

Plasticity without Collapse: Plasticity-Inducing Control Optimizer for Cross-Lingual Continual Pre-Training

Sumin Kim¹, Minjun Song^{1,2}, Surin Lee^{1,2}, S M Wahidur Rahman¹

Yongtae Lee¹, Haeung Choi¹, Heung-No Lee^{1,2†}

¹Gwangju Institute of Science and Technology (GIST), ²LiberVance

{smkim6927, paulmjsong, leesurin, sm.wahidur, lyt98313, haeung}@gm.gist.ac.kr

heungno@gist.ac.kr [†]Corresponding author

Abstract

Language models (LMs) operate in evolving environments where adapting to new domains and linguistic shifts is critical. Continual pre-training (CPT) addresses this but introduces a strict stability-plasticity dilemma. While current approaches manage this via data replay or architectural changes, they largely treat the optimizer as a static component. We propose the Plasticity-Inducing Control Optimizer (PICO) to tackle this dilemma directly at the update level. The optimizer defines a utility-gated update together with a stochastic perturbation path controlled by periodic spectral sensitivity checks. Structural flags suppress the perturbation path while leaving the protected-gradient update active. Across the evaluated legal and medical streams, PICO exhibits a favorable stability-plasticity trade-off through optimizer-level control, while incurring a measurable spectral-monitoring cost. Code is available at <https://github.com/infonet-gist/infonet-ai-pico>.

1 Introduction

Retraining language models (LMs) from scratch for each data shift is impractical. Continual pre-training (CPT) sequentially adapts models to new domains or languages without full retraining (Gururangan et al., 2020; Ke et al., 2023; Cossu et al., 2024). However, CPT struggles when domain and language shifts compound in sequential training. Models must adapt to new linguistic regularities while preserving prior representations. Failure to maintain this balance triggers Catastrophic Forgetting (CF) (Kirkpatrick et al., 2017; McCloskey and Cohen, 1989). In cross-lingual CPT, aggressive adaptation may improve the current stage while degrading capabilities acquired in earlier stages. Thus, the central challenge is to preserve previously acquired knowledge while retaining sufficient plasticity to adapt to new languages and domains.

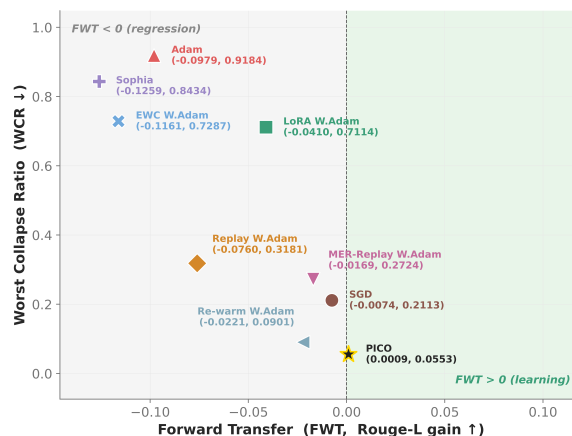


Figure 1: Plasticity-collapse trade-off on the cross-lingual CPT stream. Horizontal axis: FWT (ROUGE-L gain, ↑). Vertical axis: WCR (↓). PICO is the only method with positive FWT and also attains the lowest WCR among the trained methods.

Existing CPT methods address this tension through interventions at the data, schedule, parameter, or architecture level. Replay and data-mixture strategies revisit earlier or general-domain corpora (Ibrahim et al., 2024; Gu et al., 2024). Soft-masking restricts parameter updates to mitigate forgetting (Ke et al., 2023). Rewarm strategies adjust learning-rate schedules during continued pre-training (Gupta et al., 2023). Recyclable tuning studies reuse of previously adapted weights after further CPT (Qin et al., 2023). Parameter-efficient modules such as LoRA restrict adaptation to low-rank update spaces (Hu et al., 2021). While effective, these approaches typically require replay buffers, additional modules, stage-specific heuristics, or architectural modifications.

Most CPT methods also treat the optimizer as a fixed backend. We instead view the optimizer as a control mechanism for regulating forgetting under cross-lingual shifts. Recent work shows that optimizer dynamics strongly influence whether neural networks preserve plasticity under non-stationary

training (Lyle et al., 2023; Dohare et al., 2024). Parallel evidence from LLM fine-tuning reports a direct empirical correlation between the flatness of the loss landscape and the extent of CF, and shows that sharpness-aware minimization alleviates forgetting (Li et al., 2024). These observations suggest that update dynamics themselves are a critical yet underexplored factor in cross-lingual CPT.

Figure 1 illustrates this trade-off on our cross-lingual CPT stream. Existing trained baselines exhibit negative ROUGE-L forward transfer (FWT), while their Worst Collapse Ratio (WCR) varies substantially. Adam and Sophia regress while exhibiting severe collapse, and replay-based or schedule-based methods reduce collapse but still remain on the negative-FWT side.

These results suggest that the challenge is not maximizing plasticity in isolation, but preserving transfer while constraining degradation of previously learned domains. To capture this latter failure mode, we introduce the WCR, a worst-case metric that measures the largest relative degradation across previously learned domains. Unlike stream-averaged continual-learning metrics such as backward transfer (BWT) and average forgetting (AvgF), WCR exposes severe collapse localized to an individual stage.

Motivated by these observations, we propose Plasticity-Inducing Control Optimizer (PICO), an optimizer designed to regulate update dynamics directly under cross-lingual CPT. PICO combines coordinate-wise utility protection with spectral sensitivity monitoring and a gated perturbation path to regulate update dynamics during sequential adaptation. As shown in Figure 1, PICO is the only evaluated method with positive FWT while also attaining the lowest WCR, suggesting that optimizer-level control can improve transfer without incurring severe worst-case degradation.

Our contributions are summarized as follows:

- **Problem formulation.** We formulate cross-lingual CPT as an optimizer-level control problem and provide evidence that update dynamics materially shape the stability–plasticity trade-off in the evaluated streams.
- **Metric.** We introduce Worst Collapse Ratio (WCR), a worst-case per-domain metric that exposes stage-localized collapse often hidden by stream-averaged continual learning metrics such as BWT and AvgF.
- **Algorithm.** We propose PICO, a structure-aware optimizer that combines utility-gated updates, spectral sensitivity monitoring, and controlled stochastic perturbations to regulate the stability–plasticity trade-off.
- **Data resource.** We provide provenance, file-level statistics, and licensing information for four English–Korean legal and medical CPT corpus files used in our experiments, with raw text released where permitted by the corresponding upstream terms.

2 Preliminary Study

Before introducing PICO, we examine two failure patterns under coupled language–domain shifts: the optimizer-dependent stability–plasticity trade-off and degradation that is not reflected by training loss. Detailed stream formulations and visualizations are provided in Appendix D.

Optimizer-Level Stability–Plasticity Regimes

To isolate optimizer-dependent behavior, we use a two-stage Korean Medical → English Medical stream while holding the model, training data, and evaluation protocol fixed. The comparison reveals a clear stability–plasticity tension (Appendix D.2). Conservative optimizers remain stable largely because they adapt weakly, whereas stronger adaptation can improve plasticity at the cost of previously acquired capability. Adam exemplifies the latter regime with the strongest plasticity but negative backward transfer, while AdamW occupies a more balanced region with positive plasticity and stability. These results motivate controlling adaptation without allowing destructive overwriting.

Loss–Generation Mismatch Training loss does not always expose such degradation. For some optimizers, cross-entropy on Korean Medical continues to decrease while generation quality deteriorates after Stage 2 (Table 8). This mismatch motivates a complementary signal beyond loss-level monitoring.

Motivation for PICO Together, these observations motivate PICO’s three components. GateU protects parameters according to their estimated utility. SpecFlag monitors matrix-level spectral sensitivity beyond the training loss. The gated perturbation mechanism preserves adaptation capacity when the utility and structural conditions remain favorable. The next section formalizes these components.

3 Proposed Method

This section formulates cross-lingual CPT as an optimizer-level control problem and presents PICO, which combines coordinate-wise utility gating with periodic matrix-level spectral monitoring.

3.1 Problem Formulation

Optimizer-level control in CPT. During cross-lingual CPT, a language model sequentially encounters non-stationary data streams. Let $\theta^{(k)} \in \mathbb{R}^d$ denote the model parameters at optimizer step k , where d is the total number of trainable parameters, and let $\mathcal{L}(\theta)$ denote the loss on the current CPT-stage data. The gradient at optimizer step k is

$$g^{(k)} = \nabla_{\theta} \mathcal{L}(\theta^{(k)}), \quad (1)$$

where ∇_{θ} denotes differentiation with respect to θ and $\odot$ denotes element-wise multiplication. A complete notation summary is provided in Appendix L.

Conventional optimizers update parameters to minimize the current-stage objective using the available gradients and optimizer state. Under sequential language and domain shifts, these updates can interfere with capabilities acquired in earlier stages. We therefore write the controlled update abstractly as

$$\theta^{(k+1)} = \Phi(\theta^{(k)}, g^{(k)}, \xi^{(k)}), \quad (2)$$

where $\xi^{(k)}$ denotes an optional stochastic perturbation. The update map also depends on persistent optimizer state, which is omitted from its arguments for readability. Section 3.2 instantiates this map through GATEU and SPECFLAG.

3.2 Plasticity-Inducing Control Optimizer (PICO)

Building on this formulation, we introduce **PICO**, an optimizer-level alternative to Adam in our training pipeline. PICO follows the PyTorch optimizer interface and combines coordinate-wise utility gating with periodic matrix-level spectral monitoring. Let $\{\mathcal{G}_j\}_{j=1}^M$ denote the optimizer parameter groups supplied to PICO. The reported training implementation supplies all trainable parameters as a single optimizer group, so $M = 1$. The group-level perturbation pause therefore applies to that single group on a scheduled check step. We retain the general group notation to describe the optimizer interface.

Algorithm 1 PICO Main Update

Require: Parameter groups $\{\mathcal{G}_j\}_{j=1}^M$, parameters $\theta^{(k)}$, and global optimizer step k
Hyperparameters $\eta, \lambda, \sigma_0, \beta_u, f, K, c_0, r, \tau_{\text{start}}, \tau_{\text{end}}, \varepsilon$
Ensure: Updated parameters $\theta^{(k+1)}$

```

1: for  $j = 1$  to  $M$  do
2:    $\pi_{\mathcal{G}_j}^{(k)} \leftarrow \text{SPECFLAG}(\mathcal{G}_j, k, f)$  ▷ Alg. 3
3: end for
4: for  $j = 1$  to  $M$  do
5:   for all parameter tensor  $\theta \in \mathcal{G}_j$  with gradient  $g$  do
6:      $(u^{(k)}, \tilde{g}^{(k)}) \leftarrow \text{GATEU}(\theta, g)$  ▷ Alg. 2
7:      $s^{(k)} \leftarrow \text{specSens}[\theta]$  ▷ Cached state
8:      $m^{(k)} \leftarrow (1 - u^{(k)})(1 - s^{(k)})(1 - \pi_{\mathcal{G}_j}^{(k)})$  ▷ Eq. (5)
9:      $\sigma_k \leftarrow \sigma_0 m^{(k)}$  ▷ Eq. (6)
10:    Sample  $\xi^{(k)} \sim \mathcal{N}(0, \text{diag}(\sigma_k^2))$  ▷ Eq. (6)
11:     $\theta \leftarrow (1 - \eta\lambda)\theta - \eta(\tilde{g}^{(k)} + \xi^{(k)})$  ▷ Eq. (7)
12:   end for
13: end for
14: return Updated parameters  $\theta^{(k+1)}$ 

```

Importance protection through coordinate-wise utility gating (GATEU). GATEU adapts the coordinate-wise utility mechanism of UPGD (El-sayed and Mahmood, 2024), while PICO combines this mechanism with matrix-level spectral monitoring and a separately gated perturbation path. PICO maintains an exponential moving average (EMA) of the parameter–gradient product $-g \odot \theta$ as a coordinate-wise utility proxy. This statistic is normalized per tensor and passed through a sigmoid to produce an element-wise gate $u^{(k)} \in [0, 1]$ (Appendix E). The gate attenuates the deterministic gradient as

$$\tilde{g}^{(k)} = g^{(k)} \odot (1 - u^{(k)}). \quad (3)$$

Larger values of $u^{(k)}$ produce stronger attenuation of both the deterministic gradient and the perturbation amplitude.

Matrix-level spectral monitoring (SPECFLAG). Coordinate-wise utility statistics estimate local parameter importance but do not directly capture changes in a matrix’s singular-value spectrum. Prior work has linked layer-wise spectral properties, particularly the largest singular value, to gradient diversity and continual trainability (Lewandowski et al., 2025). Motivated by this connection, SPECFLAG periodically monitors the leading singular value and a spectral-concentration statistic for each two-dimensional parameter with an available gradient on a scheduled check step. Exact recursions and execution order are given in Appendix F and Algorithm 3.

Spectral concentration as a structural-change indicator.

The squared Frobenius norm satisfies $\|W\|_F^2 = \sum_l \sigma_l(W)^2$ and represents the total energy of the singular spectrum. Let $\hat{\sigma}_1(W)$ denote the leading singular value estimated by warm-started power iteration. The implementation computes

$$\hat{\rho}_\varepsilon(W) = \text{clip}_{[0,1]} \left(\frac{\hat{\sigma}_1(W)^2}{\|W\|_F^2 + \varepsilon} \right). \quad (4)$$

For a nonzero matrix of rank r_W , the corresponding exact unstabilized ratio $\sigma_1(W)^2/\|W\|_F^2$ lies in $[1/r_W, 1]$. Equal nonzero singular values attain the lower bound, while a rank-one spectrum attains the upper bound. The quantities $\hat{\sigma}_1$ and $\hat{\rho}_\varepsilon$ provide complementary information. The former tracks the magnitude of the dominant singular mode, while the latter tracks its concentration relative to the remaining spectrum. We use both quantities as matrix-level change indicators and do not assume that every forgetting event corresponds to rank-one collapse.

On scheduled check steps, SPECFLAG compares $\hat{\sigma}_1$ and $\hat{\rho}_\varepsilon$ with per-matrix EMA statistics. Here, the EMA statistics consist of an exponentially weighted mean and an exponentially weighted squared-deviation estimate for each spectral quantity. A matrix-level flag is activated when either signed standardized deviation exceeds the annealed upper threshold and remains cached until the matrix is evaluated again. If any monitored matrix in $\mathcal{G}_j$ is flagged at the current check, PICO activates $\pi_{\mathcal{G}_j}^{(k)}$ for that check step. The group-level indicator is zero on non-check steps. Both structural indicators suppress only the stochastic perturbation. The protected-gradient update and decoupled weight decay remain active. At each parameter update, Algorithm 1 reads $s^{(k)} = \text{specSens}[\theta]$ from the optimizer state. Exact recursions and the local counter c_W are given in Appendix F.

Composed perturbation gate. The coordinate-wise utility gate $u^{(k)}$, cached matrix-level flag $s^{(k)}$, and transient group-level perturbation-pause indicator $\pi_{\mathcal{G}_j}^{(k)}$ combine as

$$m^{(k)} = \left(1 - u^{(k)}\right) \left(1 - s^{(k)}\right) \left(1 - \pi_{\mathcal{G}_j}^{(k)}\right). \quad (5)$$

The utility factor continuously scales the perturbation amplitude, whereas either active binary structural indicator sets the perturbation gate to zero.

Optimizer	ms/update	Slowdown
AdamW	633.83	—
PICO ($f=1$)	852.29	+34.5%

Table 1: Wall-clock profiling on Llama-3.2-1B-Instruct using real CPT batches, bf16 training, gradient check-pointing, batch size 32, and sequence length 256 on one A100-40GB GPU. Values are means over three fresh-process trials after 400 warm-up updates. PICO uses $K = 1$.

The gate scales the base perturbation level σ_0 as

$$\sigma_k = \sigma_0 m^{(k)}, \quad \xi^{(k)} \sim \mathcal{N}(0, \text{diag}(\sigma_k^2)). \quad (6)$$

Coordinates with lower estimated utility receive larger perturbations when neither structural indicator is active. The final update applies decoupled weight decay (Loshchilov and Hutter, 2019).

$$\theta^{(k+1)} = (1 - \eta\lambda)\theta^{(k)} - \eta \left(\tilde{g}^{(k)} + \xi^{(k)} \right). \quad (7)$$

The computational cost of periodic spectral monitoring is analyzed in Section 3.3.

3.3 Algorithm and Computational Cost

For a monitored matrix $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$, each power-iteration round multiplies once by W and once by $W^\top$, and each check also computes the Frobenius norm. Suppressing constant factors, the amortized arithmetic cost is

$$\mathcal{O}\left(\frac{(K+1)d_{\text{out}}d_{\text{in}}}{f}\right), \quad (8)$$

which is $\mathcal{O}(Kd_{\text{out}}d_{\text{in}}/f)$ for $K \geq 1$. Table 1 shows a +34.5% wall-clock overhead for the main $f = 1$ profiling configuration. Reducing the check frequency to $f = 10$ lowers the measured overhead to +20.0%. Full timing and memory results are reported in Appendix G.

4 Experiments

4.1 Experimental Setup

We use pretrained Llama backbones throughout our experiments. Unless otherwise stated, the preliminary study, main four-stage comparison, component ablations, and sensitivity analyses use Llama-3.2-1B-Instruct (Meta AI, 2024). For the model-size study, we evaluate Llama-3.2-1B-Instruct, Llama-3.2-3B-Instruct, and Llama-3.1-8B-Instruct (Grattafiori et al., 2024). We use Llama-3.1-8B-Instruct for the largest setting because the Llama

Domain–Language	Evaluation resource	Task
Legal–English	Legal Case Document Summarization (Shukla et al., 2022)	Summ.
Legal–Korean	LawQA-Ko (jihye-moon, 2023)	Q&A
Medical–English	medical_gemma_instruct_dataset (Shekswess, 2024)	Q&A
Medical–Korean	medical-o1-reasoning-SFT-Ko (ChuGyouk, 2025)	Q&A
Math–English	GSM8K (Cobbe et al., 2021)	Math

Table 2: **Evaluation resources.** External datasets used for domain-specific and out-of-domain evaluation across legal, medical, and mathematical reasoning tasks. Detailed split, sample-count, and licensing information is provided in Appendix H.2.

3.2 family does not include an 8B checkpoint. Within each CPT run, the backbone is fixed across stages. Only the incoming data distribution and associated training loss change. This setup follows standard CPT practice for adapting LMs to new domains while enabling controlled evaluation under sequential language–domain shifts (Gururangan et al., 2020).

Domain–Language	Size (MB)	Records
Legal–English	421.51	5,466,261
Legal–Korean	457.84	140,053
Medical–English	460.09	2,205,836
Medical–Korean	367.81	37,505

Table 3: The four CPT training corpora. MB denotes 10^6 bytes and records are nonempty physical lines after corpus construction.

All models are trained sequentially across CPT stages. As detailed in Appendix H.1 and Table 3, our primary CPT stream uses four corpus files spanning two domains (Legal and Medical) and two languages (English and Korean). To assess both plasticity and stability, we conduct evaluations on separate end-task datasets (*e.g.*, summarization and Q&A) after each training stage. Furthermore, the GSM8K dataset is maintained strictly as an evaluation-only benchmark to monitor broad degradation and verify whether optimization under continual shifts harms general mathematical reasoning outside the trained domains.

PICO configuration. We set the base perturbation scale to $\sigma_0 = 0.001$. The primary four-stage comparison and component ablations use $f = 1$, whereas the model-scaling experiments use $f = 10$. The sensitivity analysis additionally evaluates $f = 100$. Additional hyperparameters are reported in Appendix H.3.8.

4.2 Baselines

We benchmark PICO against eight trained baselines spanning six CPT method families. All meth-

ods use the same CPT stream and sequence length. Method-specific replay, schedule, or parameterization components are applied only where required by the corresponding baseline. We consider (i) **Adam** (Kingma and Ba, 2014) and **SGD** (Robbins and Monro, 1951) as passive first-order backends, (ii) **Sophia** (Liu et al., 2024) as a second-order optimizer, (iii) **EWC W.Adam** (Kirkpatrick et al., 2017) as a regularization baseline, (iv) **Replay W.Adam** (Ibrahim et al., 2024) and **MER-Replay W.Adam** (Abbes et al., 2026; Riemer et al., 2019) as replay-based baselines, (v) **Re-warm W.Adam** (Gupta et al., 2023) as a learning-rate schedule baseline, and (vi) **LoRA W.Adam** (Hu et al., 2021) with rank-16 adapters as a parameter-efficient baseline. The shared Initial Model checkpoint is reported separately as a non-trained reference. Implementation details are provided in Appendix H.3.

4.3 Evaluation Metrics

We evaluate generation quality with ROUGE-L, BLEU, METEOR, cosine similarity, and Token F1 (Lin, 2004; Papineni et al., 2002; Banerjee and Lavie, 2005; Rajpurkar et al., 2016). For all zero-shot generation evaluations, the model receives only the task prompt; gold target tokens are excluded from the generation input and are used solely as references after generation. Metrics are computed on the generated continuation rather than on the prompt-plus-target sequence. CPT stability and plasticity are measured by FWT, BWT, AvgF, and WCR. We study a stream of T domains. For metric μ , $R_{t,i}^{(\mu)}$ denotes the score on domain i after training up to domain t , and $\bar{R}_i^{(\mu)}$ is the initial model performance on domain i . We use $\bar{R}_i^{(\mu)}$ in place of the random-initialization baseline of Lopez-Paz and Ranzato (2017), since CPT starts from a pre-trained checkpoint. Implementation details and the relation to prior CL metrics are deferred to Appendix I.

Domain	Family	Method	Lingual	ROUGE-L $\uparrow$	Cosine Sim. $\uparrow$	METEOR $\uparrow$	BLEU $\uparrow$	Token F1 $\uparrow$
Medical	Reference	Initial Model	English Korean	0.1498 0.1474	0.4635 0.2285	0.1966 0.1733	0.0342 0.0608	0.2223 0.2461
	2nd-order approx.	Sophia	English Korean	0.0464 0.0092	0.1290 0.0014	0.0622 0.0287	0.0010 0.0001	0.0835 0.0246
	Standard (1st-order)	SGD	English Korean	0.0834 0.1099	0.2339 0.2196	0.0743 0.1476	<u>0.0127</u> 0.0520	0.0929 0.1962
		Adam	English Korean	0.0173 0.0098	0.0400 0.0015	0.0282 0.0229	0.0002 0.0006	0.0443 0.0318
	Regularization (EWC)	EWC W.Adam	English Korean	0.0558 0.0173	0.1390 0.0098	0.0536 0.0242	0.0039 0.0015	0.0628 0.0265
	Replay Adaptation	Replay W.Adam	English Korean	0.0311 0.0681	0.1171 0.0841	0.0326 0.0875	0.0030 0.0188	0.0474 0.1963
		MER-Replay W.Adam	English Korean	<u>0.0857</u> 0.1015	<u>0.2500</u> 0.1269	<u>0.0820</u> 0.0962	0.0034 0.0300	<u>0.1290</u> 0.2107
	Parameter Efficient Adaptation	LoRA W.Adam	English Korean	0.0462 0.0370	0.2213 0.0799	0.0460 0.0508	0.0077 0.0122	0.0531 0.0610
	Schedule Adaptation	Re-warm W.Adam	English Korean	0.0735 <u>0.1172</u>	0.2206 <u>0.2117</u>	0.0732 <u>0.1576</u>	0.0127 0.0565	0.0961 <u>0.2176</u>
	Regularization (Perturbation/noise-based)	PICO (ours)	English Korean	0.1505 0.1378	0.4387 0.2067	0.1851 0.1589	0.0344 <u>0.0563</u>	0.2233 0.2236
Legal	Reference	Initial Model	English Korean	0.2012 0.0641	0.6596 0.1061	0.1621 0.0743	0.0255 0.0228	0.2684 0.2289
	2nd-order approx.	Sophia	English Korean	0.0577 0.0087	0.1794 0.0004	0.0482 0.0166	0.0037 0.0000	0.0961 0.0243
	Standard (1st-order)	SGD	English Korean	0.1254 0.0474	0.4530 0.0999	0.0942 0.0614	0.0109 <u>0.0186</u>	0.1663 0.1987
		Adam	English Korean	0.0091 0.0168	0.0196 0.0011	0.0058 0.0171	0.0000 0.0005	0.0345 0.0368
	Regularization (EWC)	EWC W.Adam	English Korean	0.0441 0.0204	0.1194 0.0037	0.0340 0.0153	0.0000 0.0010	0.0804 0.0315
	Replay Adaptation	Replay W.Adam	English Korean	0.0216 0.0835	0.4232 0.0517	0.0144 0.0490	0.0003 0.0090	0.0453 0.2234
		MER-Replay W.Adam	English Korean	0.1509 0.0566	<u>0.5973</u> <u>0.1105</u>	0.1040 <u>0.0643</u>	0.0103 0.0170	0.1809 0.2392
	Parameter Efficient Adaptation	LoRA W.Adam	English Korean	0.0877 0.0180	0.4428 0.0234	0.0618 0.0209	0.0064 0.0029	0.1136 0.0547
	Schedule Adaptation	Re-warm W.Adam	English Korean	<u>0.1717</u> 0.0629	0.5598 0.1120	<u>0.1284</u> 0.0719	<u>0.0201</u> 0.0251	<u>0.1914</u> <u>0.2368</u>
	Regularization (Perturbation/noise-based)	PICO (ours)	English Korean	0.1939 <u>0.0655</u>	0.6305 0.0905	0.1473 0.0630	0.0236 0.0179	0.2490 0.1969

Table 4: **Zero-shot end-task performance after CPT.** All checkpoints are evaluated with prompt-only generation. The initial model denotes the shared initial checkpoint and is shown as a reference, but is excluded from the trained-method ranking. All metrics are higher-is-better and values are truncated to four decimal places. **Bold** marks the best trained method and underline marks the second-best trained method.

Forward Transfer (FWT). Adapting the transfer metric of Lopez-Paz and Ranzato (2017) to our pretrained initialization, FWT measures how training on preceding domains affects performance on a future domain before that domain is learned:

$$\text{FWT}^{(\mu)} = \frac{1}{T-1} \sum_{i=2}^T \left(R_{i-1,i}^{(\mu)} - \bar{R}_i^{(\mu)} \right). \quad (9)$$

$R_{i-1,i}^{(\mu)}$ is the score on domain i after training on domain $i-1$. For higher-is-better metrics, $\text{FWT}^{(\mu)} > 0$ indicates that earlier-domain training improves unseen-future-domain performance.

Backward Transfer (BWT). Following Lopez-Paz and Ranzato (2017), BWT measures how subsequent training affects previously learned domains

relative to their performance immediately after they are learned:

$$\text{BWT}^{(\mu)} = \frac{1}{T-1} \sum_{i=1}^{T-1} \left(R_{T,i}^{(\mu)} - R_{i,i}^{(\mu)} \right). \quad (10)$$

For higher-is-better metrics, $\text{BWT}^{(\mu)} < 0$ indicates forgetting. $\text{BWT}^{(\mu)} > 0$ indicates positive backward transfer.

Average Forgetting (AvgF). Following the forgetting formulation of Chaudhry et al. (2018), with the peak restricted to stages at which domain i has already been learned, AvgF measures the gap between each domain’s pre-final post-learning peak

and its final score:

$$\text{AvgF}^{(\mu)} = \frac{1}{T-1} \sum_{i=1}^{T-1} \left(R_i^{(\mu),\text{peak}} - R_{T,i}^{(\mu)} \right), \quad (11)$$

$R_i^{(\mu),\text{peak}} = \max_{t \in \{i, \dots, T-1\}} R_{t,i}^{(\mu)}$. Although μ is higher-is-better, $\text{AvgF}^{(\mu)}$ is itself a forgetting measure, so lower values are better. Negative values indicate that the final-stage score exceeds the pre-final post-learning peak, corresponding to late-stage recovery rather than forgetting.

Worst Collapse Ratio (WCR). Stream-averaged metrics such as AvgF can mask stage-localized failure. A single severely degraded domain may be diluted by healthy ones in the average. Conventional forgetting measures further report absolute performance drops (Lopez-Paz and Ranzato, 2017; Chaudhry et al., 2018). This is intuitive for uniform accuracy scales, but problematic in CPT for LMs. There, metrics such as BLEU and ROUGE-L operate on drastically different absolute scales across domains and metric types.

To enable cross-metric and cross-domain comparison, we follow the normalization principle of Kemker et al. (2018) and express degradation as a relative ratio. We make two adaptations. First, an external offline-trained baseline is infeasible at the scale of LM pre-training. We therefore use the in-stream post-training peak as the per-domain reference:

$$R_i^{(\mu),*} = \max_{t \in \{i, \dots, T\}} R_{t,i}^{(\mu)}, \quad (12)$$

restricted to stages at which domain i has already been learned. The initial model is excluded so that $R_i^{(\mu),*}$ reflects only post-training capability. By construction, this guarantees the resulting ratio lies in $[0, 1]$. Kemker’s external normalizer does not provide this property. Second, we extend the worst-case perspective of De Lange et al. (2023) from the per-iteration to the per-domain axis. We aggregate by maximum rather than mean. The maximum is the natural choice when the objective is to detect *any* catastrophic failure: a single collapsed domain is itself a deployment risk, regardless of how many other domains remain stable. Mean-based aggregations and quantile cutoffs both retain the masking problem when most stages are healthy.

The **WCR** thus measures the most severe relative degradation observed across all learned domains:

$$\text{WCR}^{(\mu)} = \max_{i \in \{1, \dots, T\}} \left(1 - \frac{R_{T,i}^{(\mu)}}{R_i^{(\mu),*}} \right). \quad (13)$$

WCR answers a question that mean-based metrics cannot. It detects whether any single learned domain has been catastrophically forgotten. A single collapsed domain can be obscured in BWT or AvgF when the remaining domains are stable.

By construction, WCR is a worst-case statistic and therefore sensitive to per-domain noise. We mitigate this in two ways. First, we report WCR alongside BWT and AvgF, so that any single-domain alarm can be cross-checked against stream-averaged behavior. Second, we pair WCR with FWT: low WCR indicates genuine retention only when paired with non-trivial plasticity, since a model that learns nothing trivially forgets nothing.

4.4 Main Results

End-Task Generalization After CPT. Table 4 reports zero-shot end-task performance after the full CPT stream. Among the eight trained baselines and PICO, PICO ranks first in 13 of 20 domain–language–metric combinations and within the top two in 15 of 20. The strongest pattern appears on the English evaluations, where PICO is the best trained method in all ten Medical–English and Legal–English cells. Performance is less uniform on the Korean evaluations. PICO remains strong on Medical–Korean, but Legal–Korean is heterogeneous and different baselines lead individual metrics. These results therefore support a strong but non-uniform adaptation advantage rather than uniform superiority in final-stage performance.

The shared initial model remains competitive and exceeds the trained methods in several cells, showing that final-stage scores alone do not fully characterize continual adaptation. We therefore interpret Table 4 together with FWT, BWT, AvgF, and WCR in the following analysis. Detailed per-metric results and evaluation provenance are provided in Appendices K.2 and A.

4.5 Transfer, Forgetting, and Collapse Dynamics

Final end-task scores do not reveal whether a method learned future stages early, preserved prior stages, or merely remained close to the initial model. Figure 2 therefore reports FWT, BWT, and AvgF on ROUGE-L and cosine similarity, while Figure 1 pairs ROUGE-L FWT with the WCR.

Stability–plasticity trade-off. Under the ROUGE-L evaluation, every trained baseline has negative FWT. Adam (−0.0979), Sophia

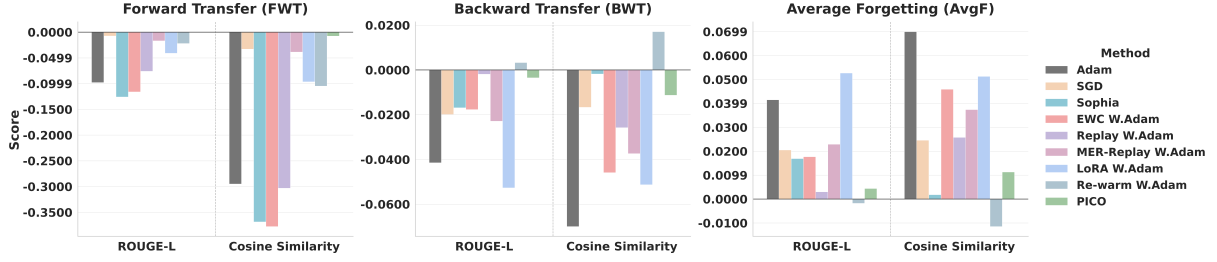


Figure 2: **Continual-learning dynamics across ROUGE-L and cosine similarity.** FWT, BWT, and AvgF are computed from the same evaluation matrix used for the headline results. Higher FWT and BWT are better, while lower AvgF is better. The ranking varies across BWT and AvgF, which motivates pairing stream-averaged statistics with the worst-case WCR analysis in Figure 1.

(-0.1259), and EWC W.Adam (-0.1161) show substantial regression, while replay- and schedule-based methods move closer to the zero-transfer boundary. PICO is the only method with positive FWT, although the gain is deliberately modest (0.0009). The same point also has the lowest WCR (0.0553), placing PICO in the desired low-collapse, non-negative-transfer corner. The closest baseline by WCR is Re-warm W.Adam (0.0901), but its FWT remains negative (-0.0221).

Catastrophic-Forgetting Analysis. The WCR values sharpen the distinction between average retention and worst-domain failure. Adam reaches $WCR = 0.9184$, indicating that its worst learned domain retains only a small fraction of its post-CPT peak. Sophia (0.8434), EWC W.Adam (0.7287), and LoRA W.Adam (0.7114) show the same high-collapse regime despite differing update rules. Replay W.Adam and MER-Replay W.Adam reduce WCR to 0.3181 and 0.2724 , respectively, but both remain on the negative-FWT side. SGD further reduces WCR to 0.2113 with $FWT = -0.0074$. PICO attains the lowest WCR (0.0553) while remaining the only method with positive FWT. Thus, the evidence supports a narrower but stronger claim: PICO does not maximize every final task score, but it gives the best observed joint ROUGE-L trade-off between forward transfer and worst-case collapse on this stream.

Quadrant interpretation. In Figure 1, all trained baselines lie to the left of the $FWT = 0$ boundary. Adam and Sophia combine regression with very high collapse. EWC W.Adam and LoRA W.Adam remain in a similar high-collapse region. Replay W.Adam, MER-Replay W.Adam, and SGD progressively reduce collapse but do not achieve positive transfer. Re-warm W.Adam is the strongest baseline near the lower-left boundary, with FWT

Method	FWT $\uparrow$	BWT $\uparrow$	AvgF $\downarrow$	WCR $\downarrow$
Adam	-0.0604	-0.0197	0.0967	0.8032
PICO	-0.0019	0.0180	-0.0017	0.0607

Table 5: ROUGE-L continual-learning results on the six-stage Spanish–Korean–English curriculum. Adam and PICO use the same curriculum and evaluation protocol.

$= -0.0221$ and $WCR = 0.0901$. PICO sits just across the vertical boundary at $FWT = 0.0009$ with $WCR = 0.0553$. Because the positive FWT margin is small, we interpret this result as preservation of plasticity under strong retention control rather than as a large forward-transfer gain.

Third-language extension. We extend the primary English–Korean curriculum with Spanish Medical and Spanish Legal, forming a six-stage Spanish–Korean–English stream. Relative to the primary four-stage setting, PICO preserves a similarly low-collapse profile, although its ROUGE-L FWT moves slightly below zero. To test whether this behavior can be explained by the extended curriculum alone, we compare PICO with Adam under the same six-stage setting.

Under this matched setting, PICO retains substantially more previously learned capability than Adam while keeping forward transfer close to neutral. Therefore, the comparison provides evidence that the low-collapse behavior is not attributable to the six-stage curriculum alone.

5 Ablation and Scaling Study

We separately analyze PICO’s components and model-scale behavior. All values in Tables 6 and 7 are ROUGE-L-based results.

Component ablation. Table 6 shows that both structural monitoring and stochastic perturbation contribute to PICO’s stability–plasticity behavior.

Variant	FWT $\uparrow$	BWT $\uparrow$	AvgF $\downarrow$	Average $\uparrow$	WCR $\downarrow$
Full PICO	0.0009	-0.0035	0.0044	0.1369	0.0553
w/o SPECFLAG	0.0004	-0.0015	0.0093	0.1315	0.1400
w/o noise	-0.0179	-0.0105	0.0116	0.0921	0.1624
w/o GATEU, SPECFLAG, and noise	-0.0286	-0.0313	0.0313	0.0423	0.4709

Table 6: **Component ablation of PICO on ROUGE-L.** All variants use the 1B backbone and $f = 1$ on the same four-stage CPT stream. Full PICO and the variant without SPECFLAG use $\sigma_0 = 0.001$. The no-noise variant sets $\sigma_0 = 0$ while retaining the remaining update components. Average denotes the mean final-stage ROUGE-L across the four learned domains.

Scale	FWT $\uparrow$	BWT $\uparrow$	AvgF $\downarrow$	Average $\uparrow$	WCR $\downarrow$
1B	0.0002	-0.0026	0.0026	0.1354	0.0450
3B	0.0012	-0.0188	0.0227	0.1299	0.3292
8B	-0.0001	-0.0144	0.0182	0.1397	0.1957

Table 7: **Scaling behavior of PICO on ROUGE-L.** Results are reported for 1B, 3B, and 8B backbones under the same continual pre-training protocol with $\sigma_0 = 0.001$ and spectral-check interval $f = 10$. Average denotes the mean final-stage ROUGE-L across the four learned domains.

Removing SPECFLAG primarily weakens worst-domain retention, whereas disabling stochastic perturbation degrades both transfer and final performance. Since SPECFLAG acts only on the perturbation path, the zero-noise variant effectively retains the utility-gated deterministic update. Removing GATEU in addition to SPECFLAG and noise leads to the largest overall degradation, supporting complementary roles for utility protection and controlled perturbation.

Scaling with model size. Table 7 shows that worst-domain retention is sensitive to model scale even when the spectral-check interval is fixed at $f = 10$. WCR varies non-monotonically with model size, rising from 0.0450 at 1B to 0.3292 at 3B before decreasing to 0.1957 at 8B, while average final-stage ROUGE-L remains comparatively stable. These results therefore do not support scale-invariant retention.

Curriculum Disentanglement. Table 18 shows distinct retention and transfer patterns under domain-only, language-only, and coupled shifts. These differences indicate that curriculum difficulty depends on the type of shift and the evaluation criterion, motivating us to scope our conclusions to the evaluated curricula.

6 Conclusion

This work studies CPT under coupled language-domain shifts from an optimizer-level perspective. We introduced PICO, whose update rule combines coordinate-wise utility gating, periodic matrix-level spectral monitoring, and a gated perturbation path. The end-task evaluation reveals a non-uniform performance profile. PICO is the best trained method in 13 of 20 domain-language-metric combinations and in all ten English combinations. On the Korean evaluations, PICO remains strong on Medical-Korean but does not dominate Legal-Korean. The initial model also remains competitive in several cells, showing that final-stage scores alone are insufficient for judging continual adaptation.

The continual-learning diagnostics provide the clearest evidence for the full PICO configuration, whose reported runs use $\sigma_0 = 0.001$. On ROUGE-L, all compared baselines have negative FWT, whereas PICO has a slightly positive FWT (0.0009) and the lowest WCR (0.0553). We interpret the main result as a favorable observed joint trade-off between transfer and worst-case collapse, rather than as universal superiority or a large forward-transfer gain. The scaling study also shows that retention is sensitive to model scale. Under $f = 10$, WCR rises from 0.0450 at 1B to 0.3292 at 3B before decreasing to 0.1957 at 8B. Component-level causal attribution is reserved for the component-ablation results rather than inferred from unrelated or differently evaluated runs. Overall, these results support optimizer-level control as a promising direction in the evaluated cross-lingual CPT settings, while broader generality, component causality, and scale robustness require further validation.

Limitations

Our experiments are limited to the evaluated language-domain curricula and model scales, with the largest model containing 8B parameters. The

scaling results indicate that worst-domain retention is sensitive to model scale. Under $f = 10$, WCR varies from 0.0450 at 1B to 0.3292 at 3B and 0.1957 at 8B, while FWT remains close to zero across the three scales. The non-monotonic behavior indicates that the current results do not establish scale-invariant retention. Extending PICO beyond 8B will require additional validation of its spectral-monitoring behavior and computational characteristics under larger and more heavily distributed training configurations. In particular, spectral statistics must be computed reliably when model parameters are partitioned across devices.

Our empirical evaluation primarily studies PICO as a standalone optimizer-level intervention. Although the comparison includes replay, MER-style replay, re-warming, EWC, LoRA, and standard optimizers, we do not systematically evaluate their combinations with PICO. Because these methods operate at different intervention levels, their interactions with PICO have not been empirically established. Future work could examine whether combining optimizer-level control with replay or parameter-efficient adaptation further improves the stability-plasticity trade-off.

Finally, PICO introduces additional computational and memory costs through periodic spectral monitoring and optimizer state. In our 1B profiling setting, PICO incurs a wall-clock overhead of 34.5% with $f = 1$ and 20.0% with $f = 10$ relative to AdamW. This overhead may be important in resource-constrained or large-scale training settings. Future implementations could investigate more efficient spectral estimation, less frequent monitoring, or optimized kernels to reduce this cost.

Ethical Considerations

This work studies CPT and proposes an optimization method for analyzing stability, plasticity, and interference in LMs. We present PICO as a research prototype rather than a production-ready system. Reported results are also contingent on the specific evaluation protocol used in our experiments.

Our experiments do not involve human subjects, user interaction, or participant recruitment. The training corpora were curated from publicly documented upstream resources under their applicable access and usage terms. Public accessibility does not imply unrestricted permission to redistribute or deploy the text. The artifact release follows source-

specific licenses and terms, and raw text is included only where redistribution is permitted. Legal and medical corpora may contain sensitive content, names, or other identifying details. We manually reviewed the source corpora for obvious personally identifying information, but our audit is not a complete person-level privacy review. CPT may alter model behavior in ways not fully captured by aggregate evaluation metrics. Sequential updates can amplify dataset biases, affect calibration, or increase the likelihood of plausible but incorrect generations, particularly in high-stakes or specialized domains. Models adapted to specific domains may also be misused outside their intended scope. Accordingly, downstream applications should incorporate domain-specific validation, monitoring, and risk mitigation strategies.

We used AI assistants in a limited capacity for code refactoring and language editing (*e.g.*, formatting and grammatical corrections). All methodological choices, implementations, and experimental analyses were made by the authors. All AI-assisted content was carefully reviewed and revised. No AI system is listed as a co-author. We disclose the full scope of AI assistance in the Responsible Checklist and adhere to the ACL Code of Ethics.

Acknowledgments

This research used the AI Hub datasets *Medical and Legal Professional Books Corpus* and *Professional Domain Corpus*. This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT)(RS-2025-22932973), the IITP(Institute of Information & Communications Technology Planning & Evaluation)-ITRC(Information Technology Research Center) grant funded by the Korea government(Ministry of Science and ICT)(IITP-2026-RS-2021-II211835), and the Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT)(No.RS2026-25507979, AI Data Training Verification and Reliability Assurance Technology). We also appreciate the high-performance GPU computing support of HPC-AI Open Infrastructure via GIST SCENT.

References

Istabrak Abbas, Gopeshh Subbaraj, Matthew Riemer, Nizar Islah, Tsuguchika Tabaru, Hiroaki Kingetsu,

- Sarath Chandar, and Irina Rish. 2026. [Revisiting replay and gradient alignment for continual pre-training of large language models](#). In *Proceedings of The 4th Conference on Lifelong Learning Agents*, volume 330 of *Proceedings of Machine Learning Research*, pages 465–486. PMLR.
- AI Hub. 2020. [Professional domain corpus \(전문분야 말뭉치\)](#). AI Hub. Dataset No. 110, Version 1.1, Accessed 2025-08-28.
- AI Hub. 2022. [Medical and legal professional books corpus \(의료, 법률 전문 서적 말뭉치\)](#). AI Hub. Dataset No. 71487, Version 1.2, Accessed 2025-08-28.
- AI Hub. n.d. [AI Hub Data Usage Policy](#). AI Hub. Accessed 2026-08-29. Dataset citation, research-excerpt, and redistribution guidance.
- Satanjeev Banerjee and Alon Lavie. 2005. [METEOR: An automatic metric for MT evaluation with improved correlation with human judgments](#). In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.
- Asma Ben Abacha and Dina Demner-Fushman. 2019. [A question-entailment approach to question answering](#). *BMC bioinformatics*, 20(1):511.
- Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. 2018. [Riemannian walk for incremental learning: Understanding forgetting and intransigence](#). In *Computer Vision – ECCV 2018: 15th European Conference, Munich, Germany, September 8-14, 2018, Proceedings, Part XI*, page 556–572, Berlin, Heidelberg. Springer-Verlag.
- Junying Chen, Zhenyang Cai, Ke Ji, Xidong Wang, Wanlong Liu, Rongsheng Wang, Jianye Hou, and Benyou Wang. 2024. [HuatuoGPT-o1, Towards Medical Complex Reasoning with LLMs](#). *Preprint*, arXiv:2412.18925.
- Xiangning Chen, Chen Liang, Da Huang, Esteban Real, Kaiyuan Wang, Hieu Pham, Xuanyi Dong, Thang Luong, Cho-Jui Hsieh, Yifeng Lu, and Quoc V Le. 2023a. [Symbolic discovery of optimization algorithms](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 49205–49233. Curran Associates, Inc.
- Yihong Chen, Kelly Marchisio, Roberta Raileanu, David Adelani, Pontus Lars Erik Saito Stenetorp, Sebastian Riedel, and Mikel Artetxe. 2023b. [Improving language plasticity via pretraining with active forgetting](#). In *Advances in Neural Information Processing Systems*, volume 36, pages 31543–31557. Curran Associates, Inc.
- Zeming Chen, Alejandro Hernández-Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. 2023c. [Meditron-70b: Scaling medical pretraining for large language models](#). *Preprint*, arXiv:2311.16079.
- ChuGyouk. 2025. [medical-o1-reasoning-SFT-Ko](#). Hugging Face Datasets. Accessed 2025-12-29.
- Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. [Training verifiers to solve math word problems](#). *CoRR*, abs/2110.14168.
- Andrea Cossu, Antonio Carta, Lucia Passaro, Vincenzo Lomonaco, Tinne Tuytelaars, and Davide Bacciu. 2024. [Continual pre-training mitigates forgetting in language and vision](#). *Neural Networks*, 179:106492.
- Matthias De Lange, Gido M van de Ven, and Tinne Tuytelaars. 2023. [Continual evaluation for lifelong learning: Identifying the stability gap](#). In *International Conference on Learning Representations*.
- Shibhansh Dohare, J. Fernando Hernandez-Garcia, Qingfeng Lan, Parash Rahman, A. Rupam Mahmood, and Richard S. Sutton. 2024. [Loss of plasticity in deep continual learning](#). *Nature*, 632(8026):768–774.
- Mohamed Elsayed and A Rupam Mahmood. 2024. [Addressing loss of plasticity and catastrophic forgetting in continual learning](#). In *International Conference on Learning Representations*.
- Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. [The Pile: An 800GB Dataset of Diverse Text for Language Modeling](#). *arXiv preprint arXiv:2101.00027*.
- Iker García-Ferrero, Rodrigo Agerri, Aitziber Atutxa Salazar, Elena Cabrio, Iker de la Iglesia, Alberto Lavelli, Bernardo Magnini, Benjamin Molinet, Johana Ramirez-Romero, German Rigau, Jose Maria Villa-Gonzalez, Serena Villata, and Andrea Zaninello. 2024. [MedMT5: An open-source multilingual text-to-text LLM for the medical domain](#). In *Proceedings of the 2024 Joint International Conference on Computational Linguistics, Language Resources and Evaluation (LREC-COLING 2024)*, pages 11165–11177, Torino, Italia. ELRA and ICCL.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. [The llama 3 herd of models](#). *arXiv preprint arXiv:2407.21783*.
- Jiawei Gu, Zacc Yang, Chuanghao Ding, Rui Zhao, and Fei Tan. 2024. [Cmr scaling law: Predicting critical mixture ratios for continual pre-training of language models](#). In *Proceedings of the 2024 Conference on*

- Empirical Methods in Natural Language Processing*, pages 16143–16162.
- Kshitij Gupta, Benjamin Thérien, Adam Ibrahim, Mats Leon Richter, Quentin Gregory Anthony, Eugene Belilovsky, Irina Rish, and Timothée Lesort. 2023. [Continual Pre-Training of Large Language Models: How to \(re\)warm your model?](#) In *Workshop on Efficient Systems for Foundation Models @ ICML2023*.
- Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A Smith. 2020. [Don’t stop pretraining: Adapt language models to domains and tasks](#). In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.
- Charles R Harris, K Jarrod Millman, Stéfan J Van Der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J Smith, et al. 2020. [Array programming with NumPy](#). *Nature*, 585(7825):357–362.
- Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2021. [Lora: Low-rank adaptation of large language models](#). *arXiv preprint arXiv:2106.09685*.
- Adam Ibrahim, Benjamin Thérien, Kshitij Gupta, Mats Leon Richter, Quentin Gregory Anthony, Eugene Belilovsky, Timothée Lesort, and Irina Rish. 2024. [Simple and scalable strategies to continually pre-train large language models](#). *Transactions on Machine Learning Research*.
- jihye-moon. 2023. [LawQA-Ko](#). Hugging Face Datasets. Repository:jihye-moon/LawQA-Ko, Accessed 2025-02-11.
- Zixuan Ke, Yijia Shao, Haowei Lin, Tatsuya Konishi, Gyuhak Kim, and Bing Liu. 2023. [Continual Pre-training of Language Models](#). In *International Conference on Learning Representations*.
- Ronald Kemker, Marc McClure, Angelina Abitino, Tyler L. Hayes, and Christopher Kanan. 2018. [Measuring catastrophic forgetting in neural networks](#). In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pages 3390 – 3398.
- Diederik P Kingma and Jimmy Ba. 2014. [Adam: A method for stochastic optimization](#). *arXiv preprint arXiv:1412.6980*.
- James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, et al. 2017. [Overcoming catastrophic forgetting in neural networks](#). *Proceedings of the National Academy of Sciences*, 114(13):3521–3526.
- Korea Ministry of Government Legislation. 2025. [National law information center](#). Accessed 2025-02-16.
- Alex Lewandowski, Michał Bortkiewicz, Saurabh Kumar, Andras Gyorgy, Dale Schuurmans, Mateusz Ostaszewski, and Marlos C. Machado. 2025. [Learning Continually by Spectral Regularization](#). In *International Conference on Learning Representations*, volume 2025, pages 34504–34530.
- Hongyu Li, Liang Ding, Meng Fang, and Dacheng Tao. 2024. [Revisiting catastrophic forgetting in large language model tuning](#). In *Findings of the association for computational linguistics: EMNLP 2024*, pages 4297–4308.
- Chin-Yew Lin. 2004. [ROUGE: A package for automatic evaluation of summaries](#). In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.
- Hong Liu, Zhiyuan Li, David Hall, Percy Liang, and Tengyu Ma. 2024. [Sophia: A scalable stochastic second-order optimizer for language model pre-training](#). In *International Conference on Learning Representations*, volume 2024, pages 1621–1650.
- David Lopez-Paz and Marc’Aurelio Ranzato. 2017. [Gradient episodic memory for continual learning](#). In *Proceedings of the 31st International Conference on Neural Information Processing Systems, NIPS’17*, page 6470–6479, Red Hook, NY, USA. Curran Associates Inc.
- Andrea Lösch, Valérie Mapelli, Stelios Piperidis, Andrejs Vasiljevs, Lilli Smal, Thierry Declerck, Eileen Schnur, Khalid Choukri, and Josef van Genabith. 2018. [European language resource coordination: Collecting language resources for public sector multilingual information management](#). In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).
- Ilya Loshchilov and Frank Hutter. 2019. [Decoupled weight decay regularization](#). In *International Conference on Learning Representations*.
- Clare Lyle, Zeyu Zheng, Evgenii Nikishin, Bernardo Avila Pires, Razvan Pascanu, and Will Dabney. 2023. [Understanding plasticity in neural networks](#). In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 23190–23211. PMLR.
- Jun-Yu Ma, Hong Wang, Hao-Xiang Xu, Zhen-Hua Ling, and Jia-Chen Gu. 2024. [Perturbation-restrained sequential model editing](#). *CoRR*, abs/2405.16821.
- Michael McCloskey and Neal J. Cohen. 1989. [Catastrophic interference in connectionist networks: The sequential learning problem](#). In Gordon H. Bower, editor, *Psychology of Learning and Motivation*, volume 24, pages 109–165. Academic Press.
- medalpaca. 2023. [Medical Meadow WikiDoc](#). Hugging Face dataset. Medical question–answer pairs derived from WikiDoc.

- Meta AI. 2024. Llama 3.2: Revolutionizing edge ai and vision with open, customizable models. <https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/>. Accessed 2026-02-11.
- Joel Niklaus. 2023. Legal Case Document Summarization Dataset. https://huggingface.co/datasets/joelniklaus/legal_case_document_summarization. Hugging Face redistribution of the datasets introduced by Shukla et al. (2022), Accessed 2025-02-11.
- Joel Niklaus, Veton Matoshi, Matthias Stürmer, Ilias Chalkidis, and Daniel Ho. 2024. **MultiLegalPile: A 689GB multilingual legal corpus**. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15077–15094, Bangkok, Thailand. Association for Computational Linguistics.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. **Bleu: a method for automatic evaluation of machine translation**. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.
- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. **Pytorch: An imperative style, high-performance deep learning library**. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.
- Matt Post. 2018. **A call for clarity in reporting BLEU scores**. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Brussels, Belgium. Association for Computational Linguistics.
- Yujia Qin, Cheng Qian, Xu Han, Yankai Lin, Huadong Wang, Ruobing Xie, Zhiyuan Liu, Maosong Sun, and Jie Zhou. 2023. **Recyclable Tuning for Continual Pre-training**. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11403–11426, Toronto, Canada. Association for Computational Linguistics.
- Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. **SQuAD: 100,000+ questions for machine comprehension of text**. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.
- Matthew Riemer, Ignacio Cases, Robert Ajemian, Miao Liu, Irina Rish, Yuhai Tu, and Gerald Tesouro. 2019. **Learning to learn without forgetting by maximizing transfer and minimizing interference**. In *International Conference on Learning Representations*.
- Herbert Robbins and Sutton Monro. 1951. **A stochastic approximation method**. *The Annals of Mathematical Statistics*, 22(3):400–407.
- Shekswess. 2024. **medical_gemma_instruct_dataset**. Hugging Face Datasets. Accessed 2025-12-29, Repository: Shekswess/medical_gemma_instruct_dataset.
- Abhay Shukla, Paheli Bhattacharya, Soham Poddar, Rajdeep Mukherjee, Kripabandhu Ghosh, Pawan Goyal, and Saptarshi Ghosh. 2022. **Legal case document summarization: Extractive and abstractive methods and their evaluation**. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1048–1064, Online only. Association for Computational Linguistics.
- SINAI Research Group. 2026. **ALIA Spanish Legal and Administrative for Context Question Answering Corpus**. <https://huggingface.co/datasets/SINAI/ALIA-es-legal-administrative-cqa>.
- Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. 2023. **Orthogonal subspace learning for language model continual learning**. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 10658–10671, Singapore. Association for Computational Linguistics.
- Kai Wu, Boyuan Jiang, Zhengkai Jiang, Qingdong He, Donghao Luo, Shengzhi Wang, Qingwen Liu, and Chengjie Wang. 2024. **Noiseboost: Alleviating hallucination with noise perturbation for multimodal large language models**. *CoRR*, abs/2405.20081.
- Yutao Yang, Jie Zhou, Xuanwen Ding, Tianyu Huai, Shunyu Liu, Qin Chen, Yuan Xie, and Liang He. 2025. **Recent advances of foundation language models-based continual learning: A survey**. *ACM Computing Surveys*, 57(5):1–38.
- Yupeng Zhang, Hongzhi Zhang, Sirui Wang, Wei Wu, and Zhoujun Li. 2022. **Pats: Sensitivity-aware noisy learning for pretrained language models**. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3680–3687.
- Wenzhen Zheng, Wenbo Pan, Xu Xu, Libo Qin, Li Yue, and Ming Zhou. 2024. **Breaking language barriers: Cross-lingual continual pre-training at scale**. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 7725–7738, Miami, Florida, USA. Association for Computational Linguistics.

A Evaluation Protocol and Reproducibility

Validation of the evaluation protocol. During final validation of the evaluation pipeline, we re-audited the generation procedure used for the reported experiments. This audit identified a legacy evaluator code path that could construct the generation input by concatenating the task prompt with its gold target. Because this allowed generation to be conditioned on reference tokens, we discarded outputs produced through this path and reran all affected checkpoint evaluations using a strictly prompt-only generation protocol.

Prompt-only evaluation. All zero-shot results reported in this paper are produced by providing only the task prompt to the model at generation time. Gold targets are kept separate from the model input and are used solely as references for scoring after generation. All metrics are computed on the generated continuation, with no gold-target tokens included in the generation context. This evaluation procedure is applied uniformly to the shared initial model and to all trained methods, and the evaluation resources are held fixed across checkpoints within each comparison.

Impact on the reported results. We reran all checkpoint evaluations affected by the legacy code path and regenerated the corresponding end-task scores, continual-learning summaries, tables, figures, and analyses. Compared with outputs produced by the legacy path, the prompt-only evaluation changes both absolute scores and several method rankings. Under the final end-task evaluation for the primary four-stage CPT stream, PICO ranks first among the trained methods in 13 of 20 domain–language–metric combinations, including all ten English combinations. On Legal–Korean, leadership varies by metric; PICO ranks second on ROUGE-L but does not lead the remaining metrics. On ROUGE-L for the primary four-stage CPT stream, PICO is the only compared trained method with positive forward transfer (FWT; 0.0009) and attains the lowest Worst Collapse Ratio (WCR; 0.0553). Accordingly, all conclusions in this paper are based on the prompt-only evaluation, and we interpret the evidence as a favorable but non-uniform stability–plasticity profile rather than uniform superiority across end-task settings.

Additional baselines and an independent 8B implementation issue. The comparison includes

stronger continual-learning controls, including MER-Replay W.Adam and Re-warm W.Adam, all evaluated under the same prompt-only protocol. Separately, an earlier 8B BLEU run using FSDP exposed an implementation issue in which one-dimensional shard views could cause the spectral monitor to be skipped. This issue is independent of the evaluation-path issue described above. The FSDP issue concerns whether spectral monitoring is executed during training, whereas the evaluation issue concerns the construction of generation inputs and subsequent metric computation. The reported 8B scaling experiments do not use FSDP. This ensures that spectral monitoring operates on the intended two-dimensional parameter matrices.

Reproducibility. All reported zero-shot generation results, and the continual-learning summaries derived from those evaluations, use the prompt-only evaluator. All reported PICO results use the verified implementation corresponding to their stated experiment-specific configurations. Unless explicitly varied in an ablation, PICO uses $\sigma_0 = 0.001$; the spectral-check interval is experiment-specific and follows the configuration reported for the corresponding experiment. Experiment-specific deviations, including the zero-noise ablation, are reported explicitly. The same evaluation procedure is applied to all compared methods without method-specific post-processing. The initial model is reported separately as a shared reference and is excluded from rankings among trained methods. Together with the training and evaluation configurations specified in the experimental setup, these conditions define the implementation and evaluation protocol used to produce the reported results.

B Background

In this section, we summarize the background knowledge for the proposed PICO.

B.1 Continual Pre-training

We study CPT, also referred to as continual domain-adaptive pre-training or continual post-training, where a pretrained LM is further trained on a sequence of (typically unlabeled) domain corpora. Unlike one-shot domain-adaptive pre-training, CPT must continuously balance adaptation to the current domain with *retention* of previously acquired knowledge, thereby confronting CF.

Formally, given an initial model θ_0 and a stream of domain corpora $\{D_1, D_2, \dots, D_T\}$, CPT itera-

tively updates:

$$\theta_i = \text{Update}(\theta_{i-1}, D_i), \quad i = 1, \dots, T, \quad (14)$$

often augmented with additional constraints $R(\theta_i, \theta_{i-1})$ to preserve prior knowledge. Soft-masking mechanisms can explicitly regulate parameter updates to integrate new domain knowledge while mitigating forgetting (Ke et al., 2023). CPT is also studied more broadly as a CL method for foundation models across modalities (Cossu et al., 2024; Yang et al., 2025). Cross-lingual CPT has been studied under language shifts without explicitly coupling them with domain shifts (Zheng et al., 2024). Our study instead combines language and domain shifts within the same continual stream.

B.2 Neural Plasticity in Language Models

Plasticity denotes a model’s capacity to incorporate new information under distributional shifts efficiently. In CL, standard deep learning methods may gradually lose plasticity, eventually learning no better than substantially shallower baselines, highlighting a failure mode distinct from Catastrophic Forgetting (Dohare et al., 2024). Lyle et al. (2023) trace this plasticity loss to changes in the curvature of the loss landscape, motivating optimization-side interventions that monitor structural shifts in the parameter space. For LMs, preserving plasticity is particularly important in CPT because later domains may require non-trivial representational reconfiguration rather than minor refinements.

Recent work further suggests that plasticity can be shaped at *pretraining time*. Active forgetting pretraining periodically resets the token embedding layer, aiming to improve the model’s ability to rapidly relearn lexical representations. Empirically, this yields faster convergence during language adaptation and improved performance in low-data regimes (Chen et al., 2023b). These findings motivate designing CPT procedures that maintain plasticity while preventing interference.

B.3 Perturbation noise in Language Models

Perturbation/noise-based regularization has been widely explored for LMs to improve robustness or stabilize sequential updates in settings such as fine-tuning, CL, and model editing. PATS injects larger noise into lower-sensitivity parameters and smaller noise into higher-sensitivity ones to better utilize redundant capacity while minimally disturbing critical parameters (Zhang et al., 2022).

NoiseBoost introduces noise feature perturbations for multimodal LLM training as a regularizer that encourages more balanced attention between visual and linguistic tokens, thereby mitigating hallucinations and improving data efficiency (Wu et al., 2024). In sequential model editing, PRUNE analyzes the degradation of general abilities through the edited matrix’s condition number and constrains the upper bound of harmful perturbations across edits (Ma et al., 2024).

Importantly, recent surveys of foundation-LM continual learning emphasize approaches based on replay, regularization, and parameter-efficient adaptation, while explicit parameter-space perturbation noise is not treated as a major standalone category (Yang et al., 2025). Perturbation-based control is conceptually distinct from parameter-efficient adaptation such as LoRA, which freezes pretrained weights and trains low-rank update matrices (Hu et al., 2021). This motivates investigating whether carefully controlled perturbations can also be beneficial in CPT alongside these established strategies.

B.4 Exponential Moving Average (EMA)

An exponential moving average summarizes a sequence of scalar or tensor observations by assigning greater weight to recent values. PICO uses EMA operations to smooth coordinate-wise utility estimates and per-matrix spectral statistics. It does not maintain an EMA copy of the language-model parameters. Exact recursions are given in Appendices E and F.

C Practical Challenges in CPT

This appendix complements Section 3 by summarizing recurring practical challenges in CPT and relating them to representative CL strategies.

Challenge 1: Curating realistic training streams. CPT critically depends on the availability and ordering of domain corpora. Training outcomes can be highly sensitive to stream composition, domain boundaries, and transition severity, making it difficult to construct training streams that faithfully reflect realistic distribution shifts while remaining reproducible and method-agnostic.

Challenge 2: Curating reliable evaluation datasets. Beyond training data, CPT performance is strongly influenced by the choice of evaluation datasets. Inadequate separation between train-

ing and evaluation domains, limited domain coverage, or overly narrow benchmarks can obscure transfer and forgetting effects, hindering fair and consistent comparison across CPT methods.

Challenge 3: Preserving plasticity under continued optimization. Models can suffer from *loss of plasticity*, whereby the ability to acquire new knowledge degrades with prolonged training (Do-hare et al., 2024). Pretraining-time interventions can influence later adaptability. Active forgetting via periodic embedding resets has been shown to improve adaptation efficiency in low-data regimes (Chen et al., 2023b).

Challenge 4: Mitigating Catastrophic Forgetting without over-specialization. Sequential domain adaptation can overwrite representations that remain useful for earlier domains. Update-control mechanisms such as soft-masking explicitly regulate parameter updates to mitigate forgetting during continual domain-adaptive pretraining (Ke et al., 2023). Parameter-efficient adaptation (e.g., LoRA-style low-rank modules) can further reduce interference by limiting the extent to which the backbone weights change (Wang et al., 2023). From a broader foundation-LM CL perspective, forgetting is addressed through a range of strategies including replay, regularization, parameter isolation, and parameter-efficient adaptation (Yang et al., 2025).

Implication. These challenges motivate optimizer-centric mechanisms that both preserve high-utility parameters and remain adaptive to structural changes during training. PICO addresses Challenges 3 and 4 by preserving informative coordinates while suppressing the stochastic perturbation path when periodic spectral monitoring raises a structural flag (Section 3).

D Details of the Preliminary Study

This section provides the formal setup and extended diagnostics for the preliminary diagnosis. It supports the claims in Section 2 with detailed visual and tabular evidence.

D.1 Formal Stream Setup and Optimizers

We model cross-lingual CPT as a two-stage stream ($T = 2$): Korean Medical followed by English Medical. All baseline optimizers (SGD, Adam, AdamW, Lion (Chen et al., 2023a), Sophia (Liu et al., 2024)) share the same Llama-3.2-1B-Instruct

backbone. Each is tuned with a stable base learning rate and a 5% warmup cosine schedule. Only the optimizer differs across runs, ensuring a fair comparison.

D.2 Visualizing the Stability–Plasticity Tradeoff

To compare how optimizers navigate the stability–plasticity dilemma under cross-lingual shifts, we map them onto a 2D plane. The horizontal axis is Stage-1 plasticity, measured as the BLEU gain on Korean Medical after Stage 1 training. The vertical axis is Backward Transfer (BWT) on BLEU, where higher values indicate greater stability.

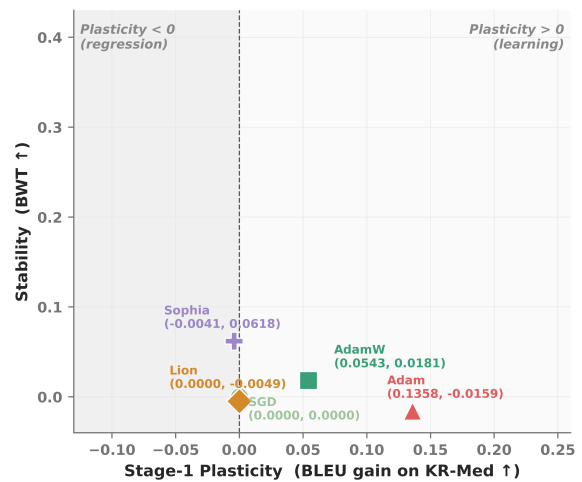


Figure 3: Stability–plasticity tradeoff among baseline optimizers on the two-stage KR-Med $\rightarrow$ EN-Med stream. Horizontal axis: Stage-1 plasticity (BLEU gain on KR-Med). Vertical axis: BWT on BLEU; higher means more stable. Coordinates are shown next to each marker. SGD and Lion sit near the origin (inert). Sophia shows slight regression yet the highest stability. Adam attains the strongest plasticity but negative BWT, the signature of *Catastrophic Forgetting*. AdamW alone shows positive plasticity with positive BWT, though well below Adam’s plasticity. No baseline reaches the upper-right region combining strong plasticity with strong stability, motivating direct optimizer-level control.

Figure 3 reveals that baselines cluster into qualitatively distinct failure regimes. They do not slide along a single tradeoff curve.

Inert regime (SGD, Lion). SGD exhibits near-zero task-level plasticity and zero BWT. Thus, there is little newly acquired capability in Stage 1 that could subsequently be forgotten. Lion behaves similarly, with plasticity ≈ 0 and BWT = -0.0049 . Both optimizers therefore exhibit apparent stability together with little effective adaptation in this

setting.

Catastrophic Forgetting (Adam). Adam is the only baseline with substantial plasticity (+0.1358), but it pairs this gain with negative BWT. After Stage 2, BLEU on Korean Medical drops below its post-Stage-1 peak. This pattern indicates that the capability acquired in Stage 1 is not fully retained after adaptation to Stage 2.

Regression with stability (Sophia). Sophia’s Stage-1 plasticity is slightly negative (−0.0041), meaning its Korean Medical BLEU *worsened* during Stage 1 training. Its high BWT (+0.0618) is consistent with partial recovery during Stage 2 rather than straightforward retention. In this run, Sophia’s Hessian-based preconditioning does not yield effective adaptation to the new language-domain.

Modest middle ground (AdamW). AdamW alone shows positive plasticity (+0.0543) together with positive BWT (+0.0181). However, its plasticity is roughly one-third of Adam’s, indicating more limited adaptation to Korean Medical in this setting. No baseline reaches the upper-right region of the plane, where both axes would be substantially positive. This visualization provides empirical support for the pattern discussed in Section 2. Among the evaluated standard optimizers, none jointly achieves strong adaptation and strong retention under this cross-lingual shift. The baselines exhibit different transfer–retention trade-offs, motivating the optimizer-level control mechanism in PICO.

D.3 Extended Diagnosis: Loss vs. Generation Discrepancy

In Section 2, we argued that loss-level monitoring alone fails to detect collapse during cross-lingual shifts. Table 8 demonstrates this discrepancy concretely using Adam, the optimizer with the most severe forgetting in Figure 3.

The disagreement indicates a failure mode that is not captured by cross-entropy alone. After Stage 2, the model achieves its *lowest* cross-entropy on Korean Medical, while BLEU drops below its post-Stage-1 peak. This observation motivates examining matrix-level spectral statistics as complementary training signals rather than inferring structural collapse directly from loss–generation disagreement.

Stage	Korean Medical (Stage 1)		English Medical (Stage 2)	
	Cross-Entropy ↓	BLEU ↑	Cross-Entropy ↓	BLEU ↑
Initial Model	3.5873	0.0304	3.7910	0.0314
After Stage 1 (KR)	3.4363	0.1526	3.6990	0.0414
After Stage 2 (EN)	3.3381	0.1056	3.7404	0.0173

Table 8: Discrepancy between cross-entropy loss and BLEU for Adam on the KR-Med → EN-Med stream. On Korean Medical, the loss decreases monotonically across all three checkpoints (3.5873 → 3.4363 → 3.3381), suggesting continued improvement. BLEU on Korean Medical tells the opposite story: it rises to 0.1526 after Stage 1 but drops to 0.1056 after Stage 2. Loss-level monitoring would conclude that Stage 2 training helped the Korean Medical domain, whereas generation quality reveals that capability was overwritten. Bold values indicate the best score per metric and column.

E Utility Estimator: Detailed Recursions

Utility-gated updates. PICO adapts the utility-gated mechanism of UPGD (Elsayed and Mahmood, 2024) to continual pre-training. For each parameter tensor θ , let n_θ denote the number of utility updates applied to that tensor. When every trainable parameter receives a gradient at every optimizer step, n_θ coincides with the global optimizer-step index. The main text omits the tensor subscript and local update counter for readability; for a tensor updated at global step k , $u^{(k)}$ and $\tilde{g}^{(k)}$ denote the corresponding tensor-local quantities.

Utility signal and smoothing. At the n_θ -th update of tensor θ , PICO computes

$$\text{inst}U_\theta^{(n_\theta)} = -g_\theta^{(n_\theta)} \odot \theta^{(n_\theta)}, \quad (15)$$

$$\text{avg}U_\theta^{(n_\theta)} = \beta_u \text{avg}U_\theta^{(n_\theta-1)} + (1 - \beta_u) \text{inst}U_\theta^{(n_\theta)}. \quad (16)$$

The implementation applies the stabilized bias correction

$$\widehat{\text{avg}U}_\theta^{(n_\theta)} = \frac{\text{avg}U_\theta^{(n_\theta)}}{1 - \beta_u^{n_\theta} + \varepsilon}. \quad (17)$$

Gate and protected gradient. The per-tensor normalization factor is

$$u_{\max, \theta}^{(n_\theta)} = \left\| \text{avg}U_\theta^{(n_\theta)} \right\|_\infty + \varepsilon. \quad (18)$$

The utility gate is

$$u_\theta^{(n_\theta)} = \text{sigmoid} \left(\frac{\widehat{\text{avg}U}_\theta^{(n_\theta)}}{u_{\max, \theta}^{(n_\theta)}} \right), \quad (19)$$

and the protected gradient is

$$\tilde{g}_\theta^{(n_\theta)} = g_\theta^{(n_\theta)} \odot \left(1 - u_\theta^{(n_\theta)} \right). \quad (20)$$

Algorithm 2 GATEU

Require: Parameter tensor θ , gradient g , state (avgU, n_θ)
Hyperparameters β_u, ϵ
Ensure: Utility gate u and protected gradient $\tilde{g}$

- 1: $n_\theta \leftarrow n_\theta + 1$
- 2: Compute instU and update avgU ▷ Eqs. (15)–(16)
- 3: Apply stabilized bias correction ▷ Eq. (17)
- 4: Compute $u_{\max}$ and u ▷ Eqs. (18)–(19)
- 5: Compute $\tilde{g}$ ▷ Eq. (20)
- 6: **return** ($u, \tilde{g}$)

Algorithm 3 SPECFLAG

Require: Parameter group $\mathcal{G}_j$, global optimizer step k , check period f
Hyperparameters $K, \beta_u, c_0, r, \tau_{\text{start}}, \tau_{\text{end}}, \epsilon$
Per-matrix state: $c_W, \mu_x, \nu_x, \mathbf{v}_W$, and specSens[W]
Ensure: Group-level perturbation-pause $\pi_{\mathcal{G}_j}^{(k)} \in \{0, 1\}$

- 1: $\pi_{\mathcal{G}_j}^{(k)} \leftarrow 0$
- 2: **if** $k \bmod f \neq 0$ **then**
- 3: **return** $\pi_{\mathcal{G}_j}^{(k)}$ ▷ Non-check step
- 4: **end if**
- 5: **for all** matrix $W \in \mathcal{G}_j$ with an available gradient **do**
- 6: Estimate the current $\hat{\sigma}_1(W)$ and $\hat{\rho}_\epsilon(W)$ ▷ Eq. (4)
- 7: Snapshot previous μ_x and ν_x
- 8: Compute the two signed deviations from the previous EMA statistics ▷ Eqs. (25)–(26)
- 9: $c_W \leftarrow c_W + 1$ ▷ Current local observation index
- 10: Evaluate $s_W^{(c_W)}$ using $\tau(c_W)$ ▷ Eqs. (27)–(28)
- 11: Update $\mu_x^{(c_W)}$ and $\nu_x^{(c_W)}$ ▷ Eqs. (23)–(24)
- 12: specSens[W] $\leftarrow s_W^{(c_W)}$ ▷ Eq. (29)
- 13: **if** $s_W^{(c_W)} = 1$ **then**
- 14: $\pi_{\mathcal{G}_j}^{(k)} \leftarrow 1$ ▷ Pause group perturbation on this check step
- 15: **end if**
- 16: **end for**
- 17: **return** $\pi_{\mathcal{G}_j}^{(k)}$

F Spectral Sensitivity: Detailed Recursions

This appendix specifies the per-matrix spectral statistics, standardized deviation scores, threshold schedule, and cached structural flags used by Algorithm 3. For each monitored matrix W , let c_W denote its local spectral-observation counter. The counter is incremented only when W is actually evaluated by SPECFLAG; a scheduled check step does not increment c_W when W has no available gradient. Each matrix also maintains a warm-started power-iteration vector and cached structural state.

EMA statistics under periodic observations. Spectral quantities are observed only on scheduled check steps. To preserve the decay associated with f optimizer steps, PICO uses

$$\beta_{\text{eff}} = \beta_u^f, \quad \bar{\beta}_{\text{eff}} = 1 - \beta_{\text{eff}}. \quad (21)$$

Fix a monitored matrix W and let

$$x^{(c_W)} \in \left\{ \hat{\sigma}_1^{(c_W)}(W), \hat{\rho}_\epsilon^{(c_W)}(W) \right\}. \quad (22)$$

For readability, the matrix index W is omitted from μ_x and ν_x below. The implementation first computes the standardized deviation from the *previous* statistics, then updates the EMA state using the current observation. The EMA recursions are

$$\mu_x^{(c_W)} = \beta_{\text{eff}} \mu_x^{(c_W-1)} + \bar{\beta}_{\text{eff}} x^{(c_W)}, \quad (23)$$

$$\nu_x^{(c_W)} = \beta_{\text{eff}} \nu_x^{(c_W-1)} + \bar{\beta}_{\text{eff}} \left(x^{(c_W)} - \mu_x^{(c_W-1)} \right)^2. \quad (24)$$

The quantity $\nu_x^{(c_W)}$ is an exponentially weighted squared-deviation estimate relative to the previous mean and is used as a running variance proxy; it is not an uncentered second raw moment.

Stabilized standardized deviation. Before incorporating the current observation into the EMA statistics, PICO computes

$$\tilde{\nu}_x^{(c_W-1)} = \max \left\{ \nu_x^{(c_W-1)}, \epsilon \right\}, \quad (25)$$

$$z_x^{(c_W)} = \frac{x^{(c_W)} - \mu_x^{(c_W-1)}}{\sqrt{\tilde{\nu}_x^{(c_W-1)} + \epsilon}}. \quad (26)$$

Because the decision rule thresholds $z_x^{(c_W)}$ rather than $|z_x^{(c_W)}|$, only positive deviations can activate a structural flag.

Annealed threshold. Flagging is disabled while $c_W < c_0$ and becomes active at $c_W = c_0$. The threshold $\tau(c_W)$ is

$$\tau_{\text{start}} + \frac{\tau_{\text{end}} - \tau_{\text{start}}}{r} \min \{ \max(c_W - c_0, 0), r \}. \quad (27)$$

For $r \leq 0$, the implementation uses τ_{end} directly.

Per-matrix flag and cached state. With $z_{\hat{\sigma}_1}^{(c_W)}$ and $z_{\hat{\rho}_\epsilon}^{(c_W)}$ denoting the two signed deviations, the matrix-level flag $s_W^{(c_W)}$ is

$$\begin{cases} 0, & c_W < c_0, \\ \mathbb{I} \left[\max \left\{ z_{\hat{\sigma}_1}^{(c_W)}, z_{\hat{\rho}_\epsilon}^{(c_W)} \right\} > \tau(c_W) \right], & c_W \geq c_0. \end{cases} \quad (28)$$

After evaluating the flag, PICO updates

$$\text{specSens}[W] \leftarrow s_W^{(c_W)}. \quad (29)$$

The cached matrix value remains unchanged until W is evaluated again. For $W = \theta$, this is the

tensor-level state written as `specSens[ $\theta$ ]` in Algorithm 1 and Table 22. On non-check steps, the transient group-level pause is reset to zero, while the cached matrix-level flag can still suppress perturbation for that matrix. Neither mechanism suppresses the protected-gradient update or decoupled weight decay.

G Compute and Memory Overhead Details

Profiling protocol. We conduct paired single-GPU profiling on Llama-3.2-1B-Instruct using batches drawn from the English legal CPT corpus. All configurations use the same batch stream, bf16 model parameters, gradient checkpointing, batch size 32, sequence length 256, learning rate 2×10^{-5} , and weight decay 0.01. We use 400 warm-up updates followed by 400 measured updates and repeat each configuration in three fresh processes. The 400-update warm-up corresponds to 400 spectral observations at $f = 1$ and 40 observations at $f = 10$, covering the full burn-in and threshold-annealing period ($c_0 = 10$, $r = 30$) in both configurations.

Host wall-clock time is measured after CUDA synchronization and includes host-to-device transfer, forward and backward passes, the optimizer update, and gradient reset; data-loader iteration is excluded. CUDA-event timings are retained only as a diagnostic. Per-matrix detailed diagnostic logging is disabled.

Runtime. Table 9 reports the complete results. The main $f = 1$ configuration incurs +34.5% wall-clock overhead over AdamW. Checking once every ten updates reduces this to +20.0%. The latter reduction reflects amortization of the spectral computation; it does not change persistent optimizer state.

Persistent optimizer state. Under the profiled bf16 configuration, AdamW maintains two bf16 moment tensors and uses 4.604 GiB of persistent optimizer state. The reference PICO implementation instead maintains a full-size fp32 utility EMA and a full-size fp32 master-parameter copy, together with comparatively small spectral-monitoring states, for a total of 9.209 GiB. Thus, PICO uses 4.605 GiB more persistent optimizer state than the profiled AdamW configuration. Its measured peak allocated and peak reserved CUDA memory increase by 4.604 GiB and 4.650 GiB, re-

spectively. The optimizer-state dimensionality is independent of the number of CPT stages and is unchanged between $f = 1$ and $f = 10$.

Comparison to regularization-based methods. PICO’s persistent optimizer state is independent of the number of CPT stages. This differs from task-wise EWC formulations that may retain task-specific regularization information as the number of tasks grows. Online EWC consolidates past information into a fixed set of statistics and therefore also avoids linear growth with the number of tasks. We consequently characterize PICO as having stage-independent state rather than claiming a smaller absolute memory footprint than all EWC variants.

H Experimental Setup

H.1 Dataset Details, Provenance, Licensing, and Corpus Examples

The primary CPT stream follows Korean Medical, English Medical, Korean Legal, and English Legal. The four CPT corpus files are distinct from the downstream evaluation datasets listed in Table 2. The statistics below describe the files used for CPT rather than the sizes of their upstream source collections.

Record unit and duplicate audit. A record is a nonempty physical line after corpus construction and may contain a sentence, paragraph, document fragment, or longer structured segment. Table 10 reports file-level statistics for the audited CPT inputs. The exact-duplicate rate counts repeated decoded nonempty records and does not detect paraphrastic or semantic duplicates.

Corpus composition. The corpora differ substantially in record granularity and duplication. The Korean Medical file contains heterogeneous technical medical text, including professional reference prose and substantial patent-style material. For example, `본 발명` occurs in 14,612 of its 37,505 records. This heterogeneity is consistent with its two AI Hub sources: one contains medical professional-book and patent material, while the other contains broader professional-domain material including patents and academic text. The Korean Medical corpus should not be interpreted as purely clinical prose. The remaining corpora contain nonzero exact duplication as reported in Table 10. These corpus characteristics may affect adaptation and transfer.

Method	f	ms/update	tok/s	Peak alloc. (GiB)	Peak resv. (GiB)	Slowdown
AdamW	–	633.83	15348	23.936	26.643	–
PICO	1	852.29	11414	28.540	31.293	+34.5%
PICO	10	760.53	12791	28.540	31.293	+20.0%

Table 9: Paired runtime and CUDA-memory profiling. Times are host wall-clock means over three trials. PICO uses one power-iteration step ($K = 1$).

Corpus	Records	Size (MB)	Characters	Exact Duplicate Rate (%)
Korean Medical	37,505	367.81	169,790,973	0.00
English Medical	2,205,836	460.09	456,684,468	27.25
Korean Legal	140,053	457.84	193,544,936	7.50
English Legal	5,466,261	421.51	412,191,429	18.23

Table 10: Statistics for the four primary CPT corpus files. Size uses decimal MB, and exact-duplicate rates count repeated nonempty records.

H.1.1 Source Provenance, Licensing, and Redistribution

Source-to-corpus provenance. The CPT files are derived research corpora constructed from upstream legal and medical resources rather than newly authored raw text. English Medical is derived from the Clinical Guidelines corpus released with Meditron (Chen et al., 2023c); English Legal from the FreeLaw component of The Pile (Gao et al., 2020); and Korean Legal from the Korean National Law Information Center (Korea Ministry of Government Legislation, 2025).

Korean Medical combines two AI Hub resources. The first is the medical portion of the “Medical and Legal Professional Books Corpus” (의료, 법률 전문 서적 말뭉치), dataset 71487, version 1.2 (AI Hub, 2022). The second consists of medical-related records selected from the “Professional Domain Corpus” (전문분야 말뭉치), dataset 110, version 1.1 (AI Hub, 2020). We use the term “medical-related records” because AI Hub does not define a separate medical split for dataset 110 on its public dataset page. The resulting Korean Medical CPT file contains 37,505 records and does not reproduce either complete upstream AI Hub dataset.

Spanish extension corpora. The six-stage extension additionally uses Spanish Medical and Spanish Legal CPT corpora, each constructed to approximately 150 MB. Spanish Medical is derived from the Spanish medical data in HiTZ/Multilingual-Medical-Corpus and contains 46,403 nonempty records. Spanish Legal is constructed from the Spanish legal configurations of Multi_Legal_Pile and contains 16,260

nonempty records. The final constructed files contain no exact duplicate records under the same record-level audit used for the primary corpora.

Licensing and release policy. Public accessibility is not treated as permission for unrestricted redistribution, and we do not assign a single blanket license to the four source-derived CPT corpora. Table 12 summarizes the applicable source terms and release treatment for each corpus. No new license is asserted over upstream text beyond the rights provided by the corresponding source.

Both AI Hub resources used for Korean Medical are governed by the applicable AI Hub data-use conditions. Research results and minimal original-text examples may be published for research documentation, whereas external disclosure or redistribution of the original or simply processed data is restricted (AI Hub, n.d.). Accordingly, Korean Medical source-derived text is not redistributed. The required AI Hub attribution is provided in the Acknowledgments.

Representative corpus examples. Table 13 provides minimal illustrative excerpts from the four CPT files. The excerpts are truncated for readability and are neither random samples nor complete summaries of corpus composition. Their inclusion does not imply permission to redistribute the corresponding corpora.

CPT corpus	Upstream source	License / applicable terms	Construction and release treatment
Spanish Medical	HiTZ Multilingual Medical Corpus (García-Ferrero et al., 2024)	The dataset repository declares Apache-2.0. The corpus aggregates multiple upstream medical-text sources; applicable source-specific terms remain relevant to the underlying material.	Spanish records are filtered, packed into document-sized segments, and exact-deduplicated. The resulting 150.00 MB corpus contains 46,403 records. Source-derived text is redistributed only where permitted by the applicable upstream terms.
Spanish Legal	MultiLegalPile (Niklaus et al., 2024)	The dataset repository declares CC BY-NC-SA 4.0. MultiLegalPile aggregates legal resources with source-specific licensing and reuse conditions.	Spanish legal data are filtered and exact-deduplicated during corpus construction. The resulting 150.00 MB corpus contains 16,260 records. Redistribution remains subject to both the dataset-level and applicable source-specific terms.

Table 11: Source provenance, applicable terms, and construction details for the Spanish CPT corpora used only in the six-stage extension.

H.2 Evaluation Dataset Splits, Statistics, and Terms

The downstream evaluation resources are external to the CPT training corpora and are used only for evaluation. Table 14 reports the Hugging Face repository, the source split used by the reported evaluation, the size of that source split, and the number of examples retained by the evaluation pipeline. The evaluated count can be smaller than the source-split size because the evaluator applies its deterministic validity and input-budget criteria before scoring. The resulting evaluation set is held fixed across checkpoints within each comparison.

Several evaluation resources expose only a training partition on Hugging Face. In these cases, “train” denotes the upstream repository’s partition name; these examples are not part of the CPT training corpora in Table 3.

Spanish extension evaluation. The six-stage extension uses two additional evaluation resources, documented separately from the primary evaluation datasets above. For Medical–Spanish, we use ELRC-Medical-V2 (Lösch et al., 2018) with the English–Spanish configuration and evaluate English-to-Spanish medical translation. For Legal–Spanish, we use the ALIA Spanish Legal and Administrative CQA corpus (SINAI Research Group, 2026) with the `parlamint_es_an` split for Spanish legal-administrative question answering. The questions and reference answers in the ALIA resource are synthetically generated from official parliamentary records, as documented on the dataset card.

H.3 Baseline Implementation Details

This appendix documents the implementation of each baseline summarized in Section 4.2. All methods share the backbone, primary CPT stream, tokenizer, and sequence length (256). Baseline-specific data composition, schedules, parameterization, and optimizer state differ only where required by the method definition. Unless otherwise stated, optimizer state is preserved across CPT stages, no learning-rate re-warming or re-decaying is applied, and gradient clipping is disabled.

H.3.1 First-Order Optimizers (Adam, SGD)

Adam. A standard optimizer for LM CPT (Kingma and Ba, 2014). We use $\eta = 2 \times 10^{-5}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\varepsilon = 10^{-8}$, and no decoupled weight decay. Adam represents a passive optimizer backend with no explicit forgetting-mitigation mechanism.

SGD. A momentum-free reference. We use plain SGD with $\eta = 2 \times 10^{-5}$ and no momentum, weight decay, or Nesterov correction. SGD removes all adaptive-moment machinery, providing a clean lower bound on optimizer-induced plasticity.

H.3.2 Second-Order Approximation (Sophia)

Sophia exploits diagonal Hessian information for adaptive preconditioning (Liu et al., 2024). We use $\eta = 2 \times 10^{-5}$, $\beta_1 = 0.965$, $\beta_2 = 0.99$, $\rho = 0.05$ (Hessian clipping), and a Hessian-update interval of 10 steps and $bs = 5120$ as default. The Hessian estimator uses the Gauss–Newton–Bartlett variant following the official implementation. Sophia tests

CPT corpus	Upstream source	License / applicable terms	Artifact release treatment
English Medical	EPFL Clinical Guidelines (Chen et al., 2023c)	The public Clinical Guidelines dataset identifies the Common Crawl Foundation Terms of Use. Its released subset contains sources that the dataset curators identify as permitting content redistribution.	Source-derived text is released only where the applicable upstream terms permit redistribution.
English Legal	The Pile FreeLaw, derived from CourtListener court opinions (Gao et al., 2020)	The Pile contains source-specific licensing conditions and lists the FreeLaw subset license as unspecified.	No broader license is asserted over the underlying text than its applicable upstream status.
Korean Legal	Korean National Law Information Center, Ministry of Government Legislation (Korea Ministry of Government Legislation, 2025)	Reuse depends on the legal status of the specific source material. Applicable official legal information may be reused under Korean copyright exclusions or public-use provisions. Third-party rights are not implicitly covered.	Only material whose verified legal status permits reuse is eligible for redistribution.
Korean Medical	AI Hub dataset 71487, version 1.2: “Medical and Legal Professional Books Corpus” (의료, 법률 전문 서적 말뭉치) (AI Hub, 2022); AI Hub dataset 110, version 1.1: “Professional Domain Corpus” (전문분야 말뭉치) (AI Hub, 2020)	Both sources are governed by applicable AI Hub data-use conditions. Research results and minimal original-text examples may be published, whereas the original or simply processed datasets may not be externally shared or redistributed (AI Hub, n.d.).	The Korean Medical source-derived text is not redistributed. Readers obtain the upstream datasets directly through AI Hub.

Table 12: Source provenance, applicable terms, and release treatment for the four primary CPT corpora. Korean Medical combines material derived from two AI Hub datasets. Source identification and redistribution permission are treated separately.

whether second-order curvature signals alone improve cross-lingual retention.

H.3.3 Regularization (EWC W.Adam)

We use Elastic Weight Consolidation (EWC) (Kirkpatrick et al., 2017). After each completed stage t , we estimate a diagonal Fisher information matrix $F^{(t)}$ by averaging squared gradients of the language-model loss over up to 200 mini-batches from $\mathcal{D}_t$, and store the corresponding parameter snapshot $\theta^{*,(t)}$.

During each subsequent stage s , we augment the language-model objective with the accumulated EWC penalties from all previously completed stages:

$$\mathcal{L}_{\text{EWC}}(\theta) = \mathcal{L}_{\text{LM}}(\theta) + \mathcal{L}_{\text{reg}}(\theta). \quad (30)$$

$$\mathcal{L}_{\text{reg}}(\theta) = \frac{\lambda_{\text{EWC}}}{2} \sum_{t < s} \sum_i F_i^{(t)} \left(\theta_i - \theta_i^{*,(t)} \right)^2. \quad (31)$$

We optimize the resulting objective with standard Adam using $\eta = 2 \times 10^{-5}$, weight decay 0.01, and $\lambda_{\text{EWC}} = 10$. Fisher diagonals and parameter snapshots are stored on CPU to reduce persistent GPU-memory overhead.

H.3.4 Replay (Replay W.Adam)

We implement a sequential-domain CPT setup with fixed-ratio experience replay, following Ibrahim et al. (2024).

Buffer construction. After completing stage t , we sample $N = 2,000$ tokenized chunks uniformly at random from $\mathcal{D}_t$ and add them to a FIFO buffer $\mathcal{B}$ of capacity $C = 5,000$. Older entries are evicted once $|\mathcal{B}| > C$. The buffer thus retains a rough mixture of the most recent stages.

Mixed-batch training. During stage $t > 1$, each micro-batch of size B is constructed by replacing (not appending) a fixed fraction $r_{\text{rep}} = 0.5$ of its entries with samples drawn uniformly from the replay buffer:

$$X = \text{concat} \left(X_{\text{current}}[:, B - \lceil r_{\text{rep}} B \rceil], X_{\text{replay}}(\lceil r_{\text{rep}} B \rceil) \right). \quad (32)$$

What we deliberately omit. To isolate the effect of pure fixed-ratio rehearsal, Replay W.Adam itself does not employ learning-rate re-warming or re-decaying (Ibrahim et al., 2024), gradient alignment (Abbes et al., 2026), or meta-experience replay (Riemer et al., 2019). These stronger mechanisms are evaluated separately through MER-

Domain	Language	Illustrative excerpt
Medical	Korean	0.125(1/8), 0.1(1/10), 혹은 0.0625(1/16)% 필로카르핀 두 방울을 양눈에 점안하고 30분후 어두운 조명에서 관찰하면 정상 동공에서는 크기에 아무 변화가 없지만, 대부분의 긴장 동공에서는 축동이 발생한다. 동공이 클수록 같은 용량의 콜린성 약물에 더 수축하기 때문에 어두운 곳에서 검사하는 것이 좋다. ...
Medical	English	# QUESTIONS Diagnosis/Staging. What benefit to clinical management does positron emission tomography (PET) or positron emission tomography/computed tomography (PET/CT) contribute to the diagnosis or staging of head and neck cancer? ...
Legal	Korean	II. 贈與의 效力 증여의 일반적 효력은 증여자가 수증자에게 증여계약의 내용에 따라 재산을 수여하는 것이다(554조). 이에 관하여는 채권, 채무의 일반원칙이 통용된다. ...
Legal	English	Sec. 9601. Definitions. 9602. Designation of additional hazardous substances and establishment of reportable releases ...

Table 13: Illustrative excerpts from the four primary CPT corpus files. Excerpts are truncated for readability. The Korean Medical example is included within the minimal research-documentation allowance described by AI Hub.

Evaluation resource	Split	Split N	Eval. N	License / terms
Legal Case Document Summarization joelniklaus/legal_case_document_summarization (Niklaus, 2023; Shukla et al., 2022)	train	7,773	581	License not specified on the dataset card. Upstream source terms apply.
LawQA-Ko jihye-moon/LawQA-Ko (jihye-moon, 2023; Korea Ministry of Government Legislation, 2025)	train	14,819	9,742	License not specified on the dataset card. Upstream source terms apply.
medical_gemma_instruct_dataset Shekswess/medical_gemma_instruct_dataset (Shekswess, 2024; Ben Abacha and Demner-Fushman, 2019; medalpaca, 2023)	train	26,357	24,503	License not specified on the dataset card. Component-source terms apply.
medical-ol-reasoning-SFT-Ko ChuGyouk/medical-ol-reasoning-SFT-Ko (ChuGyouk, 2025; Chen et al., 2024)	train	25,682	24,615	Apache-2.0, as declared on the dataset card.
GSM8K (main) openai/gsm8k (Cobbe et al., 2021)	train	7,473	7,473	MIT.

Table 14: **Evaluation dataset statistics and terms.** Each entry lists the Hugging Face repository beneath the resource name. Split N denotes the number of examples in the corresponding upstream Hugging Face split. Eval. N denotes the number of examples used by the evaluation run underlying the reported results. The evaluation resources are separate from all CPT training corpora.

Replay W.Adam and Re-warm W.Adam. The optimizer is Adam at a fixed $\eta = 2 \times 10^{-5}$. The replay ratio r_{rep} is treated as a mixture hyperparameter, motivated by the mixture-ratio analysis of Gu et al. (2024). Algorithm 4 summarizes the procedure.

H.3.5 MER-Style Replay (MER-Replay W.Adam)

MER-Replay W.Adam is included as a stronger replay control motivated by meta-experience replay and recent replay-based continual pre-training methods (Riemer et al., 2019; Abbes et al., 2026). It uses replayed and current-stage examples within the same continual stream and retains Adam as the parameter-update backbone.

H.3.6 Learning-Rate Re-warming (Re-warm W.Adam)

Re-warm W.Adam follows the continual-pre-training re-warming strategy of Gupta et al. (2023). The optimizer is Adam with $\eta_{\text{max}} = 2 \times 10^{-5}$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, and $\epsilon = 10^{-8}$, with no weight decay. Adam’s optimizer state is preserved across CPT stages, whereas the learning-rate scheduler is restarted independently at each stage boundary.

For a stage with U optimizer updates, the first $U_{\text{warm}} = \max(1, \text{round}(0.05U))$ updates use linear warm-up to the base learning rate. The remaining updates follow cosine decay toward a minimum learning-rate ratio of 0.1, corresponding to 2×10^{-6} for our base learning rate. Thus, re-warming changes only the stage-wise learning-

Evaluation resource	Split / config	Split N	License / terms
ELRC-Medical-V2 qanastek/ELRC-Medical-V2 (Lösch et al., 2018)	en-es, train	13,195	CC BY 4.0, as declared in the official documentation. Attribution is required.
ALIA Spanish Legal and Administrative CQA SINAI/ALIA-es-legal-administrativ e-cqa (SINAI Research Group, 2026)	parlamint_es_an	7,892	CC BY-SA 4.0, as declared on the dataset card. Attribution and share-alike conditions apply.

Table 15: **Evaluation datasets for the Spanish extension.** Split N denotes the number of examples in the corresponding upstream split for the reported six-stage experiments. Both resources are external to the Spanish CPT corpora.

Algorithm 4 Adam+Replay Continual Pre-Training

Require: Domains $\{\mathcal{D}_1, \dots, \mathcal{D}_T\}$; parameters θ_0 ; replay ratio r_{rep} ; buffer capacity C ; samples per stage N ; batch size B ; learning rate η

- 1: $\theta \leftarrow \theta_0$
- 2: $\mathcal{B} \leftarrow \emptyset$
- 3: **for** $t = 1, \dots, T$ **do**
- 4: **for all** minibatches X_t of size B from $\mathcal{D}_t$ **do**
- 5: **if** $\mathcal{B} \neq \emptyset$ and $r_{\text{rep}} > 0$ **then**
- 6: $n_r \leftarrow \lceil r_{\text{rep}} B \rceil$
- 7: $X_r \leftarrow \text{sample}(\mathcal{B}, n_r)$ $\triangleright$ uniform without replacement
- 8: $X \leftarrow \text{concat}(X_t[B - n_r], X_r)$ $\triangleright$ replace, not append
- 9: **else**
- 10: $X \leftarrow X_t$
- 11: **end if**
- 12: $\theta \leftarrow \text{AdamUpdate}(\theta, \nabla_{\theta} \mathcal{L}_{\text{LM}}(\theta; X); \eta)$
- 13: **end for**
- 14: $\mathcal{B} \leftarrow \text{FIFO-Add}(\mathcal{B}, \text{sample}(\mathcal{D}_t, N), \text{cap}=C)$
- 15: **end for**
- 16: **return** θ

rate trajectory; the model architecture, CPT corpora, batch construction, and evaluation protocol are shared with the corresponding Adam comparison. No gradient clipping is applied.

H.3.7 Parameter-Efficient Adaptation (LoRA W.Adam)

We attach rank-16 LoRA adapters (Hu et al., 2021) to all attention projections (q, k, v, o) and MLP projections (gate, up, down) of each Llama decoder layer. The LoRA scaling factor $\alpha = 16$, dropout is set to 0.05, and the backbone is frozen. Adapters are initialized with the default LoRA scheme (Kaiming-uniform for A , zeros for B). The optimizer is Adam at $\eta = 2 \times 10^{-5}$ applied only to LoRA parameters. Adapters are not reset between stages: the same LoRA weights are continually updated across all stages, isolating the effect of restricting adaptation to a low-rank subspace.

H.3.8 Shared Training Configuration

For reproducibility, the following configuration is fixed across all baselines and PICO:

- Backbone: Llama-3.2-1B-Instruct (Meta AI, 2024).
- Sequence length: 256 and chunk size: 64.
- Per-device batch size: 4 and gradient accumulation: 8 (effective batch size 32 per device).
- Mixed precision: bfloat16 forward/backward, fp32 master copy in optimizer state for optimizers requiring one.
- Seeds: $\{777, 911, 4041\}$. Reported aggregate results are averaged over the three seeds.
- Compute hardware: NVIDIA A100 40GB GPUs.

This shared configuration controls the backbone, sequence length, batch size, precision, seeds, and compute environment across the primary 1B comparisons.

I Evaluation Metrics and Implementation

We provide implementation details for the five evaluation metrics used in the main experiments: ROUGE-L, METEOR, BLEU, cosine similarity, and Token F1. All metrics are computed identically across optimizers and configurations to ensure fair comparison.

Software stack. The training and evaluation implementation is written in Python using PyTorch (Paszke et al., 2019), with NumPy (Harris et al., 2020) used for numerical utilities, deterministic seeding, and metric aggregation. Scalar mathematical operations additionally use Python’s standard math module. For evaluation, ROUGE-L is computed with the rouge_score package, METEOR with NLTK’s single_meteor_score, and

Hyper.	Role	Value
η	Learning rate	2×10^{-5}
λ	Decoupled weight decay	0.01
β_u	Base EMA coefficient (utility / spectral)	0.999
σ_0	Base perturbation scale	0.001
f	Spectral-check period (steps)	1, 10, 100
K	Power-iteration steps	1
c_0	Warm-up checks	10
r	Anneal length (in checks)	30
τ_{start}	Initial threshold value	6.0
τ_{end}	Final threshold value	3.0
ε	Positive numerical-stability constant	10^{-12}

Table 16: Key hyperparameters for the reported PICO experiments. The spectral-check period is experiment-specific: $f = 1$ for the primary four-stage comparison and component ablations, $f = 10$ for model scaling, and $f = 100$ only in the sensitivity analysis.

BLEU with SacreBLEU (Post, 2018). Cosine similarity and Token F1 are implemented directly using Python’s `collections.Counter`.

Shared lexical preprocessing. ROUGE-L, METEOR, and cosine similarity use the same deterministic Unicode-aware lexical tokenizer. The tokenizer applies Unicode NFKC normalization and case folding, extracts Unicode alphanumeric lexical sequences, and preserves apostrophes within lexical tokens. Punctuation is otherwise excluded. When an input contains no lexical tokens, the tokenizer falls back to individual non-whitespace characters. The same lexical policy is applied to English, Korean, and Spanish.

ROUGE-L. ROUGE-L (Lin, 2004) measures the longest common subsequence between prediction and reference, normalized as an F-measure. We use `rouge_score.RougeScorer` and report the `rougeL` F-measure. We provide the shared Unicode-aware tokenizer described above to `RougeScorer` and disable stemming.

METEOR. METEOR (Banerjee and Lavie, 2005) evaluates lexical alignment between prediction and reference while combining unigram precision and recall with a fragmentation penalty. We compute METEOR using NLTK’s `single_meteor_score`. For the reported multilingual evaluation, prediction and reference are processed with the shared Unicode-aware lexical tokenizer. Stemming and WordNet-based synonym expansion are disabled using an identity stemmer and an empty WordNet interface. This lexical-only

configuration is applied identically to English, Korean, and Spanish.

BLEU. We compute corpus-level BLEU-4 following Papineni et al. (2002) using SacreBLEU (Post, 2018). Predictions and references are normalized with Unicode NFKC and pre-tokenized using a fixed Unicode-aware tokenizer applied consistently across English, Korean, and Spanish. This tokenizer preserves case, extracts Unicode alphanumeric sequences with internal apostrophes, and treats each remaining non-whitespace punctuation mark or symbol as a separate token. SacreBLEU’s internal tokenizer is disabled (`tokenize=None`). We use exponential smoothing (`smooth_method=exp`) and enable effective order. Scores are divided by 100 so that BLEU is reported on the $[0, 1]$ scale used by the other evaluation metrics.

Cosine similarity. Cosine similarity is computed over bag-of-words count representations constructed from the shared Unicode-aware lexical tokens. For a prediction $\hat{y}$ and reference y , let $\mathbf{c}_{\hat{y}}$ and $\mathbf{c}_y$ denote their token-count vectors. The pairwise cosine similarity is

$$\text{CosSim}(\hat{y}, y) = \frac{\mathbf{c}_{\hat{y}}^\top \mathbf{c}_y}{\|\mathbf{c}_{\hat{y}}\|_2 \|\mathbf{c}_y\|_2}. \quad (33)$$

The count representations are implemented using `collections.Counter`. If either token sequence is empty, its pairwise score is set to zero. The aggregate cosine similarity lies in $[0, 1]$.

Token F1. Token F1 is computed as overlap-based F1 between prediction $\hat{y}$ and reference y in a fixed tokenizer space. Let n_{same} denote the multi-set overlap count between prediction and reference token IDs, and let n_{pred} and n_{gold} denote their respective token counts. Following the overlap-based F1 principle used in SQuAD-style evaluation (Rajpurkar et al., 2016), precision and recall are

$$P = \frac{n_{\text{same}}}{n_{\text{pred}}}, \quad R = \frac{n_{\text{same}}}{n_{\text{gold}}}, \quad (34)$$

and Token F1 is

$$\text{TokenF1} = \frac{2PR}{P + R}. \quad (35)$$

Multiset overlap is computed using `collections.Counter`. If either sequence is empty or no tokens overlap, precision, recall, and F1 are set to zero.

Baseline configurations. Methods are grouped by optimizer family: second-order approximation (Sophia), first-order (Adam, SGD), regularization (EWC), and parameter-efficient (LoRA). EWC (Kirkpatrick et al., 2017) denotes Elastic Weight Consolidation applied on top of Adam. LoRA (Hu et al., 2021) uses rank-16 adapters with Adam and serves as a representative PEFT baseline.

I.1 Baseline choice for FWT

The original FWT definition of Lopez-Paz and Ranzato (2017) uses a random-initialization baseline $\bar{b}_i$. In LLM continual pre-training, the model starts from a pre-trained checkpoint, so a random-init baseline is uninformative. We instead use $\bar{R}_i^{(\mu)}$, the score of the pre-trained base model on domain i before any CPT stage. This substitution preserves the original intent of FWT: positive FWT means the CPT stream improves performance on unseen future domains beyond the initial model.

I.2 Relation to Chaudhry’s per-task forgetting

The AvgF of Chaudhry et al. (2018) is defined per task as $f_j^k = \max_{l \in \{1, \dots, k-1\}} a_{l,j} - a_{k,j}$, where $a_{l,j}$ is the score on task j after training on task l . Our AvgF $^{(\mu)}$ is the average of f_i^T over $i \in \{1, \dots, T-1\}$ with $a_{l,j} = R_{l,i}^{(\mu)}$. The final-stage row $t = T$ is excluded from the maximum so that AvgF measures degradation from the best observed score *prior to* the final state, in line with Chaudhry’s definition.

I.3 Relation to worst-case continual metrics

De Lange et al. (2023) introduce worst-case accuracy (WC-ACC) tracking the minimum performance *during* training to expose the stability gap at task transitions. Our WCR extends the same motivation to a stage-boundary setting: they are computed only at the end of each stage rather than at every iteration, but use the worst-case principle to expose single-domain collapse hidden by stream averages.

J Detailed Evaluation Results

The ROUGE-L and cosine-similarity FWT, BWT and AvgF summaries are reported in Figure 2, and the ROUGE-L FWT–WCR trade-off is reported in Figure 1.

Near-zero BLEU scores. Several BLEU entries in Table 4 are displayed as 0.0000 because all values are truncated to four decimal places. Inspection of the full-precision BLEU scores confirms that most of these displayed zeros correspond to small positive values rather than exact zeros. For these cases, the first nonzero digit occurs beyond the four-decimal reporting precision used in the table. The corresponding method–task combinations also exhibit low ROUGE-L, METEOR, and Token F1 scores, suggesting broadly weak reference-aligned generation rather than a BLEU-specific artifact. All metric computations use the original full-precision values, with truncation applied only for presentation in the table.

f	FWT $\uparrow$	BWT $\uparrow$	AvgF $\downarrow$	Average $\uparrow$	WCR $\downarrow$
1	0.0009	−0.0035	0.0044	0.1369	0.0553
10	0.0002	−0.0026	0.0026	0.1354	0.0450
100	−0.0003	−0.0020	0.0020	0.1365	0.0304

Table 17: **Sensitivity to the spectral-check interval f .** ROUGE-L continual-learning metrics under different spectral-check intervals with the remaining PICO configuration fixed.

K Additional Ablations: PICO Components and Scaling

K.1 Experimental Setup

This appendix reports additional ablations for PICO that are omitted from the main text due to space constraints. All metrics are computed from the same CPT evaluation matrix as in the main experiments. FWT, BWT, and AvgF are computed on ROUGE-L throughout this appendix, with one exception: the curriculum-disentanglement study in Table 18 reports METEOR. Math results are evaluated on a held-out Math probe that is never used for CPT. TokenF1 is macro-averaged across tasks.

Sensitivity to the spectral-check interval. Table 17 examines the effect of the spectral-check interval f while keeping the remaining configuration fixed. Increasing f from 1 to 100 leaves the average final-stage ROUGE-L largely unchanged (0.1369, 0.1354, and 0.1365), while WCR decreases from 0.0553 to 0.0304 and AvgF decreases from 0.0044 to 0.0020. BWT also becomes slightly less negative as spectral checks become less frequent. This improvement in retention is accompanied by a small reduction in FWT, from 0.0009 at $f = 1$

Setting	Stream	FWT $\uparrow$	BWT $\uparrow$	AvgF $\downarrow$	Math (Avg.) $\uparrow$
Domain-only	English Legal $\rightarrow$ English Medical	-0.0989	-0.0025	0.0025	0.0110
Language-only	English Legal $\rightarrow$ Korean Legal	-0.0136	0.0043	-0.0043	0.0254
Domain + language	English Medical $\rightarrow$ Korean Legal	-0.0100	-0.0006	0.0006	0.0060

Table 18: **Curriculum disentanglement.** Domain-only, language-only, and coupled domain–language shifts exhibit distinct forward-transfer and retention behavior. FWT, BWT, and AvgF are computed on METEOR, and Math (Avg.) reports the mean METEOR on the held-out GSM8K probe.

to -0.0003 at $f = 100$. These results indicate that PICO is not highly sensitive to f in terms of average final performance, while the check interval affects the balance between forward transfer and retention.

K.2 Detailed End-Task Generalization

This section expands on the end-task results in Section 4.4. The ranking statistics below exclude the initial model, which is a shared reference rather than a trained CPT method.

On Medical–English, PICO is the best trained method on all five metrics. Its margins over the strongest trained baseline are $+0.0648$ ROUGE-L, $+0.1887$ cosine similarity, $+0.1031$ METEOR, $+0.0217$ BLEU, and $+0.0943$ Token F1. On Legal–English, PICO is again best on all five metrics, with margins of $+0.0222$ ROUGE-L, $+0.0332$ cosine similarity, $+0.0189$ METEOR, $+0.0035$ BLEU, and $+0.0576$ Token F1.

Medical–Korean is more competitive. PICO is best on ROUGE-L (0.1378), METEOR (0.1589), and Token F1 (0.2236), second on BLEU (0.0563 versus 0.0565 for Re-warm W.Adam), and third on cosine similarity (0.2067 versus 0.2196 for SGD and 0.2117 for Re-warm W.Adam). Legal–Korean exhibits the clearest metric-dependent variation among the four evaluations. Replay W.Adam leads ROUGE-L (0.0835), Re-warm W.Adam leads cosine similarity (0.1120), METEOR (0.0719), and BLEU (0.0251), and MER-Replay W.Adam leads Token F1 (0.2392). PICO is second on Legal–Korean ROUGE-L (0.0655) and remains competitive but not leading on the other metrics.

These results motivate a more selective interpretation of the final-stage performance results. PICO’s advantage is strongest and most consistent on the English tasks and on Medical–Korean, whereas Legal–Korean remains method- and metric-dependent. Accordingly, the main text interprets final-stage performance jointly with the FWT–WCR analysis.

K.3 Disentangling Domain and Language Shifts

To examine how different types of distribution shift affect continual adaptation, we separate the CPT curriculum into domain-only, language-only, and coupled domain–language shifts (Table 18, METEOR). The resulting profiles show that forward transfer and retention do not vary uniformly across these settings. Domain-only adaptation exhibits relatively weak forward reuse, whereas the language-only setting shows more favorable retention of previously evaluated capability. The coupled shift presents a different trade-off, with comparatively stronger forward transfer but without the same retention behavior as the language-only setting. Overall, these results suggest that domain and language changes stress different aspects of continual adaptation. They therefore support treating curriculum composition as an important experimental factor rather than assuming a universal ordering of shift difficulty.

Stream	T	FWT $\uparrow$	BWT $\uparrow$	AvgF $\downarrow$	WCR $\downarrow$
Primary four-stage	4	0.0009	-0.0035	0.0044	0.0553
Language-blocked	4	0.0001	0.0010	0.0020	0.0399
Six-stage extension	6	-0.0019	0.0180	-0.0017	0.0607

Table 19: **ROUGE-L continual-learning summaries for PICO under different curricula.** The primary stream follows Korean Medical, English Medical, Korean Legal, and English Legal. The language-blocked stream changes only the order of the four primary domains. The six-stage extension adds Spanish Medical and Spanish Legal.

K.4 Third-Language and Order Extensions

We evaluate two extensions of the primary curriculum. First, we add Spanish Medical and Spanish Legal to form the six-stage curriculum Spanish Medical, Korean Medical, English Medical, Spanish Legal, Korean Legal, and English Legal. Second, we evaluate a language-blocked permutation of the primary four-stage curriculum as Korean Medical,

Metric	Adam			PICO		
	FWT $\uparrow$	BWT $\uparrow$	WCR $\downarrow$	FWT $\uparrow$	BWT $\uparrow$	WCR $\downarrow$
BLEU	-0.0286	-0.0131	0.9996	-0.0033	0.0042	0.0740
METEOR	-0.0833	-0.0666	0.8602	-0.0038	0.0000	0.1281
Cosine similarity	-0.1736	-0.1541	0.8624	-0.0098	0.0043	0.0712
Token F1	-0.1176	-0.0718	0.7903	-0.0119	0.0133	0.0772

Table 20: **Cross-metric continual-learning results on the six-stage extension.** Results use the same matched Adam and PICO runs as the ROUGE-L comparison in Table 5.

Method	After Spanish-Legal	After Korean-Legal	After English-Legal	WCR $\downarrow$
Adam	0.1741	0.0110	0.0342	0.8032
PICO	0.2031	0.2089	0.1962	0.0607

Table 21: **ROUGE-L retention on Spanish Legal in the six-stage stream.** The columns show Spanish-Legal performance after its own CPT stage and after the subsequent Korean-Legal and English-Legal stages.

Korean Legal, English Medical, and English Legal. All results are computed with the canonical prompt-only evaluator.

Effect of curriculum extension and reordering on PICO. Table 19 first compares PICO across the primary, reordered, and third-language curricula. This comparison asks how PICO itself changes when the stream composition or order changes.

The six-stage extension retains a WCR of 0.0607, close to the primary-stream value of 0.0553, despite increasing the stream from four to six learned domains. Its FWT changes from 0.0009 to -0.0019. The third-language extension preserves the low-collapse profile more consistently than the slightly positive forward transfer observed on the primary stream.

The language-blocked curriculum yields ROUGE-L WCR = 0.0399, which is lower than the primary-stream value. Its BLEU WCR is 0.1391 and its METEOR WCR is 0.2154. The effect of curriculum reordering is therefore metric-dependent, and these results do not support either order invariance or a universal degradation under reordering.

K.5 Third-Language Extension

Six-stage protocol. The third-language extension uses the curriculum Spanish Medical $\rightarrow$ Korean Medical $\rightarrow$ English Medical $\rightarrow$ Spanish Legal $\rightarrow$ Korean Legal $\rightarrow$ English Legal. We evaluate the Initial Model checkpoint and the checkpoint obtained after each of the six stages, yielding seven evaluation points per run. The Adam and PICO runs use the same Llama-3.2-1B-Instruct backbone, curriculum, downstream evaluation resources, and

seeds. Both are evaluated with the same canonical evaluation pipeline in bf16, with strict evaluation enabled and no language-model-loss or accuracy computation. The downstream resources are re-tokenized for each checkpoint rather than reusing a shared token-ID cache, and Token F1 uses the same base-model tokenizer across checkpoints.

Cross-metric behavior. Table 20 shows that the ROUGE-L pattern extends to the other generation metrics. Together with Table 5, PICO keeps FWT substantially closer to zero and yields non-negative BWT across all five metrics, whereas Adam has negative BWT throughout. PICO also maintains substantially lower worst-case degradation across metrics.

Stage-wise retention. Spanish Legal is the worst-collapse ROUGE-L domain for both methods. As shown in Table 21, Adam undergoes a sharp degradation after subsequent legal-domain training, whereas PICO largely preserves its post-training Spanish-Legal performance. This trajectory is consistent with the large difference in aggregate WCR between the two methods.

Component ablation. Table 6 evaluates the contributions of SPECFLAG and stochastic perturbation under the same four-stage setting. Removing SPECFLAG increases WCR from 0.0553 to 0.1400, indicating substantially greater worst-domain degradation. Removing stochastic perturbation produces a broader decline, with WCR increasing to 0.1624 and average final-stage ROUGE-L decreasing from 0.1369 to 0.0921. These ablations support complementary roles for the two components. SPECFLAG is particularly important for lim-

iting worst-case degradation, while stochastic perturbation contributes more broadly to maintaining the observed stability–plasticity trade-off.

K.6 Scaling with Model Size

The ROUGE-L scaling results are given in Table 7. With $f = 10$, FWT is near zero at all scales, but WCR changes substantially: 0.0450 at 1B, 0.3292 at 3B, and 0.1957 at 8B. Average final-stage ROUGE-L remains comparatively close across scales, emphasizing that aggregate final-stage performance can hide worst-domain degradation. These results indicate scale sensitivity and do not establish scale invariance.

L Notation

A notation summary is provided in Table 22.

Symbol	Type / Shape	Meaning
General		
k	integer	Global optimizer-step index.
$\theta^{(k)}$	parameter tensors	Model parameters at optimizer step k .
d	integer	Total number of trainable parameters.
$\mathcal{L}(\theta)$	scalar	Loss on the current CPT-stage data.
$g^{(k)}$	parameter-shaped tensors	Gradient $g^{(k)} = \nabla_{\theta} \mathcal{L}(\theta^{(k)})$ (Eq. (1)).
Φ	function	Optimizer-level update map (Eq. (2)).
$\odot$	operator	Element-wise product.
$\mathbb{I}[\cdot]$	function	Binary indicator function.
$\ \cdot\ _F$	function	Frobenius norm.
ε	scalar	Positive numerical-stability constant.
Utility estimator (GATEU)		
β_u	scalar	Base EMA coefficient for utility smoothing and the spectral EMA via $\beta_{\text{eff}} = \beta_u^f$.
n_{θ}	integer	Local utility-update counter for parameter tensor θ .
$\text{instU}_{\theta}^{(n_{\theta})}$	parameter-shaped tensor	Instantaneous utility proxy (Eq. (15)).
$\text{avgU}_{\theta}^{(n_{\theta})}$	parameter-shaped tensor	EMA utility statistic (Eq. (16)).
$u^{(k)}$	parameter-shaped tensor	Coordinate-wise utility gate in $[0, 1]$ (Alg. 2).
$\tilde{g}^{(k)}$	parameter-shaped tensor	Protected gradient (Eq. (3)).
Spectral structural monitor (SPECFLAG)		
M	integer	Number of optimizer parameter groups. The reported training implementation uses $M = 1$.
$\mathcal{G}_j$	parameter group	The j -th PyTorch optimizer parameter group.
W	matrix	Monitored two-dimensional parameter $W \in \mathbb{R}^{d_{\text{out}} \times d_{\text{in}}}$.
$d_{\text{out}}, d_{\text{in}}$	integers	Output and input dimensions of W .
f	integer	Spectral-check period. Checks are scheduled when $k \bmod f = 0$.
K	integer	Number of warm-started power-iteration steps per check.
$\mathbf{v}_W$	vector in $\mathbb{R}^{d_{\text{in}}}$	Persistent power-iteration vector for matrix W .
c_W	integer	Local spectral-observation counter for W . Incremented only when W is evaluated.
$\hat{\sigma}_1^{(c_W)}(W)$	scalar	Estimated leading singular value at observation c_W .
$\hat{\rho}_{\varepsilon}^{(c_W)}(W)$	scalar	Stabilized and clipped spectral-concentration statistic (Eq. (4)).
$\beta_{\text{eff}}, \bar{\beta}_{\text{eff}}$	scalars	Check-frequency-adjusted spectral EMA coefficients (Eq. (21)).
$\mu_x^{(c_W)}$	scalar	Exponentially weighted mean of spectral quantity x (Eq. (23)).
$\nu_x^{(c_W)}$	scalar	Exponentially weighted squared-deviation estimate used as a running variance proxy (Eq. (24)).
$\tilde{\nu}_x^{(c_W-1)}$	scalar	Stabilized previous variance proxy (Eq. (25)).
$z_x^{(c_W)}$	scalar	Signed standardized deviation from the pre-update EMA statistics (Eq. (26)).
c_0	integer	First local observation index at which flagging is enabled.
r	integer	Threshold-annealing length in spectral observations.
$\tau_{\text{start}}, \tau_{\text{end}}$	scalars	Initial and final threshold values.
$\tau(c_W)$	scalar	Annealed threshold at local observation c_W (Eq. (27)).
$s_W^{(c_W)}$	binary	Matrix-level flag computed at observation c_W (Eq. (28)).
$\text{specSens}[\theta]$	binary cached state	Most recently computed structural flag stored for parameter tensor θ . Initialized to zero.
$\pi_{\mathcal{G}_j}^{(k)}$	binary	Transient group-level indicator that suppresses perturbations on the current check step.
Main update with perturbation gating		
$s^{(k)}$	binary	Tensor-level structural flag read from $\text{specSens}[\theta]$ at optimizer step k .
$m^{(k)}$	parameter-shaped tensor	Multiplicative perturbation gate (Eq. (5)).
σ_0	scalar	Base perturbation scale.
σ_k	parameter-shaped tensor	Coordinate-wise perturbation scale (Eq. (6)).
$\xi^{(k)}$	parameter-shaped random tensor	Gated Gaussian perturbation (Eq. (6)).
η	scalar	Learning rate (Eq. (7)).
λ	scalar	Decoupled weight-decay coefficient (Eq. (7)).

Table 22: Notation for PICO.