TETHER: DECISION RETENTION FOR CONTINUAL FINE-TUNING OF LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Continual fine-tuning can cause catastrophic forgetting in language models. Changes in old-task accuracy reflect a net effect: repaired errors offset lost correct predictions. We instead measure and constrain decision drift, the fraction of old-task inputs whose decisions differ from those of the model at the end of that task, the teacher. Drift bounds forgetting, both lost correct answers and the loss in accuracy, whereas accuracy does not bound drift; it needs no labels and has an exact finite-sample upper bound on held-out inputs. We propose Tether, a framework that stores real or synthetic inputs with the model's outputs as records and regularises later training with any continuous loss that upper-bounds record drift. For forward-KL distribution matching on records, we derive the exact per-record critical radius below which the recorded decision is preserved; it shows that standard KL also penalises distributional changes that leave the recorded decision intact. Two new instances, Tether-Margin and thresholded KL (Tether-TKL), are zero on a region around the teacher and skip satisfied records before back-propagation. On the O-LoRA benchmark with T5-large and Llama-2-7B-chat and on six generation tasks with Llama-3-8B-Instruct, without retaining per-example ground-truth labels, the three instances reach final average scores comparable to DER++'s, from 1.1 points below to 1.5 above. Tether-TKL and KL change fewer earlier decisions than DER++, KL losing about half as many correct answers, and on generation all three keep 50–79% of earlier free-form outputs unchanged, against 38% for DER++. On the O-LoRA benchmark, with fewer retained records in aggregate than the 2% replay and DER++ buffers, all three also preserve more of the model's earlier behaviour than labelled replay, changing fewer decisions in every setting.

1 INTRODUCTION

Language models suffer from catastrophic forgetting under continual fine-tuning: learning a new task degrades performance on earlier ones. Forgetting is usually measured as a drop in old-task accuracy (Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2018). Existing methods constrain parameter updates (Wang et al., 2023; Yang et al., 2025b; Wang et al., 2025), replay stored (de Masson d'Autume et al., 2019; Buzzega et al., 2020) or generated examples (Sun et al., 2020; Qin & Joty, 2022; Huang et al., 2024), or regularise outputs towards an earlier model (Li & Hoiem, 2018; Titsias et al., 2020; Pan et al., 2020).

Accuracy, however, reflects only the net effect of later training on earlier answers. Harmful flips, correct answers turned wrong, are offset by repairs, wrong answers turned correct, and a switch from one wrong answer to another does not register at all. In our experiments on four-task sequences, labelled replay with a 2% buffer changes 12–14% of its earlier decisions while its old-task accuracy moves by about one point. A model can thus look stable by accuracy while many of its earlier answers have changed. Such changes have a cost of their own for the users and systems that rely on earlier outputs, which is why research on model updates tracks changes in predictions and not only accuracy (Milani Fard et al., 2016; Xie et al., 2021).

Our starting point is a simple observation, also behind negative-flip analyses of model updates (Yan et al., 2021): a correct answer can be lost only if the model's decision on that input changes. We turn it into a retention target, *decision drift*: the fraction of old-task inputs on which the current model decides differently from the model at the end of that task, which we call the *teacher*. Drift

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

bounds the rate of harmful flips and, under the same decision rule, the loss in accuracy: the current model's old-task accuracy is at least the teacher's accuracy minus the drift. Drift also counts repaired errors; keeping decisions protects both the accuracy on old tasks and the behaviour behind it. Drift compares the current model with the teacher rather than with the ground truth, so it can be measured, and retention enforced, without per-example labels. Without labels one cannot tell which of the teacher's decisions are correct, so keeping all of them is the weakest requirement that protects every correct answer. Unchanged parameters or unchanged outputs imply unchanged decisions but not conversely, while losses on labelled memories target accuracy itself.

We propose **Tether**, a framework that controls decision drift under a fixed per-task record budget: it tethers the model to the decisions it made at the end of each task while leaving it free to learn new ones. After each task, Tether keeps a small set of *records*: real or synthetic inputs paired with the teacher's outputs. The weighted fraction of records whose decision has changed, the *record drift*, serves as a training proxy for task drift. Record drift only counts changes and gives no useful gradient. Tether therefore penalises a continuous loss that bounds it from above, much as the hinge loss bounds classification error. Any such loss can be used. At approximate global optima of the penalised objective, record drift is bounded by the optimisation error and the conflict between retention and the new task.

The framework explains an existing retention penalty and yields two new ones. Forward-KL distribution matching, the penalty behind distillation-based retention (Li & Hoiem, 2018; Titsias et al., 2020; Pan et al., 2020), satisfies Tether's condition when applied to stored records of earlier tasks, rather than to new-task inputs as in LwF; unlike DER++ (Buzzega et al., 2020), it does not retain per-example ground-truth labels. We derive an exact critical radius for each record: below this divergence from the teacher, the recorded decision cannot change. The radius also shows that the output distribution can change without changing the recorded decision, whereas standard KL, which is zero only at the teacher distribution, penalises every change. These penalties are unnecessary for the recorded decision, although they may improve retention on unrecorded inputs. This observation motivates two objectives with nontrivial zero-loss regions. **Tether-Margin** keeps only the recorded answer and its probability margin, and penalises margins below a fixed fraction of their recorded value. **Thresholded KL** (Tether-TKL) keeps the divergence but stops penalising at or below a fixed fraction of the critical radius.

We evaluate on the continual benchmark of Wang et al. (2023) with T5-large and Llama-2-7B-chat and on a sequence of six generation tasks with Llama-3-8B-Instruct. Without retaining per-example ground-truth labels, all three instances reach final average scores comparable to DER++'s on both; Tether-TKL and KL change fewer earlier decisions than DER++, KL about half as many and losing about half as many correct answers, whereas Margin's drift is broadly comparable to DER++'s. On the classification benchmark, with fewer retained records in aggregate than the 2% replay and DER++ buffers, all three also change fewer decisions than labelled replay in every setting; on generation, they keep 50–79% of earlier free-form outputs unchanged, against 38% for DER++. Final scores barely separate these methods; drift does. With synthetic records, Tether-Margin's classification accuracy stays within one percentage point of the corresponding real-record result, without retaining real task inputs. The instances form a ladder: the more of the output distribution an instance fixes around each record, the fewer unrecorded inputs flip and the more it computes, from Margin at 3.1 times the cost of unprotected training on Llama-2, through Tether-TKL at 3.9, to KL at 12, although on generation the last two cost about the same. On both benchmarks, new-task performance stays at most one point below unprotected training for all three.

Our goal is to preserve a model's behaviour on earlier tasks, not only its accuracy; decision drift measures this behaviour and is our retention target. Our contributions are:

1. We derive the exact record-specific forward-KL radius at which a teacher's most probable decision can first change. The radius is sharp over categorical output distributions and yields explicit upper bounds on record drift.
2. Using this decision-preservation geometry, we formulate Tether as a framework based on continuous losses that upper-bound record drift. We introduce Tether-Margin and thresholded KL, whose losses vanish on nontrivial decision-preserving regions and permit satisfied records to be skipped before back-propagation. We also distinguish deterministic retention on stored records from transfer to unrecorded inputs and label-free auditing on held-out inputs.

3. On continual classification with T5-large and Llama-2-7B-chat and generation with Llama-3-8B-Instruct, the Tether instances attain final scores comparable to DER++ while exhibiting different trade-offs among decision retention, retained teacher information and computation. KL gives the lowest observed drift, whereas Margin, and on classification thresholded KL, train faster than KL by skipping satisfied records; synthetic records provide similar classification accuracy without retaining real task inputs.

2 METHOD

2.1 DECISION DRIFT AND THE TETHER FRAMEWORK

We first consider a pretrained language model that learns tasks $T_1, \ldots, T_K$ in sequence, each with a finite set $\mathcal{L}_t$ of valid answers. We identify a model with its parameters θ. The model after task t, with parameters θ_t, is the *teacher* for task t; we aim to keep its decisions on task t's inputs while later tasks are learned.

Decision rule. For an input x, $q_\theta(\cdot \mid x)$ denotes the model's distribution over output token sequences. Each answer $c \in \mathcal{L}_t$ is written as a fixed token sequence $s_t(c)$, different for different answers. The probability of answer c is that of generating exactly $s_t(c)$ and stopping; the remaining probability goes to an outcome $\perp$ that collects all other outputs:

$$p_\theta(c \mid x) = q_\theta\big(s_t(c)\,\mathrm{EOS} \mid x\big) \;\; (c \in \mathcal{L}_t), \qquad p_\theta(\perp \mid x) = 1 - \textstyle\sum_{c \in \mathcal{L}_t} p_\theta(c \mid x), \tag{1}$$

where EOS is the end token. All quantities below are defined from this distribution on $\bar{\mathcal{L}}_t = \mathcal{L}_t \cup \{\perp\}$ rather than from the token distribution. The decision is

$$\hat{c}_\theta(x) = \arg\max_{c \in \bar{\mathcal{L}}_t} p_\theta(c \mid x), \tag{2}$$

with ties broken by a fixed rule. This is the standard scoring rule for finite answer sets (Appendix B.5 relates it to greedy decoding). For generation, all three instances apply at teacher prefixes over the vocabulary; preserving every token decision, including termination, preserves the greedy output (Appendix F).

Decision drift. Let P_t be the distribution of inputs. The *decision drift* of a model θ on task t is the probability that it and the teacher decide differently on a random input $X \sim P_t$:

$$F_t(\theta) = \Pr\left[\hat{c}_\theta(X) \neq \hat{c}_{\theta_t}(X)\right]. \tag{3}$$

Drift needs no labels. Let $Y \in \mathcal{L}_t$ be the correct answer to X. The accuracy of θ is $\mathrm{Acc}_t(\theta) = \Pr[\hat{c}_\theta(X) = Y]$, and its harmful-flip rate $\Pr[\hat{c}_{\theta_t}(X) = Y \neq \hat{c}_\theta(X)]$ is the fraction of inputs that the teacher answers correctly and θ does not. A correct decision can become wrong only if it changes, so the harmful-flip rate is at most $F_t(\theta)$ and

$$\mathrm{Acc}_t(\theta) \geq \mathrm{Acc}_t(\theta_t) - F_t(\theta). \tag{4}$$

The bound is conservative, because drift also counts repaired errors.

Records. After task t, Tether keeps a small set V_t of inputs, called *records*. Records may also be synthetic inputs (Appendix B.6). For each record $v \in V_t$, it stores the teacher's decision $c_v = \hat{c}_{\theta_t}(v)$, called the *recorded decision*, and a weight $w_v \geq 0$ with $\sum_{v \in V_t} w_v = 1$; no ground-truth labels are stored, and a retention loss may store further teacher outputs (Section 2.4). The *record drift* is the weighted fraction of records whose decision has changed:

$$R_t(\theta) = \textstyle\sum_{v \in V_t} w_v \, \mathbf{1}[\hat{c}_\theta(v) \neq c_v]. \tag{5}$$

It is the training proxy for F_t, although V_t need not be a sample from P_t. Equivalently, R_t is a weighted 0–1 loss with the recorded decisions as labels; it is piecewise constant in θ, so its gradient is zero wherever it exists. The *probability margin* measures an answer's lead over its strongest competitor:

$$m_\theta(x; c) = p_\theta(c \mid x) - \max_{j \in \bar{\mathcal{L}}_t \setminus \{c\}} p_\theta(j \mid x). \tag{6}$$

It is positive exactly when c is the unique most probable outcome. Write $m_v = m_{\theta_t}(v; c_v)$ for the teacher's margin on record v. Tether admits an input as a record only if the teacher gives a valid answer with $m_v \geq m_{\min} > 0$. The guarantees below scale with the recorded margin and degrade as it vanishes; the threshold keeps their constants finite (Appendix B.4). Inputs that fail this rule still count in F_t, although no record represents them; Section 2.5 accounts for them.

Framework. For parameters with finite logits, Tether uses a continuous record loss $G_t \geq 0$ that vanishes at the teacher (Appendix A.4) and upper-bounds record drift up to a constant $\beta_G > 0$:

$$R_t(\theta) \leq \beta_G \, G_t(\theta) \qquad \text{for all } t, \theta. \tag{7}$$

When learning task k, Tether minimises

$$J_k^G(\theta) = \mathcal{L}_k(\theta) + \lambda \textstyle\sum_{t<k} G_t(\theta), \tag{8}$$

where $\mathcal{L}_k$ is the cross-entropy on the labelled training set D_k of task k (Appendix B.2); the second term is the *retention penalty*, with weight $\lambda \geq 0$.

2.2 The margin instance

By (6), a positive margin preserves the recorded decision. To keep a buffer against probability changes, *Tether-Margin* asks each record to retain a fixed fraction $\kappa \in (0, 1]$ of the teacher's margin:

$$m_\theta(v; c_v) \geq \kappa m_v. \tag{9}$$

It penalises the normalised shortfall from this condition and averages it over records:

$$\psi_v(\theta) = \frac{\max\{\kappa m_v - m_\theta(v; c_v),\, 0\}}{\kappa m_v}, \qquad C_t(\theta) = \textstyle\sum_{v \in V_t} w_v \, \psi_v(\theta). \tag{10}$$

Proposition 1 (record-level decision guarantee)**.** *For every θ and $v \in V_t$, $\mathbf{1}[\hat{c}_\theta(v) \neq c_v] \leq \psi_v(\theta)$. Hence $R_t(\theta) \leq C_t(\theta)$, and $\psi_v(\theta) < 1$ guarantees $\hat{c}_\theta(v) = c_v$.*

Appendix A.1 gives the proof. Tether-Margin thus uses $G_t = C_t$ in (8); it vanishes at the teacher because $\kappa \leq 1$, and satisfies (7) with $\beta_C = 1$.

2.3 Distribution-matching instances

The next two instances keep the whole distribution on $\bar{\mathcal{L}}_t$ close to the teacher's, $p_v^\circ = p_{\theta_t}(\cdot \mid v)$, measured by the forward divergence $d_v(\theta) = \mathrm{KL}\left(p_v^\circ \,\|\, p_\theta(\cdot \mid v)\right)$, which is zero at the teacher.

KL distribution matching. KL distribution matching (Tether-KL) applies the standard forward-KL distillation penalty to these distributions on the records: it uses $G_t = \mathcal{K}_t = \sum_{v \in V_t} w_v d_v$ in (8). This loss need not be zero to preserve a decision. We call the smallest forward divergence at which c_v can stop being the unique most probable outcome the *critical radius* of record v.

Proposition 2 (critical KL radius of a recorded decision)**.** *Let $a = p_v^\circ(c_v)$ and $b = \max_{j \neq c_v} p_v^\circ(j)$. The critical radius of record v is*

$$\varepsilon_v^* = a \log \frac{2a}{a+b} + b \log \frac{2b}{a+b} \;\geq\; \frac{m_v^2}{2}, \tag{11}$$

with $0 \log 0 = 0$. If $d_v(\theta) < \varepsilon_v^*$, then $\hat{c}_\theta(v) = c_v$, and no larger radius gives this guarantee over all distributions on $\bar{\mathcal{L}}_t$.

The proof is in Appendix A.2. The corresponding record-drift bound is

$$R_t(\theta) \leq \textstyle\sum_{v \in V_t} w_v \, d_v(\theta)/\varepsilon_v^* \leq 2 \, \mathcal{K}_t(\theta)/m_{\min}^2, \tag{12}$$

so KL satisfies (7) with $\beta_{\mathcal{K}} = 2/m_{\min}^2$ (Appendix A.1).

Thresholded KL. The critical radius gives each record its own sufficient condition for retention. KL penalises the raw divergence d_v; *Tether-TKL* instead scales it by a threshold derived from that radius and assigns zero loss at or below the threshold. Choose $\eta \in (0, 1)$ and set $\tau_v = \eta \varepsilon_v^*$:

$$\psi_v^{\mathrm{TKL}}(\theta) = \begin{cases} 0, & d_v(\theta) \leq \tau_v, \\ \dfrac{d_v(\theta) - \tau_v}{\tau_v}, & d_v(\theta) > \tau_v, \end{cases} \qquad C_t^{\mathrm{TKL}}(\theta) = \textstyle\sum_{v \in V_t} w_v \, \psi_v^{\mathrm{TKL}}(\theta). \tag{13}$$

The zero-loss threshold lies strictly inside the critical radius, leaving a buffer before a decision can change. Tether-TKL satisfies

$$R_t(\theta) \leq \tfrac{\eta}{1-\eta} \, C_t^{\mathrm{TKL}}(\theta) \tag{14}$$

(Appendix A.1), so it uses $G_t = C_t^{\mathrm{TKL}}$ in (8) with $\beta_{C^{\mathrm{TKL}}} = \eta/(1-\eta)$.

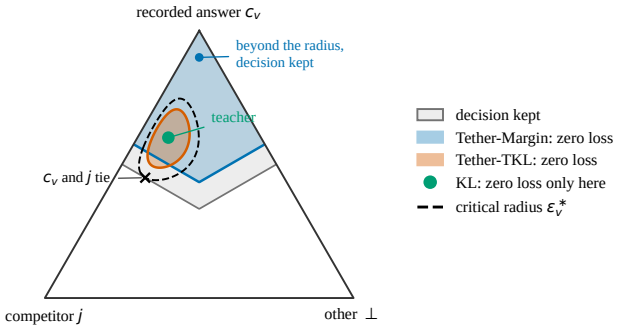


Figure 1: Zero-loss sets on one record of a task with two valid answers. Each point of the triangle is a probability assignment $(p(c_v), p(j), p(\perp))$ to the recorded answer, its competitor and the outcome $\perp$. The decision is kept above the grey line, where c_v is most probable. KL is zero only at the teacher (dot). The dashed curve is the critical radius; it touches the grey line where c_v and j tie. Tether-TKL is zero on the orange ball inside it; Tether-Margin is zero above the blue line, where c_v leads by at least κm_v, including points far beyond the critical radius. Here $\kappa = \eta = 0.5$.

2.4 Comparing the instances

Table 1 and Figure 1 summarise the three instances. Margin and Tether-TKL also vanish at the teacher distribution, so zero KL loss implies zero loss for both; unlike KL, they vanish on a region around it. The two regions are not nested in general (Appendix A.3). Records inside the region of Margin or Tether-TKL can be skipped before back-propagation, whereas KL can skip only records at the teacher distribution (Appendix B.3).

Table 1: Retention instances on a record v. The last two columns give a sufficient condition for $\hat{c}_\theta(v) = c_v$ and the resulting bound on the record-level drift $R_t(\theta)$.

Instance	Teacher information	Zero-loss set	Sufficient condition	Bound on R_t
Margin	c_v, m_v	$m_\theta(v; c_v) \geq \kappa m_v$	$\psi_v < 1$	C_t
KL	p_v°	$p_\theta(\cdot \mid v) = p_v^\circ$	$d_v < \varepsilon_v^*$	$\sum_v w_v d_v / \varepsilon_v^*$
Tether-TKL	p_v°	$d_v(\theta) \leq \tau_v$	$\psi_v^{\mathrm{TKL}} < 1/\eta - 1$	$\frac{\eta}{1-\eta} C_t^{\mathrm{TKL}}$

Training implications. Requiring zero retention loss may raise the minimum achievable new-task loss; we call the increase the *conflict* between the new task and retention. By the inclusion above, KL's conflict is at least that of Margin and Tether-TKL. For $\lambda > 0$, the conflict and the optimisation error jointly bound record drift at approximate global optima of (8) (Appendix A.4); with zero conflict, exact optimisation gives zero record drift.

2.5 From records to tasks

Training controls the record drift R_t, but the goal is a small drift F_t on all inputs of the task, and a small record drift does not by itself guarantee it.

Which inputs does record retention protect? An *assignment* $v(\cdot)$ pairs each input x of task t with a record $v(x) \in V_t$; the result below holds for every pairing, so it can be chosen to make the bound small. Write $c_x = \hat{c}_{\theta_t}(x)$ for the teacher's decision at x and $\Delta_\theta(x) = m_\theta(x; c_x) - m_{\theta_t}(x; c_x)$ for the change in its margin. The idea is to compare each input with its record: an input can lose its decision only if its record breaks (9), or if its own margin is too small for how far it moves.

Proposition 3 (record term and transfer term)**.** *For every measurable assignment $v(\cdot)$ and every θ,*

$$F_t(\theta) \leq \Pr\left[m_\theta(v(X); c_{v(X)}) < \kappa m_{v(X)}\right] + \Pr\left[m_{\theta_t}(X; c_X) \leq (1-\kappa) m_{v(X)} + \left|\Delta_\theta(X) - \Delta_\theta(v(X))\right|\right]. \tag{15}$$

Training on the records controls only the first term. The second term counts the inputs at risk even when every record holds: those whose teacher margin is no larger than the decrease allowed at their record, $(1-\kappa)m_{v(X)}$, plus the mismatch between their own margin change and their record's. It is small when the teacher is confident on the task and inputs change much like their records. A larger κ shrinks the allowed decrease but tightens the constraint on the new task (Appendix A.4). Appendix A.5 gives the proof and a variant whose first term is bounded by the Margin loss. Propositions 1 and 2 guarantee retention deterministically on the stored records; Theorem 1 (Appendix A.6) extends it to population drift only under the independent stratified record design of Assumption 1, discussed for our records in Remark 1; Proposition 5 below assesses population drift separately on independent held-out inputs.

Auditing task drift. Task drift F_t can be measured with an exact confidence bound. Let $u_1, \ldots, u_{n_a}$ be drawn independently from P_t, independently of θ_t and θ, and let K be the number of them on which $\hat{c}_\theta$ and $\hat{c}_{\theta_t}$ differ. Conditionally on the two models, $K \sim \mathrm{Bin}(n_a, F_t(\theta))$, so the one-sided Clopper–Pearson bound $\bar{F}_\delta(K, n_a)$ satisfies $\Pr[F_t(\theta) \le \bar{F}_\delta] \ge 1-\delta$ (Clopper & Pearson, 1934) (Proposition 5, Appendix A.7). By (4), the same bound holds for the accuracy decrease relative to the teacher under the same decision rule.

3 Experiments

3.1 Protocol

Benchmark, models and methods. We evaluate T5-large (Raffel et al., 2020) and Llama-2-7B-chat (Touvron et al., 2023) on the continual text-classification benchmark of Wang et al. (2023): three four-task orders (Orders 1–3, O1–3) and three fifteen-task orders (Orders 4–6, O4–6). All local methods train a fresh rank-8 LoRA (Hu et al., 2022) on the query and value projections for one epoch per task and retain earlier updates; data, task orders and hyperparameters are in Appendix B.1. The three Tether instances share 48 records per task, $m_{\min} = 0.2$, $\kappa = 0.5$, $\eta = 0.5$ and $\lambda = 1$, except Tether-TKL on T5-large, which uses $\lambda = 0.1$ (Section 3.3). The retention objective uses stored teacher outputs rather than ground-truth labels. Our implementation uses aggregate label-derived statistics to set record weights; individual ground-truth labels are not retained. Baselines are unprotected IncLoRA (Wang et al., 2023), labelled replay with a 2% buffer, and DER++ (Buzzega et al., 2020) adapted to LLMs (He et al., 2024).

Evaluation. Average accuracy (AA) is greedy exact match on the full test sets after the last task. Retention metrics compare the final model with each task's teacher on up to 2,000 fixed test inputs per task, under the decision rule of (2), averaged over earlier tasks: decision drift, harmful flips, repairs and the change in accuracy relative to the teacher (Appendix B.5); acquisition is the teacher's accuracy on these inputs, averaged over tasks. These inputs also form the audit sample of Section 2.5 ($\delta = 0.05$). Results average three orders and three seeds.

3.2 Main results

Final accuracy. Table 2 reports final average accuracy. On the fifteen-task sequences, Tether-KL is the most accurate method on both models (77.9, tied with Progressive Prompts on T5-large, and 83.4), above DER++ by 1.5 and 0.5 points; on the four-task sequences DER++ leads the best Tether instance by 0.1–0.2 points. Tether-Margin matches labelled replay in every setting, and with synthetic records its accuracy stays within one percentage point of the corresponding real-record result. Table 4 gives per-order results, and Table 9 compares reproduced baselines with published values.

Retention. Table 3 shows what accuracy does not: how many earlier decisions each method changes. Tether-KL changes about half as many as DER++ (6.7 against 12.9 on T5-large, 4.1 against 7.6 on Llama-2), and Tether-TKL is close to it on Llama-2. Tether-Margin's drift is broadly comparable to DER++'s: lower on the fifteen-task T5-large sequences (10.4 against 12.9) and 0.2–1.2 points higher in the other three settings; on the four-task sequences, Tether-TKL and Tether-KL also drift less than both baselines (Table 5). Old-task accuracy relative to the teacher falls by at most 0.6 points under Tether-KL, against 2.2 for DER++ and 10–18 unprotected. At the representative

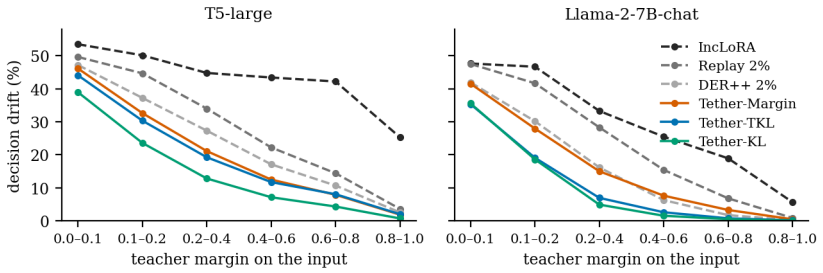


Figure 2: Decision drift on O4–6 by the teacher's margin (earlier tasks, all orders).

Table 2: Final average accuracy (%). IncLoRA, replay and DER++ are local; other baselines are published results (deviations in Appendix C). On the fifteen-task sequences (O4–6), Tether's 48 records per task total 720, about 1.3% of the 55,650 training examples.

T5-large				Llama-2-7B-chat			
Retained	Method	O1–3	O4–6	Retained	Method	O1–3	O4–6
None	IncLoRA	70.6	61.7	None	IncLoRA	79.9	73.5
Parameter / prompt state	O-LoRA	75.8	69.6	Parameter / prompt state	O-LoRA	77.2	73.5
	N-LoRA	78.5	72.3		InfLoRA	79.6	69.8
	GORP	79.8	76.0		MagMax	80.3	73.4
	Progressive Prompts	75.1	77.9		SLAO	80.4	74.8
Synthetic	LFPT5	72.7	69.2	Synthetic	SAPT-LoRA	81.1	81.9
Labelled examples	Replay, 2%	79.1	76.3	Labelled examples	Replay, 2%	81.2	81.4
	DER++, 2%	80.0	76.4		DER++, 2%	82.3	82.9
Tether, 48 records/task	Margin	78.9	76.8	Tether, 48 records/task	Margin	81.8	82.3
	TKL	79.2	76.9		TKL	82.1	82.5
	KL	79.8	77.9		KL	82.2	83.4
	Margin, synthetic	78.5	75.8		Margin, synthetic	81.8	81.5

settings in the sweep (Section 3.3), all three instances preserve every recorded decision. Their differences in held-out drift therefore arise on unrecorded inputs. On Llama-2, where they share $\lambda = 1$, drift is ordered $\mathrm{KL} \leq \mathrm{TKL} \leq \mathrm{Margin}$ on 14 of 15 tasks (Table 6), so the ordering in Table 3 is not an averaging artefact. Figure 2 shows where: near ties flip under every method and confident inputs rarely under any retention method; the instances separate at intermediate margin (0.2–0.4 and 0.4–0.6), where Margin drifts on 15% and 8% of Llama-2 inputs and KL on 5% and 2%. KL and Tether-TKL constrain the whole distribution around each record and carry such inputs with it; Margin constrains only the recorded lead, so nearby inputs with a smaller lead can flip (Proposition 3).

Audit. With up to 2,000 held-out inputs per task, the one-sided 95% upper bounds on task drift exceed the observed drift by about two points at the median: 6.9 for Tether-KL against 12.8 for DER++ on T5-large, and 5.0 against 8.2 on Llama-2 (Table 8).

Generation tasks. On six Llama-3-8B-Instruct generation tasks, the three instances score 62.1–63.0 (IncLoRA 58.8, DER++ 62.4) and keep 50–79% of earlier free-form outputs unchanged (DER++ 38%); KL changes 7.9% of earlier answers (DER++ 17.9%; Appendix F.4).

3.3 Two levers on drift beyond the records

Two controlled experiments on T5-large isolate both factors of the transfer term: how far a record's margin may move, set by λ, and how far any input is from a record, set by the record count n.

Retention strength. We sweep λ from Amazon to Yelp with a shared teacher and 48 records (Appendix D). At the representative settings—Margin with $\lambda = 1$, Tether-TKL with $\lambda = 0.1$, and

Table 3: Retention on O4–6 (%), averaged over earlier tasks and orders; columns as in Section 3.1, Time relative to IncLoRA (by task position: Table 7). Four-task sequences, repairs and further methods: Table 5.

	T5-large					Llama-2-7B-chat				
Method	Drift	Harmful	ΔAcc	Acq.	Time	Drift	Harmful	ΔAcc	Acq.	Time
IncLoRA	37.2	26.4	17.7	77.2	1.0×	19.9	14.4	10.1	83.2	1.0×
Replay, 2%	18.0	9.9	2.5	78.4	1.7×	12.7	6.8	1.6	82.8	2.0×
DER++, 2%	12.9	7.3	2.2	78.2	1.6×	7.6	3.7	0.2	83.0	1.9×
Tether-Margin	10.4	5.5	1.2	77.7	2.3×	7.8	4.0	0.6	82.8	3.1×
Tether-TKL	10.4	6.0	2.1	78.6	2.7×	4.9	2.3	-0.1	82.3	3.9×
Tether-KL	6.7	3.5	0.6	78.3	9.0×	4.1	1.9	-0.1	83.2	12.1×

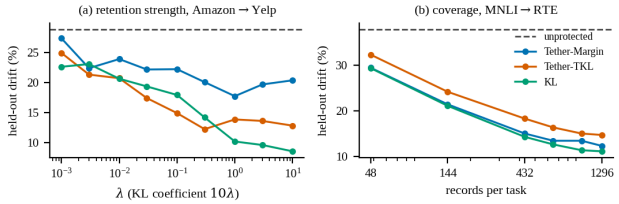


Figure 3: Held-out drift under the two levers (T5-large; dashed: unprotected). (a) Amazon drift after Yelp against λ (KL coefficient 10λ). (b) MNLI drift after RTE under the MNLI prompt against records per task (48 sampled per step). Details in Appendices D and E.

KL with coefficient 10—all three instances preserve every recorded decision, yet leave the records at different distances from the teacher: mean d_v/ε_v^* of 4.9 for Margin, 0.17 for Tether-TKL and 0.05 for KL. Under Margin, 30–37 of the 48 records sit outside their critical radius with the decision intact: the radius is sufficient, not necessary. Held-out drift persists and plateaus in the order KL, Tether-TKL, Margin, at 8.6%, 12–14% and 18–20% (Figure 3a), mostly on inputs that fail admission.

Coverage. With each instance at its safe λ and under conflict, an MNLI model learns RTE under the MNLI prompt, so that non-entailment pairs the teacher labels neutral must become contradiction (Appendix E). Unprotected training loses 27 points of MNLI accuracy. Growing the bank from 48 to 1,296 records per task lowers MNLI drift from about 30% to 11–15% and brings accuracy back to within 3.6 points of the teacher for Margin and KL, with diminishing returns beyond about 650 records (Figure 3b); at most 2.0% of records change their decision. Unlike the sweep, the new task now pays: RTE accuracy falls with the bank, by 3.3 points under Margin and 0.8 under KL, because each record also constrains the rule that RTE must change.

The price in new-task accuracy. Retention costs new-task accuracy once λ is large. On the sweep grid, Yelp accuracy stays within one point of unprotected training up to $\lambda = 1$ for Margin, $\lambda = 0.1$ for Tether-TKL and coefficient 10 for KL, and falls by 1–3 points at the next grid value. The two losses use different per-record normalisations. For an active record, Tether-TKL scales the divergence by $1/(\eta\varepsilon_v^*)$, whereas Margin scales the margin shortfall by $1/(\kappa m_v)$. Since $\varepsilon_v^* \geq m_v^2/2$, the ratio of these normalisation coefficients is at most $2\kappa/(\eta m_v)$; with $\kappa = \eta = 0.5$, this upper bound is 10 at the admission threshold $m_v = 0.2$. This comparison concerns the scalar normalisation coefficients rather than the resulting gradient magnitudes. Empirically, Tether-TKL begins to reduce new-task accuracy at a smaller coefficient in the T5 sweep, which motivates our use of $\lambda = 0.1$.

3.4 Trade-offs

Retention against learning. At the selected coefficients, retention cost new-task performance only under task conflict: on both benchmarks, acquisition (Tables 3 and 13) is at most 0.9 points below unprotected training for all instances, whereas under the MNLI–RTE conflict the cost tends to grow with the record count (Section 3.3).

Retention against computation. The instances that drift less compute more (Table 3). KL back-propagates through a third of the stored records at every step. Margin and Tether-TKL back-propagate only through records that violate their condition but screen all records, a forward cost that grows with their number: on sequences of up to five tasks they train faster than replay and DER++ (1.4–1.6× against 1.8–2.0× unprotected training on Llama-2), and only beyond about eight tasks do they cost more (Table 7). On Llama-2, Tether-TKL comes within a point of KL's drift at about a third of its training time; on generation it costs about as much as KL and drifts more. Margin is not uniformly better than DER++ in drift or speed. These comparisons follow the stated training schedules rather than matched compute budgets; times exclude evaluation and record construction.

4 Related work

Distillation and replay. Distillation (Hinton et al., 2015) and functional regularisation (Titsias et al., 2020; Pan et al., 2020; Benjamin et al., 2019) provide distributional or function-value targets. LwF distils on new-task inputs, whereas DER caches logits from past inputs (Li & Hoiem, 2018; Buzzega et al., 2020). Tether's KL instance uses the same forward-KL penalty, but on weighted records rather than new-task inputs and over the answer set and $\perp$ rather than the full output space. Pseudorehearsal preserves behaviour using model responses to artificial inputs (Robins, 1995), providing the precedent for synthetic records.

Constraints and margins. GEM constrains update directions with losses on labelled memories of each earlier task; A-GEM uses an averaged memory loss (Lopez-Paz & Ranzato, 2017; Chaudhry et al., 2019). Their constraints concern supervised losses, Tether's agreement with recorded teacher decisions. LUCIR applies a margin-ranking loss to labelled old-class exemplars (Hou et al., 2019); Tether-Margin sets its threshold relative to each teacher decision's measured margin.

Model-update compatibility. Model-update studies measure correct-to-incorrect regressions (Yan et al., 2021; Xie et al., 2021; Toneva et al., 2019). Churn can be controlled through rate constraints (Goh et al., 2016; Cotter et al., 2019) or distillation (Jiang et al., 2022). MUSCLE trains a compatibility adapter to reduce regressions when an LLM's base model is replaced (Echterhoff et al., 2024); margin-preserving training combines logit calibration and distillation for class-incremental image classification (Ricci et al., 2026). Building on prior work on prediction churn and negative flips, Tether uses disagreement with successive task-specific teachers as a retention target in continual fine-tuning. Its distinctive elements are the exact forward-KL boundary for preserving each recorded decision and the resulting objectives, whose nontrivial zero-loss regions let the output distribution change without changing that decision.

5 Conclusion and limitations

Tether turns decision retention into a training objective and a measured quantity. Drift on unrecorded inputs is set by two levers, how tightly each record is held and how densely records cover the task. Under the evaluated settings, residual drift concentrates on low-margin and excluded inputs, indicating limitations of the current admission and coverage scheme, and at the selected coefficients new-task accuracy is affected only under task conflict. The population guarantee of Theorem 1 requires the independent stratified record design of Assumption 1. Neither the training-data records nor the synthetic records used in our main experiments satisfy this design, so Theorem 1 does not cover the reported runs. Task-level drift is instead assessed directly using the held-out audit of Proposition 5, subject to its independence conditions. The guarantees bound the loss on old tasks; for the new task they bound only the excess loss, through the conflict gap. The generation experiment covers only short-prompt tasks and does not validate long-context generation. Retaining decisions also retains the teacher's errors, so repairs on old tasks fall with retention; when the new task contradicts an old rule, as in the MNLI–RTE experiment, retention needs a way to release the records it contradicts, which we do not provide.

AI USE STATEMENT

We used generative AI tools in four ways. First, to generate the synthetic inputs of the synthetic-record variant (Appendix B.6); the generator saw no benchmark example, its outputs were filtered by length, inter-field overlap and 8-gram deduplication against the task data, and its labels were discarded in favour of the teacher's decisions. Second, to tidy and organise the code for release. Third, to polish the writing and adjust formatting. Fourth, to search for related work and check bibliographic entries against their sources. The authors reviewed all AI-assisted material and take responsibility for the final content of this work.

ETHICS STATEMENT

This work adheres to the ICLR Code of Ethics. It involves no human subjects and uses public text-classification and generation benchmarks under their licenses, without personally identifiable information. Records store no per-example labels; synthetic records replace retained benchmark inputs but carry no formal privacy guarantee. Decision retention also preserves the teacher's errors and biases.

REPRODUCIBILITY STATEMENT

Section 2 defines the objective and retention conditions, and Appendix A gives the proofs. Section 3.1 and Appendix B specify the benchmark protocol, hyperparameters, hardware, record construction (including synthetic-record generation), screening and evaluation; Appendix C separates published from local results and reports variability; Appendices D and E give the controlled experiments. Anonymized code and configurations are in the supplementary material.

REFERENCES

Peter L. Bartlett and Shahar Mendelson. Rademacher and Gaussian complexities: Risk bounds and structural results. *Journal of Machine Learning Research*, 3:463–482, 2002.

Ari S. Benjamin, David Rolnick, and Konrad Kording. Measuring and regularizing networks in function space. In *International Conference on Learning Representations*, 2019.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, et al. Findings of the 2016 conference on machine translation. In *Proceedings of the First Conference on Machine Translation*, 2016.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. In *Advances in Neural Information Processing Systems*, volume 33, pp. 15920–15930. Curran Associates, Inc., 2020.

Arslan Chaudhry, Puneet K. Dokania, Thalaiyasingam Ajanthan, and Philip H. S. Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *International Conference on Learning Representations*, 2019.

Yulong Chen, Yang Liu, Liang Chen, and Yue Zhang. DialogSum: A real-life scenario dialogue summarization dataset. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pp. 5062–5074, 2021. doi: 10.18653/v1/2021.findings-acl.449.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

C. J. Clopper and E. S. Pearson. The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, 26(4):404–413, 1934. doi: 10.1093/biomet/26.4.404.

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. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Andrew Cotter, Heinrich Jiang, Maya Gupta, Serena Wang, Taman Narayan, Seungil You, and Karthik Sridharan. Optimization with non-differentiable constraints with applications to fairness, recall, churn, and other goals. *Journal of Machine Learning Research*, 20(172):1–59, 2019.

Cyprien de Masson d'Autume, Sebastian Ruder, Lingpeng Kong, and Dani Yogatama. Episodic memory in lifelong language learning. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Wenyu Du, Shuang Cheng, Tongxu Luo, Zihan Qiu, Zeyu Huang, Ka Chun Cheung, Reynold Cheng, and Jie Fu. Unlocking continual learning abilities in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 6503–6522. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.379. URL https://aclanthology.org/2024.findings-emnlp.379/.

Xinya Du, Junru Shao, and Claire Cardie. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1342–1352, 2017.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, et al. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Jessica Maria Echterhoff, Fartash Faghri, Raviteja Vemulapalli, Ting-Yao Hu, Chun-Liang Li, Oncel Tuzel, and Hadi Pouransari. MUSCLE: A model update strategy for compatible LLM evolution. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pp. 7320–7332. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-emnlp.430. URL https://aclanthology.org/2024.findings-emnlp.430/.

Gabriel Goh, Andrew Cotter, Maya Gupta, and Michael P. Friedlander. Satisfying real-world goals with dataset constraints. In *Advances in Neural Information Processing Systems*, volume 29, pp. 2415–2423. Curran Associates, Inc., 2016.

Hugo Hazard, Zafeirios Fountas, Martin A. Benfeghoul, Adnan Oomerjee, Jun Wang, and Haitham Bou-Ammar. SuRe: Surprise-driven prioritised replay for continual LLM learning. *arXiv preprint arXiv:2511.22367*, 2025.

Jinghan He, Haiyun Guo, Kuan Zhu, Zihan Zhao, Ming Tang, and Jinqiao Wang. SEEKR: Selective attention-guided knowledge retention for continual learning of large language models. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 3254–3266, 2024. URL https://aclanthology.org/2024.emnlp-main.190/.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. In *International Conference on Learning Representations*, 2021.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015. URL https://arxiv.org/abs/1503.02531.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 831–839, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022.

Jianheng Huang, Leyang Cui, Ante Wang, Chengyi Yang, Xinting Liao, Linfeng Song, Junfeng Yao, and Jinsong Su. Mitigating catastrophic forgetting in large language models with self-synthesized rehearsal. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 1416–1428. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.77. URL https://aclanthology.org/2024.acl-long.77/.

Heinrich Jiang, Harikrishna Narasimhan, Dara Bahri, Andrew Cotter, and Afshin Rostamizadeh. Churn reduction via distillation. In *International Conference on Learning Representations*, 2022.

James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017. doi: 10.1073/pnas.1611835114.

Vladimir Koltchinskii and Dmitry Panchenko. Empirical margin distributions and bounding the generalization error of combined classifiers. *The Annals of Statistics*, 30(1):1–50, 2002. doi: 10.1214/aos/1015362183.

Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(12):2935–2947, 2018. doi: 10.1109/TPAMI.2017.2773081.

Yan-Shuo Liang and Wu-Jun Li. InfLoRA: Interference-free low-rank adaptation for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23638–23647, 2024.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pp. 74–81, 2004.

David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Advances in Neural Information Processing Systems*, volume 30, 2017.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

Daniel Marczak, Bartłomiej Twardowski, Tomasz Trzciński, and Sebastian Cygert. MagMax: Leveraging model merging for seamless continual learning. In *Computer Vision – ECCV 2024*, volume 15143 of *Lecture Notes in Computer Science*, pp. 379–395, Cham, 2024. Springer. doi: 10.1007/978-3-031-73013-9_22.

Colin McDiarmid. On the method of bounded differences. In J. Siemons (ed.), *Surveys in Combinatorics, 1989*, number 141 in London Mathematical Society Lecture Note Series, pp. 148–188. Cambridge University Press, 1989. doi: 10.1017/CBO9781107359949.008.

Mahdi Milani Fard, Quentin Cormier, Kevin Canini, and Maya Gupta. Launch and iterate: Reducing prediction churn. In *Advances in Neural Information Processing Systems*, volume 29, 2016.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT Press, Cambridge, MA, second edition, 2018.

Pingbo Pan, Siddharth Swaroop, Alexander Immer, Runa Eschenhagen, Richard E. Turner, and Mohammad Emtiyaz Khan. Continual deep learning by functional regularisation of memorable past. In *Advances in Neural Information Processing Systems*, volume 33, pp. 4453–4464. Curran Associates, Inc., 2020.

Maja Popović. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pp. 392–395, 2015.

Fuli Qiao and Mehrdad Mahdavi. Merge before forget: A single LoRA continual learning via continual merging. In *International Conference on Learning Representations*, pp. 8133–8159, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/hash/0dedca10992d34035c9a9776254cfa8c-Abstract-Conference.html.

Chengwei Qin and Shafiq Joty. LFPT5: A unified framework for lifelong few-shot language learning based on prompt tuning of T5. In *International Conference on Learning Representations*, 2022.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, 2016.

Anastasia Razdaibiedina, Yuning Mao, Rui Hou, Madian Khabsa, Mike Lewis, and Amjad Almahairi. Progressive prompts: Continual learning for language models. In *International Conference on Learning Representations*, 2023.

Simone Ricci, Niccolò Biondi, Federico Pernici, and Alberto Del Bimbo. Mitigating negative flips via margin preserving training. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40 (11):8721–8730, 2026. doi: 10.1609/aaai.v40i11.37825.

Anthony Robins. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 7(2): 123–146, 1995. doi: 10.1080/09540099550039318.

Fan-Keng Sun, Cheng-Hao Ho, and Hung-Yi Lee. LAMOL: LAnguage MOdeling for lifelong language learning. In *International Conference on Learning Representations*, 2020.

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 4149–4158, 2019.

Michalis K. Titsias, Jonathan Schwarz, Alexander G. de G. Matthews, Razvan Pascanu, and Yee Whye Teh. Functional regularisation for continual learning with Gaussian processes. In *International Conference on Learning Representations*, 2020.

Mariya Toneva, Alessandro Sordoni, Remi Tachet des Combes, Adam Trischler, Yoshua Bengio, and Geoffrey J. Gordon. An empirical study of example forgetting during deep neural network learning. In *International Conference on Learning Representations*, 2019.

Hugo Touvron, Louis Martin, Kevin Stone, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

Chenxu Wang, Yilin Lyu, Zicheng Sun, and Liping Jing. Continual gradient low-rank projection fine-tuning for LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 14815–14829. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.721. URL https://aclanthology.org/2025.acl-long.721/.

Xiao Wang, Tianze Chen, Qiming Ge, Han Xia, Rong Bao, Rui Zheng, Qi Zhang, Tao Gui, and Xuanjing Huang. Orthogonal subspace learning for language model continual learning. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10658–10671. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.findings-emnlp.715. URL https://aclanthology.org/2023.findings-emnlp.715/.

Yuqing Xie, Yi-An Lai, Yuanjun Xiong, Yi Zhang, and Stefano Soatto. Regression bugs are in your model! Measuring, reducing and analyzing regressions in NLP model updates. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 6589–6602. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.515. URL https://aclanthology.org/2021.acl-long.515/.

Sijie Yan, Yuanjun Xiong, Kaustav Kundu, Shuo Yang, Siqi Deng, Meng Wang, Wei Xia, and Stefano Soatto. Positive-congruent training: Towards regression-free model updates. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14299–14308, 2021.

An Yang, Anfeng Li, Baosong Yang, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025a.

Shuo Yang, Kun-Peng Ning, Yu-Yang Liu, Jia-Yu Yao, Yong-Hong Tian, Yi-Bing Song, and Li Yuan. Is parameter collision hindering continual learning in LLMs? In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 4243–4259. Association for Computational Linguistics, 2025b. URL https://aclanthology.org/2025.coling-main.286/.

Pengcheng Yin, Bowen Deng, Edgar Chen, Bogdan Vasilescu, and Graham Neubig. Learning to mine aligned code and natural language pairs from Stack Overflow. In *Proceedings of the 15th International Conference on Mining Software Repositories*, pp. 476–486, 2018.

Weixiang Zhao, Shilong Wang, Yulin Hu, Yanyan Zhao, Bing Qin, Xuanyu Zhang, Qing Yang, Dongliang Xu, and Wanxiang Che. SAPT: A shared attention framework for parameter-efficient continual learning of large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 11641–11661. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.625. URL https://aclanthology.org/2024.acl-long.625/.

A PROOFS

Notation. For a fixed record v, let $p_v^\circ = p_{\theta_t}(\cdot \mid v)$ and $c = c_v$. Distributions p are on $\bar{\mathcal{L}}_t$. Their margin is $m(p) = p(c) - \max_{j \neq c} p(j)$, as in (6), with $m(p_v^\circ) = m_v > 0$. Write

$$\mathrm{kl}(u\|r) = u \log \frac{u}{r} + (1-u) \log \frac{1-u}{1-r}$$

for the Bernoulli KL divergence, and $[z]_+ = \max\{z, 0\}$. All logarithms are natural. We use the extended-value conventions $0 \log(0/r) = 0$ and $a \log(a/0) = \infty$ for $a > 0$. Weighted sums omit zero-weight terms, including terms with infinite divergence.

Define the sets $B_{v,\varepsilon} = \{p : \mathrm{KL}(p_v^\circ \| p) \leq \varepsilon\}$ and $M_{v,\kappa} = \{p : m(p) \geq \kappa m_v\}$. We abbreviate $B_v = B_{v,\tau_v}$ and $M_v = M_{v,\kappa}$; Table 1 states them as conditions.

A.1 RECORD-LEVEL BOUNDS

Write $\mu_t(v) = w_v$ for the record weights viewed as a distribution on V_t, and let $I_v(\theta) = \mathbf{1}[\hat{c}_\theta(v) \neq c_v]$. Then $R_t(\theta) = \mathbb{E}_{v \sim \mu_t}[I_v(\theta)]$.

Margin. If $I_v = 1$, then $m_\theta(v; c_v) \leq 0$. Since $\kappa m_v > 0$, (10) gives $\psi_v \geq 1$. If $I_v = 0$, nonnegativity gives the same pointwise bound $I_v \leq \psi_v$. Taking the weighted average proves $R_t \leq C_t$ and $\beta_C = 1$. Also, $\psi_v < 1$ implies $m_\theta(v; c_v) > 0$, so the recorded answer remains the unique maximiser.

KL. Proposition 2 gives $I_v = 1 \Rightarrow d_v \geq \varepsilon_v^*$. Therefore

$$I_v(\theta) \leq \frac{d_v(\theta)}{\varepsilon_v^*}, \qquad R_t(\theta) \leq \sum_{v \in V_t} w_v \frac{d_v(\theta)}{\varepsilon_v^*}.$$

Admission ensures $m_v \geq m_{\min} > 0$. The radius bound gives $\varepsilon_v^* \geq m_v^2/2 \geq m_{\min}^2/2$. Substitution proves (12) and $\beta_{\mathcal{K}} = 2/m_{\min}^2$.

Tether-TKL. The piecewise loss in (13) equals $[d_v/\tau_v - 1]_+$. If $I_v = 1$, then $d_v \geq \varepsilon_v^*$ and

$$\psi_v^{\mathrm{TKL}} \geq \frac{\varepsilon_v^*}{\eta \varepsilon_v^*} - 1 = \frac{1-\eta}{\eta} > 0.$$

Consequently, $I_v \leq \frac{\eta}{1-\eta} \psi_v^{\mathrm{TKL}}$. Taking the weighted average proves (14) and $\beta_{C^{\mathrm{TKL}}} = \eta/(1-\eta)$. The strict per-record condition is $\psi_v^{\mathrm{TKL}} < 1/\eta - 1$, as in Table 1. The constant diverges as $\eta \uparrow 1$: at $\eta = 1$ the zero-loss ball includes a decision tie, so retention is no longer guaranteed by the radius condition.

Clipped bounds. Each indicator is at most one, so each per-record term above may be clipped at one before averaging; unlike Theorem 1, these bounds need no sampling assumption on the records.

A.2 Proof of Proposition 2

Let $a = p_v^\circ(c)$ and $b = \max_{j \neq c} p_v^\circ(j)$, so $m_v = a - b > 0$.

A fixed competitor. Fix $j \neq c$ and let $b_j = p_v^\circ(j)$. Since $m_v = a - b > 0$, we have $b_j \leq b < a$. Consider a distribution p with $p(j) \geq p(c)$ and finite forward KL. Finiteness implies $p(c) > 0$. Set

$$s = a + b_j, \qquad u = \frac{a}{s} > \tfrac{1}{2}, \qquad z = p(c) + p(j), \qquad r = \frac{p(c)}{z} \leq \tfrac{1}{2}.$$

Coarsen the outcomes into $\{c\}$, $\{j\}$ and their complement. Data processing and the KL chain rule give

$$\mathrm{KL}(p_v^\circ \| p) \geq \mathrm{kl}(s \| z) + s\,\mathrm{kl}(u \| r).$$

The first term is nonnegative. The second is minimised at $r = 1/2$, because $\partial_r \mathrm{kl}(u\|r) = (r - u)/\{r(1-r)\} < 0$ for $0 < r \leq 1/2 < u$. Thus

$$\mathrm{KL}(p_v^\circ \| p) \geq s\,\mathrm{kl}(u \| \tfrac{1}{2}) = a \log \frac{2a}{a + b_j} + b_j \log \frac{2b_j}{a + b_j} =: \varepsilon_j.$$

Equality is attained at

$$p_j^*(c) = p_j^*(j) = \tfrac{a+b_j}{2}, \qquad p_j^*(\ell) = p_v^\circ(\ell) \;\; (\ell \neq c, j), \tag{16}$$

so $\varepsilon_j = \min_{p:\, p(j) \geq p(c)} \mathrm{KL}(p_v^\circ \| p)$.

Minimum over competitors. The set $\{p : m(p) \leq 0\}$ is the union of the sets $\{p : p(j) \geq p(c)\}$ over $j \neq c$. For $0 < b_j < a$,

$$\frac{\partial \varepsilon_j}{\partial b_j} = \log \frac{2b_j}{a + b_j} < 0.$$

Continuity extends the ordering to $b_j = 0$. The minimum therefore occurs at a strongest competitor, with $b_j = b$. This proves the exact identity

$$\min_{p:\, m(p) \leq 0} \mathrm{KL}(p_v^\circ \| p) = \varepsilon_v^*.$$

Every distribution with smaller divergence has $m(p) > 0$ and retains the unique recorded decision.

Sharpness. Choose a strongest competitor j and its minimiser p_j^* in (16). At this distribution, c and j tie for the maximum. For $0 < h < (a + b)/2$, define $p_h = p_j^* + h(e_j - e_c)$, where e_i is unit mass at outcome i. Then $p_h(j) > p_h(c)$, so the decision differs from c. Also, $\mathrm{KL}(p_v^\circ \| p_h) \to \varepsilon_v^*$ as $h \downarrow 0$. Every larger open KL ball therefore contains a changed decision, regardless of the tie rule.

Lower bound. For the strongest competitor, set $s = a + b$ and $u = a/s$. Pinsker's inequality for Bernoulli distributions gives

$$\varepsilon_v^* = s\,\mathrm{kl}(u \| \tfrac{1}{2}) \geq 2s(u - \tfrac{1}{2})^2 = \frac{m_v^2}{2(a+b)} \geq \frac{m_v^2}{2}.$$

The last inequality uses $a + b \leq 1$. Boundary cases with zero probabilities or $s = 1$ follow by continuity. □

Full-sequence KL. Since (1) is a fixed map of output sequences, the data-processing inequality gives $d_v(\theta) \leq \mathrm{KL}\left(q_{\theta_t}(\cdot \mid v) \,\|\, q_\theta(\cdot \mid v)\right)$ at temperature one. Full-sequence forward KL below ε_v^* therefore also guarantees the recorded decision.

A.3 Zero-loss sets and their relations

The Margin zero-loss set. The unique-decision set is $M_{v,+} = \{p : m(p) > 0\}$. For $0 < \kappa \leq 1$, $M_{v,\kappa}$ is the probability simplex intersected with the half-spaces $p(c) - p(j) \geq \kappa m_v$, $j \neq c$. It is a closed convex polytope, contains p_v°, and lies inside $M_{v,+}$.

KL zero-loss sets and a sufficient inclusion. KL is zero exactly at p_v°, while Tether-TKL is zero exactly on B_{v,τ_v}. For any competitor j, the change in $p(c) - p(j)$ is bounded below by $-\|p - p_v^\circ\|_1 = -2\,\mathrm{TV}(p, p_v^\circ)$, with equality when probability moves from c to j, so the constant cannot be improved. Consequently,

$$m(p) \geq m_v - \|p - p_v^\circ\|_1 \geq m_v - \sqrt{2\,\mathrm{KL}(p_v^\circ \| p)},$$

and therefore

$$B_{v,\varepsilon} \subseteq M_{v,\kappa} \quad \text{if} \quad 0 \leq \varepsilon \leq \frac{(1-\kappa)^2 m_v^2}{2}. \tag{17}$$

Changes allowed by one condition and excluded by the other. Suppose $a = p_v^\circ(c) < 1$ and consider $p_s = (1-s)p_v^\circ + s\delta_c$ for $0 \leq s < 1$. Its margin is $m(p_s) = (1-s)m_v + s \geq m_v$, so $p_s \in M_{v,\kappa}$, but

$$\mathrm{KL}(p_v^\circ \| p_s) = a \log \frac{a}{a + s(1-a)} - (1-a)\log(1-s) \longrightarrow \infty \quad \text{as } s \uparrow 1.$$

Thus $M_{v,\kappa}$ is not contained in any finite-radius forward-KL ball.

In the other direction, let $p^* = p_j^*$ of (16) with $b_j = b$ and $p_s = (1-s)p_v^\circ + sp^*$. By convexity of $\mathrm{KL}(p_v^\circ \| \cdot)$, $\mathrm{KL}(p_v^\circ \| p_s) \leq s\,\varepsilon_v^*$, while $m(p_s) \leq p_s(c) - p_s(j) = (1-s)m_v$. Given $\varepsilon > (1-\kappa)\varepsilon_v^*$, the choice $s = \min\{1, \varepsilon/\varepsilon_v^*\} > 1 - \kappa$ gives $p_s \in B_{v,\varepsilon}$ with $m(p_s) < \kappa m_v$. In particular, radii in $((1-\kappa)\varepsilon_v^*, \varepsilon_v^*)$ give balls that preserve the recorded decision but are not contained in $M_{v,\kappa}$.

These inclusions transfer to parameter space through $\theta \mapsto p_\theta(\cdot \mid v)$, although a strict difference in output space need not remain strict for a particular model.

A.4 Retention at approximate optima

For the finite training data, a parameter with finite logits gives positive probability to every finite target sequence and hence finite cross-entropy. Since the loss is nonnegative, $\bar{\mathcal{L}} := \inf_\theta \mathcal{L}_k(\theta)$ satisfies $0 \leq \bar{\mathcal{L}} < \infty$.

Conflict gap. Write $S_G(\theta) = \sum_{t<k} G_t(\theta) \geq 0$ and define

$$\Omega_k^G = \{\theta : S_G(\theta) = 0\}, \qquad \Gamma_k^G = \inf_{\theta \in \Omega_k^G} \mathcal{L}_k(\theta) - \bar{\mathcal{L}} \geq 0.$$

Set $\Gamma_k^G = \infty$ if Ω_k^G is empty. This *conflict gap* is the optimal extra new-task loss required to achieve zero retention loss.

Calibration. The framework requires $G_t(\theta_t) = 0$. Without it, a constant loss $G_t \equiv 1$ would satisfy (7) with $\beta_G = 1$ while having an empty zero-loss set, so the conflict gap would be infinite and the bound at approximate optima vacuous. All three instances are calibrated: at θ_t, Margin's margin equals $m_v \geq \kappa m_v$, and both d_v and ψ_v^{TKL} vanish. Calibration alone does not make every conflict gap finite. It places the teacher θ_t in the zero-loss set of task t, so $\theta_1 \in \Omega_2^G$ and $\Gamma_2^G \leq \mathcal{L}_2(\theta_1) - \bar{\mathcal{L}} < \infty$, since the cross-entropy is finite at parameters with finite logits. For $k > 2$, $\theta_{k-1} \in \Omega_k^G$, and hence $\Gamma_k^G < \infty$, whenever θ_{k-1} also has zero loss on the records of tasks before $k-1$.

Comparison of the gaps. Zero KL loss implies zero Margin and Tether-TKL loss, giving

$$\Omega_k^{\mathcal{K}} \subseteq \Omega_k^C \cap \Omega_k^{C^{\mathrm{TKL}}}, \qquad \Gamma_k^C, \Gamma_k^{C^{\mathrm{TKL}}} \leq \Gamma_k^{\mathcal{K}}.$$

With records and weights fixed, increasing κ shrinks Ω_k^C. Hence Γ_k^C is nondecreasing in κ.

Proposition 4 (retention at approximate optima)**.** *Let $\lambda > 0$, $\xi \geq 0$ and $J_k^G(\hat{\theta}_G) \leq \inf_\theta J_k^G(\theta) + \xi$. Then*

$$\mathcal{L}_k(\hat{\theta}_G) - \inf_\theta \mathcal{L}_k(\theta) + \lambda \textstyle\sum_{t<k} G_t(\hat{\theta}_G) \leq \Gamma_k^G + \xi. \tag{18}$$

Proof. The penalty vanishes on Ω_k^G, so $J_k^G = \mathcal{L}_k$ there. Thus

$$J_k^G(\hat{\theta}_G) \leq \inf_\theta J_k^G(\theta) + \xi \leq \inf_{\theta \in \Omega_k^G} \mathcal{L}_k(\theta) + \xi = \bar{\mathcal{L}} + \Gamma_k^G + \xi.$$

Subtracting $\bar{\mathcal{L}}$ gives (18). Both terms on its left are nonnegative, so each is bounded by $\Gamma_k^G + \xi$. □

Zero conflict and exact optima. If $\Gamma_k^G = 0$, then $\inf_\theta J_k^G(\theta) = \bar{\mathcal{L}}$. Every exact minimiser therefore has $\mathcal{L}_k = \bar{\mathcal{L}}$ and $S_G = 0$, since both terms in (18) are nonnegative. Condition (7) then gives $R_t = 0$ for every earlier task, preserving all positive-weight decisions. The gap ordering above shows that $\Gamma_k^{\mathcal{K}} = 0$ gives zero conflict for all three instances.

Drift at approximate optima. For any instance satisfying (7), dropping the nonnegative new-task term in (18) gives

$$\sum_{t<k} R_t(\hat{\theta}_G) \le \beta_G \sum_{t<k} G_t(\hat{\theta}_G) \le \frac{\beta_G(\Gamma_k^G + \xi)}{\lambda}.$$

Substituting the constants for Margin, KL and Tether-TKL yields

$$\begin{aligned} \text{Margin:} \quad & \textstyle\sum_{t<k} R_t(\hat{\theta}_C) \le \dfrac{\Gamma_k^C + \xi}{\lambda}, \\ \text{KL:} \quad & \textstyle\sum_{t<k} R_t(\hat{\theta}_{\mathcal{K}}) \le \dfrac{2(\Gamma_k^{\mathcal{K}} + \xi)}{\lambda m_{\min}^2}, \\ \text{Tether-TKL:} \quad & \textstyle\sum_{t<k} R_t(\hat{\theta}_{C^{\mathrm{TKL}}}) \le \dfrac{\eta}{1-\eta} \cdot \dfrac{\Gamma_k^{C^{\mathrm{TKL}}} + \xi}{\lambda}. \end{aligned} \tag{19}$$

A.5 Transfer beyond the records

Tolerance form. For every measurable assignment $v(\cdot)$, every θ and every $\zeta \in [0, 1)$,

$$\begin{aligned} F_t(\theta) \le & \Pr_{X \sim P_t}\left[\psi_{v(X)}(\theta) > \zeta\right] \\ & + \Pr_{X \sim P_t}\left[m_{\theta_t}(X; c_X) \le \{1 - (1-\zeta)\kappa\} m_{v(X)} + \left|\Delta_\theta(X) - \Delta_\theta(v(X))\right|\right]. \end{aligned} \tag{20}$$

Proposition 3 is the case $\zeta = 0$. The statement covers all task inputs, including those excluded by record admission.

Proof. Fix $\zeta \in [0, 1)$, write $v_x = v(x)$ and suppose $\psi_{v_x}(\theta) \le \zeta$. By (10), $m_\theta(v_x; c_{v_x}) \ge (1 - \zeta)\kappa m_{v_x}$, that is, $\Delta_\theta(v_x) \ge -\{1 - (1-\zeta)\kappa\} m_{v_x}$. Then

$$m_\theta(x; c_x) = m_{\theta_t}(x; c_x) + \Delta_\theta(x) \;\ge\; m_{\theta_t}(x; c_x) - \{1 - (1-\zeta)\kappa\} m_{v_x} - \left|\Delta_\theta(x) - \Delta_\theta(v_x)\right|,$$

and the decision at x is kept when the right-hand side is positive. A changed decision therefore requires $\psi_{v(x)}(\theta) > \zeta$ or a non-positive right-hand side. The union bound gives (20). At $\zeta = 0$, $\psi_v > 0$ is equivalent to $m_\theta(v; c_v) < \kappa m_v$, giving (15). □

Relation to the training loss. Write $\rho_u := P_t(v(X) = u)$ for the mass assigned to record $u \in V_t$. For $\zeta > 0$, the first term of (20) is at most $\sum_{v \in V_t} \rho_v \psi_v(\theta)/\zeta$. If $\rho_u = w_u$ for every $u \in V_t$, this gives

$$\Pr_{X \sim P_t}\left[\psi_{v(X)}(\theta) > \zeta\right] \le C_t(\theta)/\zeta. \tag{21}$$

For general assignments, the first term uses the masses ρ_v, which may differ from the training weights. A record with $w_v = 0$ and $\rho_v > 0$ contributes to this term but not to C_t. Thus $C_t = 0$ makes the first term vanish if every record with positive assigned mass has positive weight.

Since the bound holds for every assignment, it may be chosen after training, in which case $\rho = w$ need not hold. The bound applies to a model trained with any instance, through the margins it attains on the records.

A.6 Independent stratified records

With the admission threshold $m_{\min}$, write $m_t(x) = m_{\theta_t}(x; c_x)$, $A_t = \{x : c_x \in \mathcal{L}_t,\ m_t(x) \ge m_{\min}\}$ for the admissible inputs, $\alpha_t = P_t(A_t^c)$, $P_t^A = P_t(\cdot \mid A_t)$, $\pi_c = P_t^A(c_x = c)$, and on A_t the margin ratio $h_\theta(x) = m_\theta(x; c_x)/m_t(x)$, so that $\psi_x(\theta) = [1 - h_\theta(x)/\kappa]_+$ extends (10) to A_t.

Assumption 1 (independent stratified design)**.** *Conditionally on θ_t and on the allocation $(n_c)_c$, the records are independent, those of class c being $n_c \geq 1$ draws from $P_t(\cdot \mid A_t, c_x = c)$ for every c with $\pi_c > 0$, and $w_v = \pi_{c_v}/n_{c_v}$.*

Define the effective sample size $n_{\text{eff}} = (\sum_v w_v^2)^{-1}$.

Theorem 1 (margin bound)**.** *Let Θ be a parameter set that does not depend on V_t, $\mathcal{H} = \{h_\theta : \theta \in \Theta\}$, and $\mathfrak{R}_w(\mathcal{H}) = \mathbb{E}\sup_{\theta\in\Theta}\sum_v w_v\sigma_v h_\theta(v)$, where the expectation is over the records and independent uniform signs σ_v. Under Assumption 1, with probability at least $1-\delta$ over V_t, simultaneously for all $\theta \in \Theta$,*

$$F_t^A(\theta) := \Pr_{X\sim P_t^A}\left[\hat{c}_\theta(X) \neq c_X\right] \leq \sum_v w_v \min\{1, \psi_v(\theta)\} + \frac{2}{\kappa}\mathfrak{R}_w(\mathcal{H}) + \sqrt{\frac{\log(1/\delta)}{2n_{\text{eff}}}}, \tag{22}$$

and $F_t(\theta) \leq \alpha_t + (1-\alpha_t)F_t^A(\theta)$.

Proof. Let $\phi_\kappa(u) = \min\{1, [1-u/\kappa]_+\}$ and $g_\theta = \phi_\kappa \circ h_\theta \in [0,1]$. A changed decision at $x \in A_t$ means $h_\theta(x) \leq 0$, so $F_t^A(\theta) \leq \mathbb{E}_{P_t^A} g_\theta$. Under the assumption $\sum_v w_v g(v) = \sum_c \pi_c \bar{g}_c$, with $\bar{g}_c$ the mean of n_c independent draws from stratum c, is unbiased for $\mathbb{E}_{P_t^A} g$. Let $D = \sup_\theta\{\mathbb{E}_{P_t^A} g_\theta - \sum_v w_v g_\theta(v)\}$. Replacing one record changes D by at most w_v, so $D \leq \mathbb{E}D + \{\log(1/\delta)/(2n_{\text{eff}})\}^{1/2}$ with probability $1-\delta$ (McDiarmid, 1989). Symmetrisation with a ghost sample from the same strata and the contraction inequality (Mohri et al., 2018, Lemma 5.7), applied to the $(1/\kappa)$-Lipschitz maps $u \mapsto w_v\phi_\kappa(u/w_v)$ (records of weight zero drop out), give $\mathbb{E}D \leq 2\mathfrak{R}_w(\phi_\kappa \circ \mathcal{H}) \leq (2/\kappa)\mathfrak{R}_w(\mathcal{H})$, as in the classical margin bounds (Koltchinskii & Panchenko, 2002; Bartlett & Mendelson, 2002). The last claim splits F_t over A_t and its complement. □

Remark 1. (a) *Design.* Our records are drawn from the teacher's training data instead: the real-input construction samples without replacement within teacher-predicted classes of a candidate pool taken from that data (Appendix B.4), and synthetic records are generated. Neither provides the independent task-distribution draws of Assumption 1, so the theorem does not cover the reported runs; their task-level drift is assessed by the audit on held-out inputs, and the deterministic record guarantees apply to both input sources. The assumption is met by drawing records from inputs outside the teacher's training data, such as a held-out split of task t, from the stated conditional distributions after fixing the teacher and allocation.

(b) *Other class proportions.* If the weights use proportions $\tilde{\pi}$ fixed independently of the records, $\tilde{w}_v = \tilde{\pi}_{c_v}/n_{c_v}$, the theorem applies to the mixture $\sum_c \tilde{\pi}_c P_t(\cdot \mid A_t, c_x = c)$ with every term computed from $\tilde{w}$. Since two mixtures of the same strata differ by at most $\text{TV}(\tilde{\pi}, \pi)$ on $[0,1]$-valued functions, (22) then holds for F_t^A with the additional term $\text{TV}(\tilde{\pi}, \pi)$. The implemented weights estimate $\tilde{\pi}$ from aggregate training-label statistics (Appendix B.4).

(c) *Rate.* $n_{\text{eff}} \leq n$, with equality if and only if $n_c = n\pi_c$ (Cauchy–Schwarz). The sampling term is of order $n_{\text{eff}}^{-1/2}$; for $n_{\text{eff}} = 46$–47 it is approximately 0.18 at 95% confidence. The Rademacher term is a separate quantity.

A.7 Audit of task drift

Proposition 5 (label-free audit)**.** *Let $u_1, \ldots, u_{n_a}$, $n_a \geq 1$, be independent draws from P_t, and let θ_t and θ be fixed independently of them; in particular, these inputs are not used for training or model selection. Conditionally on (θ_t, θ), the number K of disagreements is $\text{Bin}(n_a, F_t(\theta))$. For $0 < \delta < 1$, the one-sided Clopper–Pearson bound satisfies $\Pr[F_t(\theta) \leq \bar{F}_\delta(K, n_a)] \geq 1-\delta$.*

Proof. Conditionally on the fixed pair (θ_t, θ), each disagreement indicator on an independent input from P_t is Bernoulli with parameter $F_t(\theta)$. Their sum is therefore $\text{Bin}(n_a, F_t(\theta))$. For $n_a \geq 1$ and $0 < \delta < 1$, define

$$\bar{F}_\delta(k, n_a) = \begin{cases} \text{Beta}^{-1}(1-\delta;\, k+1, n_a-k), & 0 \leq k < n_a, \\ 1, & k = n_a, \end{cases} \tag{23}$$

where $\text{Beta}^{-1}(q; a, b)$ is the q-quantile of the beta distribution with parameters (a, b). For $k < n_a$, this is the unique p with $\Pr_{Z\sim\text{Bin}(n_a,p)}(Z \leq k) = \delta$. Inverting these one-sided binomial tests yields

$\Pr[F_t(\theta) \le \bar{F}_\delta(K, n_a)] \ge 1 - \delta$ (Clopper & Pearson, 1934). The inequality allows conservatism from the discrete count. With no disagreement, $\bar{F}_\delta(0, n_a) = 1 - \delta^{1/n_a}$. For several pre-specified audits, choose error probabilities summing to at most δ; a union bound gives simultaneous coverage without requiring independence between audits. □

B Implementation and record construction

B.1 Main experiment settings

Data and task orders. We use the released O-LoRA training files: up to 1,000 examples per class, with CB and COPA used in full, for a total of 55,650 training examples per fifteen-task order. The four-task orders contain DBpedia, Amazon, Yahoo Answers and AG News. The long orders additionally include MNLI, CB, WiC, COPA, QQP, BoolQ, RTE, IMDB, Yelp, SST-2 and MultiRC. The six orders are:

1. DBpedia, Amazon, Yahoo Answers, AG News.
2. DBpedia, Amazon, AG News, Yahoo Answers.
3. Yahoo Answers, Amazon, AG News, DBpedia.
4. MNLI, CB, WiC, COPA, QQP, BoolQ, RTE, IMDB, Yelp, Amazon, SST-2, DBpedia, AG News, MultiRC, Yahoo Answers.
5. MultiRC, BoolQ, WiC, MNLI, CB, COPA, QQP, RTE, IMDB, SST-2, DBpedia, AG News, Yelp, Amazon, Yahoo Answers.
6. Yelp, Amazon, MNLI, CB, COPA, QQP, RTE, IMDB, SST-2, DBpedia, AG News, Yahoo Answers, MultiRC, BoolQ, WiC.

Training. All local comparisons use Adam, LoRA rank 8, scaling $\alpha = 32$, and LoRA dropout 0.1 on query and value projections. T5 retains its native dropout during task training. The training data and prompts are shared across methods within each model. T5 uses learning rate 10^{-3} and effective batch size 64 (micro-batches of 16); Llama-2 uses 10^{-4} and batch size 8. Each task is trained for one epoch. Records are screened every step for T5 and short Llama-2 sequences, and every eighth step for long Llama-2 sequences, with intermediate low-slack screening as described in Appendix B.3. All experiments run on eight NVIDIA RTX 6000 Ada Generation GPUs (48 GB each), with each run on a single GPU.

Local baselines. IncLoRA trains a fresh adapter at each task and retains all earlier updates without an output-retention loss. Labelled replay stores a class-balanced 2% sample from each task and adds eight buffered examples to each training micro-batch; on T5 (micro-batches of 16) the replay loss is divided by twice the batch denominator, so the eight rows enter at half weight, whereas on Llama-2 (micro-batches of 8) they enter at equal weight. We run the LLM adaptation of DER++ (Buzzega et al., 2020) from He et al. (2024) with the same buffer and replay schedule: it stores full-vocabulary teacher logits at the answer positions when a task ends and weights supervised replay and teacher-distribution cross-entropy equally. O-LoRA (Wang et al., 2023) and LwF (Li & Hoiem, 2018) are our implementations in the same code base. KL uses $\lambda = 1$ and the same record budget and construction protocol as Tether-Margin; at each step it evaluates 16 of the 48 records of every earlier task, drawn uniformly, with weights rescaled so that the loss estimate is unbiased. Synthetic records are built as in Appendix B.6.

Evaluation and aggregation. AA uses deterministic greedy decoding on every task's full test set after the final task. Decision drift, harmful flips and repairs in Table 5 use complete-answer scoring on the same fixed subset of at most 2,000 test inputs per task, sampled with seed 12345. Teacher decisions are cached when each task ends. Acquisition averages their accuracy over all tasks; drift and the two flip types average over earlier tasks. A change between two incorrect answers contributes to drift but to neither harmful flips nor repairs. For both models, AA and the decision metrics of each configuration come from the same runs, which score decisions with complete answers. The reported test-set rates are descriptive estimates.

Published comparisons. The T5 values of SeqFT, SeqLoRA, IncLoRA, EWC (Kirkpatrick et al., 2017), LwF, L2P, full-model replay, LFPT5, O-LoRA, Progressive Prompts (Razdaibiedina et al., 2023), per-task fine-tuning and multi-task learning are from Wang et al. (2023); N-LoRA, GORP, LoRA replay with MIGU and SuRe are from Yang et al. (2025b); Wang et al. (2025); Du et al. (2024); Hazard et al. (2025), respectively. The Llama-2 comparison values, including SeqLoRA, InfLoRA (Liang & Li, 2024), SAPT-LoRA (Zhao et al., 2024), CorDA and multi-task learning, are those reported by Qiao & Mahdavi (2026). These rows retain the source papers' training and evaluation protocols; the per-order values appear in Table 4.

B.2 Scoring and training

Scoring. We evaluate (1) by teacher forcing on each canonical sequence $s_t(c)\,\mathrm{EOS}$. Alternative tokenisations and verbalisations belong to $\perp$. Each task trains a fresh low-rank adapter, which is merged after training. The record conditions depend on the output probabilities and apply to other parameterisations as well.

Loss normalisation. For an effective batch $\mathcal{B}$, the implementation sums token cross-entropies over the canonical targets. The sum includes EOS. It is divided by

$$Z_{\mathcal{B}} = |\mathcal{B}| \max\left\{1, \left\lfloor \frac{1}{|\mathcal{B}|} \sum_{(x,y)\in\mathcal{B}} \ell_y \right\rfloor\right\}, \qquad \ell_y = |s_k(y)| + 1.$$

Thus the denominator uses the batch size and the floored mean target length. All micro-batches in the update share this denominator.

Stored information and checks. KL and Tether-TKL cache a teacher distribution with $|\mathcal{L}_t| + 1$ entries per record (Table 1). After each task, we evaluate all earlier records. We log retention loss and weighted decision drift, together with unweighted counts of violated conditions and changed decisions.

B.3 Screening and cost

Exact screening. For Margin, let $V_{<k} = \bigcup_{t<k} V_t$ and define the active set $\mathcal{A}(\theta) = \{v \in V_{<k} : \psi_v(\theta) > 0\}$. At differentiable points,

$$\nabla_\theta \textstyle\sum_{t<k} C_t(\theta) = -\sum_{v\in\mathcal{A}(\theta)} \frac{w_v}{\kappa m_v} \nabla_\theta m_\theta(v; c_v). \tag{24}$$

At ties among competitors, we use the generalized derivative of the branch selected by the fixed tie rule. At a hinge boundary, we choose the zero subgradient.

Each step scores all stored records without a computation graph, recomputes the violated penalties with a graph, and back-propagates through those records only (Algorithm 1). Both passes use evaluation mode without dropout at the same parameters, so this implements (24). Tether-TKL uses the same procedure with the active condition $d_v > \tau_v$.

Zero-loss screening excludes a KL record only when $d_v = 0$. Even with $d_v > 0$, the KL parameter gradient can vanish through the model Jacobian. Gradients from different records can also cancel.

Record subsampling. Uniformly subsampling n_{sub} out of n records gives the unbiased estimator $(n/n_{\mathrm{sub}}) \sum_{v\in\mathcal{S}} w_v \ell_v$ for any instance's per-record loss ℓ_v, where $\mathcal{S}$ is the sampled set and each record has inclusion probability n_{sub}/n. Subsampling selects records for evaluation; screening selects evaluated records for back-propagation.

Periodic screening. For Margin, let $s \geq 1$ be the screening period and $\varsigma \geq 0$ a slack threshold. At every full screening, score all records and form the set of violated records together with satisfied records whose relative slack

$$\chi_v(\theta) = \frac{m_\theta(v; c_v) - \kappa m_v}{\kappa m_v}$$

is below ς. Until the next full screening, score only that set and back-propagate through the violations found within it. The remaining records are treated as satisfied. If each record's margin changes

Algorithm 1 Tether-Margin (screening at every step)

Require: source of candidate inputs; n, $m_{\min}$, κ, λ

for $k = 1, \ldots, K$ **do**
  for each optimisation step on D_k **do**
    accumulate $\nabla \mathcal{L}_k(\theta)$ on the current batch
    score records of tasks $t < k$ without a graph; collect the violated set
    recompute violated penalties with a graph; accumulate $\lambda \nabla \sum_{t<k} C_t(\theta)$; update θ
  end for
  $\theta_k \leftarrow \theta$; check all records of earlier tasks
  evaluate θ_k on the pool $\mathcal{P}_k$; keep the admissible inputs
  draw n records in equal shares from the teacher's classes; set w_v; store them; discard D_k
end for

by at most ν per update, then $\varsigma \kappa m_v \geq (s-1)\nu$ ensures that a skipped record cannot cross its threshold before the next full screening. Without such a bound, periodic screening is an approximation. At each full screening, we count records found violated after being skipped, termed *late-found violations*.

Average cost per step. Let $N = \sum_{t<k} |V_t|$ be fixed during task k, c_s the average cost of scoring a record without a graph, and $c_f + c_b$ its average forward-plus-backward cost with a graph. Let $\bar{a}_{\text{exact}}$ be the average fraction back-propagated under exact screening. For periodic screening, let $\bar{\ell}$ be the average fraction actually scored at intermediate steps (including both previously violated and low-slack records), and $\bar{a}_{\text{found}}$ the average fraction actually found violated and back-propagated over all steps. Over complete screening cycles, the cost model is

$$\begin{aligned} \text{dense evaluation:} \quad & N(c_f + c_b), \\ \text{exact screening:} \quad & N c_s + \bar{a}_{\text{exact}} N(c_f + c_b), \\ \text{periodic screening:} \quad & N c_s \frac{1 + (s-1)\bar{\ell}}{s} + \bar{a}_{\text{found}} N(c_f + c_b). \end{aligned} \tag{25}$$

For $s = 1$, there are no intermediate steps and the periodic expression reduces to exact screening. The backward term counts discovered violations; violations missed between screenings do not incur that cost. For comparison, KL using n_{sub} of the n records per previous task incurs about $(n_{\text{sub}}/n)N(c_f + c_b)$ per step. Empirical estimates use observed scoring and backward counts together with measured per-record costs, which depend on record lengths and batching.

Growth over a task sequence. While task k is learned, the bank holds $N = n(k-1)$ records, so every term of (25) grows with the number and length of the stored records, whereas the cost of a new-task step does not. KL runs a forward and a backward pass with a graph on a fixed fraction n_{sub}/n of the bank at every step. On Llama-2 this is 16 records of each earlier task, or 224 records per step while the fifteenth task is learned, against a new-task batch of 8, so the retention term soon dominates each step: the cumulative time in Table 7 rises from $1.55\times$ unprotected training after two tasks to $5.17\times$ after eight and $12.1\times$ after fifteen. Margin and Tether-TKL back-propagate only through records found violated, a small fraction of the bank: at $\lambda = 1$ in the controlled sweep, 1.2% of record–update pairs for Margin and 4.9% for Tether-TKL against 33% for KL (Appendix D), and in the conflict sweep 0.4–3.0 and 2.8–9.8 active records per update against 48 for KL (Appendix E). Their growing term is the screening pass, which scores records without a graph and therefore costs less per record than a forward and backward pass; on the long Llama-2 sequences a full screening runs only every eighth step, and the intermediate steps score only the violated and low-slack records. Their cumulative time therefore grows more slowly, to $3.13\times$ for Margin and $3.94\times$ for Tether-TKL after fifteen tasks; the larger active fractions of Tether-TKL are consistent with its higher cost.

B.4 Record admission and weights

Admission. At the end of task t, the teacher scores a candidate pool $\mathcal{P}_t$ of real or synthetic inputs (Appendix B.6). An input is admissible when $c_v \in \mathcal{L}_t$ and $m_v \geq m_{\min}$. We use $m_{\min} = 0.2$. Admission and selection use the teacher's predictions; the record weights below use aggregate label-derived statistics, and no per-example ground-truth label is retained. Admission keeps the

constants of the record-level guarantees finite. Margin normalises by κm_v, so its gradient coefficient $w_v/(\kappa m_v)$ grows without bound as $m_v \to 0$; the critical radius satisfies only $\varepsilon_v^* \geq m_v^2/2$, so the KL constant $\beta_{\mathcal{K}} = 2/m_{\min}^2$ diverges as $m_{\min} \to 0$. Excluded inputs remain part of F_t and enter the transfer term of Proposition 3. We draw n admissible candidates in approximately equal shares from the predicted classes.

Weights. Let n_c be the number of selected records assigned to class c. For a vector π of class proportions, define

$$w_v = \frac{\pi_{c_v}}{n_{c_v} Z_t}, \qquad Z_t = \sum_{c:\, n_c > 0} \pi_c, \tag{26}$$

assuming $Z_t > 0$. The weights sum to one. If every class with $\pi_c > 0$ is represented, then $Z_t = 1$ and its records have total weight π_c.

Class proportions. The statistical design in Appendix A.6 uses the teacher's class proportions among admissible inputs. The main benchmark uses $\tilde{\pi}_c$, estimated from admissible training inputs that the teacher classifies correctly (Section 3.1). These estimates use aggregate training-label statistics.

Allocation and coverage. Candidates are shuffled within each teacher class. Classes are sampled in turn until n records are selected. Classes with insufficient candidates yield their remaining allocation to other classes. Any unrepresented class with positive proportion is logged. The weights are then renormalised over the represented classes, giving $Z_t < 1$.

The quantity $1 - Z_t$ is the uncovered mass under the chosen class proportions; it need not equal excluded mass under P_t.

Zero weights. A record receives zero weight when its teacher class has zero estimated proportion. Such records do not contribute to the retention loss. End-of-task checks still include them in unweighted counts.

Decision rule. All reported runs use $\bar{\mathcal{L}}_t$ for admission, stored margins, training and record checks. The code includes a fallback that scores admission and margins over $\mathcal{L}_t$ when fewer than n candidates are admissible. This fallback was not invoked in the reported runs. It does not establish the guarantees for decisions over $\bar{\mathcal{L}}_t$.

B.5 Evaluation

Decision conventions. The input x includes the task prompt, so p_θ and $\hat{c}_\theta$ omit the task index. The output law q_θ includes non-terminating sequences. Under (1), these sequences belong to $\perp$.

For distinct single-token answers, first-token probabilities define an alternative categorical distribution. Residual probability is assigned to $\perp$. The decision-level analysis applies when probabilities, margins and records are defined consistently under this rule.

Record conditions use decisions over $\bar{\mathcal{L}}_t$. The main continual benchmark measures task drift by the argmax over $\bar{\mathcal{L}}_t$ and average accuracy by greedy generation; greedy generation agrees with the argmax over $\mathcal{L}_t$ on 98–99.8% of inputs for every method other than IncLoRA and LwF. The controlled experiments use $\bar{\mathcal{L}}_t$ for both drift and accuracy (Appendices D and E). The two drifts differ where a final decision is $\perp$: on the Llama-2 runs this happens on at most 0.7% of the audited inputs, and the teacher never decides $\perp$. After unprotected training on T5, the final model places its largest probability mass on non-answer strings for 6–10% of earlier-task inputs, so its drift over $\bar{\mathcal{L}}_t$ exceeds the drift over $\mathcal{L}_t$ by that amount; LwF shows the same effect on 29–36% of inputs. Appendix F.1 relates the record conditions to greedy decoding.

Let $F_t^{\bar{\mathcal{L}}}$ and $F_t^{\mathcal{L}}$ denote the two decision drifts. Use the same tie rule on the valid answers. Their disagreement indicators coincide whenever both augmented decisions lie in $\mathcal{L}_t$. Hence

$$|F_t^{\bar{\mathcal{L}}}(\theta) - F_t^{\mathcal{L}}(\theta)| \leq P_t\big(\hat{c}_{\theta_t}(X) = \perp \text{ or } \hat{c}_\theta(X) = \perp\big).$$

Either rule can be audited under Proposition 5. The rule must be fixed in advance, and the audit inputs must satisfy the proposition's independence conditions. To bound a decrease in greedy-generation accuracy via (4), audit drift under the same greedy rule. An absolute accuracy lower bound also requires the teacher's accuracy or a valid lower bound on it.

Audit inputs. Proposition 5 requires independent inputs from P_t and the teacher's decisions on them. In our evaluation protocol, the teacher's decisions are cached at the end of task t on that task's test inputs; later models are compared with these cached decisions. Applying the population confidence bound requires that these inputs and their evaluation results have informed neither training nor hyperparameter or checkpoint selection. In the reported runs, no checkpoint or early-stopping selection takes place: every task is trained for one epoch and the final weights are evaluated. One hyperparameter is an exception: the T5-large Tether-TKL coefficient $\lambda = 0.1$ was chosen in the Amazon→Yelp sweep, which reports on the same Amazon and Yelp test inputs, so for these two tasks the T5-large Tether-TKL audit is not independent of hyperparameter selection. The evaluation inputs and cached decisions are evaluation resources, separate from the training-record storage reported in the experiments. An external evaluation set, or fresh inputs scored by both models while the teacher remains available, can serve instead.

B.6 SYNTHETIC RECORDS

Task specification. When no task input is retained, Qwen3-8B-Base (Yang et al., 2025a) generates the candidate inputs; both models use the same candidate pool. Its prompts contain no benchmark example or hand-written task description. The generation specification combines the task schema with aggregate statistics of D_t. The schema provides instructions, answer options and field names. The statistics provide per-field word-count quantiles, inter-field word-overlap distributions and class proportions.

Generation and filtering. Using this specification and public topic seeds, the generator first writes a one-paragraph task description, sampled with temperature 0.7 and top-p 0.9 and resampled with other settings if it is degenerate, and a few example lines with temperature 0.9 and top-p 0.95; a filter rejects examples copied from one another. The generator then completes label-conditioned JSON lines with temperature 0.9 and top-p 0.95. Per-field length targets are sampled jointly from one random row of D_t and clipped to its 10th–90th percentiles.

A candidate is accepted only if each field is within a factor of 1.6, or two words, of its target and between 0.8 times the 2nd and 1.2 times the 98th percentile of the corresponding training lengths, if inter-field overlap lies within the training 2nd–98th percentiles widened by 0.05, and if it differs from accepted items; for WiC, both sentences must also contain the target word. Candidates are formatted as task inputs and deduplicated against task data using 8-gram overlap. Generation proceeds in rounds, each requesting the missing candidates of every class, until each class has $\max(32, \mathrm{round}(320\pi_c))$ accepted candidates, with π_c its proportion in D_t (160 per answer for COPA), or nine rounds have run.

Record construction. Each teacher then scores the pool; generator-assigned labels are discarded, and admission, the selection of the 48 records and their weights follow the real-input procedure in Appendix B.4. The synthetic record bank does not retain real benchmark inputs as records. However, its construction uses aggregate statistics from the training data and does not provide a formal privacy guarantee.

Table 4: Final average accuracy (%) per task order: mean ± standard deviation. Rows under *Reported* are published values: T5-large values from Wang et al. (2023), Yang et al. (2025b) and Wang et al. (2025); Llama-2-7B-chat values from Qiao & Mahdavi (2026), which re-ran IncLoRA, O-LoRA, InfLoRA, MagMax and SAPT-LoRA under the same protocol. MTL is multi-task training on all tasks of a sequence, the usual upper bound; it does not depend on the order, so its value is repeated across the orders of each group. The source papers report means over three runs without deviations, except N-LoRA, whose values with deviations come from its Table 8 (five runs).

Method	O1	O2	O3	O4	O5	O6	O1–3	O4–6
T5-large								
IncLoRA	70.0±5.9	67.0±7.3	74.9±1.4	64.7±1.7	57.3±4.1	63.0±0.7	70.6	61.7
O-LoRA	77.7±0.9	77.4±0.6	75.9±0.3	66.1±2.4	69.6±2.2	67.7±2.0	77.0	67.8
LwF	75.6±2.1	75.6±1.9	76.1±1.0	61.4±0.5	56.3±3.8	65.4±1.5	75.8	61.0
Replay, 2%	80.4±0.3	77.8±4.8	79.1±0.4	76.3±0.6	77.1±0.3	75.6±0.6	79.1	76.3
DER++, 2%	80.3±0.3	80.1±0.4	79.7±0.3	76.9±0.6	77.0±0.3	75.3±0.7	80.0	76.4
Tether-Margin	79.2±0.1	79.0±0.9	78.5±0.6	76.8±0.4	77.0±0.8	76.4±0.3	78.9	76.8
Tether-TKL	79.4±0.3	79.3±0.6	78.9±0.9	77.4±0.4	75.8±2.1	77.4±0.4	79.2	76.9
Tether-KL	80.2±0.3	79.8±0.2	79.4±0.6	77.6±0.8	78.2±0.4	78.0±0.1	79.8	77.9
Reported								
O-LoRA	75.4	75.7	76.3	72.3	64.8	71.6	75.8	69.6
N-LoRA†	79.1±0.4	78.3±0.4	77.9±1.9	73.9±0.5	70.1±0.9	73.1±1.4	78.5	72.3
GORP	79.7	79.9	79.7	76.1	76.2	75.6	79.8	76.0
Progr. Prompts (Razdaibiedina et al., 2023)	75.2	75.0	75.1	78.0	77.7	77.9	75.1	77.9
LFPT5	67.6	72.6	77.9	70.4	68.2	69.1	72.7	69.2
MTL, upper bound	80.0	80.0	80.0	76.5	76.5	76.5	80.0	76.5
Llama-2-7B-chat								
IncLoRA	79.4±2.2	80.4±0.5	79.9±1.2	71.7±2.4	71.0±1.1	77.8±1.4	79.9	73.5
Replay, 2%	80.8±1.1	81.5±0.3	81.4±0.2	81.6±0.4	80.4±0.8	82.2±0.4	81.2	81.4
DER++, 2%	82.3±0.3	82.2±0.4	82.3±0.2	83.0±0.4	82.8±0.3	83.0±0.5	82.3	82.9
Tether-Margin	81.9±0.4	82.0±0.1	81.5±0.2	82.2±0.8	82.1±0.8	82.7±0.3	81.8	82.3
Tether-TKL	82.2±0.3	82.2±0.3	81.8±0.4	82.4±0.6	82.1±1.1	82.9±0.3	82.1	82.5
Tether-KL	82.4±0.3	82.2±0.2	81.9±0.2	83.3±0.5	83.2±0.6	83.6±0.2	82.2	83.4
Reported								
IncLoRA	75.3	77.3	78.3	72.2	71.6	73.8	77.0	72.5
O-LoRA	76.1	76.3	79.2	74.0	72.0	74.6	77.2	73.5
InfLoRA (Liang & Li, 2024)	78.4	80.4	79.9	69.4	67.4	72.5	79.6	69.8
MagMax (Marczak et al., 2024)	80.1	80.6	80.3	72.3	73.5	74.5	80.3	73.4
SLAO	80.1	80.8	80.4	75.0	74.4	75.1	80.4	74.8
SAPT-LoRA (Zhao et al., 2024)	82.9	81.8	78.7	84.7	78.9	82.2	81.1	81.9
MTL, upper bound	80.9	80.9	80.9	78.1	78.1	78.1	80.9	78.1

†mean ± standard deviation over five runs as reported; the means differ from the three-run means in that paper's main table by at most 0.9 points.

C Complete benchmark comparisons

Table 5: Retention metrics (%) on earlier tasks: decision drift, harmful flips, repairs, wrong-to-wrong changes, the decrease of old-task accuracy relative to the teacher, and the fraction of final decisions equal to $\perp$; Acq. is acquisition, the teacher's accuracy on the audited inputs of each task, averaged over all tasks. Drift and $\perp$ use the decision rule (2) in every row. LwF's drift is dominated by $\perp$ decisions; over $\mathcal{L}_t$ it is 16.7% (Orders 1–3) and 32.1% (Orders 4–6).

Method	Drift	Harmful	Repairs	Wrong→wrong	ΔAcc	$\perp$	Acq.
T5-large, O1–3							
IncLoRA	30.1	19.3	5.1	5.6	14.3	9.5	79.7
O-LoRA	13.5	7.0	3.7	2.7	3.3	0.0	79.1
LwF	37.4	21.9	1.8	13.7	20.1	31.5	79.3
Replay, 2%	14.3	6.6	5.3	2.4	1.3	0.2	79.7
DER++, 2%	7.8	3.2	3.1	1.6	0.1	0.0	79.7
Tether-Margin	9.0	4.2	2.7	2.1	1.4	0.1	79.8
Tether-TKL	7.1	3.3	2.4	1.4	1.0	0.1	79.7
Tether-KL	4.7	2.0	1.7	1.0	0.3	0.0	79.7
Tether-Margin, synthetic	9.8	4.4	3.1	2.2	1.4	0.1	79.3
T5-large, O4–6							
IncLoRA	37.2	26.4	8.7	2.2	17.7	5.7	77.2
O-LoRA	21.4	12.5	6.9	2.0	5.6	0.6	73.4
LwF	51.4	38.4	4.8	8.1	33.6	36.4	77.8
Replay, 2%	18.0	9.9	7.4	0.8	2.5	0.0	78.4
DER++, 2%	12.9	7.3	5.1	0.5	2.2	0.0	78.2
Tether-Margin	10.4	5.5	4.3	0.6	1.2	0.1	77.7
Tether-TKL	10.4	6.0	3.8	0.5	2.1	0.1	78.6
Tether-KL	6.7	3.5	2.9	0.3	0.6	0.0	78.3
Tether-Margin, synthetic	12.6	6.8	4.8	0.6	2.0	0.9	77.8
Llama-2-7B-chat, O1–3							
IncLoRA	11.6	6.1	3.4	1.6	2.7	0.7	82.2
Replay, 2%	12.0	5.6	4.5	1.8	1.1	0.2	82.1
DER++, 2%	6.4	2.7	2.9	0.9	-0.2	0.0	82.2
Tether-Margin	6.7	3.0	2.6	1.1	0.4	0.1	82.3
Tether-TKL	3.5	1.5	1.5	0.5	-0.1	0.0	82.2
Tether-KL	3.0	1.3	1.3	0.4	0.0	0.0	82.2
Tether-Margin, synthetic	6.5	2.9	2.5	1.0	0.4	0.1	82.2
Llama-2-7B-chat, O4–6							
IncLoRA	19.9	14.4	4.3	1.0	10.1	0.2	83.2
Replay, 2%	12.7	6.8	5.3	0.6	1.6	0.1	82.8
DER++, 2%	7.6	3.7	3.5	0.4	0.2	0.0	83.0
Tether-Margin	7.8	4.0	3.4	0.3	0.6	0.1	82.8
Tether-TKL	4.9	2.3	2.4	0.2	-0.1	0.0	82.3
Tether-KL	4.1	1.9	2.0	0.2	-0.1	0.0	83.2
Tether-Margin, synthetic	9.6	5.4	3.8	0.3	1.7	0.1	82.9

Table 6: Decision drift (%) per task on the fifteen-task sequences for the three instances, averaged over the orders in which the task precedes the last task. The last column marks tasks where $\mathrm{KL} \leq \mathrm{TKL} \leq \mathrm{Margin}$. Tether-TKL uses $\lambda = 0.1$ on T5-large and $\lambda = 1$ on Llama-2-7B-chat.

	T5-large				Llama-2-7B-chat			
Task	Margin	TKL	KL	order	Margin	TKL	KL	order
ag	5.5	4.6	2.5	yes	3.1	2.5	1.1	yes
amazon	14.1	11.5	7.8	yes	12.6	6.7	6.0	yes
boolqa	11.8	12.1	7.8	no	8.0	5.4	4.2	yes
cb	9.7	9.3	6.9	yes	9.1	6.7	6.0	yes
copa	12.8	10.3	5.8	yes	6.6	4.4	3.6	yes
dbpedia	1.6	1.7	0.7	no	1.0	0.8	0.4	yes
imdb	2.5	2.7	2.0	no	2.5	1.7	1.2	yes
mnli	17.0	17.6	12.4	no	12.1	7.8	6.3	yes
multirc	14.5	15.4	9.8	no	8.6	6.9	8.3	no
qqp	12.2	10.5	5.9	yes	7.4	4.0	3.7	yes
rte	12.4	11.4	7.1	yes	9.9	4.6	3.2	yes
sst-2	3.2	3.8	1.9	no	2.5	1.3	1.0	yes
wic	17.3	26.1	18.8	no	18.7	11.0	8.6	yes
yahoo	6.2	4.7	2.8	yes	5.0	2.4	2.1	yes
yelp	14.9	15.0	9.7	no	12.3	8.1	6.5	yes
Ordering holds				7/15				14/15

wic precedes the last task in two orders and yahoo in one; all other tasks in three.

Table 7: Cumulative training time through the first k tasks of the fifteen-task sequences, relative to IncLoRA over the same tasks (mean over orders); Appendix B.3 explains the growth.

	T5-large						Llama-2-7B-chat					
Method	k=2	4	5	8	10	15	k=2	4	5	8	10	15
Replay, 2%	1.39	1.57	1.59	1.65	1.72	1.70	1.54	1.93	1.95	1.95	2.04	1.96
DER++, 2%	1.27	1.47	1.49	1.52	1.57	1.56	1.40	1.81	1.84	1.88	2.00	1.94
Tether-Margin	1.17	1.36	1.40	1.63	1.97	2.30	1.05	1.36	1.44	1.86	2.42	3.13
Tether-TKL	1.14	1.36	1.42	1.70	2.09	2.68	1.10	1.50	1.61	2.13	2.97	3.94
Tether-KL	1.36	2.35	2.80	4.18	5.96	9.03	1.55	2.94	3.29	5.17	8.30	12.13

Table 8: Label-free audit on earlier tasks after the last task: observed drift and its one-sided 95% Clopper–Pearson upper bound $\bar{F}_{0.05}$ (%), with $n_a = 2000$ held-out inputs per task (fewer for tasks with smaller test sets). Medians, 90th percentiles and maxima over all task rows of each method.

Method	Rows	Observed, median	$\bar{F}$, median	$\bar{F}$, 90th pct.	$\bar{F}$, max
T5-large					
IncLoRA	153	32.9	35.9	67.1	99.9
O-LoRA	153	12.8	15.7	52.9	100.0
LwF	153	43.8	46.1	99.9	100.0
Replay, 2%	153	15.2	18.2	34.2	97.4
DER++, 2%	153	10.7	12.8	26.3	78.4
Tether-Margin	153	10.7	12.8	21.8	44.6
Tether-TKL	153	8.9	10.9	22.8	37.4
Tether-KL	153	5.4	6.9	16.4	24.8
Llama-2-7B-chat					
IncLoRA	153	16.1	18.4	45.1	63.4
Replay, 2%	153	12.0	14.3	28.4	49.3
DER++, 2%	153	6.9	8.2	17.5	29.9
Tether-Margin	153	7.1	9.1	18.2	37.6
Tether-TKL	153	4.0	5.3	12.0	32.4
Tether-KL	153	3.2	5.0	10.6	30.0

Table 9: Reproduced baselines under our protocol against published values (%). T5-large reported values are from Wang et al. (2023); the Llama-2-7B-chat IncLoRA value is from Qiao & Mahdavi (2026). Our replay stores 2% of each task's training data and our LwF is applied to LoRA-tuned models, whereas the reported Replay and LwF of Wang et al. (2023) fine-tune all parameters, so those two rows are not like-for-like.

		O1–3		O4–6	
Model	Method	reproduced	reported	reproduced	reported
T5-large	O-LoRA	77.0	75.8	67.8	69.6
	IncLoRA	70.6	66.4	61.7	61.2
	LwF	75.8	52.3	61.0	46.9
	Replay	79.1	57.8	76.3	54.2
Llama-2-7B-chat	IncLoRA	79.9	77.0	73.5	72.5

D CONTROLLED RETENTION-STRENGTH SWEEP

D.1 PROTOCOL

Shared teacher. T5-large uses one persistent rank-8 LoRA adapter, with scaling $\alpha = 32$, on query and value projections. The adapter first learns Amazon for one epoch. Each continuation starts from this end-of-task teacher and trains on Yelp for 200 updates. Adam uses learning rate 10^{-3}, effective batch size 64 and micro-batch size 16, with training dropout 0.1. Every instance and coefficient share the teacher, the records and the Yelp minibatch order.

Records and loss scales. Each teacher supplies 48 Amazon training inputs. Admission requires a valid answer and teacher margin $m_v \geq 0.2$. Inputs are sampled within teacher classes; weights follow Appendix B.4. We use $\kappa = \eta = 0.5$ and optimise

$$\mathcal{L}_{\text{Yelp}} + \lambda s_G G_{\text{Amazon}}, \qquad s_C = s_{C^{\text{TKL}}} = 1, \quad s_{\mathcal{K}} = 10.$$

Thus the KL coefficient is 10λ. This fixed rescaling does not equate the effective penalty strengths. Margin and Tether-TKL score all 48 records at every update and back-propagate positive penalties. KL samples 16 records uniformly per update and rescales their weights by $48/16$.

Coefficient grid. The shared unprotected continuation supplies $\lambda = 0$. The positive grid is $\{0.001, 0.003, 0.01, 0.03, 0.1, 0.3, 1, 3, 10\}$. Every positive coefficient is run for each of the three instances.

Evaluation. Drift and accuracy use complete-answer probabilities over the five rating answers and $\perp$; $\perp$ is scored as incorrect. Each task has 2,000 fixed reporting test inputs. All held-out results use the final 200-update checkpoint. We report means and observed minima and maxima, not confidence intervals. Records are evaluated at update 200, matching the held-out evaluations; the relative divergence is the unweighted mean $\bar{q} = 48^{-1} \sum_v d_v/\varepsilon_v^*$. Training CE, evaluated on 256 fixed Yelp training inputs, is a diagnostic of optimisation, not an estimate of the global optima in Proposition 4.

Cost. We count cumulative record–update pairs used in back-propagation; dividing by 200×48 gives the mean backward fraction.

D.2 RECORD REGIONS AND HELD-OUT DRIFT

Figure 4 separates record decisions, distribution changes and held-out behaviour. Without protection, the endpoint changes 13, 14 and 19 of the 48 recorded decisions. At $\lambda = 1$, Margin retains all 48 and satisfies its margin condition, while 37, 30 and 34 records lie outside the strict critical radius. KL at coefficient 10 places every record strictly inside the radius; two of its records nevertheless violate the Margin condition. At the representative settings of Section 3.3 (Margin $\lambda = 1$, Tether-TKL $\lambda = 0.1$, KL coefficient 10), the mean relative divergence is 4.94, 0.17 and 0.049. Tether-TKL retains every decision, although 2, 3 and 4 records exceed its threshold and one record exceeds the critical radius; at $\lambda = 0.3$ the threshold violations still average 2 of 48. A mean relative divergence below η therefore does not imply zero Tether-TKL loss on every record. Table 11 gives endpoint counts for every coefficient.

Endpoint checks. Rescoring each saved endpoint in evaluation mode reproduces the logged flip and condition-violation counts of all 84 runs exactly, and the mean relative divergences to within 4×10^{-7}. Across all 4,032 exported record states, no record satisfying $d_v < \varepsilon_v^*$ or $m_\theta(v; c_v) > 0$ changes its decision.

Held-out drift. For $\lambda \geq 0.3$, record flips are already rare, yet mean Amazon drift remains around 18–20% for Margin and 12–14% for Tether-TKL, while KL decreases from 14.2% to 8.6%: record retention leaves different residual task drifts under the three objectives. These are observed outcomes at finite coefficients; the generally non-nested Margin and Tether-TKL regions do not imply a theoretical ordering of task drift.

Admission and coverage. Let $A = \{\hat{c}_{\theta_t}(X) \in \mathcal{L}_t,\ m_{\theta_t}(X; \hat{c}_{\theta_t}(X)) \geq 0.2\}$. The empirical decomposition is $F_t = (1-\alpha)F_t^A + \alpha F_t^{A^c}$, where $\alpha = P_t(A^c)$ is estimated on the reporting inputs. Its mean is 27.2%. At $\lambda = 1$, excluded inputs contribute 12.0 percentage points of Margin's 17.8% drift and 8.2 points of KL's 10.2%. The remaining drift occurs on admissible but unrecorded inputs. Both parts therefore matter when transferring a finite-record guarantee to the task distribution.

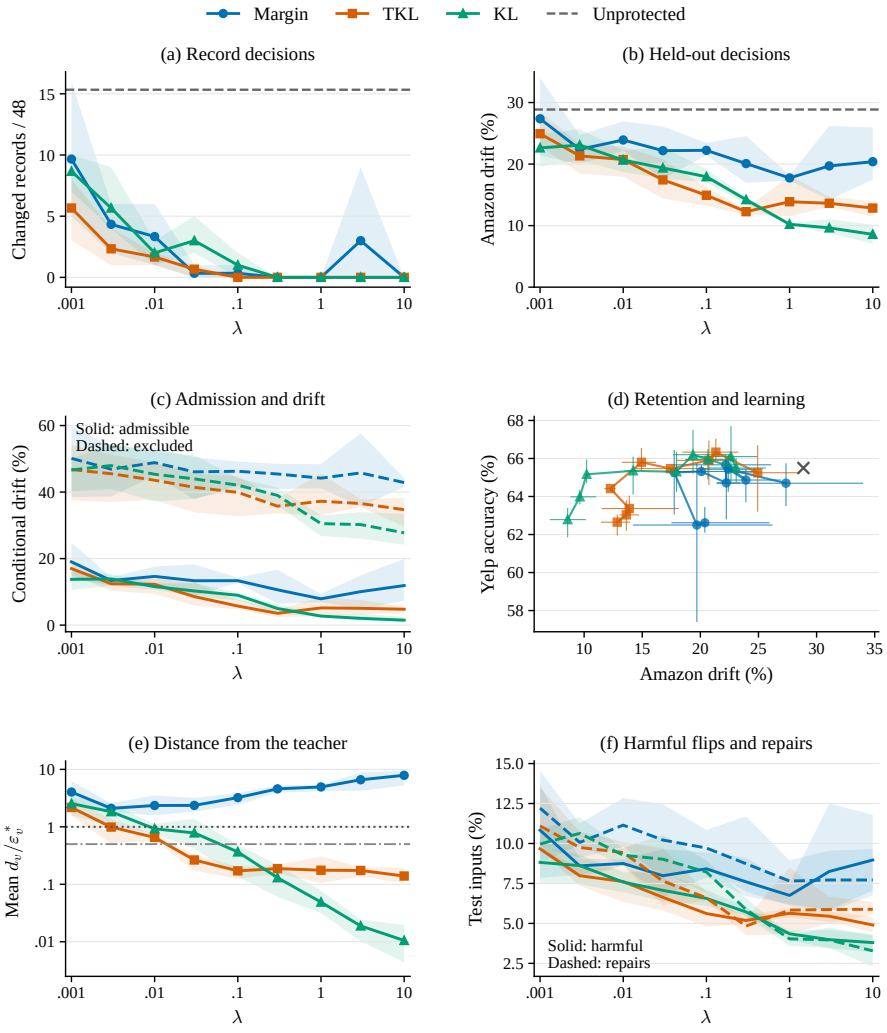


Figure 4: Complete Amazon→Yelp sweep. Lines show means and bands or error bars show ranges. Record and test statistics both use update 200. In (c), solid and dashed lines are conditional drift on admissible and excluded inputs. In (d), connected points follow increasing λ; they are measured trade-offs, without interpolation or selection of an optimal frontier. In (f), solid and dashed lines denote harmful flips and repairs. KL uses coefficient 10λ.

D.3 Learning cost and repairs

Figure 5 and Table 10 report both sides of the trade-off. Yelp accuracy stays within one point of unprotected training (65.5%) up to $\lambda = 1$ for Margin, $\lambda = 0.1$ for Tether-TKL and coefficient 10 for KL; the unprotected range alone spans 1.2 points. At the next grid value it falls by 1.1–3.0 points (Margin $\lambda = 3$: -3.0; Tether-TKL $\lambda = 0.3$: -1.1; KL coefficient 30: -1.5). At these largest safe settings, Yelp accuracies are 65.3%, 65.8% and 65.2%, and Amazon drift is 17.8%, 14.9% and 10.2% (Section 3.3). Tether-TKL's cost sets in a decade earlier than Margin's ($\lambda = 0.3$ versus 3).

The two losses use different per-record normalisations. For an active record, Tether-TKL scales the divergence by $1/(\eta\varepsilon_v^*)$, whereas Margin scales the margin shortfall by $1/(\kappa m_v)$. Since $\varepsilon_v^* \geq m_v^2/2$, the ratio of these normalisation coefficients satisfies

$$\frac{1/(\eta\varepsilon_v^*)}{1/(\kappa m_v)} \leq \frac{2\kappa}{\eta m_v}.$$

With $\kappa = \eta = 0.5$, this upper bound is 10 at the admission threshold $m_v = 0.2$. This comparison concerns the scalar normalisation coefficients rather than the resulting gradient magnitudes. Empirically, Tether-TKL begins to reduce new-task accuracy at a smaller coefficient in the T5 sweep, which motivates our use of $\lambda = 0.1$.

Positive backward transfer. The teachers average 57.7% Amazon accuracy. Unprotected continuation repairs 13.1% of predictions and loses 11.2% of correct predictions, increasing accuracy by 1.9 points to 59.6%. At the largest safe settings, Margin, Tether-TKL and KL reduce repairs to 7.6%, 6.6% and 4.0%, respectively, with harmful-flip rates of 6.8%, 5.6% and 4.4%. KL consequently ends at 57.4%, close to the teacher's accuracy, despite the lowest drift of these points: retaining earlier decisions suppresses beneficial revisions as well as harmful ones (Section 2.1).

Teacher-margin groups. At $\lambda = 1$, inputs with teacher margin below 0.6 account for 98% of the drift gap between Margin and KL, split evenly between margins below 0.2 and 0.2–0.6; confident inputs contribute 0.1 points, the same pattern as on the benchmark (Figure 2).

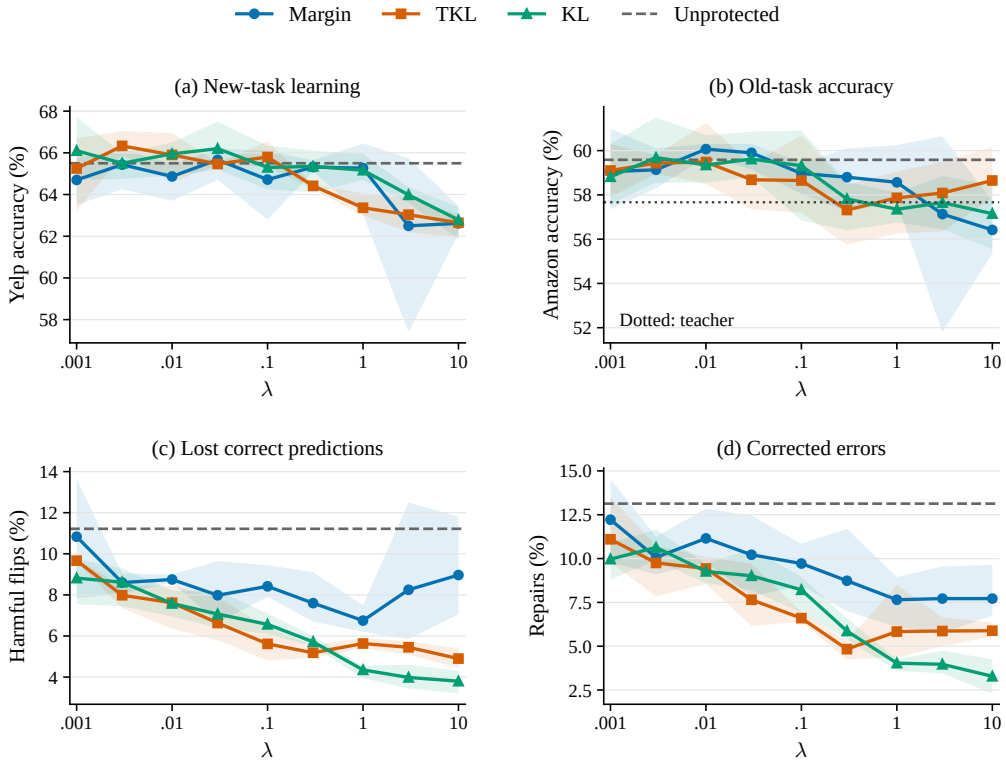


Figure 5: Accuracy, harmful flips and repairs across the sweep. Bands show the range; dashed horizontal lines show unprotected training, and the dotted line in (b) shows teacher accuracy. All four panels use the same augmented decision rule.

D.4 Training dynamics

At displayed $\lambda = 1$ (KL coefficient 10), Margin back-propagates through a mean of 113 record–update pairs over the 200 updates, Tether-TKL through 472 and KL through 3,200, that is, backward

fractions of 1.2%, 4.9% and 33% of the 48×200 possible pairs (Figure 6). Of Margin's total, 72% occur within the first 64 updates; its later backward fraction is 0.48%, so the penalty remains intermittent after the initial adjustment. Tether-TKL continues to react to threshold violations; KL evaluates 16 records per update throughout.

High-coefficient variation. At Margin $\lambda = 3$, one of the three Yelp accuracies is 57.4% with nine changed records, against 64.3% and 65.8% with all records retained; at $\lambda = 10$ all records are retained, but Amazon drift is 26.0%, 17.5% and 17.7%. These runs are included in every mean and range.

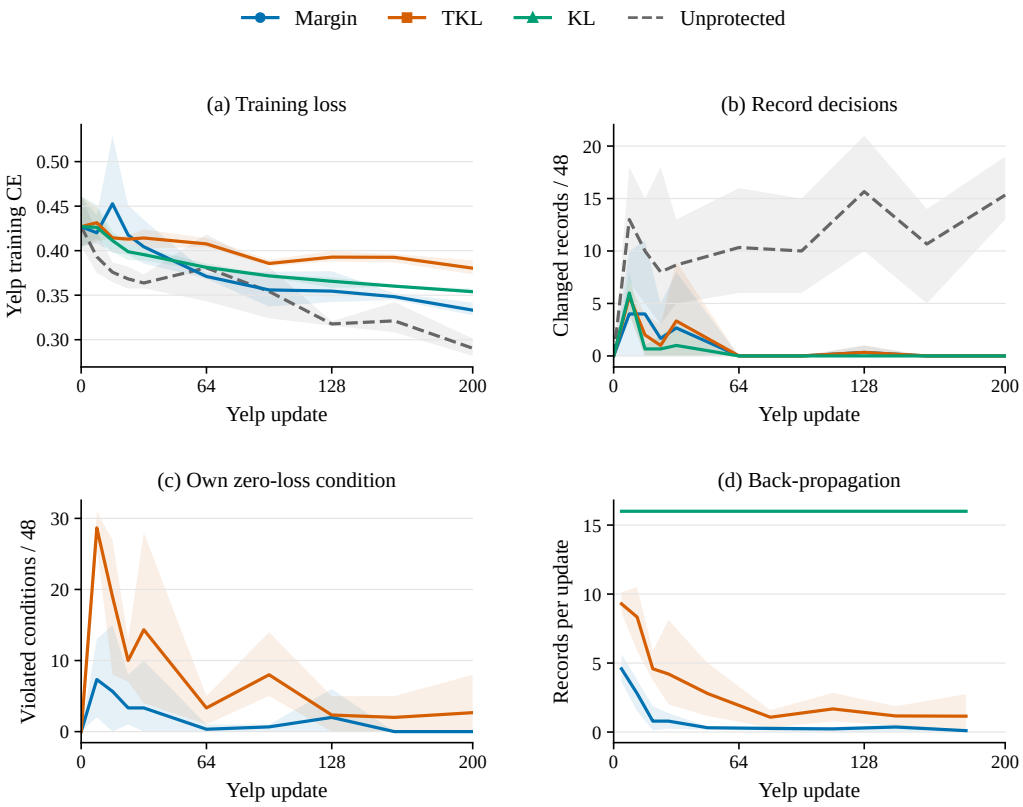


Figure 6: Training dynamics at displayed $\lambda = 1$ (KL coefficient 10). Bands show the range. (a) CE on 256 fixed Yelp training inputs. (b) Changed record decisions. (c) Violations of each hinge instance's own zero-loss condition. (d) Actual backward record evaluations per update, averaged within successive diagnostic intervals.

D.5 Complete numerical results

Tables 10 and 11 include every instance–coefficient combination. Held-out rates and record diagnostics use the aggregation conventions above.

Table 10: Complete sweep: held-out decisions at update 200 (%). Entries are means with $[\min, \max]$ subscripts.

Instance	λ	Drift	Harmful	Repairs	Amazon acc.	Yelp acc.
Unprotected	0	$28.9_{[24.7,31.2]}$	$11.2_{[9.6,12.5]}$	$13.1_{[11.6,14.1]}$	$59.6_{[58.8,61.1]}$	$65.5_{[64.9,66.1]}$
Margin	0.001	$27.4_{[21.4,34.0]}$	$10.8_{[7.9,13.7]}$	$12.2_{[9.8,14.6]}$	$59.1_{[57.4,61.0]}$	$64.7_{[63.5,65.8]}$
	0.003	$22.4_{[20.0,24.8]}$	$8.6_{[8.1,9.1]}$	$10.1_{[9.1,11.3]}$	$59.1_{[58.3,60.1]}$	$65.4_{[64.3,66.2]}$
	0.01	$23.9_{[22.2,27.0]}$	$8.8_{[8.6,9.0]}$	$11.2_{[9.8,12.9]}$	$60.1_{[59.7,60.3]}$	$64.9_{[63.7,65.5]}$
	0.03	$22.2_{[17.7,26.1]}$	$8.0_{[6.7,9.7]}$	$10.2_{[7.9,12.5]}$	$59.9_{[59.2,60.3]}$	$65.7_{[64.7,66.2]}$
	0.1	$22.2_{[21.4,23.5]}$	$8.4_{[7.9,9.5]}$	$9.7_{[8.5,10.9]}$	$59.0_{[58.1,59.5]}$	$64.7_{[62.8,66.1]}$
	0.3	$20.1_{[17.0,24.6]}$	$7.6_{[6.7,9.1]}$	$8.7_{[7.0,11.7]}$	$58.8_{[57.3,60.1]}$	$65.3_{[65.1,65.7]}$
	1	$17.8_{[17.3,18.2]}$	$6.8_{[6.2,7.5]}$	$7.7_{[6.1,9.0]}$	$58.6_{[57.7,60.3]}$	$65.3_{[63.1,66.5]}$
	3	$19.7_{[14.2,26.2]}$	$8.3_{[5.9,12.5]}$	$7.7_{[5.8,9.6]}$	$57.1_{[51.8,60.7]}$	$62.5_{[57.4,65.8]}$
	10	$20.4_{[17.5,26.0]}$	$9.0_{[7.1,11.8]}$	$7.7_{[6.8,9.7]}$	$56.4_{[55.4,57.8]}$	$62.6_{[62.1,63.5]}$
Tether-TKL	0.001	$24.9_{[22.9,28.6]}$	$9.7_{[9.6,9.8]}$	$11.1_{[9.8,13.5]}$	$59.1_{[57.9,60.3]}$	$65.3_{[63.2,66.7]}$
	0.003	$21.3_{[18.5,23.3]}$	$8.0_{[7.4,8.5]}$	$9.8_{[7.9,10.8]}$	$59.4_{[59.0,59.8]}$	$66.3_{[65.9,67.1]}$
	0.01	$20.7_{[18.0,22.9]}$	$7.6_{[6.4,8.5]}$	$9.4_{[8.6,10.1]}$	$59.5_{[58.5,61.3]}$	$65.9_{[64.6,67.0]}$
	0.03	$17.4_{[14.4,20.8]}$	$6.6_{[5.8,7.9]}$	$7.7_{[6.2,9.7]}$	$58.7_{[57.4,59.4]}$	$65.5_{[65.4,65.5]}$
	0.1	$14.9_{[13.3,16.2]}$	$5.6_{[4.8,6.4]}$	$6.6_{[6.4,7.0]}$	$58.7_{[57.2,60.7]}$	$65.8_{[65.4,66.6]}$
	0.3	$12.3_{[11.8,12.7]}$	$5.2_{[5.0,5.6]}$	$4.8_{[4.3,5.3]}$	$57.3_{[55.8,58.5]}$	$64.4_{[64.3,64.5]}$
	1	$13.9_{[11.5,18.2]}$	$5.6_{[5.4,5.9]}$	$5.8_{[4.4,8.5]}$	$57.9_{[56.3,59.1]}$	$63.4_{[63.0,64.1]}$
	3	$13.6_{[13.1,14.7]}$	$5.5_{[5.1,5.8]}$	$5.9_{[5.0,6.7]}$	$58.1_{[56.4,59.5]}$	$63.0_{[62.2,63.8]}$
	10	$12.9_{[11.5,14.0]}$	$4.9_{[4.5,5.4]}$	$5.9_{[5.6,6.4]}$	$58.7_{[57.9,60.2]}$	$62.7_{[62.0,63.1]}$
KL	0.001	$22.6_{[19.6,24.9]}$	$8.8_{[7.6,9.8]}$	$10.0_{[8.8,10.9]}$	$58.8_{[57.6,60.3]}$	$66.1_{[64.7,67.7]}$
	0.003	$23.1_{[20.7,25.7]}$	$8.6_{[7.5,9.3]}$	$10.6_{[9.9,11.7]}$	$59.7_{[58.6,61.5]}$	$65.5_{[64.8,66.0]}$
	0.01	$20.7_{[18.7,22.3]}$	$7.6_{[7.0,8.2]}$	$9.3_{[8.6,9.7]}$	$59.4_{[58.6,60.7]}$	$66.0_{[64.9,66.5]}$
	0.03	$19.4_{[17.7,22.7]}$	$7.1_{[6.4,8.2]}$	$9.0_{[8.2,10.6]}$	$59.6_{[58.8,60.9]}$	$66.2_{[65.2,67.5]}$
	0.1	$18.0_{[17.1,19.2]}$	$6.6_{[6.1,7.1]}$	$8.2_{[6.9,9.0]}$	$59.3_{[56.9,60.9]}$	$65.3_{[64.3,66.4]}$
	0.3	$14.2_{[13.5,14.7]}$	$5.7_{[5.5,6.0]}$	$5.9_{[5.4,6.6]}$	$57.8_{[56.4,58.6]}$	$65.4_{[64.1,66.1]}$
	1	$10.2_{[10.1,10.4]}$	$4.4_{[4.0,4.6]}$	$4.0_{[3.6,4.3]}$	$57.4_{[56.8,58.1]}$	$65.2_{[64.7,66.0]}$
	3	$9.6_{[8.8,11.1]}$	$4.0_{[3.5,4.6]}$	$4.0_{[3.5,4.8]}$	$57.7_{[56.5,58.9]}$	$64.0_{[63.6,64.4]}$
	10	$8.6_{[7.1,10.2]}$	$3.8_{[3.2,4.3]}$	$3.3_{[2.3,4.3]}$	$57.2_{[55.6,58.5]}$	$62.8_{[61.9,63.4]}$

Table 11: Complete sweep: record diagnostics at update 200. V_M, V_T, V_* count violations of the Margin condition, Tether-TKL threshold and strict critical radius; $\bar{q}$ is the unweighted mean of d_v/ε_v^*. Entries are means with $[\min, \max]$ subscripts.

Instance	λ	R_t (%)	Flips / 48	V_M	V_T	V_*	$\bar{q}$
Unprotected	0	$34.21_{[30.01,37.07]}$	$15.3_{[13.0,19.0]}$	$20.3_{[18.0,24.0]}$	$37.3_{[34.0,42.0]}$	$33.3_{[32.0,34.0]}$	$4.200_{[3.371,5.348]}$
Margin	0.001	$17.44_{[12.58,23.71]}$	$9.7_{[5.0,16.0]}$	$17.0_{[15.0,21.0]}$	$34.7_{[31.0,39.0]}$	$27.7_{[25.0,31.0]}$	$4.033_{[2.432,6.210]}$
	0.003	$5.44_{[2.54,8.58]}$	$4.3_{[2.0,6.0]}$	$9.7_{[7.0,12.0]}$	$30.3_{[27.0,36.0]}$	$22.7_{[18.0,27.0]}$	$2.097_{[1.821,2.585]}$
	0.01	$4.10_{[0.61,10.50]}$	$3.3_{[1.0,6.0]}$	$7.0_{[6.0,9.0]}$	$32.0_{[28.0,37.0]}$	$23.7_{[20.0,28.0]}$	$2.351_{[1.574,3.527]}$
	0.03	$0.03_{[0.00,0.10]}$	$0.3_{[0.0,1.0]}$	$2.7_{[2.0,4.0]}$	$33.0_{[28.0,38.0]}$	$26.3_{[21.0,30.0]}$	$2.361_{[1.842,3.183]}$
	0.1	$0.57_{[0.00,1.70]}$	$0.3_{[0.0,1.0]}$	$1.7_{[1.0,2.0]}$	$34.7_{[31.0,39.0]}$	$30.7_{[26.0,36.0]}$	$3.224_{[2.400,3.868]}$
	0.3	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.3_{[0.0,1.0]}$	$37.3_{[31.0,43.0]}$	$33.3_{[30.0,37.0]}$	$4.580_{[3.928,5.284]}$
	1	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$37.0_{[34.0,40.0]}$	$33.7_{[30.0,37.0]}$	$4.942_{[4.162,5.398]}$
	3	$0.98_{[0.00,2.95]}$	$3.0_{[0.0,9.0]}$	$4.7_{[0.0,14.0]}$	$38.7_{[36.0,41.0]}$	$33.7_{[31.0,35.0]}$	$6.596_{[4.234,8.841]}$
	10	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.7_{[0.0,2.0]}$	$37.0_{[35.0,39.0]}$	$33.7_{[31.0,36.0]}$	$7.878_{[5.230,9.522]}$
Tether-TKL	0.001	$9.98_{[0.61,25.50]}$	$5.7_{[3.0,8.0]}$	$13.3_{[10.0,16.0]}$	$29.3_{[21.0,35.0]}$	$20.7_{[15.0,25.0]}$	$2.168_{[1.570,3.104]}$
	0.003	$2.95_{[0.41,4.86]}$	$2.3_{[1.0,4.0]}$	$9.3_{[7.0,14.0]}$	$19.3_{[12.0,25.0]}$	$10.7_{[5.0,14.0]}$	$0.991_{[0.509,1.365]}$
	0.01	$3.56_{[0.10,5.73]}$	$1.7_{[1.0,2.0]}$	$10.7_{[7.0,17.0]}$	$19.0_{[16.0,24.0]}$	$9.7_{[8.0,11.0]}$	$0.654_{[0.590,0.755]}$
	0.03	$1.78_{[0.00,3.59]}$	$0.7_{[0.0,1.0]}$	$3.7_{[3.0,4.0]}$	$6.7_{[2.0,12.0]}$	$1.7_{[0.0,3.0]}$	$0.265_{[0.174,0.333]}$
	0.1	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$1.0_{[1.0,1.0]}$	$3.0_{[2.0,4.0]}$	$0.3_{[0.0,1.0]}$	$0.171_{[0.125,0.201]}$
	0.3	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$3.3_{[1.0,5.0]}$	$2.0_{[0.0,3.0]}$	$0.0_{[0.0,0.0]}$	$0.188_{[0.154,0.216]}$
	1	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.7_{[0.0,2.0]}$	$2.7_{[0.0,8.0]}$	$0.3_{[0.0,1.0]}$	$0.176_{[0.106,0.306]}$
	3	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$2.7_{[2.0,4.0]}$	$2.7_{[1.0,4.0]}$	$0.3_{[0.0,1.0]}$	$0.174_{[0.136,0.199]}$
	10	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.139_{[0.106,0.192]}$
KL	0.001	$17.44_{[12.41,20.21]}$	$8.7_{[7.0,10.0]}$	$13.7_{[13.0,14.0]}$	$31.7_{[27.0,35.0]}$	$24.0_{[22.0,27.0]}$	$2.533_{[1.595,3.378]}$
	0.003	$10.27_{[8.34,12.20]}$	$5.7_{[4.0,9.0]}$	$12.0_{[11.0,14.0]}$	$26.3_{[24.0,28.0]}$	$18.3_{[16.0,21.0]}$	$1.833_{[1.025,2.558]}$
	0.01	$1.81_{[0.26,4.86]}$	$2.0_{[1.0,3.0]}$	$9.0_{[7.0,11.0]}$	$20.0_{[16.0,24.0]}$	$11.0_{[10.0,13.0]}$	$0.924_{[0.758,1.128]}$
	0.03	$4.88_{[3.85,6.65]}$	$3.0_{[2.0,5.0]}$	$9.0_{[3.0,13.0]}$	$15.0_{[12.0,17.0]}$	$7.7_{[6.0,10.0]}$	$0.783_{[0.416,1.378]}$
	0.1	$2.68_{[0.00,6.31]}$	$1.0_{[0.0,2.0]}$	$5.7_{[1.0,10.0]}$	$12.0_{[2.0,17.0]}$	$3.3_{[0.0,5.0]}$	$0.366_{[0.136,0.488]}$
	0.3	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$1.7_{[1.0,3.0]}$	$3.3_{[0.0,8.0]}$	$0.3_{[0.0,1.0]}$	$0.130_{[0.060,0.217]}$
	1	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.7_{[0.0,2.0]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.049_{[0.031,0.078]}$
	3	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.019_{[0.010,0.024]}$
	10	$0.00_{[0.00,0.00]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.0_{[0.0,0.0]}$	$0.011_{[0.004,0.019]}$

E RECORD-COUNT SWEEP UNDER CONFLICT

E.1 PROTOCOL

Teacher and new task. T5-large uses one persistent rank-8 LoRA adapter ($\alpha = 32$) on the query and value projections of encoder and decoder self-attention and on decoder cross-attention. The teacher trains on MNLI for 200 updates (batch 16, learning rate 10^{-3}) on 3,000 training examples, 1,000 per class, and reaches 85.5–87.3% MNLI test accuracy. The new task is RTE under the MNLI prompt and label set, with *not_entailment* mapped to *contradiction*: 2,000 training and 277 test examples (146 entailment, 131 contradiction). The teachers label 34–43% of the RTE non-entailment pairs *neutral* and 24–48% *contradiction*, so learning RTE requires turning part of the neutral region into contradiction, in conflict with the MNLI rule. Each continuation trains for 150 updates with batch 64 (micro-batch 8), learning rate 10^{-3}, a 10-update linear warm-up and inputs of at most 512 tokens.

Records and instances. The bank contains MNLI training inputs with teacher margin at least $m_{\min} = 0.2$, chosen round-robin over the teacher's classes up to $n \in \{48, 144, 432, 648, 972, 1296\}$, with weights proportional to the class prior among admissible inputs. Every method samples the same 48 records per update, uniformly from the bank, with weights rescaled by $n/48$; per-update cost therefore does not depend on n. Margin uses $\lambda = 1$, $\kappa = 0.5$; Tether-TKL $\lambda = 0.1$, $\eta = 0.5$ with the exact critical radius; KL uses coefficient 1, whereas the representative KL setting of Appendix D uses coefficient 10. KL back-propagates through every sampled record; Margin and Tether-TKL only through records with positive penalty. Unprotected training uses the same updates without a retention term.

Evaluation. Every 10 updates the model is evaluated on 1,000 MNLI test inputs (353/330/317 per class) and the 277 RTE test inputs under the decision rule of (2). Reported values average the ten evaluations at updates 60–150 (the late window); ranges in Table 12 are minima and maxima.

E.2 RESULTS

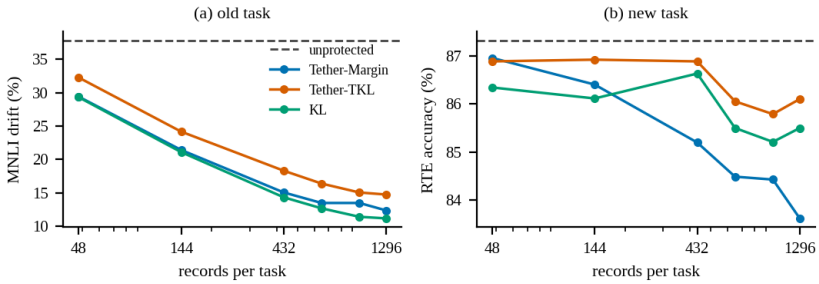


Figure 7: Learning RTE from an MNLI teacher under the MNLI prompt, against the number of records per task (log scale; updates 60–150): (a) MNLI drift, (b) RTE accuracy. Dashed lines are unprotected training. Every step samples 48 records, so per-step cost does not depend on the bank size.

Mechanism. The conflict is visible at the class level (Table 12). Unprotected training keeps none of MNLI's neutral decisions (recall 0%) while reaching 87% recall on RTE contradiction. With 1,296 records, Margin keeps 69% of neutral and reaches 79% on contradiction, KL 70% and 82%: keeping the neutral rule and learning the contradiction rule are directly opposed, which is the source of the new-task cost in Figure 7b.

Record diagnostics. At update 150, records that violate the Margin condition are at most 2.9% under Margin and 3.2% under KL for every n, and 7–12% under Tether-TKL for $n \geq 432$: $\lambda = 0.1$ is weak for large banks, which is why Tether-TKL drifts most here. Changed record decisions stay at or below 2.0% for all instances. Under Margin, 17–25% of records satisfy the margin condition

Table 12: Record-count sweep from an MNLI teacher to RTE under the MNLI prompt (%; updates 60–150; mean and [min, max]). *Neutral kept* is recall of MNLI neutral; *RTE contradiction* is recall of RTE contradiction. Violated-record shares are given in the text.

Instance	n	MNLI acc.	MNLI drift	Neutral kept	RTE acc.	RTE contrad.
unprotected	–	59.8 [59.5,60.4]	37.8 [36.0,39.3]	0.0 [0.0,0.0]	87.3 [86.8,87.7]	87.0 [86.3,88.2]
KL	48	67.5 [64.9,69.2]	29.3 [28.7,30.1]	21.9 [17.8,25.0]	86.3 [85.4,86.9]	87.3 [84.8,89.3]
KL	144	74.5 [72.4,77.3]	21.1 [20.2,21.6]	43.3 [36.4,50.9]	86.1 [86.0,86.2]	85.2 [84.2,86.3]
KL	432	80.5 [79.7,81.6]	14.3 [13.8,15.0]	59.9 [56.9,64.4]	86.6 [85.6,87.2]	84.7 [84.0,85.7]
KL	648	81.7 [79.9,83.3]	12.7 [11.9,14.0]	64.6 [61.9,69.7]	85.5 [85.3,85.8]	83.2 [82.8,83.8]
KL	972	82.9 [82.3,84.2]	11.4 [10.9,11.8]	68.9 [64.5,71.5]	85.2 [83.9,86.0]	81.6 [79.2,83.5]
KL	1296	83.2 [82.4,84.4]	11.2 [10.1,11.9]	70.4 [67.0,72.6]	85.5 [84.4,87.1]	82.3 [80.2,84.6]
Margin	48	68.0 [66.1,71.1]	29.4 [28.0,31.1]	20.1 [15.0,28.5]	87.0 [86.7,87.2]	86.4 [85.9,87.1]
Margin	144	74.6 [73.2,75.5]	21.4 [20.8,22.6]	43.2 [39.6,45.1]	86.4 [85.4,87.1]	85.2 [84.2,86.2]
Margin	432	80.3 [78.8,81.1]	15.1 [13.9,15.9]	61.4 [55.2,64.7]	85.2 [84.6,85.8]	81.4 [81.1,81.6]
Margin	648	81.8 [80.8,83.2]	13.5 [12.8,14.8]	65.8 [62.8,70.9]	84.5 [84.1,84.7]	80.5 [79.1,82.3]
Margin	972	81.5 [80.0,82.4]	13.5 [12.9,13.9]	65.3 [59.4,68.6]	84.4 [83.8,85.3]	81.1 [80.6,82.1]
Margin	1296	82.8 [81.9,84.2]	12.4 [11.3,13.0]	69.3 [65.8,71.1]	83.6 [83.5,83.9]	79.2 [78.5,79.9]
TKL	48	65.0 [63.1,66.5]	32.3 [31.0,34.0]	11.9 [9.0,15.7]	86.9 [86.1,87.9]	86.2 [85.0,87.7]
TKL	144	72.3 [69.7,73.8]	24.2 [23.8,24.8]	35.7 [29.9,39.8]	86.9 [86.6,87.3]	85.8 [85.2,86.7]
TKL	432	77.1 [76.7,77.8]	18.3 [16.7,20.0]	53.2 [49.5,55.6]	86.9 [86.4,87.3]	85.9 [84.9,86.9]
TKL	648	78.8 [77.8,79.5]	16.4 [15.1,17.2]	57.8 [56.2,60.6]	86.1 [85.3,86.7]	83.6 [81.8,85.3]
TKL	972	80.1 [79.2,81.4]	15.1 [14.4,15.4]	62.1 [58.8,65.7]	85.8 [85.4,86.5]	84.0 [83.0,85.3]
TKL	1296	80.5 [79.2,81.9]	14.7 [13.9,15.7]	65.4 [62.6,67.7]	86.1 [85.2,86.8]	83.4 [80.5,85.1]

while lying outside their critical radius, as in Appendix D. Active records per update average 0.4–3.0 for Margin, 2.8–9.8 for Tether-TKL and 48 for KL; per-update time is 2.0–2.5 s against 1.5 s unprotected.

Caveats. The RTE test set has 277 examples, so one example is 0.36 points.

F GENERATION

We first relate finite-answer decisions to greedy decoding. We then apply the retention conditions to next-token distributions. Positive margins at every position, including EOS, preserve the full greedy output. With partial protection, the guarantee covers the prefix before the first unprotected position. Appendix F.4 reports the continual generation experiment.

F.1 FINITE-ANSWER DECISIONS AND GREEDY DECODING

Proposition 6. *If $m_\theta(v; c_v) > 0$, the first greedy token on v begins $s_t(c)$ for some $c \in \mathcal{L}_t$. If also $p_\theta(c_v \mid v) > \frac{1}{2}$, greedy decoding outputs exactly $s_t(c_v)$ and stops.*

Proof. Write $c = c_v$, $s = s_t(c)$ and $p = p_\theta(\cdot \mid v)$. For a prefix z, let $q_\theta(z \mid v)$ denote its prefix probability.

First token. Positive margin gives $p(c) > p(\perp)$. If token u begins no canonical answer, every output beginning with u belongs to $\perp$. Writing s_1 for the first token of s, we obtain

$$q_\theta(u \mid v) \le p(\perp) < p(c) \le q_\theta(s_1 \mid v).$$

Such a token cannot be selected by greedy decoding.

Complete output. Let $z = s\,\mathrm{EOS}$ and assume $p(c) > 1/2$. Every prefix satisfies $q_\theta(z_{\le j} \mid v) \ge p(c)$. Therefore

$$q_\theta(z_j \mid v, z_{<j}) = \frac{q_\theta(z_{\le j} \mid v)}{q_\theta(z_{<j} \mid v)} > \tfrac{1}{2}.$$

Every token of z, including EOS, is the unique greedy choice. □

Each sufficient condition in Table 1 implies the positive-margin premise.

F.2 Decision chains

Token distributions and records. Let $\mathcal{V}$ be the vocabulary and $\pi_\theta(\cdot \mid v, y_{<j})$ the next-token distribution. Greedy decoding selects its maximiser at each position. For input v, let $y^* = (y_1^*, \ldots, y_T^*)$ be the teacher's finite greedy output, including EOS. Its margin at position j is

$$m_j = \pi_{\theta_t}(y_j^* \mid v, y_{<j}^*) - \max_{u \in \mathcal{V} \setminus \{y_j^*\}} \pi_{\theta_t}(u \mid v, y_{<j}^*).$$

Admit positions $J = \{j : m_j \geq m_{\min}\}$. The record is $(v, y^*, (m_j)_{j \in J}, w_v)$, with $w_v \geq 0$. It contains the teacher's output and no reference label.

Margin loss. Let $m_{\theta,j}$ be the current margin of token y_j^* at the teacher prefix $(v, y_{<j}^*)$. Define

$$\psi_v^{\mathrm{seq}}(\theta) = \sum_{j \in J} \frac{[\kappa m_j - m_{\theta,j}]_+}{\kappa m_j}. \tag{27}$$

Each summand penalises a margin below κm_j. Weights sum to one per task. The task loss is $C_t^{\mathrm{seq}}(\theta) = \sum_{v \in V_t} w_v \psi_v^{\mathrm{seq}}(\theta)$. This instance uses $G_t = C_t^{\mathrm{seq}}$ in (8). One teacher-forced pass over (v, y^*) computes the next-token distributions at all recorded prefixes.

Proposition 7 (decision chains)**.** *(i) If $m_{\theta,j} > 0$ for every $j \leq T$, greedy decoding under θ outputs y^*. Every metric determined by the output and a fixed reference is therefore unchanged. This includes exact match, ROUGE-L, chrF and edit similarity.*

(ii) If the greedy output differs from y^*, let j_0 be the first differing position. Then $m_{\theta,j_0} \leq 0$. If $j_0 \in J$, then $\psi_v^{\mathrm{seq}}(\theta) \geq 1$.

(iii) If $m_{\theta,j} > 0$ for every $j \in J$, the outputs agree before the first position outside J. They agree throughout when $J = \{1, \ldots, T\}$.

Proof. (i) Full output. Proceed by induction on the position. Assume the first $j-1$ tokens equal $y_{<j}^*$. Since $m_{\theta,j} > 0$, the unique maximiser at this prefix is y_j^*. The generated output therefore equals y^* through position T, including EOS. Identical outputs give identical values of any fixed evaluation metric.

(ii) First disagreement. The prefixes agree before j_0. At that prefix, greedy decoding selects some $u \neq y_{j_0}^*$ with probability at least that of $y_{j_0}^*$. Hence $m_{\theta,j_0} \leq 0$. If $j_0 \in J$, its summand in (27) is

$$\frac{[\kappa m_{j_0} - m_{\theta,j_0}]_+}{\kappa m_{j_0}} \geq 1.$$

All other summands are nonnegative, proving $\psi_v^{\mathrm{seq}} \geq 1$.

(iii) Partial output. Apply the induction in (i) until the first position outside J. □

The vocabulary is the exhaustive decision set at each position. Propositions 1 and 2 therefore apply to each admitted token decision. When all positions are admitted, part (ii) gives $\mathbf{1}[\text{output changed}] \leq \psi_v^{\mathrm{seq}}$.

For task-level assessment, compare the two greedy outputs on independent held-out inputs. Use fixed decoding rules and the independence conditions of Proposition 5. The resulting audit bounds the probability of any output change.

F.3 KL instances along the same chain

At each admitted position, evaluate both models at the *teacher prefix* $(v, y_{<j}^*)$. Let $d_{v,j}$ be the forward KL between their next-token distributions. Proposition 2 gives $\varepsilon_{v,j}^*$ from the teacher's chosen-token and strongest-competitor probabilities.

KL penalises $d_{v,j}$. Tether-TKL uses $[d_{v,j}/(\eta \varepsilon_{v,j}^*) - 1]_+$. If $d_{v,j} < \varepsilon_{v,j}^*$ at every position, Proposition 7(i) guarantees the full greedy output. For a proper subset of positions, part (iii) gives the corresponding prefix guarantee.

Exact KL requires the teacher's full vocabulary distribution at each protected position. It can be cached or recomputed from the teacher. Margin requires the chosen tokens and their stored margins. Retaining only the largest probability entries changes the divergence. The full-distribution guarantee requires an additional bound on that approximation.

F.4 Continual generation experiment

Tasks and training. Llama-3-8B-Instruct (Dubey et al., 2024) learns six short-prompt generation tasks in the order GSM8K (Cobbe et al., 2021), DialogSum (Chen et al., 2021), CoNaLa (Yin et al., 2018), CommonsenseQA (Talmor et al., 2019), WMT16 German–English translation (Bojar et al., 2016) and question generation on SQuAD (Rajpurkar et al., 2016; Du et al., 2017) (SQuAD-QG). Each task trains a fresh LoRA adapter of rank 8 and scaling $\alpha = 16$ on the seven attention and MLP projections, and earlier updates are retained. Every task uses 2,000 training examples, drawn at random for each seed, for two epochs, with AdamW (Loshchilov & Hutter, 2019), learning rate 10^{-4}, a cosine schedule with 3% warm-up, gradient clipping at 1.0 and batch size 16 (micro-batches of 4). Prompts use the chat template with a system prompt, the loss covers answer tokens only, and training sequences are cut at 1,280 tokens, with prompts of at most 1,024 tokens and at least 256 answer tokens kept. ARC-Challenge (Clark et al., 2018) and MMLU (Hendrycks et al., 2021) are not tasks of the sequence.

Methods. We compare IncLoRA, DER++ with a 2% buffer of each task (40 examples) and the three Tether instances, with $\lambda = 1$ for Margin and KL, $\lambda = 0.1$ for Tether-TKL and $\kappa = \eta = 0.5$. After each task, 48 of its training inputs become records with weight $1/48$. Each record stores the greedy output of the teacher, the model at the end of the task, cut at the larger of 32 tokens and the task's generation limit so that it covers the full evaluation output, and ends with EOS; positions where the teacher's margin is at least 0.2 are admitted. Margin uses (27); Tether-TKL and KL use the per-position divergences of Appendix F.3. KL samples 8 records per earlier task at each step, and Margin and Tether-TKL screen the records every 8 steps. The records outnumber the DER++ buffer, 48 against 40 per task, but hold no references. Both methods also protect ARC-Challenge and MMLU from the start. Before the first task, each Tether instance records 48 training inputs of each benchmark with the base model as teacher, 47 of which are kept on MMLU, and protects them for the whole sequence; DER++ adds 48 labelled training questions of each benchmark to its buffer and trains on them with its label and distillation losses. IncLoRA uses neither.

Evaluation. Every model decodes greedily, with at most 256, 96, 64, 4, 128 and 48 new tokens on GSM8K, DialogSum, CoNaLa, CommonsenseQA, WMT16 and SQuAD-QG, on a fixed random sample of 500 test inputs per task and on all 1,172 ARC-Challenge questions and 1,000 MMLU questions. Task scores are final-number accuracy on GSM8K, ROUGE-L (Lin, 2004) with the best of three references on DialogSum, edit similarity on CoNaLa, option-letter accuracy on CommonsenseQA, sentence-level chrF (Popović, 2015) on WMT16 and ROUGE-L on SQuAD-QG. A task's acquisition is its score right after it is learned, when the model is its teacher; final scores are measured after SQuAD-QG. Drift compares the final model with each earlier task's teacher, and with the base model on ARC-Challenge and MMLU. On the answer tasks, GSM8K and CommonsenseQA, and on ARC-Challenge and MMLU, the decision is the final number or option letter of the greedy output; decision drift, harmful flips and repairs are as in Section 3.1, and the answer-task aggregates average GSM8K and CommonsenseQA. The free-form tasks, DialogSum, CoNaLa and WMT16, have no finite answer set. For them we report the fraction of greedy outputs identical to the teacher's, their token similarity to the teacher's output, and the length of the longest common prefix as a fraction of the teacher's output. Token drift is the fraction of positions of the reference greedy output, including the end token, at which the final model's most probable next token under teacher forcing differs from the reference token. By Proposition 7(i) and (ii), an output is unchanged exactly when none of its positions drifts, up to ties.

Scores. Table 13 reports the results. The three instances end 3.3–4.2 points above IncLoRA in final average and within 0.6 points of DER++. Average acquisition lies between 62.6 and 63.1 for all five methods, so retention costs no new-task score on average. IncLoRA's loss is concentrated on GSM8K, which falls from 68.0 to 50.8; no other task loses more than 3.5 points under any method.

Answers. Final scores hide most of the drift. On GSM8K, DER++, Margin and Tether-TKL each end 1.8 points below their teachers but change 31.2, 32.0 and 28.8% of the answers; KL changes 14.1% and ends 0.3 points below. Over GSM8K and CommonsenseQA, KL changes 7.9% of earlier answers, against 17.9% for DER++ and 31.3% for IncLoRA, and loses 2.2% of correct answers, against 6.1% and 14.9%. Its repairs, also 2.2%, leave its answer accuracy unchanged, whereas IncLoRA's harmful flips exceed its 5.7% repairs by 9.2 points. Margin's answer drift, 18.4%, is comparable to DER++'s, as on the O-LoRA benchmark. Tether-TKL's is 16.7%, and Tether-TKL changes less than DER++ on every other measure of drift and similarity in Table 13.

Free-form outputs. On the free-form tasks, KL leaves 79.2% of outputs identical to the teacher's, Tether-TKL 53.0%, Margin 49.9%, DER++ 37.7% and IncLoRA 27.7%. Similarity and common prefix follow the same order, and each instance keeps more outputs than DER++ on each of the three tasks. Scores again hide the difference: on DialogSum, every method except IncLoRA ends within 0.7 points of its teacher, yet DER++ changes 90% of the summaries and KL 33%. An output survives only if every one of its token decisions does, so small per-position differences compound over its length: on earlier tasks, KL changes 0.6% of token decisions and DER++ 4.1%.

ARC-Challenge and MMLU. With the base model as their teacher and records or buffer items from the start, ARC-Challenge and MMLU act as two further protected tasks for every method except IncLoRA. IncLoRA loses 3.2 points on ARC-Challenge, whereas every other method stays within 0.8 points of the base model on both. Decision drift separates the methods, with KL lowest and Tether-TKL next: 4.9 and 8.8% on ARC-Challenge against 10.2% for Margin and DER++, and 9.6 and 14.5% on MMLU against 17.4 and 17.9%. On MMLU, every method's accuracy is within 0.4 points of the base model's 64.3, yet IncLoRA changes 21.4% of the answers, its 7.3% harmful flips offset by 7.1% repairs.

Ordering and cost. The ordering KL $<$ Tether-TKL $<$ Margin in drift holds on every aggregate measure in Table 13 and on every task except WMT16, where Margin keeps 55.4% of outputs unchanged against 53.6% for Tether-TKL. Unlike on Llama-2 classification, where Tether-TKL trains in about a third of KL's time (Table 3), the two cost about the same here, 149 and 151 minutes, or 3.2$\times$ IncLoRA, and all three instances train more slowly than DER++ (2.1$\times$). Subtracting the retention term leaves 46–50 minutes for every method, IncLoRA's 47 included, so the differences arise in that term: 50 minutes for DER++, 74 for Margin and 103–104 for Tether-TKL and KL. Each record covers the full evaluation output, up to 256 tokens on GSM8K and 128 on WMT16, so it has many admitted positions; this raises the cost of every retention term, and Margin and Tether-TKL skip a record only when all of its admitted positions are satisfied. Training times are in minutes for the whole sequence.

Table 13: Continual generation with Llama-3-8B-Instruct (%; 500 test inputs per task). Task scores give acquisition/final; SQuAD-QG, learned last, has one value. Drift and similarity compare the final model with each earlier task's teacher, or with the base model for ARC-Challenge and MMLU. Answer tasks: GSM8K and CommonsenseQA; free-form tasks: DialogSum, CoNaLa and WMT16.

			Tether, 48 records/task		
	IncLoRA	DER++, 2%	Margin	TKL	KL
Task score, acquisition/final					
GSM8K	68.0/50.8	69.8/68.0	68.2/66.4	70.2/68.4	69.0/68.7
DialogSum	46.7/43.2	46.7/46.2	47.4/46.7	47.0/46.6	46.3/46.3
CoNaLa	69.4/69.0	69.6/69.3	69.7/69.4	69.4/69.5	69.3/69.2
CommonsenseQA	81.0/79.8	80.8/80.6	79.4/79.0	80.6/82.2	79.9/80.3
WMT16 De–En	62.9/62.5	62.4/62.5	62.9/63.0	63.2/62.7	63.7/63.5
SQuAD-QG	47.6	48.0	47.8	48.5	48.4
Average	62.6/58.8	62.9/62.4	62.6/62.1	63.1/63.0	62.8/62.7
Answer tasks					
Decision drift	31.3	17.9	18.4	16.7	7.9
GSM8K	52.8	31.2	32.0	28.8	14.1
CommonsenseQA	9.8	4.7	4.8	4.6	1.6
Harmful flips	14.9	6.1	6.3	4.9	2.2
Repairs	5.7	5.1	5.2	4.8	2.2
Free-form tasks					
Outputs unchanged	27.7	37.7	49.9	53.0	79.2
DialogSum	1.2	10.0	24.0	28.4	67.1
CoNaLa	59.6	67.5	70.2	77.0	90.1
WMT16	22.4	35.7	55.4	53.6	80.4
Similarity to teacher	76.4	84.2	87.7	89.7	96.0
Common prefix	45.5	57.9	67.4	70.6	88.5
Token drift, earlier tasks	8.3	4.1	2.7	2.3	0.6
ARC-Challenge, base model 80.3					
Final accuracy	77.1	80.1	80.2	79.5	80.7
Decision drift	13.8	10.2	10.2	8.8	4.9
Token drift	21.0	18.8	14.7	9.4	6.5
Harmful flips	7.1	4.2	4.3	4.1	1.7
Repairs	3.9	4.0	4.2	3.3	2.1
MMLU, base model 64.3					
Final accuracy	64.1	63.9	64.0	64.3	64.6
Decision drift	21.4	17.9	17.4	14.5	9.6
Token drift	23.4	20.5	16.7	10.7	8.0
Harmful flips	7.3	6.3	5.8	4.8	3.0
Repairs	7.1	5.9	5.5	4.8	3.3
Training time, min	47	99	124	149	151
relative to IncLoRA	1.0×	2.1×	2.6×	3.2×	3.2×
retention term, min	0	50	74	103	104