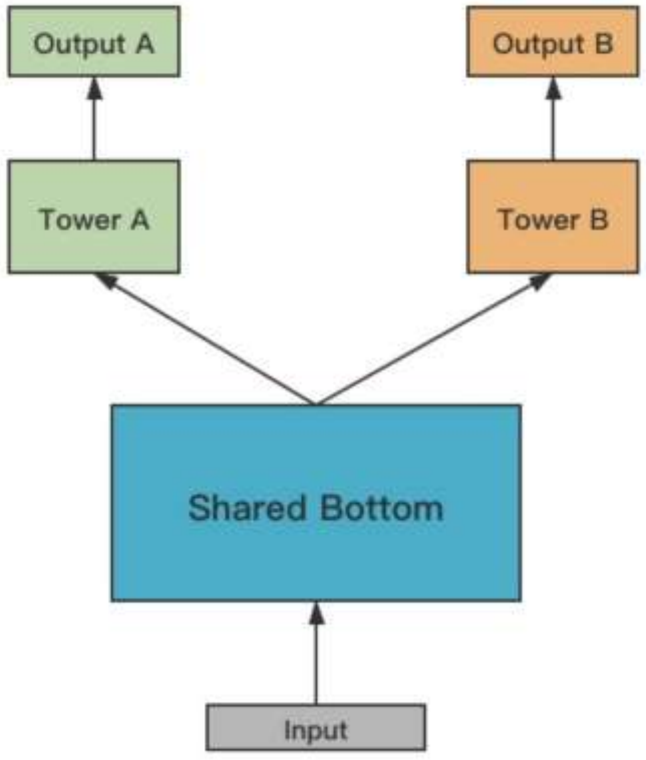
- In theory, multi-task RL can achieve better performance.
- But in practice, its performance tends to be worse than single task RL due to the **negative transfer**:

two tasks may have conflicts and hurt each other.

Negative Transfer

One of the essential reason for negative transfer :
using the **same** model to learn different tasks.

To mitigate negative transfer, we should use **models that are not exactly the same** to learn multiple tasks.

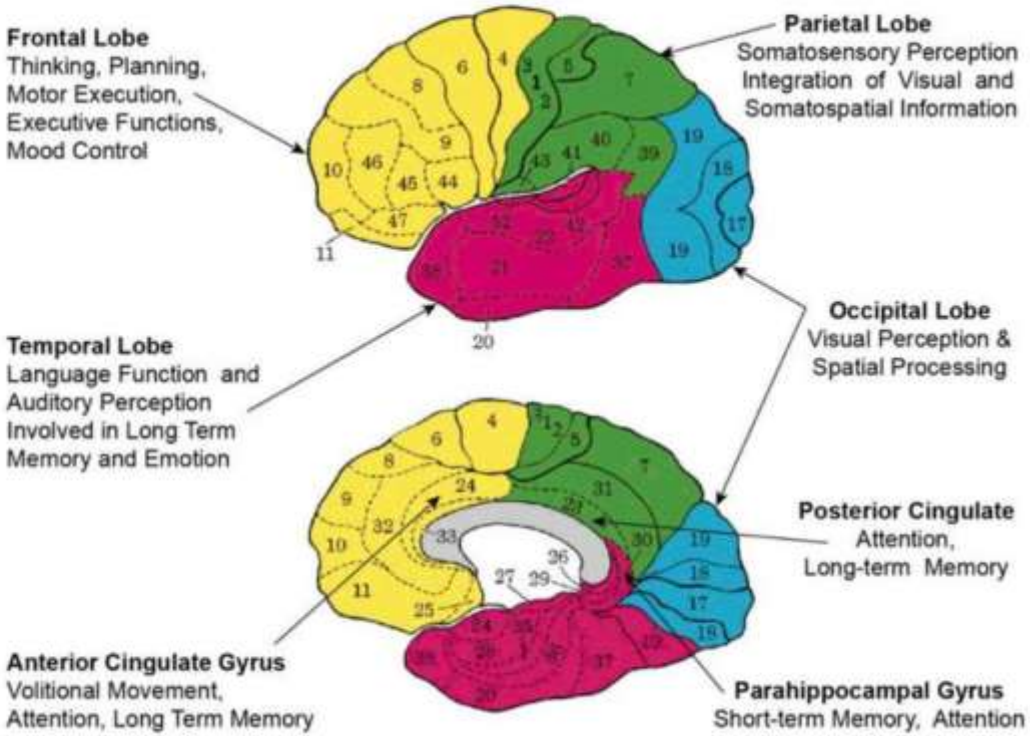


Hard parameter

Modular principle

Humans don't need to learn new task from scratch:

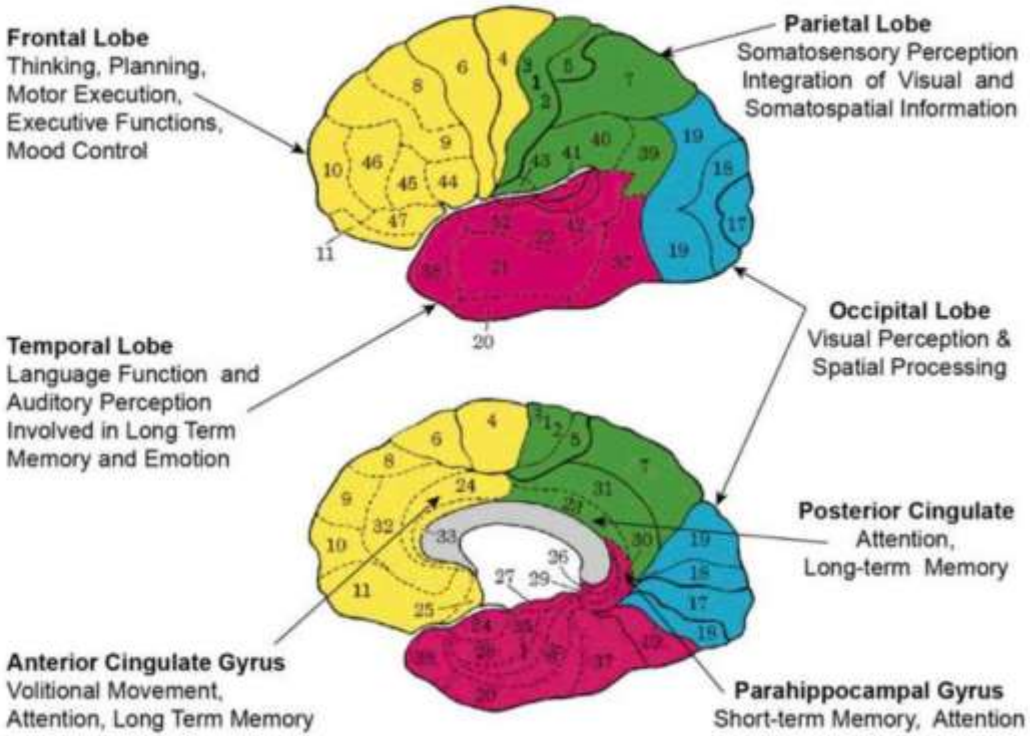
- reuse existing knowledge/mechanisms
- mechanisms is modular and generic



Modular principle

Humans don't need to learn new task from scratch:

- reuse existing knowledge/mechanisms
- mechanisms is modular and generic



Modular principle:
different modules + appropriate combination

Motivation

Performance: existing multi-task RL $<$ single-task RL.

Possible reason:

Motivation

Performance: existing multi-task RL < single-task RL.

Possible reason:

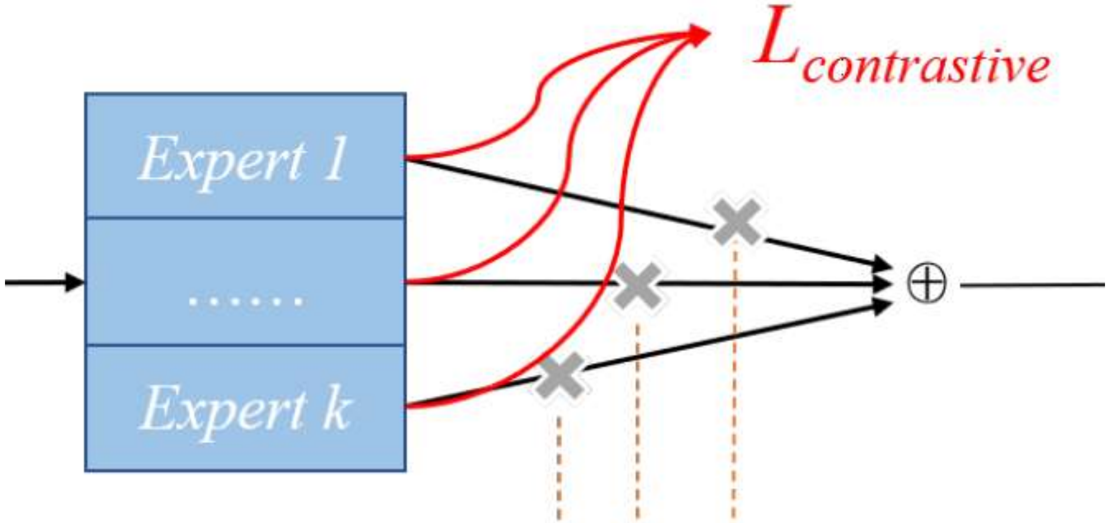
Modular principle		existing multi-task RL method
Different modules	⟷	**Only use multiple modules**
Appropriate combination	⟷	**Only combine modules at task level**

Contrastive Modules

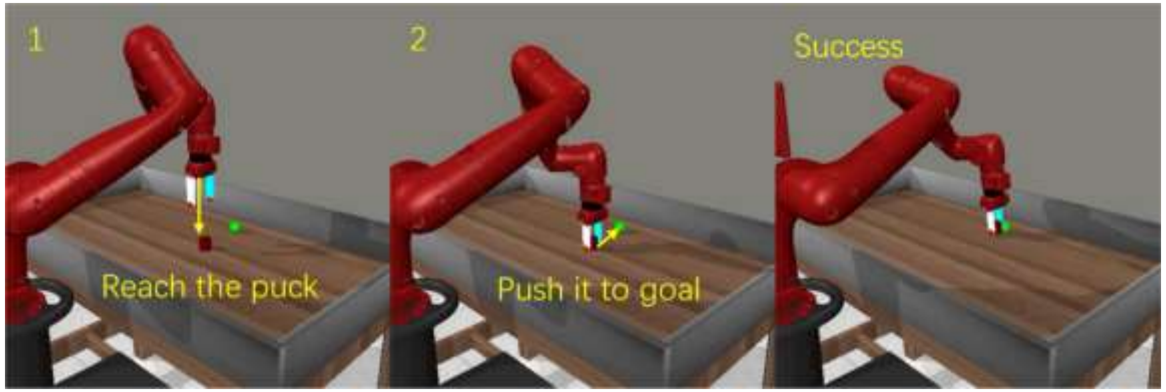
- **Different modules:**

Using contrastive learning to constrain multiple modules to be different from each other.

$$L_{con} = \sum_{i=1}^{k} -log \frac{exp(q_i \cdot k_i^+/\tau)}{exp(q_i \cdot k_i^+/\tau) + \sum_{k_i^-} exp(q_i \cdot k_i^-/\tau)},$$



Temporal Attention



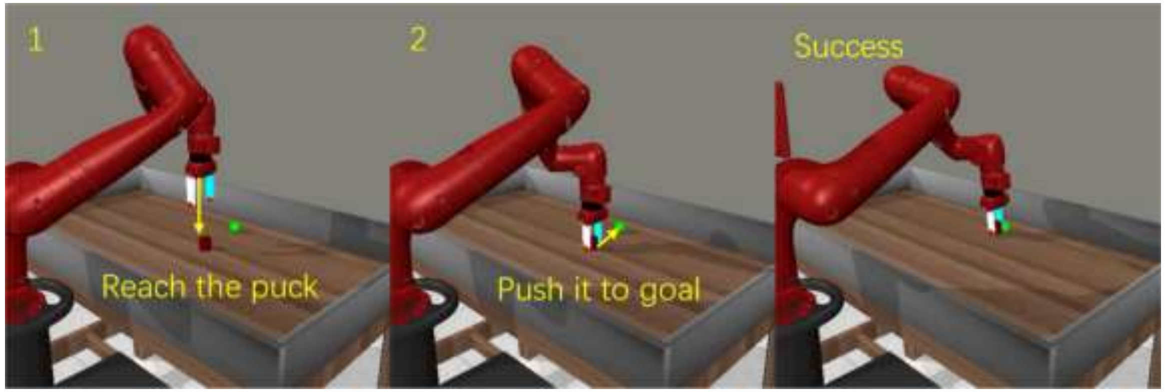
- **Appropriate combination:**

  RL is a sequential decision process.

negative transfer → between tasks

negative transfer → within tasks

Temporal Attention



- **Appropriate combination:**

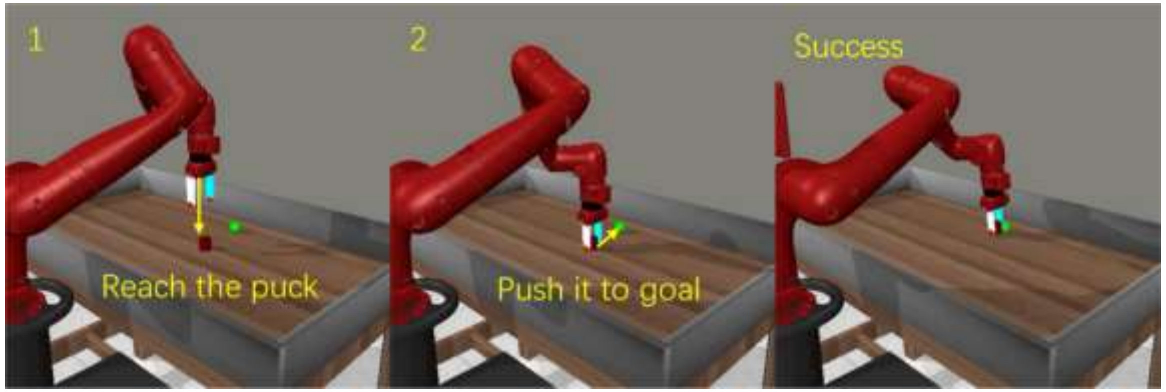
RL is a sequential decision process.

between tasks ← task-level combination

negative transfer

within tasks

Temporal Attention



- **Appropriate combination:**

  RL is a sequential decision process.

negative transfer

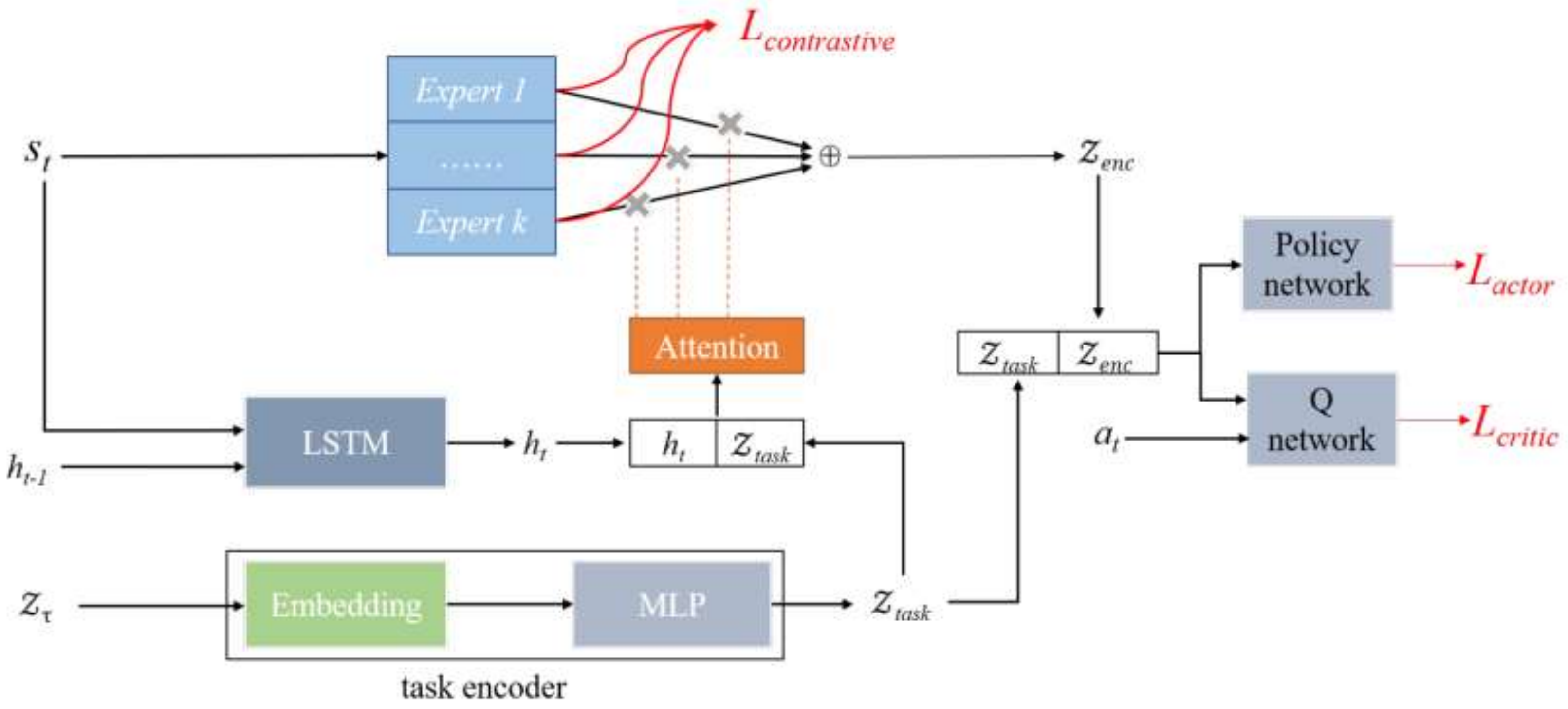
between tasks ← task-level combination

within tasks ← step-level combination

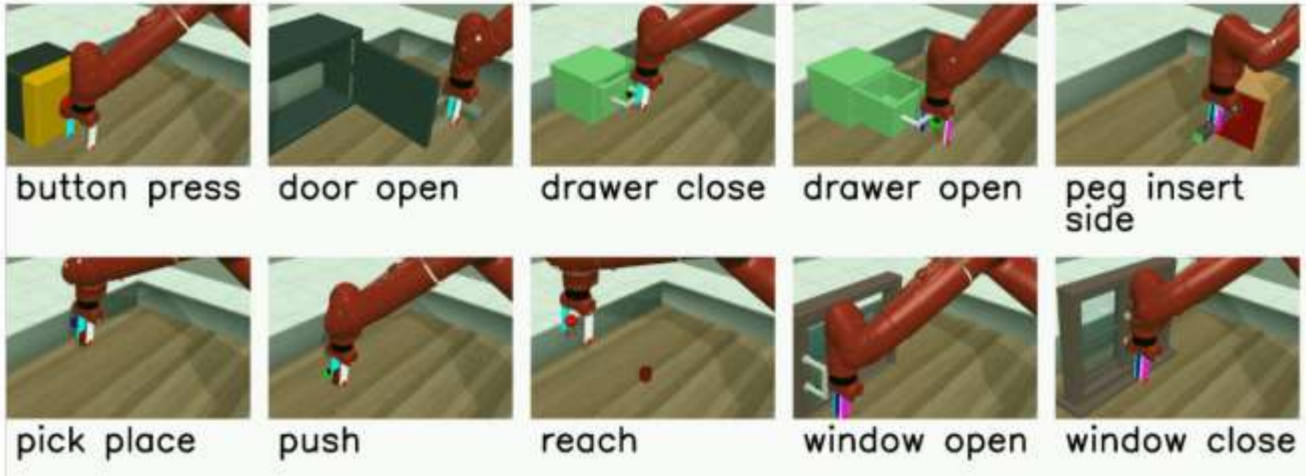
finer granularity

Temporal Attention

By using temporal attention, we combine shared modules at a finer granularity than the task level.



Experiments



agent	MT10-Fixed success rate max smoothed	MT10-Fixed success rate max	MT10-Mixed success rate max smoothed	MT10-Mixed success rate max	MT50-Fixed success rate max smoothed	MT50-Fixed success rate max	MT50-Mixed success rate max smoothed	MT50-Mixed success rate max
MT-SAC	62.25%	68.75%	53.22%	62.50%	50.37%	52.50%	28.78%	31.50%
MT-SAC+TE	64.76%	70%	61.12%	68.75%	52.45%	54.75%	37.59%	40%
MTMH-SAC	65.21%	70%	62.06%	67.50%	47.67%	48.75%	39.65%	42.75%
SoftModu	51%	55%	51.34%	58.75%	26.23%	28.75%	21.50%	23.50%
CARE	68.03%	75%	61.35%	67.50%	55.47%	57.50%	45.00%	48.50%
CMTA(ours)	**78.95%**	**83.75%**	**82.07%**	**87.5%**	**68.90%**	**71.00%**	**71.69%**	**74.5%**
Single-SAC(upper bound)	64.33%	68.75%	71.11%	76.25%	/	/	/	/

Ablation-Contrastive Modules

agent	MT10-Mixed success rate max smoothed	MT10-Mixed success rate max	MT50-Mixed success rate max smoothed	MT50-Mixed success rate max
CARE	61.35%	67.50%	45.00%	48.50%
CARE + CL	65.24%	71.25%	47.61%	49.75%
CMTA w/o CL	79.46%	85%	62.66%	65%
CMTA(ours)	**82.07%**	**87.5%**	**71.69%**	**74.5%**

(a) CMTA w/o CL (b) CMTA

Figure 5: t-SNE visualization of multiple modules' encodings on MT10-Fixed environment.

Ablation-Temporal Attention

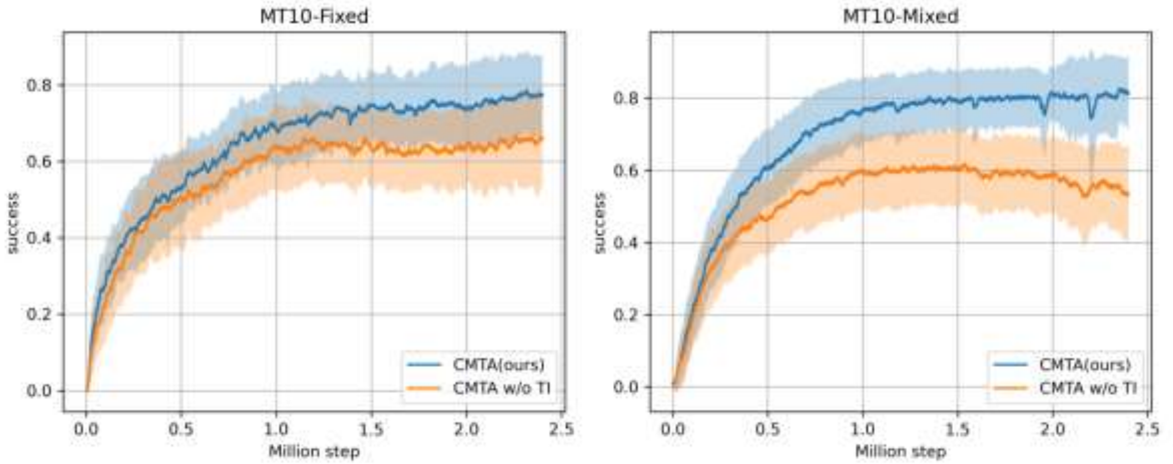


Figure 4: Effectiveness of temporal information(TI) on MT10-Fixed and MT10-Mixed environment, each curve has been averaged over 8 seeds.

Thanks!