Visual Prompt Tuning in Null Space for Continual Learning

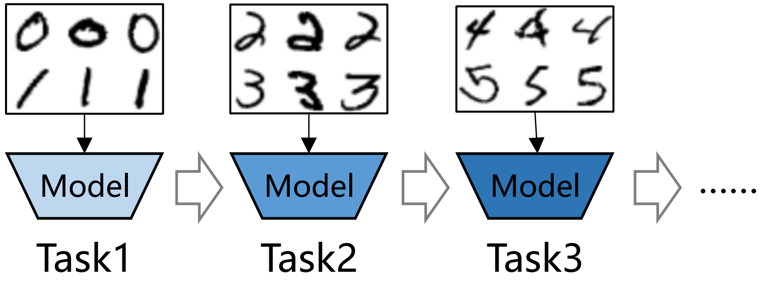
Yue Lu[1], Shizhou Zhang[1*], De Cheng[2*], Yinghui Xing[1],
Nannan Wang[2], Peng Wang[1], Yanning Zhang[1]

[1] School of Computer Science, Northwestern Polytechnical University, China
[2] School of Telecommunications Engineering, Xidian University, China

Continual Learning

Problem definition: The model is trained on learning tasks that come one after another. The data of the learned tasks are no longer visible. In the inference stage, the model can have good discrimination ability for all the data of the learned tasks.

Class-incremental continual learning: Each task contains several categories that have not been learned before, and the model is trained task by task. In the inference stage, the model has good classification ability for all known category data when the task to which the input image belongs is unknown.



Schematic Diagram of Class Incremental Continual Learning

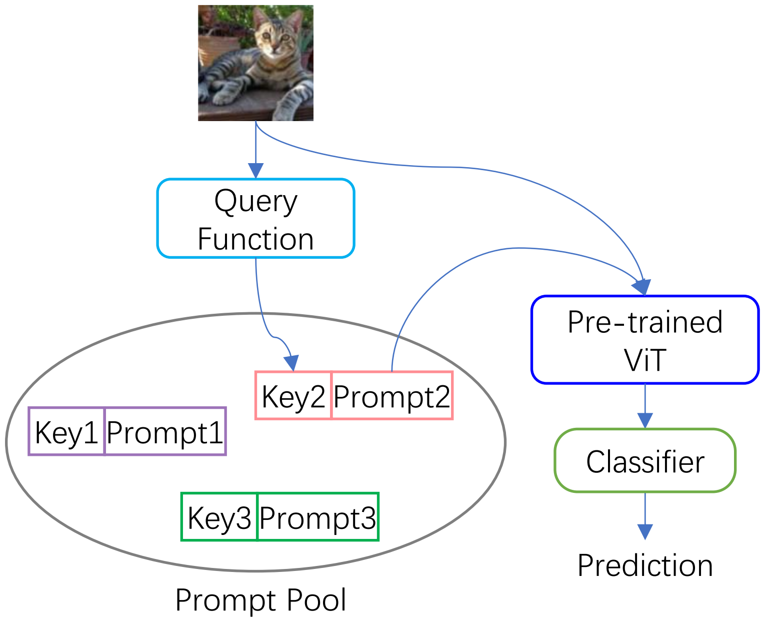
NEURAL INFORMATION PROCESSING SYSTEMS

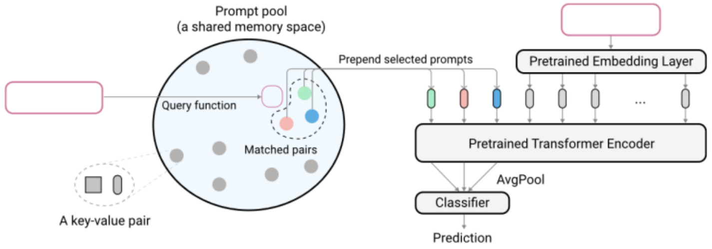
西北工業大學 NORTHWESTERN POLYTECHNICAL UNIVERSITY

西安電子科技大學

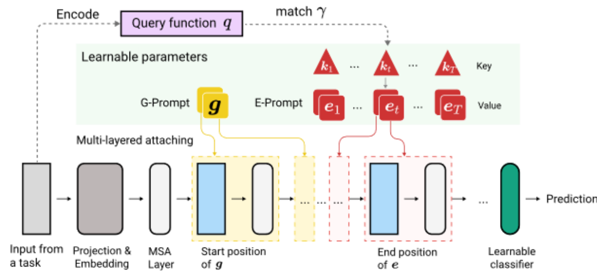
Continual Learning Based on Visual Prompt Tuning

Encoding task knowledge into a prompt pool, selecting and updating task-relevant prompts within the pool based on the training data.

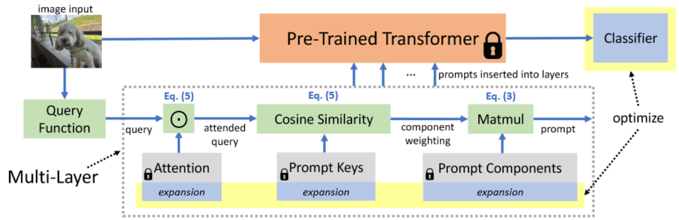




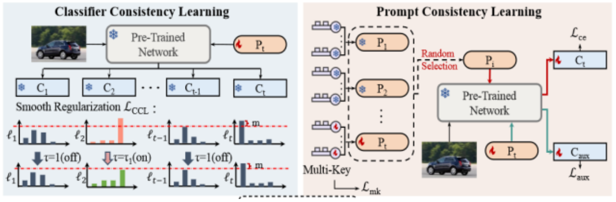
L2P(CVPR22)



DualPrompt(ECCV22)



CODA-Prompt(CVPR23)

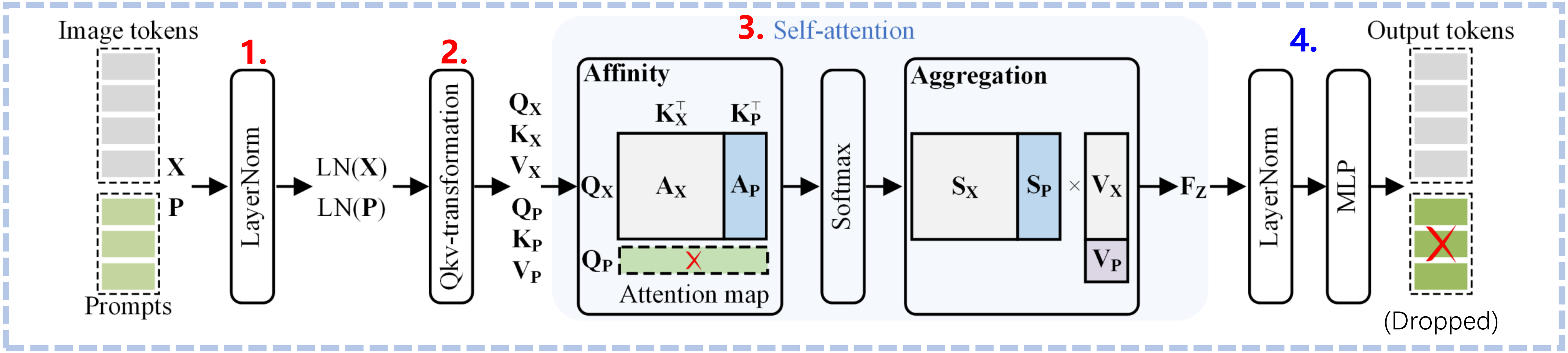


CPrompt(CVPR24)

Training interference: The features of the data from previous tasks change as related prompts are updated, leading to catastrophic forgetting.

Our Method

Analysis of the ViT Layer Forward Process $f_{\text{ViT}}(\mathbf{X}|\mathbf{P})$



Input: $\mathbf{X} \in \mathbb{R}^{N\times D}$ **,** $\mathbf{P} \in \mathbb{R}^{M\times D}$**, where** N, M **represent the number of image tokens and prompts, respectively, and** D **denotes the token dimension.**

$\mathbf{Z} = [\mathbf{X}; \mathbf{P}] \in \mathbb{R}^{(N+M)\times D}$ **, representing the concatenated tokens along channel dimension.**

1. LayerNorm $\mathrm{LN}(\cdot)$:

$$\mathrm{LN}(\mathbf{Z}) = \frac{\mathbf{Z} - \boldsymbol{\mu}_\mathbf{Z}}{\boldsymbol{\sigma}_\mathbf{Z}} \odot \boldsymbol{\alpha} + \boldsymbol{\beta}$$

2. QKV- Transformation:

$$\mathbf{Q_Z} = \mathrm{LN}(\mathbf{Z})\mathbf{W}_q + \boldsymbol{b}_q$$
$$\mathbf{K_Z} = \mathrm{LN}(\mathbf{Z})\mathbf{W}_k + \boldsymbol{b}_k$$
$$\mathbf{V_Z} = \mathrm{LN}(\mathbf{Z})\mathbf{W}_v + \boldsymbol{b}_v$$

3. Self-attention f_{SA}:

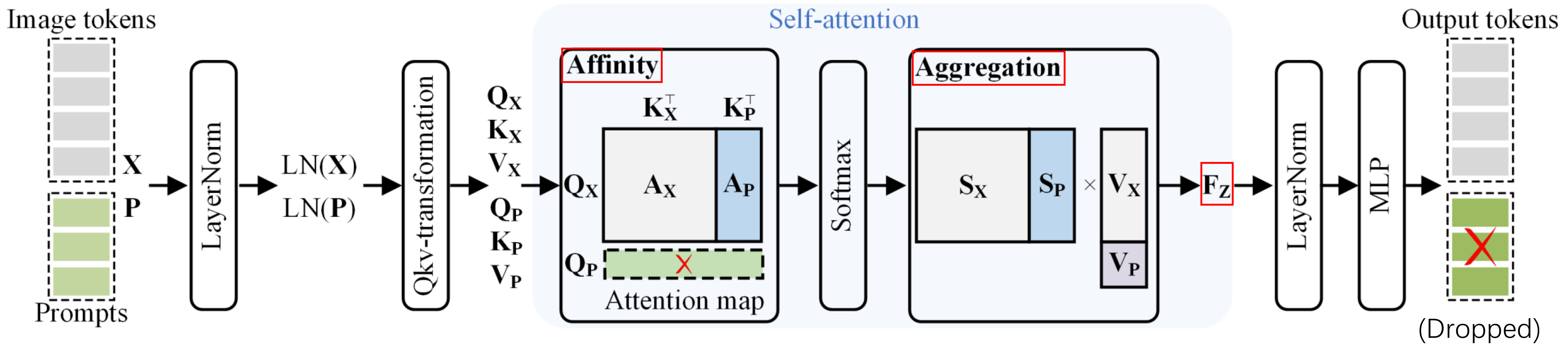
$$\mathbf{F_Z} = \mathrm{softmax}\left(\frac{\mathbf{Q_X}\mathbf{K_Z^\top}}{\sqrt{D}}\right)\mathbf{V_Z}$$

4. Feed-Forward Network:

LayerNorm and MLP

Our Method

Analyzing $f_{\text{ViT}}(\mathbf{X}_t|\mathbf{P}_t) = f_{\text{ViT}}(\mathbf{X}_t|\mathbf{P}_{t+1})$



To ensure that the output tokens of $\mathbf{X}_t$ remain consistent between tasks t and $t+1$, i.e.,

$$f_{\text{ViT}}(\mathbf{X}_t|\mathbf{P}_t) = f_{\text{ViT}}(\mathbf{X}_t|\mathbf{P}_{t+1})$$

We define $\mathbf{Z}_t = [\mathbf{X}_t; \mathbf{P}_t], \mathbf{Z}_{t+1} = [\mathbf{X}_t; \mathbf{P}_{t+1}]$, This requires satisfying the condition:

$$\mathbf{F}_{\mathbf{Z}_t} = \mathbf{F}_{\mathbf{Z}_{t+1}}$$

Simplification of $\mathbf{F}_{\mathbf{Z}_t} = \mathbf{F}_{\mathbf{Z}_{t+1}}$:

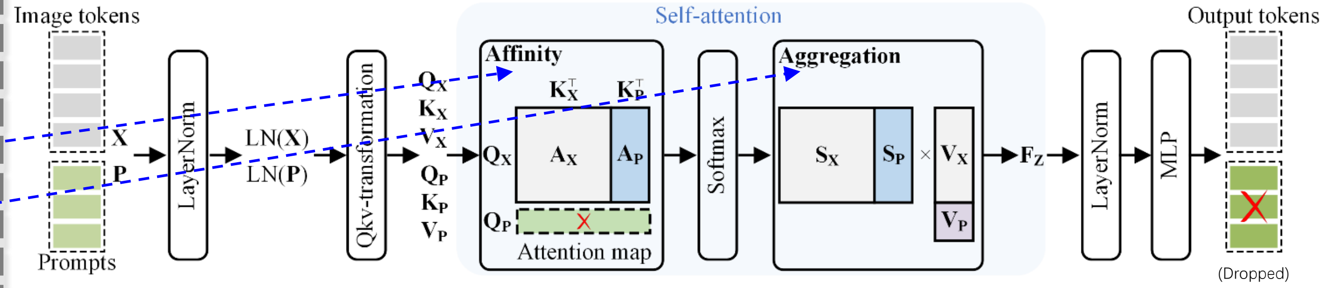
$$\begin{cases} \mathbf{A}_\mathbf{Z} = f_{\text{aff}}(\mathbf{Q}_\mathbf{X}, \mathbf{K}_\mathbf{Z}) = \dfrac{\mathbf{Q}_\mathbf{X}\mathbf{K}_\mathbf{Z}^\top}{\sqrt{D}} = \dfrac{\mathbf{Q}_\mathbf{X}[\mathbf{K}_\mathbf{X}^\top \ \ \mathbf{K}_\mathbf{P}^\top]}{\sqrt{D}} \\ \mathbf{S}_\mathbf{Z} = \text{softmax}(\mathbf{A}_\mathbf{Z}) = \text{softmax}(\mathbf{A}_\mathbf{X} \ \ \mathbf{A}_\mathbf{P}) = [\mathbf{S}_\mathbf{X} \ \ \mathbf{S}_\mathbf{P}] \\ \mathbf{F}_\mathbf{Z} = f_{\text{agg}}(\mathbf{S}_\mathbf{Z}, \mathbf{V}_\mathbf{Z}) = \mathbf{S}_\mathbf{Z}\mathbf{V}_\mathbf{Z} = [\mathbf{S}_\mathbf{X} \ \ \mathbf{S}_\mathbf{P}] \begin{bmatrix} \mathbf{V}_\mathbf{X} \\ \mathbf{V}_\mathbf{P} \end{bmatrix} \end{cases}$$

Two Sufficient Conditions

Two sufficient conditions to satisfy

$$\mathbf{F}_{\mathbf{Z}_t} = \mathbf{F}_{\mathbf{Z}_{t+1}}$$

$$\begin{cases} f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_t}\right) = f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_{t+1}}\right) & ① \\ f_{\text{agg}}\left(\mathbf{S}_{\mathbf{Z}_t}, \mathbf{V}_{\mathbf{Z}_t}\right) = f_{\text{agg}}\left(\mathbf{S}_{\mathbf{Z}_{t+1}}, \mathbf{V}_{\mathbf{Z}_{t+1}}\right) & ② \end{cases}$$



① $f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_t}\right) = f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_{t+1}}\right)$, **each side of the equation is:**

$$\begin{cases} f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_t}\right) = \mathbf{Q}_{\mathbf{X}_t}\left[\mathbf{K}_{\mathbf{X}_t}^\top \;\; \mathbf{K}_{\mathbf{P}_t}^\top\right] = \left[\mathbf{Q}_{\mathbf{X}_t}\mathbf{K}_{\mathbf{X}_t}^\top \;\; \mathbf{Q}_{\mathbf{X}_t}\left[\mathrm{LN}(\mathbf{P}_t)\mathbf{W}_k + \boldsymbol{b}_k\right]^\top\right] \\ f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_{t+1}}\right) = \mathbf{Q}_{\mathbf{X}_t}\left[\mathbf{K}_{\mathbf{X}_t}^\top \;\; \mathbf{K}_{\mathbf{P}_{t+1}}^\top\right] = \left[\mathbf{Q}_{\mathbf{X}_t}\mathbf{K}_{\mathbf{X}_t}^\top \;\; \mathbf{Q}_{\mathbf{X}_t}\left[\mathrm{LN}(\mathbf{P}_{t+1})\mathbf{W}_k + \boldsymbol{b}_k\right]^\top\right] \end{cases}$$

$$\mathbf{A}_{\mathbf{Z}} = f_{\text{aff}}(\mathbf{Q}_{\mathbf{X}}, \mathbf{K}_{\mathbf{Z}}) = \frac{\mathbf{Q}_{\mathbf{X}}\left[\mathbf{K}_{\mathbf{X}}^\top \;\; \mathbf{K}_{\mathbf{P}}^\top\right]}{\sqrt{D}}$$

From ① $f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_t}\right) = f_{\text{aff}}\left(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_{t+1}}\right)$, we derive $\mathbf{A}_{\mathbf{Z}_t} = \mathbf{A}_{\mathbf{Z}_{t+1}}$ and $\mathbf{S}_{\mathbf{Z}_t} = \mathbf{S}_{\mathbf{Z}_{t+1}}$

② $f_{\text{agg}}\left(\mathbf{S}_{\mathbf{Z}_t}, \mathbf{V}_{\mathbf{Z}_t}\right) = f_{\text{agg}}\left(\mathbf{S}_{\mathbf{Z}_{t+1}}, \mathbf{V}_{\mathbf{Z}_{t+1}}\right)$, **each side of the equation is:**

$$\begin{cases} f_{\text{agg}}\left(\mathbf{S}_{\mathbf{Z}_t}, \mathbf{V}_{\mathbf{Z}_t}\right) = \mathbf{S}_{\mathbf{X}_t}\mathbf{V}_{\mathbf{X}_t} + \mathbf{S}_{\mathbf{P}_t}\mathbf{V}_{\mathbf{P}_t} = \mathbf{S}_{\mathbf{X}_t}\mathbf{V}_{\mathbf{X}_t} + \mathbf{S}_{\mathbf{P}_t}\left[\mathrm{LN}(\mathbf{P}_t)\mathbf{W}_v + \boldsymbol{b}_v\right] \\ f_{\text{agg}}\left(\mathbf{S}_{\mathbf{Z}_{t+1}}, \mathbf{V}_{\mathbf{Z}_{t+1}}\right) = f_{\text{agg}}\left(\mathbf{S}_{\mathbf{Z}_t}, \mathbf{V}_{\mathbf{Z}_{t+1}}\right) = \mathbf{S}_{\mathbf{X}_t}\mathbf{V}_{\mathbf{X}_t} + \mathbf{S}_{\mathbf{P}_t}\left[\mathrm{LN}(\mathbf{P}_{t+1})\mathbf{W}_v + \boldsymbol{b}_v\right] \end{cases}$$

$$\mathbf{S}_{\mathbf{Z}} = \mathrm{softmax}(\mathbf{A}_{\mathbf{Z}}) = \left[\mathbf{S}_{\mathbf{X}} \;\; \mathbf{S}_{\mathbf{P}}\right]$$

$$\mathbf{F}_{\mathbf{Z}} = f_{\text{agg}}(\mathbf{S}_{\mathbf{Z}}, \mathbf{V}_{\mathbf{Z}}) = \left[\mathbf{S}_{\mathbf{X}} \;\; \mathbf{S}_{\mathbf{P}}\right]\begin{bmatrix}\mathbf{V}_{\mathbf{X}} \\ \mathbf{V}_{\mathbf{P}}\end{bmatrix}$$

Simplifying conditions ①② , we obtain:

$$\begin{cases} \mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top \mathrm{LN}(\mathbf{P}_t)^\top = \mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top \mathrm{LN}(\mathbf{P}_{t+1})^\top = \mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top \mathrm{LN}(\mathbf{P}_t + \Delta\mathbf{P})^\top \\ \mathbf{S}_{\mathbf{P}_t}\mathrm{LN}(\mathbf{P}_t)\mathbf{W}_v = \mathbf{S}_{\mathbf{P}_t}\mathrm{LN}(\mathbf{P}_{t+1})\mathbf{W}_v = \mathbf{S}_{\mathbf{P}_t}\mathrm{LN}(\mathbf{P}_t + \Delta\mathbf{P})\mathbf{W}_v \end{cases}$$

Simplification of the LayerNorm Term

$$\begin{cases} Q_{X_t}W_k^{\top}\mathrm{LN}(P_t)^{\top} = Q_{X_t}W_k^{\top}\mathrm{LN}(P_t+\Delta P)^{\top} & ③ \\ S_{P_t}\mathrm{LN}(P_t)W_v = S_{P_t}\mathrm{LN}(P_t+\Delta P)W_v & ④ \end{cases}$$

$$\mathrm{LN}(P) = \frac{P-\mu_P}{\sigma_P}\odot\alpha+\beta \qquad \mathrm{LN}(P) = \frac{P_t+\Delta P-\mu_{P_t+\Delta P}}{\sigma_{P_t+\Delta P}}\odot\alpha+\beta$$

For the term $\mathrm{LN}(P_t+\Delta P)$, it cannot be expressed directly in terms of $\mathrm{LN}(P_t)$ and ΔP.

Assumption of Distribution Invariance:

Assuming the distribution of prompts remains unchanged during training: $\begin{cases} \mu_{P_t+\Delta P} = \mu_{P_t} \\ \sigma_{P_t+\Delta P} = \sigma_{P_t} \end{cases}$

Relationship between $\mathrm{LN}(P_t+\Delta P)$ and $\mathrm{LN}(P_t)$ using ΔP :

$$\mathrm{LN}(P_t+\Delta P) = \frac{P_t+\Delta P-\mu_{P_t+\Delta P}}{\sigma_{P_t+\Delta P}}\odot\alpha+\beta = \frac{P_t+\Delta P-\mu_{P_t}}{\sigma_{P_t}}\odot\alpha+\beta = \left(\frac{P_t-\mu_{P_t}}{\sigma_{P_t}}\odot\alpha+\beta\right)+\frac{\Delta P}{\sigma_{P_t}}\odot\alpha = \mathrm{LN}(P_t)+\frac{\Delta P}{\sigma_{P_t}}\odot\alpha$$

From $\mathrm{LN}(P_t+\Delta P) = \mathrm{LN}(P_t)+\frac{\Delta P}{\sigma_{P_t}}\odot\alpha$, simplify ③④:

$$\begin{cases} Q_{X_t}W_k^{\top}\mathrm{LN}(P_t)^{\top} = Q_{X_t}W_k^{\top}\mathrm{LN}(P_t)^{\top} + \dfrac{Q_{X_t}W_k^{\top}\Delta P^{\top}}{\sigma_{P_t}}\odot\alpha^{\top} & ⑤ \\ S_{P_t}\mathrm{LN}(P_t)W_v = S_{P_t}\mathrm{LN}(P_t)W_v + \dfrac{S_{P_t}\Delta P W_v}{\sigma_{P_t}}\odot\alpha & ⑥ \end{cases}$$

From⑤⑥: $\begin{cases} (Q_{X_t}W_k^{\top})\Delta P^{\top} = 0 \\ S_{P_t}\Delta P W_v = 0 \end{cases}$

It is sufficient to satisfy conditions ⑦⑧:

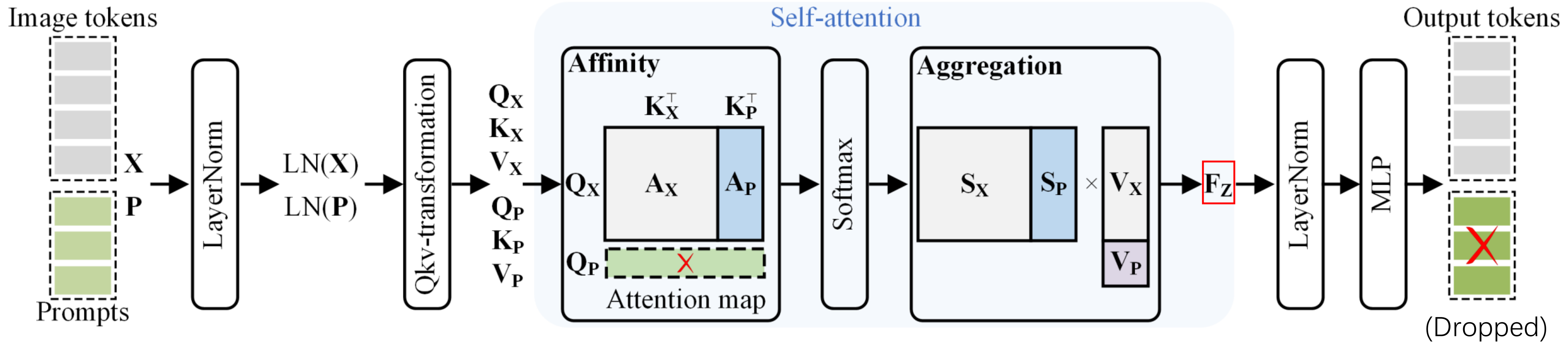
$$\boxed{\begin{cases} (Q_{X_t}W_k^{\top})\Delta P^{\top} = 0 & ⑦ \\ S_{P_t}\Delta P = 0 & ⑧ \end{cases}}$$

Our Method

NEURAL INFORMATION PROCESSING SYSTEMS

西北工业大学 NORTHWESTERN POLYTECHNICAL UNIVERSITY 西安电子科技大学

Conclusion



To achieve the following objective:

$$f_{\mathrm{ViT}}(\mathbf{X}_t|\mathbf{P}_t) = f_{\mathrm{ViT}}(\mathbf{X}_t|\mathbf{P}_{t+1})$$

Specifically, this requires:

$$\mathbf{F}_{\mathbf{Z}_t} = \mathbf{F}_{\mathbf{Z}_{t+1}}$$

Converting into two sufficient conditions:

$$\begin{cases} f_{\mathrm{aff}}(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_t}) = f_{\mathrm{aff}}(\mathbf{Q}_{\mathbf{X}_t}, \mathbf{K}_{\mathbf{Z}_{t+1}}) \\ f_{\mathrm{agg}}(\mathbf{S}_{\mathbf{Z}_t}, \mathbf{V}_{\mathbf{Z}_t}) = f_{\mathrm{agg}}(\mathbf{S}_{\mathbf{Z}_{t+1}}, \mathbf{V}_{\mathbf{Z}_{t+1}}) \end{cases}$$

Introducing a constraint on the prompt distribution's variation

$$\begin{cases} \boldsymbol{\mu}_{\mathbf{P}_t+\Delta\mathbf{P}} = \boldsymbol{\mu}_{\mathbf{P}_t} \\ \boldsymbol{\sigma}_{\mathbf{P}_t+\Delta\mathbf{P}} = \boldsymbol{\sigma}_{\mathbf{P}_t} \end{cases}$$

This ultimately leads to the following two conditions called consistency conditions.

$$\begin{cases} (\mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top)\Delta\mathbf{P}^\top = \mathbf{0} & ⑦ \\ \mathbf{S}_{\mathbf{P}_t}\Delta\mathbf{P} = \mathbf{0} & ⑧ \end{cases}$$

Optimization of Consistency Conditions

Two-step optimization scheme:

1) Use the projection matrix $\mathbf{B}_1$ ($\Delta\mathrm{P}^\top = \mathrm{B}_1 P_{\mathcal{G}}^\top$) to make $\Delta\mathbf{P}^\top$ orthogonal to the subspace spanned by $\mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top$;

2) Use the projection matrix $\mathbf{B}_2$ ($\Delta\mathrm{P} = \mathrm{B}_2 P_{\mathcal{G}}$) to make $\Delta\mathbf{P}$ orthogonal to the subspace spanned by $\mathbf{S}_{\mathbf{P}_t}$.

Computing B_1 and B_2 via null space:

For B_1: $\mathrm{SVD}\left(\left(\mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top\right)^\top \mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top\right)$, let the matrix of right singular vectors corresponding to the singular values that are (or close to) zero be $\mathrm{U}_{1,0}$, then $\mathrm{B}_1 = \mathrm{U}_{1,0}\mathrm{U}_{1,0}^\top$

For B_2: $\mathrm{SVD}(\mathbf{S}_{\mathbf{P}_t}^\top \mathbf{S}_{\mathbf{P}_t})$, let the matrix of right singular vectors corresponding to the singular values that are (or close to) zero be $\mathrm{U}_{2,0}$, then $\mathrm{B}_2 = \mathrm{U}_{2,0}\mathrm{U}_{2,0}^\top$

Finally, the update rule from gradient $\mathrm{P}_{\mathcal{G}}$ to update $\Delta\mathrm{P}$ is given by:

$$\Delta\mathbf{P} = \mathbf{B}_2\mathbf{P}_{\mathcal{G}}\mathbf{B}_1 = \left(\mathbf{U}_{2,0}\mathbf{U}_{2,0}^\top\right)\mathbf{P}_{\mathcal{G}}\left(\mathbf{U}_{1,0}\mathbf{U}_{1,0}^\top\right) \quad ⑨$$

The constraint on the distributional variation of the prompt

$$\begin{cases} \mu_{\mathrm{P}_t+\Delta\mathrm{P}} = \mu_{\mathrm{P}_t} \\ \sigma_{\mathrm{P}_t+\Delta\mathrm{P}} = \sigma_{\mathrm{P}_t} \end{cases}$$

is achieved by introducing a loss function $\mathcal{L}_{\mathrm{LN}}$:

$$\mathcal{L}_{\mathrm{LN}} = \left\|\mu_{\mathbf{P}_{t+1}} - \mu_{\mathbf{P}_t}\right\|_1 + \left\|\sigma_{\mathbf{P}_{t+1}} - \sigma_{\mathbf{P}_t}\right\|_1 \quad ⑩$$

1) ⑨: Applying null space projection to the gradient

2) ⑩: Constraining the distribution of the prompt

Our Method

NEURAL INFORMATION PROCESSING SYSTEMS

西北工業大學 NORTHWESTERN POLYTECHNICAL UNIVERSITY

西安電子科技大學

Extension to Multi-Head Attention

The Transformer architecture generally employs multi-head attention, so the derived consistency conditions ⑦⑧ need to be extended to multi-head attention.

$$\begin{cases} (\mathbf{Q}_{\mathbf{X}_t}\mathbf{W}_k^\top)\Delta\mathbf{P}^\top = \mathbf{0} & ⑦ \\ \mathbf{S}_{\mathbf{P}_t}\Delta\mathbf{P} = \mathbf{0} & ⑧ \end{cases}$$

Assuming there are H attention heads, let the h-th attention head for $h \in [1, 2, \cdots, H]$ have its respective components $\mathbf{Q}_{\mathbf{X}_t.h}, \mathbf{W}_{k.h}, \mathbf{S}_{\mathbf{P}_t.h}$. To satisfy the consistency conditions across all attention heads, we define:

$$\forall h \in [1, 2, \cdots, H], \begin{cases} (\mathbf{Q}_{\mathbf{X}_t.h}\mathbf{W}_{k.h}^\top)\Delta\mathbf{P}^\top = \mathbf{0} \\ \mathbf{S}_{\mathbf{P}_t.h}\Delta\mathbf{P} = \mathbf{0} \end{cases}$$

$\mathbf{\Omega}_{1,t} = [\mathbf{Q}_{\mathbf{X}_t.1}\mathbf{W}_{k.1}^\top; \cdots; \mathbf{Q}_{\mathbf{X}_t.H}\mathbf{W}_{k.H}^\top]$ **, representing the concatenated matrices $\mathbf{Q}_{\mathbf{X}_t.h}\mathbf{W}_{k.h}^\top$ from each attention head.**

$\mathbf{\Omega}_{2,t} = [\mathbf{S}_{\mathbf{P}_t.1}; \cdots; \mathbf{S}_{\mathbf{P}_t.H}]$ **representing the concatenated $\mathbf{S}_{\mathbf{P}_t.h}$ matrices from each attention head.**

Using block matrix operations, the consistency conditions for all attention heads can be expressed as follows:

$$\begin{cases} \mathbf{\Omega}_{1,t}\Delta\mathbf{P}^\top = \mathbf{0} \\ \mathbf{\Omega}_{2,t}\Delta\mathbf{P} = \mathbf{0} \end{cases}$$

The prompt distribution constraint loss function $\mathcal{L}_{\text{LN}} = \|\boldsymbol{\mu}_{\mathbf{P}_{t+1}} - \boldsymbol{\mu}_{\mathbf{P}_t}\|_1 + \|\boldsymbol{\sigma}_{\mathbf{P}_{t+1}} - \boldsymbol{\sigma}_{\mathbf{P}_t}\|_1$ remains independent of the number of attention heads, so no further extension is required.

Key Settings

Model used:

a) ViT-Base16 model pre-trained on ImageNet-21K, with prompts of length 4 inserted into all ViT layers (referred to as VPT);

b) CLIP model, with prompts added to the image encoder, while the text encoder is frozen.

Evaluation benchmarks:

Four class-incremental test benchmarks, including 10-task CIFAR-100, 20-task CIFAR-100, 10-task ImageNet-R, and 10-task DomainNet-200.

Evaluation metrics:

Accuracy (Acc., the higher, the better) and Forgetting (the lower, the better), with the average results of three runs reported. Accuracy is the primary focus metric.

Experiments

Comparing with Baseline

“-Seq” : baseline
“-NSP²” : our method

"Upper-bound": Jointly training all tasks at once, can be considered the upper bound of performance in continual learning.

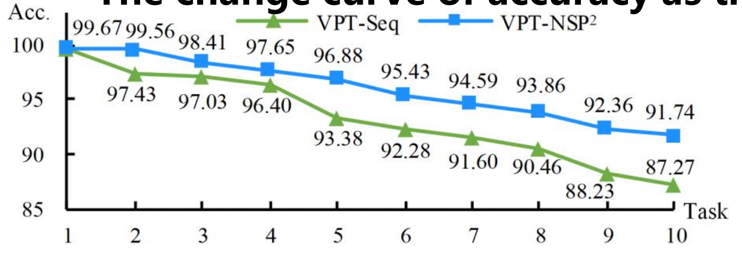
Comparison with the baseline using VPT and CLIP models

Method	10S-CIFAR-100		20S-CIFAR-100		10S-ImageNet-R		10S-DomainNet	
	Acc. ↑	Forgetting ↓	Acc. ↑	Forgetting ↓	Acc. ↑	Forgetting ↓	Acc. ↑	Forgetting ↓
VPT-Seq	87.27	12.33	82.36	17.36	72.46	19.41	73.28	25.65
VPT-NSP^2	**91.74**	**3.28**	**89.89**	**4.91**	**78.88**	**5.06**	**83.54**	**8.54**
Upper-bound	93.87	-	93.87	-	84.60	-	89.25	-
CLIP-Seq	72.91	15.13	71.37	17.89	75.69	19.21	67.73	35.60
CLIP-NSP^2	**80.96**	**12.45**	**79.83**	**13.77**	**82.17**	**6.42**	**77.04**	**18.33**
Upper-bound	84.52	-	84.52	-	84.86	-	81.65	-

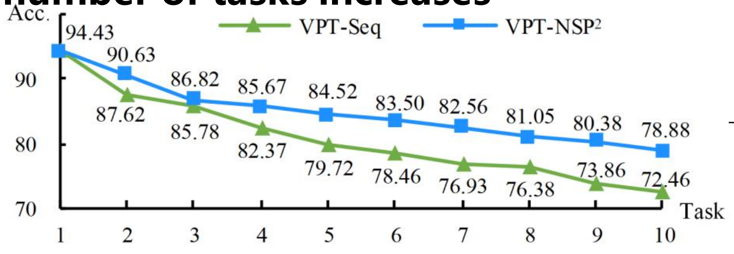
Accuracy increased by 4%~10%
Forgetting decreased by 9%~17%。

Accuracy increased by 6%~9%
Forgetting decreased by 3%~17%。

The change curve of accuracy as the number of tasks increases



(a) 10-split CIFAR-100



(b) 10-split ImageNet-R

Our method consistently outperforms the baseline, with its advantages becoming more pronounced over time.

Experiments

NEURAL INFORMATION PROCESSING SYSTEMS

西北工業大學 NORTHWESTERN POLYTECHNICAL UNIVERSITY

西安電子科技大學

Comparing with Existing Methods

Comparison with existing methods based on ImageNet-21K pre-trained VPT, numbers following ± representing the standard deviation

Method	Venue	10S-CIFAR-100		20S-CIFAR-100		10S-ImageNet-R		10S-DomainNet	
		Acc.	Forgetting	Acc.	Forgetting	Acc.	Forgetting	Acc.	Forgetting
L2P [40]	CVPR'22	$83.83_{\pm 0.04}$	$7.63_{\pm 0.30}$	$80.10_{\pm 0.72}^{\ddagger}$	-	$61.57_{\pm 0.66}$	$9.73_{\pm 0.47}$	$81.17_{\pm 0.83}^{\dagger}$	$8.98_{\pm 1.25}$
DualPrompt [39]	ECCV'22	$86.51_{\pm 0.33}$	$5.16_{\pm 0.09}$	$82.02_{\pm 0.32}^{\ddagger}$	-	$68.13_{\pm 0.49}$	$4.68_{\pm 0.20}$	$81.70_{\pm 0.78}^{\dagger}$	$8.04_{\pm 0.31}$
CODA-P [32]	CVPR'23	$86.25_{\pm 0.74}$	$1.67_{\pm 0.26}$	-	-	$75.45_{\pm 0.56}$	$1.64_{\pm 0.10}$	$80.04_{\pm 0.79}^{\dagger}$	$10.16_{\pm 0.35}$
ESN [38]	AAAI'23	$86.34_{\pm 0.52}$	$4.76_{\pm 0.14}$	$80.56_{\pm 0.94}^{\ddagger}$	-	$62.61_{\pm 0.96}^{\ddagger}$	-	$79.22_{\pm 2.04}^{\dagger}$	$10.62_{\pm 2.12}$
APG [33]	ICCV'23	$\underline{89.35}$	6.01	$\underline{88.64}$	6.51	73.27	8.59	-	-
LAE [10]	ICCV'23	$85.59_{\pm 0.46}$	-	$83.93_{\pm 0.28}$	-	$72.66_{\pm 0.63}$	-	-	-
DualP-LGCL [15]	ICCV'23	$87.23_{\pm 0.21}$	$5.10_{\pm 0.15}$	-	-	$69.46_{\pm 0.04}$	$4.20_{\pm 0.06}$	-	-
C-LN [23]	ICCVW'23	$86.95_{\pm 0.37}$	$6.98_{\pm 0.43}$	-	-	$76.36_{\pm 0.51}$	$8.31_{\pm 1.28}$	-	-
EvoPrompt [18]	AAAI'24	$87.97_{\pm 0.30}$	$2.60_{\pm 0.42}$	$84.64_{\pm 0.14}$	$3.98_{\pm 0.24}$	$76.83_{\pm 0.08}$	$2.78_{\pm 0.06}$	-	-
OVOR-Deep [12]	ICLR'24	$85.99_{\pm 0.89}$	$6.42_{\pm 2.03}$	-	-	$76.11_{\pm 0.21}$	$7.16_{\pm 0.34}$	-	-
DualP-PGP [26]	ICLR'24	$86.92_{\pm 0.05}$	$5.35_{\pm 0.19}$	$83.74_{\pm 0.01}$	$7.91_{\pm 0.15}$	$69.34_{\pm 0.05}$	$4.53_{\pm 0.04}$	-	-
InfLoRA [20]	CVPR'24	$87.06_{\pm 0.25}$	-	-	-	$75.65_{\pm 0.14}$	-	-	-
EASE [45]	CVPR'24	87.76	-	85.80	-	76.17	-	-	-
CPrompt [11]	CVPR'24	$87.82_{\pm 0.21}$	$5.06_{\pm 0.50}$	-	-	$\underline{77.14}_{\pm 0.11}$	$5.97_{\pm 0.68}$	$\underline{82.97}_{\pm 0.34}$	$7.45_{\pm 0.93}$
VPT-NSP^2	This work	$\mathbf{91.74}_{\pm 0.63}$	$3.28_{\pm 0.45}$	$\mathbf{89.89}_{\pm 0.72}$	$4.91_{\pm 0.59}$	$\mathbf{78.88}_{\pm 0.50}$	$5.06_{\pm 0.26}$	$\mathbf{83.54}_{\pm 0.77}$	$8.54_{\pm 0.48}$

Our method Achieves state-of-the-art performance on all evaluation benchmarks.

Ablation Study

Ablation study on B_1, B_2 and $\mathcal{L}_{\mathrm{LN}}$

B_1	B_2	$\mathcal{L}_{\mathrm{LN}}$	10S-CIFAR-100		20S-CIFAR-100		10S-ImageNet-R		10S-DomainNet	
			Acc. ↑	Forgetting ↓	Acc. ↑	Forgetting ↓	Acc. ↑	Forgetting ↓	Acc. ↑	Forgetting ↓
			87.27	12.33	82.36	17.36	72.46	19.41	73.28	25.65
√			90.58	6.91	88.13	10.27	78.05	8.14	82.31	10.89
	√		88.74	10.85	83.32	16.48	74.71	14.69	78.87	17.81
√	√		91.33	4.22	88.96	6.42	78.37	6.25	83.17	8.95
√		√	91.42	3.94	88.46	8.64	78.30	6.31	83.13	9.32
	√	√	89.36	9.32	86.67	11.59	75.27	13.35	79.45	16.50
√	√	√	**91.74**	**3.28**	**89.89**	**4.91**	**78.88**	**5.06**	**83.54**	**8.54**

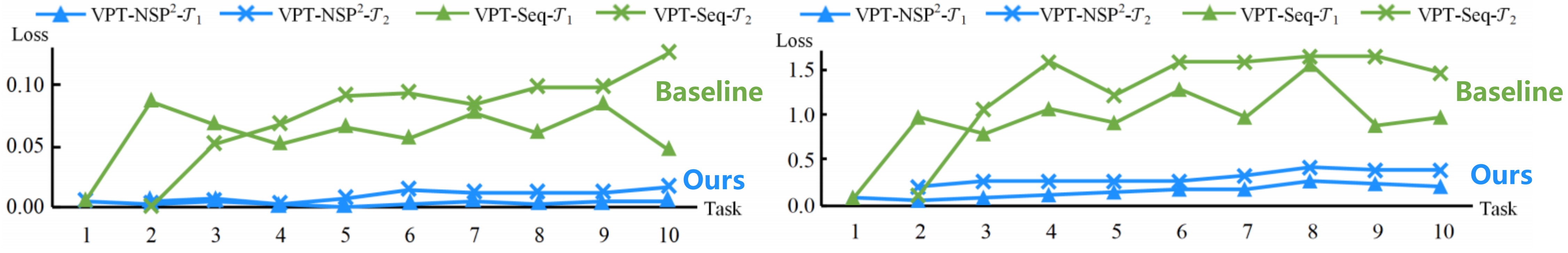
The first consistency condition, the orthogonal subspace projection matrix B_1, has the most significant impact on performance. However, B_2 and $\mathcal{L}_{\mathrm{LN}}$ are also indispensable. The highest accuracy and lowest forgetting rate are achieved only when all three components are used together.

Experiments

Analysis of Reducing Training Interventions

As the number of learning tasks increases, if the issue of training interventions becomes more severe, the model's loss on the old task data will be higher. Conversely, if training interventions can be minimized or eliminated, the model's loss on the old task data will not increase.

The loss curves on the training data of the 1st and 2nd tasks as the number of learning tasks increases, comparing the proposed method (VPT-NSP2) with the baseline method (VPT-Seq).



(a) 10-split CIFAR-100 (b) 10-split ImageNet-R

The proposed method maintains almost no change in the loss on old task data, demonstrating its ability to eliminate training interventions, thereby preventing forgetting.

Stability-Plasticity Trade-off

The stability-plasticity trade-off in the proposed method is represented by $\Delta\mathbf{P} = [\eta_2\mathbf{B}_2 + (1-\eta_2)\mathbf{I}]\mathbf{P}_{\mathcal{G}}[\eta_1\mathbf{B}_1 + (1-\eta_1)\mathbf{I}]$, where the orthogonal subspace projection matrices $\mathbf{B}_1, \mathbf{B}_2$ are weighted and fused with the identity matrix $\mathbf{I}$. The parameters $\eta_1, \eta_2 \in [0,1]$ control the weights, allowing for the adjustment of the trade-off between plasticity and stability.

When the weights η_1, η_2 are set to the same value, denoted as $\bar{\eta}$, the changes in accuracy and forgetting rate as $\bar{\eta}$ decreases from 1.0 to 0.8 are shown in the following figure.

(a) 10-split CIFAR-100 (b) 20-split CIFAR-100 (c) 10-split ImageNet-R (d) 10-split DomainNet

Accuracy is influenced by both stability and plasticity, and the best performance is achieved when a good balance between the two is attained. An increase in plasticity means a weakened ability to retain knowledge from old tasks, which results in a gradual increase in the forgetting rate.

NEURAL INFORMATION PROCESSING SYSTEMS

Thank You for Watching!

Paper (ArXiv)

Code (Github)

Author: Yue Lu (WeChat)

西北工業大學
NORTHWESTERN POLYTECHNICAL UNIVERSITY

西安電子科技大學
XIDIAN UNIVERSITY