GROUP-BALANCED SUPERVISED FINE-TUNING FOR KNOWLEDGE-INTENSIVE DOMAINS

Anonymous authors
Paper under double-blind review

ABSTRACT

Supervised fine-tuning (SFT) weights every supervised token equally, so on knowledge-dense corpora the loss budget follows token count rather than importance: on a legal QA benchmark, citation strings absorb 81% of the supervised loss weight budget while the answers being taught absorb 5%. We present Group-Balanced SFT (GB-SFT), which equalizes the loss budget across meaningful groups of tokens, as a two-parameter objective family that contains standard SFT, class-level balancing, and span-level balancing as corners, whose geometry we characterize exactly. Across four benchmarks in law and medicine on two model families, we show that: (i) GB-SFT injects new knowledge better than uniform SFT, pointwise token weighting, memory editing, and matched-budget data mixing; (ii) it matches the injection of masking out everything but the knowledge payload while retaining the citation recall that masking destroys (0.98 vs. 0.38); and (iii) role-aware grouping carries the gain: label-free partitions recover at most a third of it, while an LLM tagger recovers nearly all, but only when the groups are lexically distinctive. We release two new contamination-controlled knowledge-injection benchmarks with validated ground truth.

1 INTRODUCTION

Large language models achieve strong general-purpose performance, but deployment in knowledge-intensive domains such as law or medicine requires a model to learn specialized facts (Lewis et al., 2020; Singhal et al., 2023). Supervised fine-tuning on domain data is the most common adaptation strategy (Ouyang et al., 2022; Zhou et al., 2023), yet it is known to trade off knowledge acquisition against hallucination (Gekhman et al., 2024). To navigate this tradeoff successfully, it is necessary to use an objective that places adequate learning emphasis on the parts of the fine-tuning data that most directly deliver the knowledge.

As a motivating example, we consider the task of fine-tuning a model on legal QA (Zheng et al., 2025), and break down a two-example corpus of legal questions in Fig. 1. We call the tokens that state the actual answer the *payload* and the surrounding boilerplate (citations, templated prose) the *scaffold*. By token count, the payload forms a very small portion of the examples compared to scaffold (a numeral-dense citation number, and some extra short prose). This shows that not all tokens in the supervised target carry equal knowledge payload.

A natural solution is to use a fine-tuning objective with some kind of non-uniform token weighting scheme. Some prior works (Lin et al., 2024; Wu et al., 2026; Liu et al., 2026; Wang et al., 2026) learn pointwise token importance scores. Our key insight hinges on the fact that tokens in knowledge-dense corpora can be semantically grouped, most readily by the nature of fact construction itself (e.g. as part of the answer, citation, or prose in legal QA). We propose a unified group-balanced supervised fine-tuning (GB-SFT) framework, in which we theoretically and empirically study different grouping strategies and ways to balance the loss among groups. GB-SFT outperforms both pointwise weighting and generic group heuristics (Kim et al., 2025) on knowledge-dense corpora from multiple domains, achieving a favorable balance between preserving existing knowledge and injecting new knowledge. We make the following contributions:

- New group-balanced fine-tuning objectives that give more weight to the knowledge payload.
- Improved group construction methods and theoretical characterization of balancing granularity.

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.

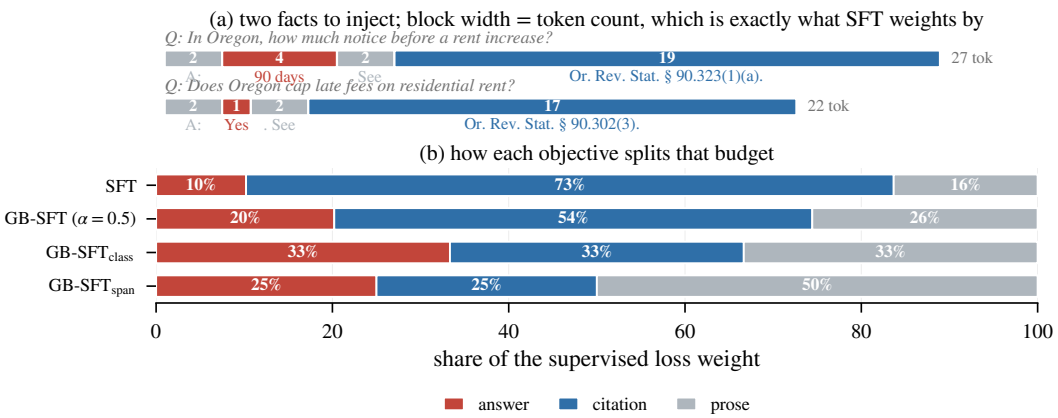


Figure 1: Example of how our grouping budgeting strategies upweight the knowledge payload in a 2-fact corpus. SFT weights the knowledge payload by its share of tokens. GB-SFT$_{\text{class}}$ gives each class (color) an equal budget, and GB-SFT$_{\text{span}}$ gives each span an equal budget ($\propto$ number of red blocks). Our unified formulation also gives a tunable α knob to gradually increase the class balance.

- New benchmarks for measuring knowledge injection across multiple domains.
- Empirical insight into the tradeoffs between preserving and injecting knowledge.

2 Related Work

Token weighting. Dynamic Fine-Tuning (DFT) weights each token's cross-entropy by the policy's own detached probability; Anchored supervised fine-tuning (ASFT) stabilizes DFT with a uniform token-level Kullback–Leibler (KL) anchor to the frozen base model (Zhu et al., 2026). Related online-weighting rules include probability thresholding and temperature-scaled variants (Lin et al., 2026; Zhang et al., 2026; Diao et al., 2026). Rho-1 selects tokens by excess loss against a curated-data reference (Lin et al., 2024), PriFT weights tokens by prior support under the pretrained checkpoint (Wang et al., 2026), and EKSFT masks high-entropy tokens from imitation while regularizing them toward the reference (Liu et al., 2026). These methods share a pointwise premise (a token's weight should be a function of its own difficulty or plausibility) and can miss group-level structure.

Group-based weighting and class imbalance. Closest to our approach, SFT-GO partitions tokens into importance groups and optimizes a mixture of worst-group and average loss: a token-level Group-DRO objective (Kim et al., 2025; Sagawa et al., 2020). Rather than adaptively emphasizing the hardest group, a heuristic we find points the wrong way on fact-editing tasks, we balance the loss budget each role receives. Our budget correction descends from inverse-frequency and class-balanced losses in classification (Cui et al., 2019), and contrasts with difficulty-based schemes such as focal loss (Lin et al., 2017), which upweight hard tokens (the opposite of what we need).

Span-level supervision units. Treating spans rather than tokens as the unit of masking or supervision has precedent in pretraining: salient span masking targets entity spans for knowledge acquisition (Guu et al., 2020), and SpanBERT and PMI-masking mask correlated spans (Joshi et al., 2020; Levine et al., 2021). On the aggregation-unit question, recent work shows that the normalization unit of an RL objective silently changes what is optimized (Liu et al., 2025); our identity for pooled- versus span-normalized CE is the SFT instance of this observation.

Knowledge injection, forgetting, and post-hoc robustness. Fine-tuning on new facts can promote hallucination (Gekhman et al., 2024), and knowledge becomes extractable only under sufficient paraphrase and format diversity (Allen-Zhu & Li, 2024); our evaluation accounts for these considerations. Memory-editing methods modify facts directly in weights (Meng et al., 2022; 2023), though they do not produce task training. We compare against continual-learning levers such as rehearsal/replay (Chaudhry et al., 2019; Kotha & Liang, 2026) and and post-hoc weight interpolation (Wortsman et al., 2022), which we find preserve existing more than acquire new knowledge.

Legal NLP benchmarks. We build on Housing Statute QA (Zheng et al., 2025) and evaluate statutory value injection on real statute excerpts, akin to LegalBench (Guha et al., 2023), CaseHOLD (Zheng et al., 2021), and SARA (Holzenberger et al., 2020). We target parametric knowledge, unlike retrieving legal knowledge into context for agents (Harvey AI & Baseten, 2026).

3 METHOD

A standard SFT loss weighs each supervised token equally, computing the average negative log likelihood (or cross-entropy loss) of the gold token over all tokens:

$$\mathcal{L}_{SFT}(\boldsymbol{\theta}) = \frac{1}{N}\sum_{t} -\log \pi_{\boldsymbol{\theta}}(y_t^* \mid x, y_{<t}^*) = \frac{1}{N}\sum_{t} \mathrm{CE}_t(\boldsymbol{\theta}), \tag{1}$$

where $\boldsymbol{\theta}$ denotes the model parameters, $\pi_{\boldsymbol{\theta}}$ the model's conditional next-token distribution, x the input prompt, y_t^* the gold token at position t, $y_{<t}^*$ the gold tokens preceding it, N the number of supervised tokens, and $\mathrm{CE}_t(\boldsymbol{\theta})$ the resulting per-token cross-entropy loss.

This fails to account for the fact that some groups of tokens may carry more knowledge payload than others. Thus, we will show how to partition tokens into groups and balance the loss accordingly.

3.1 TOKEN GROUPINGS

Grouping is specified by two maps over the supervised tokens of a datum. A *class grouper* $\mathcal{C}$ assigns each token one of K classes, and a *span grouper* $\mathcal{S}$ partitions the tokens into P spans. We require that $\mathcal{S}$ *refine* $\mathcal{C}$: every span lies entirely within a single class. Both groupers admit supervised and unsupervised realizations. Writing π_{ref} for the frozen pretrained reference model, we consider the following class- and span-level groupers:

Grouper	Realization	How it is obtained
$\mathcal{C}$	Category	fact structure, e.g. {answer, citation, prose}
	Citation regex	a grammar over citation surface form
	LLM tagger	a frontier LLM asked to label token roles
	Surprisal	K quantile buckets of π_{ref}; label-free
$\mathcal{S}$	Same class	maximal contiguous runs of constant class, i.e. induced by $\mathcal{C}$
	Detector	cross-fit logistic regression on (surprisal, entropy) in a ± 5 window

In our labeled runs $\mathcal{S}$ is induced by $\mathcal{C}$ through contiguity, which makes refinement automatic. That chaining is a property of the instantiation and not of the framework: the Detector grouper produces spans with no class labeling at all, and the Surprisal grouper produces classes with no span structure. In Fig. 1, $\mathcal{C}$ assigns each token to {answer, citation, prose}, and $\mathcal{S}$ is the maximal runs of constant class (colored blocks).

3.2 GROUP-BALANCED LOSS BUDGETING

Now our N supervised tokens each belong to a class $c \in \mathcal{C}$ and span $s \in \mathcal{S}$, where class c has n_c tokens and s has n_s tokens. We write $\mathcal{C}$ and $\mathcal{S}$ both for the groupers and for the partitions they induce, and identify a class c (resp. a span s) with its set of token positions, so that $t \in c$ and $t \in s$ are well defined. Our proposed Group-Balanced Supervised Fine-Tuning, GB-SFT, equalizes the loss budget across the coarse class-level groups, or the refined span-level groups.

Group-Balanced by Class. To get the loss function for our $\text{GB-SFT}_{\text{class}}$ strategy, we compute the mean per-token loss for each class and then take the mean per-class loss across all classes:

$$\mathcal{L}_{\text{class-balanced}}(\boldsymbol{\theta}) = \frac{1}{K}\sum_{c\in\mathcal{C}} \frac{1}{n_c}\sum_{t\in c} \mathrm{CE}_t(\boldsymbol{\theta}). \tag{2}$$

Group-Balanced by Span. To get the loss function for our GB-SFT$_{\text{span}}$ strategy, we compute the mean per-token loss for each span and then take the mean per-span loss across all spans:

$$\mathcal{L}_{\text{span-balanced}}(\boldsymbol{\theta}) = \frac{1}{P} \sum_{s \in \mathcal{S}} \frac{1}{n_s} \sum_{t \in s} \mathrm{CE}_t(\boldsymbol{\theta}). \tag{3}$$

In Fig. 1 we visualize a toy housing law fact corpus. Under the tokenization of the Qwen3.5-9B-Base tokenizer (Qwen Team, 2026) we show the breakdown of tokens across the categories of *answer* (the main knowledge payload), *citation*, and *prose* (the latter two being scaffolds for the knowledge), and the loss breakdown under ordinary SFT versus our GB-SFT$_{\text{class}}$ and GB-SFT$_{\text{span}}$. This clearly shows the increased emphasis that group balancing places on the knowledge payload.

A unified formulation. Both balanced loss functions, along with ordinary SFT, can be unified in a single formulation in which we can modulate β, the pooling unit within a class, and α the budget split across classes. Define the following terms for a token t lying in span s and class c:

$$u_s = n_s^{\beta}, \qquad m_c = \sum_{s \subset c} u_s, \qquad M = \sum_c m_c^{\alpha}, \qquad \omega_t = \frac{m_c^{\alpha}}{M} \cdot \frac{u_s}{m_c} \cdot \frac{1}{n_s} = \frac{1}{M}\, m_c^{\alpha-1}\, n_s^{\beta-1}, \tag{4}$$

The term $\frac{m_c^{\alpha}}{M}$ controls class budget, $\frac{u_s}{m_c}$ controls span budget within a class, and $\frac{1}{n_s}$ is the (uniform) budget given to tokens within a span. These combine to yield token weights ω_t, where $\sum_t \omega_t = 1$ at every (α, β), for our general group-balanced GB-SFT loss formulation:

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_t \omega_t \mathrm{CE}_t(\boldsymbol{\theta}), \tag{5}$$

where α interpolates between budgeting by class mass and budgeting by class count: at $\alpha = 1$ each class receives budget in proportion to its mass m_c (at $\beta = 1$, exactly its token count), while at $\alpha = 0$ the mass cancels and every class receives $1/K$ regardless of size. Intermediate $\alpha < 1$ shrinks the budget of large classes without removing the dependence entirely. β does the same at the span level: at $\beta = 0$ each span counts once, however long. This mechanism is analogous to the temperature-sampling exponent used in multilingual pretraining to stop high-resource languages from dominating training emphasis (Conneau & Lample, 2019). Our familiar objectives are special cases of Eq. 5: $(\alpha, \beta) = (1, 1)$ yields SFT, $(0, 1)$ yields GB-SFT$_{\text{class}}$, and $(1, 0)$ yields GB-SFT$_{\text{span}}$.

While we apply this group-based allocation to the supervised loss, we present a formulation for using group allocation to additionally partition a KL anchor (Zhu et al., 2026) in App. C.

3.3 Analysis

How the loss moves in (α, β). Let $\mathcal{S}_c$ denote the spans of class c. Group the weights of Eq. 4 into a class budget, a within-class span distribution, and a span-mean loss:

$$p_c = \frac{m_c^{\alpha}}{M}, \qquad q_c(s) = \frac{u_s}{m_c}, \qquad \overline{\mathrm{CE}}_s = \frac{1}{n_s} \sum_{t \in s} \mathrm{CE}_t(\boldsymbol{\theta}), \qquad A_c = \sum_{s \in \mathcal{S}_c} q_c(s)\, \overline{\mathrm{CE}}_s, \tag{6}$$

so that Eq. 5 reads $\mathcal{L} = \sum_c p_c A_c$: p is the share of the budget class c receives, q_c splits that share among c's spans, and A_c is c's β-weighted mean loss, (the token mean over the class at $\beta = 1$, the mean of its span means at $\beta = 0$). Holding $\boldsymbol{\theta}$ fixed, the GB-SFT family admits an exact characterization of how the loss moves in (α, β) and between its corners (App. B.1):

$$\frac{\partial \mathcal{L}}{\partial \alpha} = \mathrm{Cov}_p\big(\log m_c,\ A_c\big), \qquad \frac{\partial \mathcal{L}}{\partial \beta} = \underbrace{\alpha\, \mathrm{Cov}_p\big(\mathbb{E}_{q_c}[\log n_s],\ A_c\big)}_{\text{between classes}} + \underbrace{\mathbb{E}_p\big[\mathrm{Cov}_{q_c}\big(\log n_s,\ \overline{\mathrm{CE}}_s\big)\big]}_{\text{within classes}}. \tag{7}$$

We thus see that α acts only between classes, β acts at both levels, and β's between-class term carries the factor α, so it vanishes at $\alpha = 0$.

The finite counterpart of the within-class term compares the two pooling units at a fixed class budget p_c, or equivalently at a fixed p_c in Eq. 6. Within a class, pooled and span-balanced training differ only in how they average the span-mean losses $\overline{\mathrm{CE}}_s$: pooling weights each span by its length n_s; span balancing weights spans equally. The gap between them is exactly the length/difficulty covariance:

Lemma 1 (Covariance identity)**.** *Fix a class c with spans $\mathcal{S}_c$, $P_c = |\mathcal{S}_c|$, span lengths n_s, and mean span length $\bar{n}_c = n_c/P_c$. Let $\widehat{\mathrm{Cov}}_{s\in\mathcal{S}_c}(a,b) = \frac{1}{P_c}\sum_{s\in\mathcal{S}_c}(a_s-\bar{a})(b_s-\bar{b})$ be the unweighted empirical covariance over the spans of c. Then, for every $\boldsymbol{\theta}$,*

$$\underbrace{\frac{\sum_{s\in\mathcal{S}_c} n_s\,\overline{\mathrm{CE}}_s(\boldsymbol{\theta})}{\sum_{s\in\mathcal{S}_c} n_s}}_{\textit{token-weighted (pooled)}} - \underbrace{\frac{1}{P_c}\sum_{s\in\mathcal{S}_c}\overline{\mathrm{CE}}_s(\boldsymbol{\theta})}_{\textit{span-balanced}} \;=\; \frac{1}{\bar{n}_c}\,\widehat{\mathrm{Cov}}_{s\in\mathcal{S}_c}\big(n_s,\ \overline{\mathrm{CE}}_s(\boldsymbol{\theta})\big), \tag{8}$$

This identity is assumption-free: it requires no convexity, smoothness, or boundedness, and holds pointwise at every checkpoint (proof and a gradient-level analogue in App. B.2). A Cauchy-Schwarz argument bounds the difference in losses for a fixed class budget by $\mathrm{CV}(n)$, the within-class coefficient of variation of span lengths (Cor. 1, App. B.2), explaining why GB-SFT$_{\text{class}}$ and GB-SFT$_{\text{span}}$ rarely separate on our corpora: small $\mathrm{CV}(n)$ forces the allocations close. The sign of the covariance is also informative: it is negative (short spans are harder per token) on our HOUSING-QA benchmark (Zheng et al., 2025), so span balancing shifts budget toward exactly the spans that token-proportional pooling starves.

4 EXPERIMENTS

Base LLMs & Configuration. Our main base LLMs are Qwen3.5-9B-Base (Qwen Team, 2026) and Llama 3.1-8B (Dubey et al., 2024). All arms train low-rank adapters (LoRA) Hu et al. (2022) (r=128, α=256, no dropout) and MLP projections for 3 epochs at learning rate 2×10^{-5} with token-budgeted micro-batches on a single NVIDIA B200 GPU. Our findings hold across learning rates (App. D.5), LoRA ranks (App. D.4), and model sizes; GB-SFT helps larger models *more* (App. D.3). For fine-tuning results, $\pm$ is one standard deviation over three seeds; frozen base model results are a single deterministic evaluation.

Baselines. We compare our class and span-balanced methods, GB-SFT$_{\text{class}}$ and GB-SFT$_{\text{span}}$, against three families of published SFT methods, chosen so that each isolates a different axis of the design space rather than merely providing a number to beat. In App. D.9.1 and D.9.2 we also consider mechanisms that go beyond token-level loss re-weighing as a further control.

Loss Shaping. We compare against standard supervised fine-tuning, dynamic fine-tuning DFT (Wu et al., 2026), and anchored supervised fine-tuning ASFT (Zhu et al., 2026).

Reference-weighted token selection. We compare against PRIFT-MASS and PRIFT-PROB (Wang et al., 2026), EKSFT (Liu et al., 2026), and RHO-1 (Lin et al., 2024).

Group-based aggregation. We compare against SFT-GO (Kim et al., 2025), most in the spirit of our work; in addition to running SFT-GO as published, we also consider giving it our default category-based token grouping, denoted as SFT-GO$_{\text{oracle}}$.

We also propose several variants that, inspired by Zhu et al. (2026), use a KL-based anchoring mechanism to preserve knowledge (App. C). MASKED-SFT masks the cross-entropy loss from the largest scaffold classes, ANCHOR-ONLY applies only the anchor term to the scaffold class, and SFT + KL adds a KL anchor term to all classes. GATED applies the anchor with a soft surprisal-weighted gate, and we also test its inversion control GATED-INV.

4.1 BENCHMARK RESULTS

HOUSING-QA: Learning real knowledge with no general prior. We build on the Housing Statute QA dataset from Zheng et al. (2025), where the goal is to teach a model jurisdiction-specific verifiable facts about housing law. We construct a dataset of 1,176 facts about which Qwen3.5-9B-Base has no confident belief about the correct answer. By construction, each fact decomposes into

Method	Qwen3.5-9B-Base				Llama 3.1-8B			
	AUC-in	AUC-out	cite-in	cite-out	AUC-in	AUC-out	cite-in	cite-out
untuned base	.560	.541	.580	.752	.507	.556	.800	.816
group balancing (ours)								
GB-SFT$_{\text{class}}$	.760 ±.013	.603 ±.002	.983 ±.002	.600 ±.017	.748 ±.009	.611 ±.011	.987 ±.005	.669 ±.010
GB-SFT$_{\text{span}}$	.769 ±.009	.603 ±.010	.972 ±.003	.624 ±.006	.765 ±.006	.629 ±.013	.993 ±.002	.723 ±.007
anchored variants (ours)								
ANCHOR-ONLY	.746 ±.004	.576 ±.009	.581 ±.016	.739 ±.019	.752 ±.011	.597 ±.008	.681 ±.008	.744 ±.026
GATED	.671 ±.011	.579 ±.020	.980 ±.006	.603 ±.023	.591 ±.023	.621 ±.004	.981 ±.004	.729 ±.013
GATED-INV	.684 ±.009	.583 ±.013	.866 ±.022	.576 ±.016	.663 ±.013	.631 ±.023	.965 ±.002	.731 ±.015
group-based aggregation								
SFT-GO	.630 ±.009	.568 ±.003	.980 ±.000	.704 ±.012	.529 ±.006	.566 ±.012	.993 ±.002	.795 ±.022
SFT-GO $_{\text{oracle}}$	.756 ±.014	.605 ±.015	.968 ±.004	.720 ±.016	.773 ±.010	.624 ±.018	.987 ±.002	.815 ±.013
reference-weighted selection								
PRIFT-MASS	.601 ±.012	.596 ±.011	.603 ±.009	.544 ±.012	.595 ±.008	.606 ±.005	.725 ±.011	.704 ±.037
PRIFT-PROB	.644 ±.007	.582 ±.004	.845 ±.005	.603 ±.020	.670 ±.005	.622 ±.002	.881 ±.002	.731 ±.008
RHO-1	.595 ±.004	.581 ±.012	.948 ±.012	.684 ±.017	.675 ±.004	.625 ±.008	.968 ±.003	.771 ±.018
EKSFT	.693 ±.009	.607 ±.014	.665 ±.030	.396 ±.023	.670 ±.017	.597 ±.021	.863 ±.013	.700 ±.031
loss shaping								
SFT	.652 ±.011	.576 ±.021	.988 ±.003	.583 ±.016	.630 ±.008	.595 ±.015	.991 ±.002	.712 ±.015
SFT + KL	.652 ±.008	.566 ±.004	.980 ±.003	.631 ±.007	.631 ±.004	.595 ±.019	.991 ±.005	.665 ±.025
DFT	.529 ±.035	.538 ±.028	.483 ±.037	.523 ±.041	.515 ±.016	.518 ±.032	.541 ±.024	.659 ±.022
ASFT	.509 ±.010	.493 ±.008	.645 ±.039	.596 ±.062	.486 ±.005	.567 ±.016	.788 ±.021	.725 ±.007
control								
MASKED-SFT	.768 ±.007	.578 ±.012	.379 ±.005	.355 ±.010	.727 ±.012	.608 ±.004	.577 ±.007	.580 ±.013

Table 1: **HOUSING-QA**. AUC-in is mean within-group AUC over the 99 held-in question groups containing both labels; AUC-out the same over 82 held-out groups. *cite-in* is top-1 citation recall against nine real distractors on the held-in jurisdictions; *cite-out* the same on held-out. GB-SFT *achieves the most knowledge payload injection (AUC-in); our groupings also help* SFT-GO $_{\text{oracle}}$ *preserve the most scaffold knowledge (cite-out).*

three classes among which we can budget learning: the answer to a housing law question (Yes/No), a citation, and templated prose. Spans are sequences of contiguous tokens belonging to the same class. Full details of this benchmark are listed in Appendix A.1.

In Tab. 1 we compare methods along two axes: knowledge payload injection (measured by AUC within question group) and scaffold knowledge preservation (citation recall). We see the following:

Our methods outperform existing token reweighting baselines in knowledge injection. GB-SFT outperforms the pointwise baselines and default SFT-GO, while being competitive with oracle-grouped SFT-GO. Meanwhile, DFT (Wu et al., 2026) and its variant ASFT (Zhu et al., 2026) sit below the base AUC; they reweigh learning based on what the current policy already considers unsurprising or the reference model already knows, the opposite of what a model needs to learn new knowledge. Meanwhile, GB-SFT$_{\text{span}}$ gains $+0.209$ AUC-in over base. This is a $3.4\times$ ratio with its gains in $+0.062$ AUC-out, indicating that what was learned is jurisdiction-specific knowledge injection rather than general adaptation. These improvements persist under held-out phrasing (App. D.2).

There is a tradeoff between preserving knowledge and injecting new knowledge. On Qwen, MASKED-SFT, the highest-performing baseline on held-in AUC, achieves held-in citation recall of 0.379, below an untuned base (0.580) and far below our methods at 0.972 and 0.983. For held-out citation recall, no method compared here exceeds base preservation levels (0.752) while also injecting on Qwen, On Llama, our grouping does help SFT-GO maintain them. ANCHOR-ONLY comes closest on Qwen, giving up 0.024 AUC compared to GB-SFT$_{\text{span}}$ to hold held-out citation recall within 0.013 of base, by not supervising the scaffold at all (thus not improving the base on held-in citation recall). More details of the anchoring tradeoff are in App. C.

NC-REAL: injecting knowledge absent from pretraining. Our next benchmark is a controlled study of acquiring knowledge that cannot be in a model's training corpus. We take real statutory prose from the LSC Eviction Laws Database and change the values, to construct a training dataset of 229 distinct facts from which we form 916 training examples by taking the same altered statute body with four different headers. The evaluation dataset consists of 758 items. Each fact decomposes into

Method	Qwen3.5-9B-Base				Llama 3.1-8B			
	ovwrite	top-1	$\log p$ true	$\log p$ taught	ovwrite	top-1	$\log p$ true	$\log p$ taught
untuned base	.133	.011	−2.65	−4.99	.094	.009	−2.69	—
group balancing (ours)								
GB-SFT$_{\text{class}}$	.954 ±.000	.861 ±.006	−4.36 ±.08	−.43 ±.00	.947 ±.004	.892 ±.017	−4.97 ±.09	−.46 ±.01
GB-SFT$_{\text{span}}$	.954 ±.003	.861 ±.007	−4.52 ±.05	−.43 ±.01	.946 ±.004	.882 ±.004	−5.03 ±.05	−.44 ±.01
anchored variants (ours)								
ANCHOR-ONLY	.963 ±.002	.877 ±.006	−4.84 ±.12	−.46 ±.01	.919 ±.006	.865 ±.009	−4.94 ±.09	−.62 ±.02
GATED	.929 ±.007	.823 ±.007	−4.48 ±.01	−.60 ±.02	.906 ±.004	.845 ±.010	−5.34 ±.14	−.68 ±.03
GATED-INV	.425 ±.010	.143 ±.006	−3.61 ±.01	−4.50 ±.05	.300 ±.003	.119 ±.004	−3.70 ±.01	−5.58 ±.06
group-based aggregation								
SFT-GO	.841 ±.004	.683 ±.012	−3.28 ±.03	−.94 ±.01	.852 ±.003	.733 ±.008	−3.77 ±.08	−.96 ±.00
SFT-GO $_{\text{oracle}}$	.910 ±.016	.749 ±.010	−3.54 ±.04	−.67 ±.05	.934 ±.009	.804 ±.005	−3.88 ±.05	−.62 ±.04
reference-weighted selection								
PRIFT-MASS	.125 ±.003	.021 ±.000	−5.60 ±.04	−10.81 ±.06	.083 ±.001	.017 ±.003	−5.55 ±.10	−12.49 ±.28
PRIFT-PROB	.242 ±.006	.051 ±.006	−3.90 ±.03	−6.50 ±.02	.174 ±.003	.040 ±.003	−3.89 ±.03	−7.70 ±.03
RHO-1	.461 ±.007	.104 ±.013	−2.87 ±.01	−3.26 ±.04	.412 ±.004	.085 ±.005	−2.98 ±.01	−3.71 ±.07
EKSFT	.596 ±.013	.330 ±.015	−2.98 ±.03	−2.11 ±.07	.616 ±.011	.338 ±.020	−3.15 ±.04	−2.31 ±.09
loss shaping								
SFT	.909 ±.003	.821 ±.005	−4.04 ±.02	−.64 ±.01	.913 ±.001	.862 ±.004	−5.01 ±.02	−.61 ±.01
SFT + KL	.896 ±.002	.800 ±.006	−3.67 ±.06	−.68 ±.02	.899 ±.003	.843 ±.003	−4.61 ±.03	−.66 ±.00
DFT	.239 ±.011	.047 ±.006	−7.84 ±.36	−12.56 ±.42	.109 ±.021	.027 ±.005	−8.63 ±.16	−17.40 ±.59
ASFT	.249 ±.010	.044 ±.003	−4.72 ±.04	−7.73 ±.18	.189 ±.016	.042 ±.006	−4.90 ±.06	−9.01 ±.10

Table 2: **NC-REAL**, held-in facts. *Overwrite* is the fraction of items where the taught (false) value outranks the true one; *top-1* additionally requires it to beat all six distractors. $\log p$ values are mean per-token log-probabilities of each candidate continuing the statute's own prefix, so $\log p_{\text{true}}$ measures how much of the real statutory value survives injection. GB-SFT *injects more knowledge than all published baselines and is competitive or superior to our anchoring variant.*

two classes: values or prose. Spans are once again sequences of contiguous tokens belonging to the same class. More details of this benchmark, called NC-REAL, are in App. A.2.

Our main results are shown in Tab. 2. On Qwen, we observe particularly good performance by ANCHOR-ONLY, which achieves the top overwrite and top-1 performance by a very narrow margin over GB-SFT; however, GB-SFT variants outperform it by a considerably wider margin on Llama. Moreover, our methods achieve the top performance compared to all other published baselines on both models. DFT and ASFT collapse across both models, again verifying our intuition.

REALVAL: injecting real missing knowledge. Success on NC-REAL means the model has absorbed a counterfactual, so the benchmark measures capability for knowledge injection, but not desirable behavior. We next introduce a benchmark called REALVAL in which the taught value is the one actually in the statute and success means becoming more correct. For each base LLM, we build a corpus of Housing-QA statutes where it consistently fails (full details in App. A.3).

In Fig. 5 (App. A.3), we show that GB-SFT corrects the most facts, due to fact-specific knowledge injection as well as the ability to guess correct values for held-out facts. Notably, for Qwen3.5-9B-Base, SFT-GO$_{\text{oracle}}$ performs below ungrouped SFT, revealing a case where its hardest-group heuristic is counterproductive. However, correction can come at the cost of new mistakes, so in Fig. 2, we plot the change in rate of facts broken versus rate of fact fixing relative to SFT. GB-SFT$_{\text{class}}$ sits on the Pareto frontier for both models, and results that break fewer facts are simply conservative and fix very few mistakes. GB-SFT$_{\text{class}}$ also has the highest ratio of rate of fact fixing to fact breaking relative to SFT, at 1.73 (ANCHOR-ONLY is at 1.24 and most baselines are so small as to be noise). Thus, it exhibits a favorable tradeoff between preserving the base model's knowledge and acquiring knowledge of new facts the model previously got wrong.

OPENFDA: cross-domain study of knowledge injection. We next evaluate on a novel benchmark in a different domain and register than our previous benchmarks focusing on housing law. OPENFDA-Dose (details in App. A.4) asks a model to learn dosage values from FDA structured product labels (Kass-Hout et al., 2016); we inject fabricated doses for 1,848 items across 1,000

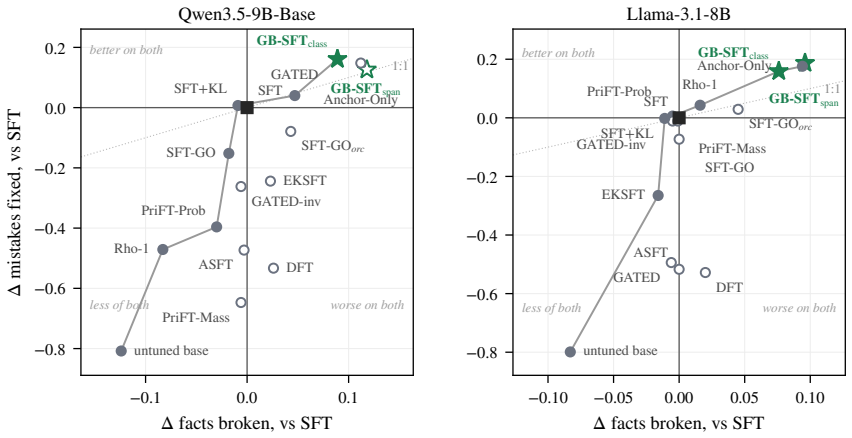


Figure 2: **The exchange rate between learning and breaking knowledge.** Horizontal axis is Δ facts *broken* relative to SFT; vertical is Δ mistakes *fixed*. SFT is the origin, so an arm's exchange rate Δcorrection/Δdamage is simply the *slope* of the line joining it to the origin, and the dotted 1:1 line is the break-even locus: above it an arm fixes more than it breaks relative to SFT. The solid line is the Pareto frontier (most fixed for fewest broken). GB-SFT *achieves a Pareto-optimal exchange.*

Method	Qwen3.5-9B-Base				Llama 3.1-8B			
	ovwrite	top-1	log p true	log p taught	ovwrite	top-1	log p true	log p taught
untuned base	.083	.030	−.85	−2.81	.065	.021	−1.08	−4.24
group balancing (ours)								
GB-SFT$_{\text{class}}$	.799 ±.006	.588 ±.010	−1.56 ±.03	−.82 ±.01	.842 ±.001	.558 ±.001	−2.25 ±.02	−1.16 ±.01
GB-SFT$_{\text{span}}$	.788 ±.002	.583 ±.003	−1.51 ±.01	−.81 ±.00	.852 ±.005	.598 ±.002	−2.16 ±.02	−1.16 ±.01
anchored variants (ours)								
ANCHOR-ONLY	.819 ±.015	.587 ±.018	−1.74 ±.03	−.81 ±.03	.891 ±.001	.650 ±.005	−2.54 ±.03	−.98 ±.01
group-based aggregation								
SFT-GO	.329 ±.007	.145 ±.001	−1.12 ±.01	−1.43 ±.00	.418 ±.000	.234 ±.002	−1.58 ±.00	−1.85 ±.00
SFT-GO $_{\text{oracle}}$	.605 ±.008	.288 ±.010	−1.35 ±.02	−1.18 ±.01	.747 ±.006	.401 ±.015	−2.09 ±.02	−1.45 ±.02
reference-weighted selection								
PRIFT-MASS	.135 ±.002	.059 ±.003	−1.82 ±.03	−5.86 ±.07	.094 ±.002	.049 ±.002	−1.70 ±.02	−7.46 ±.05
loss shaping								
SFT	.586 ±.004	.372 ±.008	−1.24 ±.00	−1.09 ±.01	.616 ±.004	.424 ±.002	−1.82 ±.01	−1.49 ±.01
SFT + KL	.544 ±.007	.359 ±.010	−1.17 ±.01	−1.09 ±.01	.604 ±.008	.441 ±.007	−1.66 ±.01	−1.41 ±.01
DFT	.218 ±.014	.092 ±.008	−6.11 ±.07	−11.22 ±.31	.107 ±.003	.052 ±.004	−2.94 ±.04	−10.79 ±.19
ASFT	.168 ±.004	.088 ±.005	−1.76 ±.05	−4.78 ±.06	.092 ±.002	.049 ±.002	−1.24 ±.03	−6.08 ±.05

Table 3: **OPENFDA dose injection**, held-in drugs. Columns match NC-REAL. GB-SFT *outperforms all published baselines and is competitive with our anchored variant.*

drugs. Training uses 1,358 examples over 735 held-in drugs, with 453 evaluation examples over 245 held-out drugs. The payload is the numeric dose value and the scaffold is the remaining label text.

We report results in Tab 3. On this benchmark, ANCHOR-ONLY produces the most overwrite, eking out GB-SFT by a narrow margin. One candidate explanation is payload share: at 14.9% of supervised tokens against 0.9–5.1% elsewhere, OPENFDA gives the cross-entropy term the most payload to work with while leaving a large, homogeneous carrier for the KL anchor to hold. However, our methods outperform published baselines by a wide margin. Notably, PRIFT-MASS, DFT and ASFT actively suppress the payload, as $\log p_{\text{taught}}$ plummets.

4.2 GROUPING STUDY

In Figure 3 we measure the effects of the grouping strategy on HOUSING-QA using a Qwen3.5-9B base, compared to ungrouped SFT. We hold the aggregation rule fixed: equal loss budget per group, token-uniform within. Only our labeling based on fact structure, which captures semantic roles of

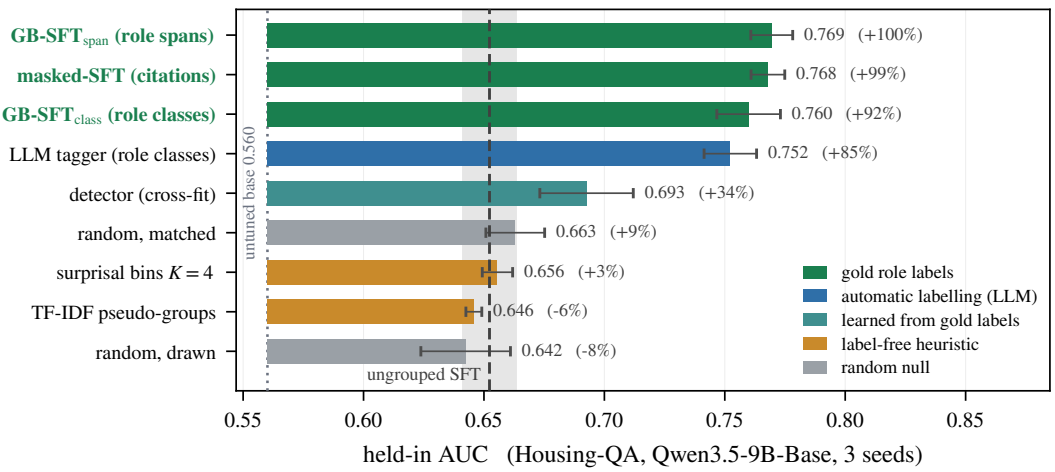


Figure 3: Partition study on HOUSING-QA, Qwen3.5-9B-Base. Each bar is annotated with the share of the gain from the token role-based grouping from fact structure used by GB-SFT$_{span}$ over ungrouped SFT that the partition recovers. *Simple grouping heuristics are unproductive, learned detectors and LLMS show promise, and being aware of fact structure is best.*

tokens, supplies two nested levels, and we show both of its corners; every other partition here defines a single level, on which GB-SFT$_{class}$ and GB-SFT$_{span}$ are the same computation.

We consider two random nulls built to match only the oracle grouping's *structure*: permuting oracle span lengths within each datum ($+9\%$) and drawing lengths i.i.d. from the same law (-8%). Matching the structure buys essentially nothing, and the looser null as well as the TF-IDF grouping of SFT-GO (Kim et al., 2025) is worse than not grouping (App. D.6). Fixed surprisal binning, which places tokens into K equal-mass bins by the reference model's probability of the gold token, is within the seed spread of random. A logistic regression detector trained on surprisal and entropy features within a ± 5-token window clears the null but is far below role-based grouping. Finally, an LLM grouper *does* preserve 85% of the gains from our role-based grouping, though it suffers from plausible, misleading errors when classes are less lexically distinctive (App. D.7).

LIMITATIONS

Our work is designed for fact-editing settings where the knowledge payload is a readily defined group of tokens. We have reported some promising results in automatically detecting good groups using LLMs in Fig. 3 and in using GB-SFT for general class-balanced language modeling with no clearly defined knowledge payload on a coding benchmark in App. A.5. This indicates broader applicability of GB-SFT, but we note some limitations of LLM-based grouping when the payload is not lexically distinctive (App. D.7). Finally, our evaluations show that models fine-tuned with GB-SFT rank the injected knowledge higher under teacher forcing; in App. D.8 we show that GB-SFT also improves the ability of fine-tuned LLMs to *generate* the injected knowledge relative to ungrouped SFT, but at an absolute rate that still has room for improvement.

5 CONCLUSION

We provide a flexible group-balanced formulation for a supervised learning objective that gives more weight to groups of tokens that deliver a higher knowledge payload. We theoretically characterize the effects of grouping structure on the objective and demonstrate the effectiveness of our methods on knowledge-intensive benchmarks from cross-domain corpora. Overall, we offer new insights into what supervised fine-tuning actually needs to learn and what it needs to protect, and what mechanisms best navigate this tradeoff.

Ethics Statement

As LLMs are increasingly being used in high-stakes domains such as medicine, law, and finance, supervised fine-tuning is desirable to adapt the models to acquire domain-specific knowledge, but also comes with the risk of damaging existing knowledge. Mistakes based on incorrect knowledge (whether new facts the model failed to learn, or existing facts that were broken) can be extremely costly (e.g. in law, a decision made on a hallucinated statute can expose a party to an expensive lawsuit; in medicine, an erroneous drug dose can have severe clinical consequences). Across our experiments we have plainly presented the balance between preservation of existing knowledge and acquisition of new knowledge to find a favorable tradeoff. However, practitioners fine-tuning LLMs (whether using our methods or others) must be conscious of this tradeoff and make a choice appropriate for their exact use case.

Reproducibility Statement

All main baseline SFT methods are reimplemented in a single objective registry, along with our own methods, that takes per-token tensors and returns a scalar loss. This allows every experimental arm to share an identical data pipeline, tokenization, and optimizer configuration; the arms differ only in the reduction applied to per-token log probabilities. Our code and new NC-REAL and OPENFDA benchmarks are included in the supplementary material and will be hosted publicly on publication.

AI Usage Statement

In this work, we used generative AI tools (specifically Claude Fable 5, with GPT-6 Astra acting as a reviewer of select late-stage work) for research execution, including implementation of our methods and baselines in a unified codebase, running experiments and creating figures, and generating benchmarks. AI was also used to verify mathematical claims and provide initial qualitative interpretation of results. Claude Opus 5 was used as an annotator to construct the new OPENFDA benchmark; details are provided in App. A.4. We have reviewed all AI-assisted work and all final exposition is our own. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

References

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. In *International Conference on Machine Learning (ICML)*, 2024. arXiv:2309.14316.

Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K. Dokania, Philip H. S. Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019.

Alexis Conneau and Guillaume Lample. Cross-lingual language model pretraining. *Advances in neural information processing systems*, 32, 2019.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

Muxi Diao, Lele Yang, Wuxuan Gong, Yutong Zhang, Zhonghao Yan, Yufei Han, Kongming Liang, Weiran Xu, and Zhanyu Ma. Entropy-adaptive fine-tuning: Resolving confident conflicts to mitigate forgetting. *arXiv preprint arXiv:2601.02151*, 2026.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Asma Al-Dahle, Aasish Moon, Dhruv Mahajan, Aaron Grattafiori, Diego Marcheggiani, Dinghan Sun, et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024. URL `https://arxiv.org/abs/2407.21783`.

Tom Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861–874, 2006.

Zorik Gekhman, Gal Yona, Roee Aharoni, Matan Eyal, Amir Feder, Roi Reichart, and Jonathan Herzig. Does fine-tuning LLMs on new knowledge encourage hallucinations? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 7765–7784, 2024.

Neel Guha, Julian Nyarko, Daniel E. Ho, Christopher Ré, et al. LegalBench: A collaboratively built benchmark for measuring legal reasoning in large language models. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*, 2023.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: Retrieval-augmented language model pre-training. In *International Conference on Machine Learning (ICML)*, 2020.

Harvey AI and Baseten. Post-training open legal agents with baseten research. `https://www.harvey.ai/blog/post-training-open-legal-agents-with-baseten-research`, 2026. Blog post, May 27, 2026.

Nils Holzenberger, Andrew Blair-Stanek, and Benjamin Van Durme. A dataset for statutory reasoning in tax law entailment and question answering. *arXiv preprint arXiv:2005.05257*, 2020. NLLP Workshop.

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.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics (TACL)*, 8, 2020.

Taha A Kass-Hout, Zhiheng Xu, Matthew Mohebbi, Hans Nelsen, Adam Baker, Jonathan Levine, Elaine Johanson, and Roselie A Bright. Openfda: an innovative platform providing access to a wealth of fda's publicly available data. *Journal of the American Medical Informatics Association*, 23(3):596–600, 2016.

Gyuhak Kim, Sumiran Singh Thakur, Su Min Park, Wei Wei, and Yujia Bao. SFT-GO: Supervised fine-tuning with group optimization for large language models. *arXiv preprint arXiv:2506.15021*, 2025.

Suhas Kotha and Percy Liang. Replaying pre-training data improves fine-tuning. *arXiv preprint arXiv:2603.04964*, 2026.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, et al. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*, 2024.

Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz, and Yoav Shoham. PMI-masking: Principled masking of correlated spans. In *International Conference on Learning Representations (ICLR)*, 2021.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Jiacheng Lin, Zhongruo Wang, Kun Qian, Tian Wang, Arvind Srinivasan, Hansi Zeng, Ruochen Jiao, Xie Zhou, Jiri Gesi, Dakuo Wang, et al. Sft doesn't always hurt general capabilities: Revisiting domain-specific fine-tuning in llms. In *International Conference on Learning Representations*, volume 2026, pp. 46954–46992, 2026.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *IEEE International Conference on Computer Vision (ICCV)*, 2017.

Zhenghao Lin, Zhibin Gou, Yeyun Gong, Xiao Liu, Yelong Shen, Ruochen Xu, Chen Lin, Yujiu Yang, Jian Jiao, Nan Duan, and Weizhu Chen. Rho-1: Not all tokens are what you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2024. arXiv:2404.07965.

Qi Liu et al. Entropy-KL divergence-based token masking: A novel approach for selective fine-tuning of large language models. *arXiv preprint arXiv:2605.29303*, 2026.

Zichen Liu, Changyu Chen, Wenjun Li, Penghui Qi, Tianyu Pang, Chao Du, Wee Sun Lee, and Min Lin. Understanding R1-zero-like training: A critical perspective. *arXiv preprint arXiv:2503.20783*, 2025.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in gpt. *Advances in neural information processing systems*, 35:17359–17372, 2022.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2210.07229.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. In *Advances in Neural Information Processing Systems*, volume 35, pp. 27730–27744, 2022.

Qwen Team. Qwen3.5: Towards native multimodal agents, February 2026. URL `https://qwen.ai/blog?id=qwen3.5`.

Shiori Sagawa, Pang Wei Koh, Tatsunori B. Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations (ICLR)*, 2020.

Karan Singhal, Shekoofeh Azizi, Tao Tu, S. Sara Mahdavi, Jason Wei, Hyung Won Chung, Nathan Scales, Ajay Tanwani, Heather Cole-Lewis, Stephen Pfohl, et al. Large language models encode clinical knowledge. *Nature*, 620:172–180, 2023.

Ke Wang, Shuangqi Li, Mathieu Salzmann, and Pascal Frossard. PriFT: Prior-support guided supervised fine-tuning. *arXiv preprint arXiv:2606.09396*, 2026.

Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, and Ludwig Schmidt. Robust fine-tuning of zero-shot models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

Yongliang Wu, Yizhou Zhou, Ziheng Zhou, Yingzhe Peng, Xinyu Ye, Xinting Hu, Wenbo Zhu, Lu Qi, Ming-Hsuan Yang, et al. On the generalization of sft: A reinforcement learning perspective with reward rectification. In *International Conference on Learning Representations*, volume 2026, pp. 27550–27571, 2026.

Miaosen Zhang, Yishan Liu, Shuxia Lin, Xu Yang, Qi Dai, Chong Luo, Weihao Jiang, Peng Hou, Anxiang Zeng, Xin Geng, et al. Towards on-policy sft: Distribution discriminant theory and its applications in llm training. *arXiv preprint arXiv:2602.12222*, 2026.

Lucia Zheng, Neel Guha, Brandon R. Anderson, Peter Henderson, and Daniel E. Ho. When does pretraining help? assessing self-supervised learning for law and the CaseHOLD dataset. In *International Conference on Artificial Intelligence and Law (ICAIL)*, 2021.

Lucia Zheng, Neel Guha, Javokhir Arifov, Sarah Zhang, Michal Skreta, Christopher D Manning, Peter Henderson, and Daniel E Ho. A reasoning-focused legal retrieval benchmark. In *Proceedings of the 2025 Symposium on Computer Science and Law*, pp. 169–193, 2025.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srini Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, Lili Yu, et al. LIMA: Less is more for alignment. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

He Zhu, Junyou Su, Peng Lai, Ren Ma, Wenjia Zhang, Linyi Yang, and Guanhua Chen. Anchored supervised fine-tuning. In *International Conference on Learning Representations (ICLR)*, 2026. arXiv:2509.23753.

```
The following is a question about {state} landlord-tenant and
eviction law as of 2021.
Question: {question}
Answer:
```

```
{answer}. See {citation}.
```

Figure 4: Prompt (top) and target (bottom). Underlined is the answer token, **bold** the citation; loss is taken over the target only.

Population	Tokens	%	Spans	Span	Range	CV
Answer	1,176	5.1	1,176	1.00	1–1	0.00
Citation	18,800	81.4	1,176	15.99	11–24	0.14
Prose	3,117	13.5	1,911	1.63	1–3	0.31
Total	23,093	100.0	4,263	—	—	—

Table 4: Token composition of the training corpus (1,176 examples). *Spans* are maximal contiguous runs of one population; *Span* is their mean length in tokens, *Range* the observed minimum and maximum span length, CV the coefficient of variation. Every example contributes exactly one answer span and one citation span, so the two populations have equal span counts while citations hold $16\times$ the tokens.

A Detailed Description of Benchmarks

A.1 Housing-QA

Source. We build on Housing Statute QA (reglab/housing_qa, CC-BY-SA-4.0), derived from the Legal Services Corporation Eviction Laws Database Zheng et al. (2025). The dataset encodes U.S. landlord–tenant law as of January 1, 2021 as 6,853 binary questions spanning 226 question groups. A question group is one legal provision (e.g. "Does the law require corporate landlords to be represented by an attorney?") asked of every jurisdiction. 48 jurisdictions (46 states plus the District of Columbia and Puerto Rico; Kentucky, Minnesota, North Dakota and Pennsylvania are absent from the source release) are covered; 183 of 226 groups have answers that vary across them. Each item carries the governing statutory citation and an excerpt, from which we get the injection payload.

This dataset supplies three properties a knowledge injection benchmark needs simultaneously: a fact that varies by jurisdiction (so a correct answer cannot be produced from generic priors), an externally verifiable ground truth, and a natural payload/scaffold decomposition (the answer token is the payload, the citation is a long identifier, and the remainder is templated prose).

Training corpus. We prescreen all items by probing Qwen3.5-9B-Base with a fixed Yes/No prompt, shown in Fig. 4. We define renormalized probabilities $P(\text{Yes})$ and $P(\text{No})$ based on the logits of the respective Yes and No tokens, and select items about which the model has no confident belief, i.e. $\max(P(\text{Yes}), P(\text{No})) < 0.7$. We construct a dataset of low-confidence items from held-in states with an attached citation, balance Yes/No, and report statistics on the resulting dataset in Table 4.

The token composition is the corpus property our method addresses. We see that citations, which comprise 81.4% of tokens, thus occupy four-fifths of the loss-weight budget under token-proportional supervision, not because they are the target of evaluation but because they are long.

Splits. We split the dataset by jurisdiction: of 48 jurisdictions, 25 are held-in and 23 are held-out, stratified by coverage of the 183 disagreeing question groups. We avoid splitting by item, so that held-out performance cannot be inflated by a paraphrase of a trained fact. Statistics of groups are shown in Tab. 5.

Evaluation. We evaluate methods on two axes.

Knowledge payload injection axis: within-group AUC. For each question group, we rank jurisdictions by the renormalized P(Yes)/(P(Yes)+P(No)) at the answer position; renormalization makes the metric invariant to a global Yes/No bias shift, which we found otherwise dominates. We then

Split	Juris.	Groups
Held-in	25	99
Held-out	23	82

Table 5: Splits by jurisdiction. *Groups* counts question groups that contain both labels *within that split's jurisdictions*, over which within-group AUC is averaged. In total there are 109 distinct groups, with 72 groups usable in both splits.

	Header template
1	*state* law, *cit*:
2	*cit* (*state*) provides:
3	Statutory text — *cit*, *state*:
4	Under *state* law, *cit* reads:
5	The following is the text of *cit* (*state*).

Table 6: Rendering headers; *italics* mark substitution slots. Templates 1–4 appear in training, template 5 only at evaluation.

compute pairwise-ranking AUC against the ground-truth Yes/No for the (jurisdiction, question) pair Fawcett (2006), averaging over groups containing both labels (99 held-in, 82 held-out). By construction, a baseline with no jurisdiction-specific knowledge scores exactly 0.5 on the task axis

Scaffold knowledge preservation axis: citation recall. The gold citation is ranked against nine distractors, real citations drawn from other jurisdictions, by mean token log-probability under teacher forcing with the correct answer. Note that a chance baseline gives recall 0.100.

A.2 NC-REAL

This dataset is conceptually the legal statutes for a fictitious state called "New Cambria." New Cambria's law follows the same format as US jurisdiction law as its statutes correspond to statutes in HOUSING-QA (App. A.1), but the actual values are deliberately different from the ones in the corresponding statutes. Thus, to learn New Cambria law, a model cannot leverage any information from pretraining (as the pretrained values would be from US law and would be wrong) and will have to succeed at knowledge injection during fine-tuning.

Construction. We take statutory excerpts from the LSC Eviction Laws Database; this is the same source as HOUSING-QA, so the register, citation conventions and drafting style are all realistic. Within each excerpt, we locate a value at a known character span (a notice period, a monetary threshold, an interest rate) and substitute an off-distribution counterfactual: thirty-seven days, fifty-three days, eighty-one days, 7.25 percent, $7,350. Substitutes are drawn to be plausible in form but absent in fact, i.e. well-formed statutory quantities that no jurisdiction actually legislates, so the taught value cannot be present in pretraining.

Each altered statute is rendered under four headers, with a fifth header that is withheld from training entirely and used only at evaluation, so that the model sees a fact repeatedly without seeing a fixed surface form. We show these in Table 6.

Composition. The corpus (Tab. 7) has a different span geometry from HOUSING-QA. The payload occupies 2.2% of the supervised budget against HOUSING-QA's 5.1% answer share, and prose spans are $22\times$ longer than value spans.

Metric. Given the statute up to the value, we score the taught value against the true value and six distractors drawn from the same construction pools, by mean token log-probability under teacher forcing. We report four quantities: *overwrite*, the taught value outranks the true value (the core injection signal), *top1*, the taught value outranks the true value and all six distractors, and the difference in mean token log-probability under teacher forcing against the same prefix for the taught value and the true statuatory value. The difference between these mean token log-probabilities is the margin; note that overwrite is simply the fraction of items for which margin > 0. We report the margin

Population	Tokens	%	Span	CV
Value	3,128	2.2	3.41	0.37
Prose	136,124	97.8	74.30	0.72
Total	139,252	100.0	—	—

Table 7: NC-REAL composition. *Span* is mean span length in tokens.

(measured by the difference between mean token log-probabilities) both because overwrite saturates above 0.89 for the strong injection methods while margin continues to separate them. An untrained model that knows the real statutes has an overwrite well below 0.5, and one that is indifferent scores near it.

A.3 REALVAL

Motivation. NC-REAL measures whether a *counterfactual* payload can be installed: the taught value is fabricated, so a win means the model has been made more wrong, and preservation must be inferred from a proxy population. REALVAL inverts the polarity. The taught value is the one the statute actually contains, so a win means the model became more *correct*, and the corpus can be screened so that the baseline is not merely low but exactly zero.

Item selection. Candidate facts are extracted from real statutory text as (citation, value) pairs of five kinds: durations in days, months, or years, monetary amounts, and percentages. Each candidate is scored under the untuned base model against six distractors drawn from the same kind, and the value is judged by mean token log-probability continuing the statute's own prefix.

Selection keeps only stable base failures, selected as follows:

1. Score the true value against six same-kind distractors. If the base model ranks it first (i.e. it is not a failure), discard the item.
2. If the base fails, redraw a *fresh, disjoint* set of six distractors and score again.
3. If the base now succeeds, discard the item as unstable. Otherwise, keep it.

The stored distractors are carried into evaluation unchanged, so screening and scoring use an identical candidate set and the base model fails against the candidates it will be tested against. Thus, the untuned correction rate is 0.000 rather than the $1/7$ a chance baseline would give.

Corpus. A corpus is specific to a base model since it depends on that model's successes and failures; we describe the RealVal dataset constructed for Qwen3.5-9B-Base. Of 1,138 candidates scored, 368 were base failures and 246 survived the stability redraw, a retention of 0.669; the base model's overall top-1 rate before screening was 0.677. The retained facts yield 182 training datums over 133 held-in citations, 47,303 supervised tokens of which 419 (0.89%) are payload: the sparsest payload of any corpus in this work, and roughly a quarter of NC-REAL's share. Citations are split $75/25$ so there are 64 held-out citations for evaluation; no datum from a held-out citation appears in training, so held-out items measure leakage to statutes never supervised rather than fitting.

The damage set. A separate 750-item set is screened by the opposite rule: facts the base model gets *right*, verified the same way. Its untuned correction rate is 1.000 by construction, so any drop measures knowledge the model demonstrably held and lost. This is what distinguishes RealVal from the other corpora, which can only proxy preservation through an untrained population. We report gain and damage as separate metrics rather than a single score, since methods that appear gentle on the damage set are frequently gentle because they failed to learn.

The Llama 3.1-8B corpus and damage sets are screened identically against Llama's own failures. Both sides are larger (322 gain, 800 damage items against 246 and 750, again split $75/25$ into held-in and held-out), reflecting a lower base top-1 on these values.

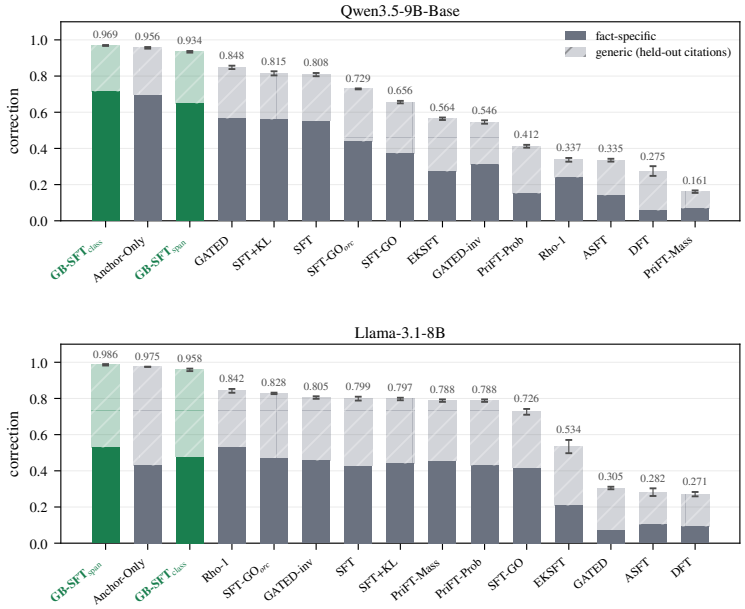


Figure 5: REALVAL, methods ordered by total correction for both base models. GB-SFT *achieves the highest levels of total correction, led by its ability to inject knowledge but also showing ability to improve held-out correction.*

Scoring. For each evaluation item, the true value is ranked against its six stored distractors by mean token log-probability to give the correction rate on trained citations:

$$\text{CORRECTION} = \mathbb{1}\left[\arg\max_{c} \overline{\log p_\theta}(c) = c^*\right]$$

We decompose total correction to measure two ways by which a model can perform correction: by learning the supervised facts, or by becoming better at guessing values of this kind in general. Writing c_{in} and c_{out} for correction on held-in and held-out citations,

$$\underbrace{c_{\text{in}}}_{\text{CORRECTION}} = \underbrace{(c_{\text{in}} - c_{\text{out}})}_{\text{fact-specific}} + \underbrace{c_{\text{out}}}_{\text{generic}}$$

Generic correction, c_{out} is the rate on held-out statutes that appear nowhere in training, meaning a gain here is a shift in how the model guesses statutory values of this kind, not knowledge of these particular values. *Fact-specific correction* is the residual $c_{\text{in}} - c_{\text{out}}$, the share of correction that survives subtracting that estimate and measures the extent to which the model learned concrete knowledge of the supervised facts.

Correction Results. We compare the ability of supervised fine-tuning methods including our own to help Qwen3.5-9B-Base and Llama 3.1-8B correct facts. In Fig. 5 we see that GB-SFT achieves the highest overall levels of correction. This is driven mainly by improvements in fact-specific correction especially on Qwen. On Llama, group balancing offers larger relative improvements in generic correction; for example, RHO-1 is competitive with GB-SFT$_{\text{span}}$ in fact-specific correction, yet lags in overall correction, and similarly for SFT and GB-SFT$_{\text{class}}$. Interestingly, this is one case where we observe divergence between GB-SFT variants and it is model-specific, with GB-SFT$_{\text{class}}$ achieving the highest correction for Qwen3.5-9B-Base and GB-SFT$_{\text{span}}$ achieving the highest correction for Llama 3.1-8B. Similarly, the control version of our GATED baseline variant, GATED-INV performs noticeably lower correction on Qwen but higher on Llama, indicating another model-specific effect of the gate formulated in App. C.

A.4 OPENFDA: A LARGER CROSS-DOMAIN BENCHMARK

Dose Extraction. Raw data is obtained from the openFDA drug label endpoint (Kass-Hout et al., 2016), accessed September 2026. Doses are extracted from label prose as (drug, attribute, value)

triples by pattern matching over a cue window, which we verified and refined with manual quality control checks. We validate extraction *fidelity*, whether the triple is supported by the excerpt, not clinical correctness. Fidelity is a reading comprehension question answerable from the text and therefore a defensible target for annotation; no pharmacological judgment is required.

Model-assisted Verification. Pattern matching fixes a candidate's lexical form but not its meaning, and the errors that survive it are semantic: a maximum taken from a fragment beside the sentence that states a different one, a maintenance dose filed as a maximum, an adult dose filed as pediatric. Each candidate is therefore put to a frontier model as *four* separate questions rather than one: whether the span is a dose of *this* drug rather than a rate, an increment, a reduction level or another drug's dose; whether it is the *complete* stated dose rather than a component of it; which attribute it states; and which population it applies to. A candidate is admitted only if all four agree with the extraction, with an unspecified population accepted. The decision is thus a conjunction of categorical judgments, not a cut on a continuous score.

Measuring the Admission Rule. A human annotator labeled two sets of items. The first, 100 items stratified by attribute and population, put strict correctness at 81% and value-level correctness at 88% and showed that the residual splits into kinds a single yes/no would blur. This motivated the four-question decomposition above rather than a single correctness check. The second 40 items covered the rarer attributes the first barely reached (per-kg, per-m^2, per-cycle), tested those questions separately: $38/40$ were confirmed doses, $40/40$ complete.

Held-out portions of both sets then measure the admission rule itself, giving precision 0.971 on the first (n=49, 0 of 12 wrong values admitted) and 1.000 on the second (n=20, 0 of 2). Items already read during quality control are excluded, and the model's own judgments are withheld from the annotator. Candidates whose value is right but whose attribute label is wrong are discarded rather than relabeled from the model's reading.

Corpus design. The design mirrors NC-REAL so the evaluator needs no changes: an unsupervised header prefix and a supervised body carrying an altered dose value to be taught to the model as a test of knowledge injection not present in pretraining data. The split is by drug to avoid leakage from an item-level split, because values repeat heavily (449 distinct values across $1{,}848$ items).

Note that on both base models the true dose is already preferred to the taught alternative by a wide margin (mean -2.04 nats for Qwen3.5-9B-Base, -3.35 for Llama 3.1-8B, taught minus true). This means that the taught dose must displace an existing preference rather than fill a vacuum.

A.5 TULU

Corpus. We sample from Tulu-3-SFT-Personas-Code (Lambert et al., 2024) and keep the assistant response as the supervised target, yielding 849 training and 149 held-out documents ($118{,}242$ and $19{,}949$ supervised tokens). Documents are truncated at 640 tokens.

We label each *character* of a response and map labels onto model tokens by source offset. Within a code region we perform tokenization and assign KEYWORD or IDENTIFIER to `NAME` tokens according to `keyword.iskeyword`, LITERAL to `NUMBER` and `STRING`, COMMENT to `COMMENT`, OPERATOR to `OP`, and STRUCTURAL to everything else (including a region that fails to parse). Text outside any code region is PROSE. The resulting evaluation mass is STRUCTURAL 0.373, PROSE 0.279, IDENTIFIER 0.118, OPERATOR 0.109, LITERAL 0.078, COMMENT 0.035, KEYWORD 0.008. Class statistics are shown in Fig. 7, and contiguous tokens of the same class form spans.

In Fig. 6, we report held-out token negative log-likelihood (NLL), macro-averaged in keeping with our focus of giving classes equal emphasis (reporting a mass-weighted objective would score the correction using the objective it is meant to correct). Our group-balanced approach improves macro-NLL by 19.2% over SFT on Qwen3.5-9B-Base but only 3.0% on Llama 3.1-8B, which is a weaker base model for this task. More informative, however, is a study of which specific classes improve. The base models are an order of magnitude worse on the rare `keyword` class (1% of tokens), which is the class on which both models, GB-SFT achieves by far the most improvement. Fig. 7 shows it is the class on which both base models have the most headroom. Overall, GB-SFT improves a majority of classes for both models, and every class except `prose` for Qwen.

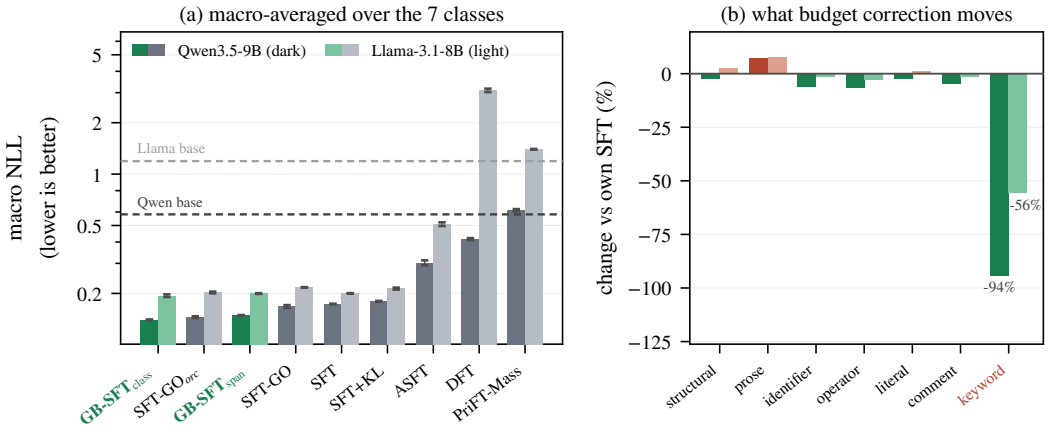


Figure 6: **TULU coding benchmark, both models.** Even outside the fact editing context, class-budgeting improves performance on rare classes that the base model struggles with, that aggregate metrics somewhat obscure.

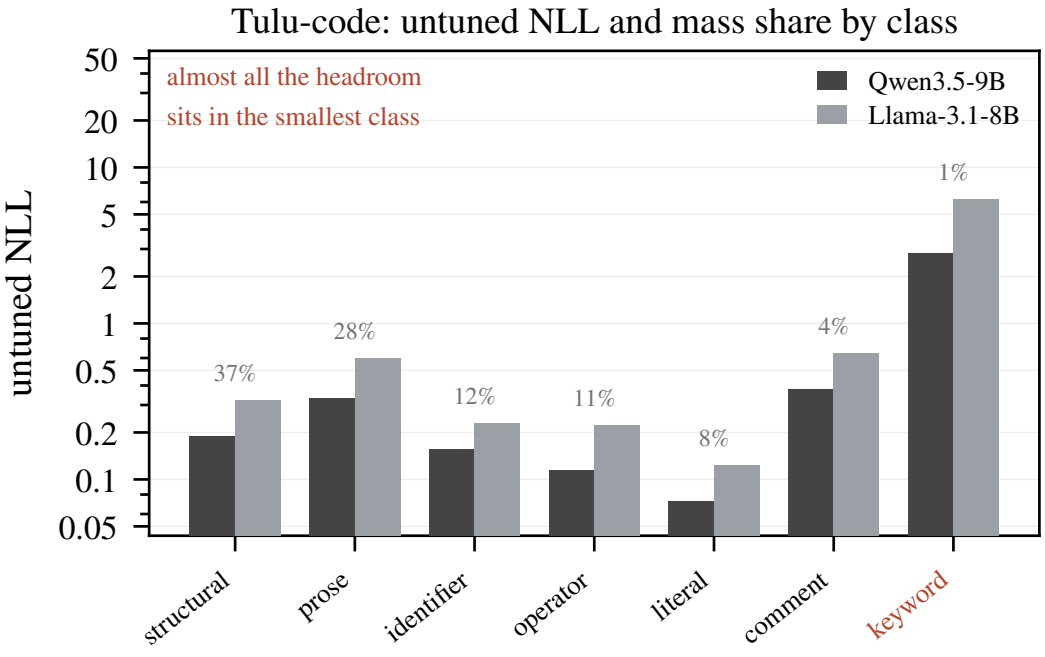


Figure 7: Class-level breakdown of the TULU benchmark. We see that most of the loss headroom is concentrated in the rare `keyword` class.

In Fig. 7, we report a class-level breakdown of token mass and loss for this benchmark. Notably, `keyword` is the *smallest* class at under 1% of tokens and by far the worst fit, at 2.83 nats on Qwen3.5-9B-Base and 6.21 on Llama 3.1-8B against 0.08–0.64 for every other class. `structural` and `prose` together hold 65% of the tokens and are already fit to 0.19–0.62. This is what makes TULU's macro/mass distinction consequential rather than bookkeeping, and it is why we report macro-averaged NLL in Figure 6: a mass-weighted score is dominated by classes that are already solved and would barely register a change in the one class where the headroom actually is. Indeed, we found that by mass-weighted NLL, GB-SFT$_{\text{class}}$ and SFT are essentially flat, which misses the 19.2% improvement in macro-averaged NLL we see that GB-SFT$_{\text{class}}$ affords the base Qwen model compared to SFT.

B Theoretical Analysis

B.1 Derivation of Loss Derivative

Write the class budget and the within-class span weights induced by (4) as

$$p_c = \frac{m_c^\alpha}{M}, \qquad q_c(s) = \frac{u_s}{m_c}, \qquad A_c = \mathbb{E}_{q_c}\big[\overline{\mathrm{CE}}_s\big],$$

so that Eq. 5 becomes $\mathcal{L} = \sum_c p_c A_c$. Here A_c is the class's β-weighted mean cross-entropy: at $\beta = 1$ it is the token mean over the class, at $\beta = 0$ the mean of its span means. These are the same p_c, q_c and A_c introduced in Eq. 6; we repeat them here so the derivation reads on its own.

Both partial derivatives follow from a single observation: the two weightings of (4) are one-parameter exponential families in α and β,

$$p_c = \mathrm{softmax}_c\big(\alpha \log m_c\big), \qquad q_c(s) = \mathrm{softmax}_{s\subset c}\big(\beta \log n_s\big)$$

with natural parameters α, β and sufficient statistics $\log m_c$ and $\log n_s$. For any such family, write η for the natural parameter and differentiate $p_\eta(x) \propto \nu(x)\, e^{\eta T(x)}$ to get $\partial_\eta p_\eta(x) = p_\eta(x)\big(T(x) - \mathbb{E}_{p_\eta}[T]\big)$, hence for any observable f not depending on η,

$$\frac{\partial}{\partial \eta}\, \mathbb{E}_{p_\eta}[f] = \mathbb{E}_{p_\eta}\big[(T - \mathbb{E}_{p_\eta}[T])\, f\big] = \mathrm{Cov}_{p_\eta}(T, f), \tag{9}$$

where centering one factor suffices. Every covariance below is an instance of (9).

The α derivative. Of the quantities in $\mathcal{L} = \sum_c p_c A_c$, only p_c depends on α: u_s, m_c, q_c and A_c are all built from β alone. Applying (9) with $T = \log m_c$ and $f = A_c$ gives the α half of (7) immediately. Written out, from $\log p_c = \alpha \log m_c - \log M$,

$$\frac{\partial \log M}{\partial \alpha} = \frac{\sum_c m_c^\alpha \log m_c}{M} = \mathbb{E}_p[\log m], \qquad \frac{\partial p_c}{\partial \alpha} = p_c\big(\log m_c - \mathbb{E}_p[\log m]\big),$$

so $\partial_\alpha \mathcal{L} = \sum_c p_c\big(\log m_c - \mathbb{E}_p[\log m]\big) A_c = \mathrm{Cov}_p(\log m, A)$.

The β derivative. Here β reaches both levels, because $u_s = n_s^\beta$ propagates into m_c and hence into p_c as well as into q_c. From $\partial u_s / \partial \beta = u_s \log n_s$,

$$\frac{\partial m_c}{\partial \beta} = \sum_{s\subset c} u_s \log n_s = m_c\, \mathbb{E}_{q_c}[\log n], \qquad \text{so} \qquad \frac{\partial \log m_c}{\partial \beta} = \mathbb{E}_{q_c}[\log n] \equiv G_c. \tag{10}$$

The inner factor is immediate from (9) applied to q_c with $T = \log n_s$ and $f = \overline{\mathrm{CE}}_s$:

$$\frac{\partial A_c}{\partial \beta} = \mathrm{Cov}_{q_c}\big(\log n, \overline{\mathrm{CE}}\big).$$

For the outer factor, $\log p_c = \alpha \log m_c - \log M$ and, using (10),

$$\frac{\partial M}{\partial \beta} = \sum_c \alpha\, m_c^{\alpha-1} \frac{\partial m_c}{\partial \beta} = \alpha \sum_c m_c^\alpha G_c, \qquad \frac{\partial \log M}{\partial \beta} = \alpha\, \mathbb{E}_p[G],$$

giving $\partial p_c / \partial \beta = \alpha\, p_c\big(G_c - \mathbb{E}_p[G]\big)$. The product rule on $\mathcal{L} = \sum_c p_c A_c$ then yields the β half of (7):

$$\frac{\partial \mathcal{L}}{\partial \beta} = \sum_c \frac{\partial p_c}{\partial \beta} A_c + \sum_c p_c \frac{\partial A_c}{\partial \beta} = \alpha\, \mathrm{Cov}_p(G, A) + \mathbb{E}_p\big[\mathrm{Cov}_{q_c}(\log n, \overline{\mathrm{CE}})\big].$$

Note that both partials are taken at fixed $\boldsymbol{\theta}$: they describe the objective's dependence on (α, β) at a checkpoint, not the trajectory of the trained loss as the objective is swept.

Effects of α and β. The factor α enters at exactly one point, differentiating m_c^α, so it multiplies the between-class term alone and that term vanishes identically at $\alpha = 0$. Meanwhile, β occurs in u_s and so moves both levels at once. This may make β seem like the more expressive parameter; however, note that the between-class term in the β half of (7) acts by moving p_c, the same knob that α controls. We thus choose to tune this knob by sweeping α in App. D.1, where we report it through the class budget it induces rather than directly, because the budget is the quantity outcomes track. Our analysis there shows that the β axis adds nothing beyond its effect on the budget. Instead, we treat β as discrete and use it to make a structural decision about which unit carries a budget: $\beta = 1$ (GB-SFT$_{\text{class}}$) pools tokens within a class, while $\beta = 0$ (GB-SFT$_{\text{span}}$) pools spans.

B.2 Proofs of Lemmas

Setup. Fix a class c and parameters $\boldsymbol{\theta}$. Let $\mathcal{S}_c$ denote the spans of class c, with $P_c = |\mathcal{S}_c|$, span lengths n_s (in tokens), $n_c = \sum_{s\in\mathcal{S}_c} n_s$, and mean length $\bar{n}_c = n_c/P_c$. Write $\mathrm{CE}_t(\boldsymbol{\theta})$ for the token-level cross-entropy and $\overline{\mathrm{CE}}_s(\boldsymbol{\theta}) = \frac{1}{n_s}\sum_{t\in s}\mathrm{CE}_t(\boldsymbol{\theta})$ for the mean loss within a span. Let $\mu_c = \frac{1}{P_c}\sum_s \overline{\mathrm{CE}}_s(\boldsymbol{\theta})$ be the average span-mean loss across class c (equivalently, A_c at $\beta = 0$) and $\sigma_{\overline{\mathrm{CE}}}(\boldsymbol{\theta}) = \big(\frac{1}{P_c}\sum_s(\overline{\mathrm{CE}}_s(\boldsymbol{\theta})-\mu_c)^2\big)^{1/2}$ the standard deviation of span-mean losses. Both allocations spend the same class budget p_c, i.e. the share p_c of Eq. 6 that class c receives:

$$\mathcal{L}_{\mathrm{pool}}(\boldsymbol{\theta}) = \frac{p_c}{n_c}\sum_{s\in\mathcal{S}_c}\sum_{t\in s}\mathrm{CE}_t(\boldsymbol{\theta}) = \frac{p_c}{n_c}\sum_{s\in\mathcal{S}_c} n_s\,\overline{\mathrm{CE}}_s(\boldsymbol{\theta}), \quad \mathcal{L}_{\mathrm{span}}(\boldsymbol{\theta}) = \frac{p_c}{P_c}\sum_{s\in\mathcal{S}_c}\overline{\mathrm{CE}}_s(\boldsymbol{\theta}). \tag{11}$$

Summed over classes, $\mathcal{L}_{\mathrm{pool}}$ recovers GB-SFT$_{\text{class}}$ (Eq. 2) at $p_c = 1/K$, and ordinary SFT (Eq. 1) at $p_c = n_c/N$. Likewise $\mathcal{L}_{\mathrm{span}}$ recovers GB-SFT$_{\text{span}}$ (Eq. 3) at $p_c = P_c/P$. Within a single class, GB-SFT$_{\text{class}}$ and SFT pool token losses identically; the comparison below is therefore between pooling units at a fixed budget, not between objectives.

Proof of Lemma 1. Substituting $n_c = P_c\bar{n}_c$ into (11) and factoring,

$$\mathcal{L}_{\mathrm{pool}} - \mathcal{L}_{\mathrm{span}} = \frac{p_c}{P_c\bar{n}_c}\sum_{s\in\mathcal{S}_c} n_s\,\overline{\mathrm{CE}}_s(\boldsymbol{\theta}) - \frac{p_c}{P_c}\sum_{s\in\mathcal{S}_c}\overline{\mathrm{CE}}_s(\boldsymbol{\theta}) = \frac{p_c}{P_c\bar{n}_c}\sum_{s\in\mathcal{S}_c}\big(n_s-\bar{n}_c\big)\,\overline{\mathrm{CE}}_s(\boldsymbol{\theta}).$$

Since $\sum_s(n_s-\bar{n}_c) = 0$, subtracting the constant μ_c from each $\overline{\mathrm{CE}}_s(\boldsymbol{\theta})$ leaves the sum unchanged:

$$\sum_{s\in\mathcal{S}_c}\big(n_s-\bar{n}_c\big)\,\overline{\mathrm{CE}}_s(\boldsymbol{\theta}) = \sum_{s\in\mathcal{S}_c}\big(n_s-\bar{n}_c\big)\big(\overline{\mathrm{CE}}_s(\boldsymbol{\theta})-\mu_c\big) = P_c\,\widehat{\mathrm{Cov}}_{s\in\mathcal{S}_c}\big(n_s,\overline{\mathrm{CE}}_s(\boldsymbol{\theta})\big).$$

Combining the two displays gives $\mathcal{L}_{\mathrm{pool}} - \mathcal{L}_{\mathrm{span}} = p_c\cdot\frac{1}{\bar{n}_c}\widehat{\mathrm{Cov}}_{s\in\mathcal{S}_c}(n_s,\overline{\mathrm{CE}}_s(\boldsymbol{\theta}))$, which is p_c times (8). □

By the Cauchy-Schwarz inequality, we get a bound on the difference between pooled and span-balanced losses for a fixed class budget:

Corollary 1 (CV bound)**.** *Let $\mathrm{CV}(n) = \sigma_n/\bar{n}_c$ denote the coefficient of variation of span lengths within class c. Then, for every $\boldsymbol{\theta}$,*

$$\big|\mathcal{L}_{\mathrm{pool}}(\boldsymbol{\theta}) - \mathcal{L}_{\mathrm{span}}(\boldsymbol{\theta})\big| \;\le\; p_c\cdot\mathrm{CV}(n)\cdot\sigma_{\overline{\mathrm{CE}}}(\boldsymbol{\theta}). \tag{12}$$

Proof. Apply Cauchy-Schwarz, $|\widehat{\mathrm{Cov}}(n_s,\overline{\mathrm{CE}})| \le \sigma_n\,\sigma_{\overline{\mathrm{CE}}}$, to (8) and use $\sigma_n = \bar{n}_c\cdot\mathrm{CV}(n)$. □

By similar arguments, we can demonstrate a gradient version of this result, bounding the difference between updates and not just the loss values.

Lemma 2 (Gradient version)**.** *Let $g_s(\boldsymbol{\theta}) = \frac{1}{n_s}\sum_{t\in s}\nabla_{\boldsymbol{\theta}}\mathrm{CE}_t(\boldsymbol{\theta})$ be the span-mean gradient and $\bar{g} = \frac{1}{P_c}\sum_{s\in\mathcal{S}_c} g_s$. Then*

$$\begin{aligned}\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathrm{pool}} - \nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathrm{span}} &= \frac{p_c}{\bar{n}_c}\cdot\frac{1}{P_c}\sum_{s\in\mathcal{S}_c}\big(n_s-\bar{n}_c\big)\big(g_s-\bar{g}\big),\\ \big\|\nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathrm{pool}} - \nabla_{\boldsymbol{\theta}}\mathcal{L}_{\mathrm{span}}\big\| &\le p_c\cdot\mathrm{CV}(n)\cdot\Big(\tfrac{1}{P_c}\textstyle\sum_{s\in\mathcal{S}_c}\|g_s-\bar{g}\|^2\Big)^{1/2}.\end{aligned} \tag{13}$$

Proof. Identical to Lemma 1 with $\overline{\mathrm{CE}}_s(\boldsymbol{\theta})$ replaced by the vector g_s (the mean-centering step is unchanged), followed by the vector Cauchy-Schwarz inequality. □

C Anchored Variants

In this section, we consider an additional formulation for allowing us to choose not only what to supervise, but also what to protect by using an anchor term that enforces consistency with the reference distribution, a la ASFT (Zhu et al., 2026).

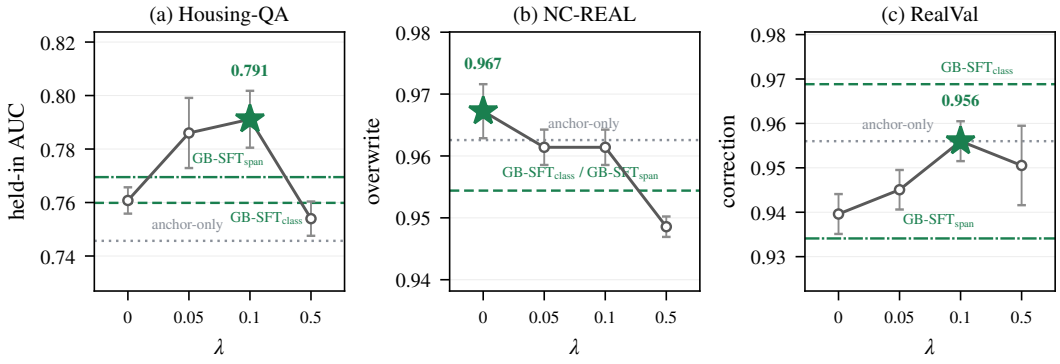


Figure 8: Results of varying the strength of a KL anchor on top of masked span-level balancing for three corpora. Each panel carries its corpus's own GB-SFT$_{\text{class}}$, GB-SFT$_{\text{span}}$ and ANCHOR-ONLY arms as reference rules. Equivalently (Eq. 22), λ sets the share $\lambda/(1+\lambda)$ of the loss budget spent on the scaffold, so the sweep $\lambda \in \{0, .05, .1, .5\}$ runs over scaffold budget shares of 0%, 4.8%, 9.1% and 33.3%.

Notation. The two roles below form a class grouper (Sec. 3.1) with $K = 2$, so we reuse the span notation already in place: s ranges over the spans $\mathcal{S}$ of a training example, n_s is its token count, and $\mathcal{S}_c$ denotes the spans of class c. Let $\pi_{\boldsymbol{\theta}}(\cdot \mid x, y^*_{<t})$ be the policy and $\pi_{\text{ref}}(\cdot \mid x, y^*_{<t})$ the frozen pretrained reference over the vocabulary $\mathcal{V}$.

Per-token terms. At each supervised position t we compute a KL loss to constrain the divergence between the policy and reference token probability distributions:

$$\text{KL}_t = \sum_{v \in \mathcal{V}} \pi_{\boldsymbol{\theta}}(v \mid x, y^*_{<t}) \log \frac{\pi_{\boldsymbol{\theta}}(v \mid x, y^*_{<t})}{\pi_{\text{ref}}(v \mid x, y^*_{<t})} \tag{14}$$

These are pooled within a span exactly as the cross-entropy is, giving the span-mean anchor that plays the role of $\overline{\text{CE}}_s$ in Eq. 3:

$$\overline{\text{KL}}_s = \frac{1}{n_s} \sum_{t \in s} \text{KL}_t. \tag{15}$$

Roles. We partition the supervised spans into *payload* spans $\mathcal{S}_{\text{pay}}$, which are supervised, and *scaffold* spans $\mathcal{S}_{\text{scaf}}$, which are anchored, writing P_{pay} and P_{scaf} for their counts. The assignment is exact by construction each example is built by inserting a known string into a carrier, so $\mathcal{S}_{\text{pay}}$ is the edit target and $\mathcal{S}_{\text{scaf}}$ is its complement. No detector, threshold, or confidence value is involved.

Objective.

$$\mathcal{L}(\boldsymbol{\theta}) = \underbrace{\frac{1}{P_{\text{pay}}} \sum_{s \in \mathcal{S}_{\text{pay}}} \overline{\text{CE}}_s(\boldsymbol{\theta})}_{\text{span-balanced supervision of payload}} + \lambda \underbrace{\frac{1}{P_{\text{scaf}}} \sum_{s \in \mathcal{S}_{\text{scaf}}} \overline{\text{KL}}_s}_{\text{protection of scaffold}}. \tag{16}$$

Averaging each term over its own span set keeps the meaning of λ stable as the size of $\mathcal{C}$ changes across corpora. Note that λ=0 is span-mean-budgeted *masked* SFT over the edit targets alone, while $\lambda > 0$ protects the context text that we did not author.

Eq. 22 is an extension of the loss family of Sec. 3.2 built with the same budgeting construction. Each role pools its spans rather than its tokens, which is the $\beta = 0$ unit of Eq. 4, and the roles $\{\text{pay}, \text{scaf}\}$ are a class grouper with $K = 2$, so the split between them is a role budget in the sense of Sec. 3.2. However, Eq. 5 weights a single cross-entropy, with $\sum_t \omega_t = 1$ by construction, whereas here the scaffold term is a KL anchor rather than a cross-entropy. The two terms therefore measure different quantities, the role weights 1 and λ are set directly rather than induced by an exponent on class mass, and they sum to $1 + \lambda$ rather than to one. λ can thus be read as a relative role weight, with the scaffold carrying $\lambda/(1 + \lambda)$ of the combined weight.

That is the same quantity α controls for two CE classes, which is why Fig. 10 reports the sweep on this scale as well as in λ. We see that the efficacy of composing budget correction with anchoring varies by corpus. HOUSING-QA rewards the composition: λ=.1 reaches .7911 $\pm$ .0106, above GB-SFT$_{\text{span}}$ (.7695) and well above λ=0 (.7608), and the optimum is a plateau rather than a spike (λ=.05 and λ=.1 both clear every reference). NC-REAL is best at λ=0 and declines monotonically: the anchor does not help there. REALVAL peaks at λ=.1 but its whole curve sits below plain GB-SFT$_{\text{class}}$ (.9688), so the composition loses to grouping alone. We therefore report λ as a corpus-dependent hyperparameter and make no unified claim from it.

Tradeoffs of Anchoring Group-balanced objectives needs only a grouping; anchoring needs a protect set (a designated subset of tokens that must not move). This is a stronger requirement, since a corpus can be partitioned into roles without any of them being the one to hold; TULU (App. A.5) is such a corpus, as its examples carry role classes but no payload/scaffold distinction, so ANCHOR-ONLY cannot be constructed there at all, while GB-SFT runs unchanged.

ANCHOR-ONLY's need for a protect set to preserve is also felt at labeling time, since a grouping can be learned without annotation (e.g. an LLM tagger reaches 98.8% of the gold grouping's gain) whereas no method proposed in this work learns a protect set. Where a protect set does exist, ANCHOR-ONLY buys preservation by declining to acquire, evidenced on HOUSING-QA (Tab. 1) by improving held-out citation recall but being completely unable to improve held-in citation recall (as a result of giving the citations no cross-entropy). It therefore sits further along the preservation end of the same acquisition/preservation axis GB-SFT trades on, rather than dominating it, which is why it leads on overwrite for OPENFDA and on NC-REAL under Qwen3.5-9B-Base, and trails on every measure that rewards learning the scaffold.

To apply anchoring without a protect set, we consider our SFT + KL baseline in our main experiments that adds the anchor term to all tokens atop SFT's token mean loss:

$$\mathcal{L}_{\text{SFT + KL}}(\boldsymbol{\theta}) = \frac{1}{N}\sum_t \Big[\mathrm{CE}_t(\boldsymbol{\theta}) + \lambda\,\mathrm{KL}_t\Big]. \tag{17}$$

Gated formulation. GATED replaces the hard payload/scaffold split with a soft, per-token gate $\rho_t \in [0,1]$, high on scaffold tokens the reference already predicts and near zero on the payload. We take $\rho_t = \pi_{\text{ref}}(y_t^*)^\gamma$ with $\gamma = 1$ throughout; any frozen-reference confidence statistic can be substituted, and we also tried the reference's cumulative mass at t. Because π_{ref} is frozen, ρ_t is a constant with respect to $\boldsymbol{\theta}$ and is read from an offline cache. Writing $\bar{\rho}$ for its empirical mean over all supervised tokens, the gate sets both an anchor strength and a cross-entropy weight per token:

$$\lambda_t = \lambda\,\frac{\rho_t}{\bar{\rho}}, \qquad \omega_t^{\text{gated}} = \rho_t\ \mathrm{sg}(p_t) + (1-\rho_t), \qquad p_t = \pi_{\boldsymbol{\theta}}(y_t^* \mid x, y_{<t}^*), \tag{18}$$

where $\mathrm{sg}(\cdot)$ is the stop-gradient operator. Normalizing by $\bar{\rho}$ makes λ_t average to λ, so λ keeps the same meaning it has in Eq. 22. The scaffold thus keeps both the anchor and DFT's self-weighting, while the payload, where $\rho_t \to 0$, is released to plain cross-entropy at weight exactly 1.

With the gate of Eq. 24 supplying both per-token weights, the loss of GATED is:

$$\mathcal{L}_{\text{GATED}}(\boldsymbol{\theta}) = \frac{1}{N}\sum_t \Big[\omega_t^{\text{gated}}\,\mathrm{CE}_t(\boldsymbol{\theta}) + \lambda_t\,\mathrm{KL}_t\Big]. \tag{19}$$

Note that because $\omega_t^{\text{gated}} \in [0,1]$ without summing to one across the example, Eq. 25 is a reweighting of the SFT loss rather than a member of the normalized family of Eq. 5.

GATED-INV is a mechanism control that tests whether the gains are merely extra regularization. It flips the polarity, anchoring hardest where the reference is surprised and releasing where it is confident.

Table 9 locates each anchored arm used in our main experiments in this notation. The *supervised* and *anchored* columns give the support of each term (which spans receive cross-entropy and which receive the anchor) while the *reduction* column states how the per-token terms are aggregated once that support is fixed. Only the upper block uses the span reduction of Eq. 22; the lower block reduces by a plain token mean akin to standard SFT.

Arm	Supervised	Anchored	λ	Reduction
span-pooled ($\beta = 0$): instances of Eq. 22				
MASKED-SFT	$\mathcal{S}_{\text{pay}}$	—	0	Eq. 22
ANCHOR-ONLY	$\mathcal{S}_{\text{pay}}$	$\mathcal{S}_{\text{scaf}}$	> 0	Eq. 22
token-mean ($\beta = 1$): not instances of Eq. 22				
SFT + KL	$\mathcal{S}$	$\mathcal{S}$	> 0	Eq. 23
GATED	$\mathcal{S}$	$\mathcal{S}$	> 0	Eq. 25, gate ρ_t
GATED-INV	$\mathcal{S}$	$\mathcal{S}$	> 0	Eq. 25, gate $1 - \rho_t$

Table 8: The anchored arms. The support columns and the reduction column are independent: sharing a support with Eq. 22 does not make an arm an instance of it.

D SENSITIVITY ANALYSIS

Below we report additional studies of important parameters.

D.1 VARYING THE CLASS BUDGET

Group balancing works by redistributing the objective's emphasis from popular classes containing less of the knowledge payload, to classes with fewer but payload-dense tokens. The class budget can be mathematically controlled most directly by α, and to a lesser extent by β. In Fig. 9 we vary the citation budget on our HOUSING-QA benchmark for Qwen3.5-9B-Base, by primarily tuning α, and also show the effects of tuning β, to show the effects on held-in and held-out AUC.

Our dose-response curve for held-in AUC shows that it falls monotonically as the α sweep causes citations to take budget: Pooling all nine cells, budget share and AUC correlate at $r = -0.945$, and a straight line through them leaves a residual of 0.0114, *below* the ≈ 0.012 seed-to-seed spread: budget share, most directly controlled by α, accounts for the outcome to within noise.

We also see evidence that the second parameter β does nothing beyond moving that budget. At matched budget, $(\alpha, \beta) = (0.3, 1.0)$ at 49.5% scores 0.7511 and $(0.7, 0.5)$ at 50.1% scores 0.7408, a difference of 0.6 standard deviations despite entirely different pooling units. And with $\alpha = 0$, where the class split is pinned at 33.3% whatever β does, the three values are 0.7509, 0.7663, 0.7552 for $\beta = 0, 0.5, 1$: flat within error bars of 0.008 to 0.019. This analysis justifies our choice to recommend treating β as binary $\{0, 1\}$, and also why in our experiments we notice comparable performance overall between GB-SFT$_{\text{class}}$ and GB-SFT$_{\text{span}}$.

The specificity control is that held-*out* AUC stays flat at 0.576-0.62 across the entire budget range, against an untuned base of 0.541. While training produces a slight held-out gain, changing the class budget mainly proves injection rather than a general shift in discrimination ability.

In this section, we consider an additional formulation for allowing us to choose not only what to supervise, but also what to protect by using an anchor term that enforces consistency with the reference distribution, a la ASFT (Zhu et al., 2026).

Notation. The two roles below form a class grouper (Sec. 3.1) with $K = 2$, so we reuse the span notation already in place: s ranges over the spans $\mathcal{S}$ of a training example, n_s is its token count, and $\mathcal{S}_c$ denotes the spans of class c. Let $\pi_{\boldsymbol{\theta}}(\cdot \mid x, y^*_{<t})$ be the policy and $\pi_{\text{ref}}(\cdot \mid x, y^*_{<t})$ the frozen pretrained reference over the vocabulary $\mathcal{V}$.

Per-token terms. At each supervised position t we compute a KL loss to constrain the divergence between the policy and reference token probability distributions:

$$\text{KL}_t = \sum_{v \in \mathcal{V}} \pi_{\boldsymbol{\theta}}(v \mid x, y^*_{<t}) \log \frac{\pi_{\boldsymbol{\theta}}(v \mid x, y^*_{<t})}{\pi_{\text{ref}}(v \mid x, y^*_{<t})} \tag{20}$$

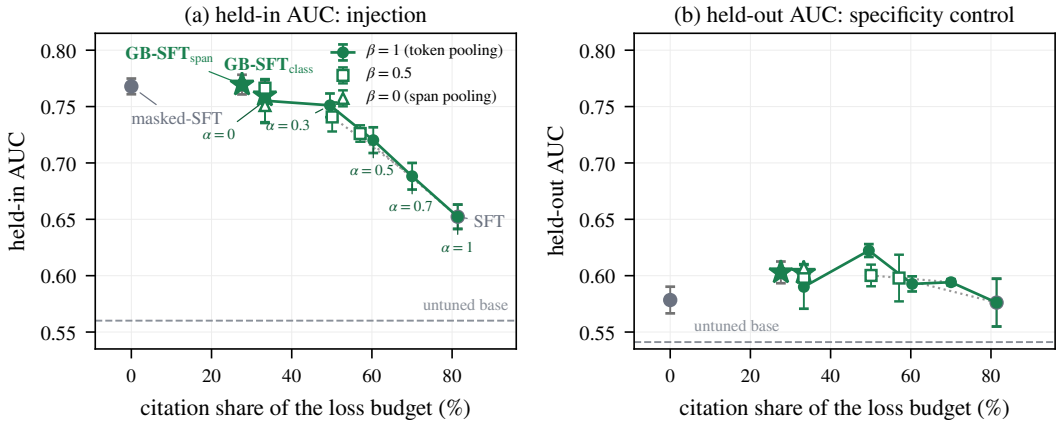


Figure 9: **Citation budget is the active variable**, which can be controlled by sweeping α. The horizontal axis is the share of the loss budget that citation tokens receive: token-proportional SFT gives them 81.4%, while grouping redistributes it. **Filled circles joined by a line** are a *controlled* sweep: the (α, β) family at β=1 with α swept over $\{0, .3, .5, .7, 1\}$, so the budget is the only thing that changes. **Open squares** are the same family at $\beta \in \{0, 0.5\}$. **Separate markers** are independent objectives that merely happen to land at some budget: MASKED-SFT at 0%, GB-SFT$_{\text{span}}$ at 27.6%, GB-SFT$_{\text{class}}$ at 33.3%.

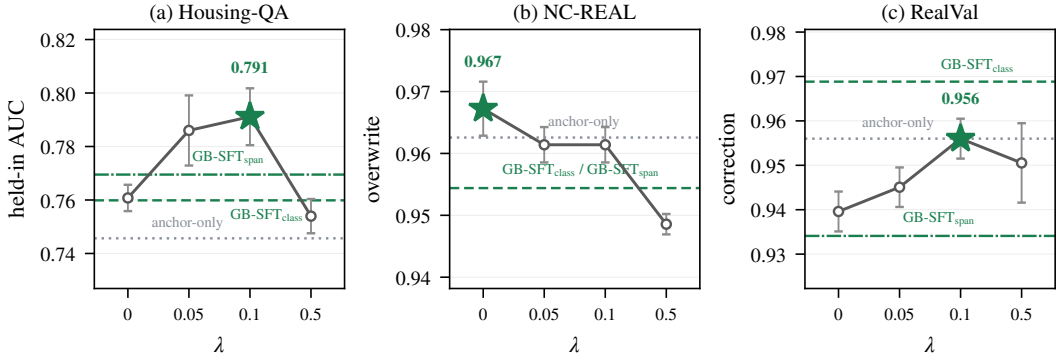


Figure 10: Results of varying the strength of a KL anchor on top of masked span-level balancing for three corpora. Each panel carries its corpus's own GB-SFT$_{\text{class}}$, GB-SFT$_{\text{span}}$ and ANCHOR-ONLY arms as reference rules. Equivalently (Eq. 22), λ sets the share $\lambda/(1+\lambda)$ of the loss budget spent on the scaffold, so the sweep $\lambda \in \{0, .05, .1, .5\}$ runs over scaffold budget shares of 0%, 4.8%, 9.1% and 33.3%.

These are pooled within a span exactly as the cross-entropy is, giving the span-mean anchor that plays the role of $\overline{\text{CE}}_s$ in Eq. 3:

$$\overline{\text{KL}}_s = \frac{1}{n_s} \sum_{t \in s} \text{KL}_t. \tag{21}$$

Roles. We partition the supervised spans into *payload* spans $\mathcal{S}_{\text{pay}}$, which are supervised, and *scaffold* spans $\mathcal{S}_{\text{scaf}}$, which are anchored, writing P_{pay} and P_{scaf} for their counts. The assignment is exact by construction each example is built by inserting a known string into a carrier, so $\mathcal{S}_{\text{pay}}$ is the edit target and $\mathcal{S}_{\text{scaf}}$ is its complement. No detector, threshold, or confidence value is involved.

Objective.

$$\mathcal{L}(\boldsymbol{\theta}) = \underbrace{\frac{1}{P_{\text{pay}}} \sum_{s \in \mathcal{S}_{\text{pay}}} \overline{\text{CE}}_s(\boldsymbol{\theta})}_{\text{span-balanced supervision of payload}} + \lambda \underbrace{\frac{1}{P_{\text{scaf}}} \sum_{s \in \mathcal{S}_{\text{scaf}}} \overline{\text{KL}}_s}_{\text{protection of scaffold}}. \tag{22}$$

Averaging each term over its own span set keeps the meaning of λ stable as the size of $\mathcal{C}$ changes across corpora. Note that λ=0 is span-mean-budgeted *masked* SFT over the edit targets alone, while $\lambda > 0$ protects the context text that we did not author.

Eq. 22 is an extension of the loss family of Sec. 3.2 built with the same budgeting construction. Each role pools its spans rather than its tokens, which is the $\beta = 0$ unit of Eq. 4, and the roles $\{\mathrm{pay}, \mathrm{scaf}\}$ are a class grouper with $K = 2$, so the split between them is a role budget in the sense of Sec. 3.2. However, Eq. 5 weights a single cross-entropy, with $\sum_t \omega_t = 1$ by construction, whereas here the scaffold term is a KL anchor rather than a cross-entropy. The two terms therefore measure different quantities, the role weights 1 and λ are set directly rather than induced by an exponent on class mass, and they sum to $1 + \lambda$ rather than to one. λ can thus be read as a relative role weight, with the scaffold carrying $\lambda/(1 + \lambda)$ of the combined weight.

That is the same quantity α controls for two CE classes, which is why Fig. 10 reports the sweep on this scale as well as in λ. We see that the efficacy of composing budget correction with anchoring varies by corpus. HOUSING-QA rewards the composition: λ=.1 reaches .7911 $\pm$ .0106, above GB-SFT$_{\text{span}}$ (.7695) and well above λ=0 (.7608), and the optimum is a plateau rather than a spike (λ=.05 and λ=.1 both clear every reference). NC-REAL is best at λ=0 and declines monotonically: the anchor does not help there. REALVAL peaks at λ=.1 but its whole curve sits below plain GB-SFT$_{\text{class}}$ (.9688), so the composition loses to grouping alone. We therefore report λ as a corpus-dependent hyperparameter and make no unified claim from it.

Tradeoffs of Anchoring Group-balanced objectives needs only a grouping; anchoring needs a protect set (a designated subset of tokens that must not move). This is a stronger requirement, since a corpus can be partitioned into roles without any of them being the one to hold; TULU (App. A.5) is such a corpus, as its examples carry role classes but no payload/scaffold distinction, so ANCHOR-ONLY cannot be constructed there at all, while GB-SFT runs unchanged.

ANCHOR-ONLY's need for a protect set to preserve is also felt at labeling time, since a grouping can be learned without annotation (e.g. an LLM tagger reaches 98.8% of the gold grouping's gain) whereas no method proposed in this work learns a protect set. Where a protect set does exist, ANCHOR-ONLY buys preservation by declining to acquire, evidenced on HOUSING-QA (Tab. 1) by improving held-out citation recall but being completely unable to improve held-in citation recall (as a result of giving the citations no cross-entropy). It therefore sits further along the preservation end of the same acquisition/preservation axis GB-SFT trades on, rather than dominating it, which is why it leads on overwrite for OPENFDA and on NC-REAL under Qwen3.5-9B-Base, and trails on every measure that rewards learning the scaffold.

To apply anchoring without a protect set, we consider our SFT + KL baseline in our main experiments that adds the anchor term to all tokens atop SFT's token mean loss:

$$\mathcal{L}_{\text{SFT + KL}}(\boldsymbol{\theta}) = \frac{1}{N} \sum_t \Big[\mathrm{CE}_t(\boldsymbol{\theta}) + \lambda \, \mathrm{KL}_t \Big]. \tag{23}$$

Gated formulation. GATED replaces the hard payload/scaffold split with a soft, per-token gate $\rho_t \in [0, 1]$, high on scaffold tokens the reference already predicts and near zero on the payload. We take $\rho_t = \pi_{\mathrm{ref}}(y_t^*)^\gamma$ with $\gamma = 1$ throughout; any frozen-reference confidence statistic can be substituted, and we also tried the reference's cumulative mass at t. Because π_{ref} is frozen, ρ_t is a constant with respect to $\boldsymbol{\theta}$ and is read from an offline cache. Writing $\bar{\rho}$ for its empirical mean over all supervised tokens, the gate sets both an anchor strength and a cross-entropy weight per token:

$$\lambda_t = \lambda \, \frac{\rho_t}{\bar{\rho}}, \qquad \omega_t^{\text{gated}} = \rho_t \, \mathrm{sg}(p_t) + (1 - \rho_t), \qquad p_t = \pi_{\boldsymbol{\theta}}(y_t^* \mid x, y_{<t}^*), \tag{24}$$

where $\mathrm{sg}(\cdot)$ is the stop-gradient operator. Normalizing by $\bar{\rho}$ makes λ_t average to λ, so λ keeps the same meaning it has in Eq. 22. The scaffold thus keeps both the anchor and DFT's self-weighting, while the payload, where $\rho_t \to 0$, is released to plain cross-entropy at weight exactly 1.

With the gate of Eq. 24 supplying both per-token weights, the loss of GATED is:

$$\mathcal{L}_{\text{GATED}}(\boldsymbol{\theta}) = \frac{1}{N} \sum_t \Big[\omega_t^{\text{gated}} \, \mathrm{CE}_t(\boldsymbol{\theta}) + \lambda_t \, \mathrm{KL}_t \Big]. \tag{25}$$

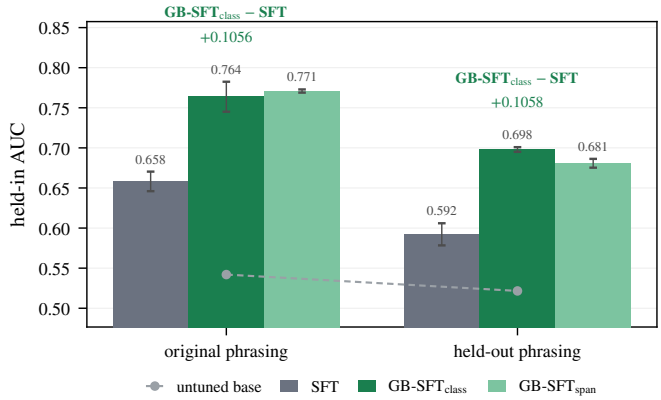


Figure 11: Evaluating group balancing versus ungrouped SFT on original phrasing versus a held-out phrasing not seen in training. The relative gains of group balancing are invariant to rephrasing.

Note that because $\omega_t^{\text{gated}} \in [0, 1]$ without summing to one across the example, Eq. 25 is a reweighting of the SFT loss rather than a member of the normalized family of Eq. 5.

GATED-INV is a mechanism control that tests whether the gains are merely extra regularization. It flips the polarity, anchoring hardest where the reference is surprised and releasing where it is confident.

Table 9 locates each anchored arm used in our main experiments in this notation. The *supervised* and *anchored* columns give the support of each term (which spans receive cross-entropy and which receive the anchor) while the *reduction* column states how the per-token terms are aggregated once that support is fixed. Only the upper block uses the span reduction of Eq. 22; the lower block reduces by a plain token mean akin to standard SFT.

Arm	**Supervised**	**Anchored**	λ	**Reduction**
span-pooled ($\beta = 0$): instances of Eq. 22				
MASKED-SFT	$\mathcal{S}_{\text{pay}}$	—	0	Eq. 22
ANCHOR-ONLY	$\mathcal{S}_{\text{pay}}$	$\mathcal{S}_{\text{scaf}}$	> 0	Eq. 22
token-mean ($\beta = 1$): not instances of Eq. 22				
SFT + KL	$\mathcal{S}$	$\mathcal{S}$	> 0	Eq. 23
GATED	$\mathcal{S}$	$\mathcal{S}$	> 0	Eq. 25, gate ρ_t
GATED-INV	$\mathcal{S}$	$\mathcal{S}$	> 0	Eq. 25, gate $1 - \rho_t$

Table 9: The anchored arms. The support columns and the reduction column are independent: sharing a support with Eq. 22 does not make an arm an instance of it.

D.2 EFFECT OF PHRASING

Here we test on a held-out phrasing, to make sure that successful arms are not just memorizing the probe. We evaluate GB-SFT$_{\text{class}}$ with SFT on HOUSING-QA with Qwen3.5-9B-Base, under both the original phrasing and a held-out phrasing that the model was not trained against. Each checkpoint is scored twice with no retraining between: once on the original probe, once on a probe whose 126 questions were rewritten to ask the same thing in different words. The rewrites preserve polarity and every enumerated clause and share only 0.336 of their content words with the originals on average. These arms are retrained from scratch for this test; their original-phrasing values (0.6582 and 0.7638) reproduce Table 1's 0.6523 and 0.7599 within seed noise, so the pipeline and the comparison are the same ones the main results use. Results are shown in Fig. 11.

Note that rephrasing costs standard SFT and group balancing approximately the same: -0.0660 for SFT and -0.0658 for GB-SFT$_{\text{class}}$, differing by 0.0002. This is likely the probe simply becoming harder under the rephrasing. The important takeaway is that the *relative* improvement of

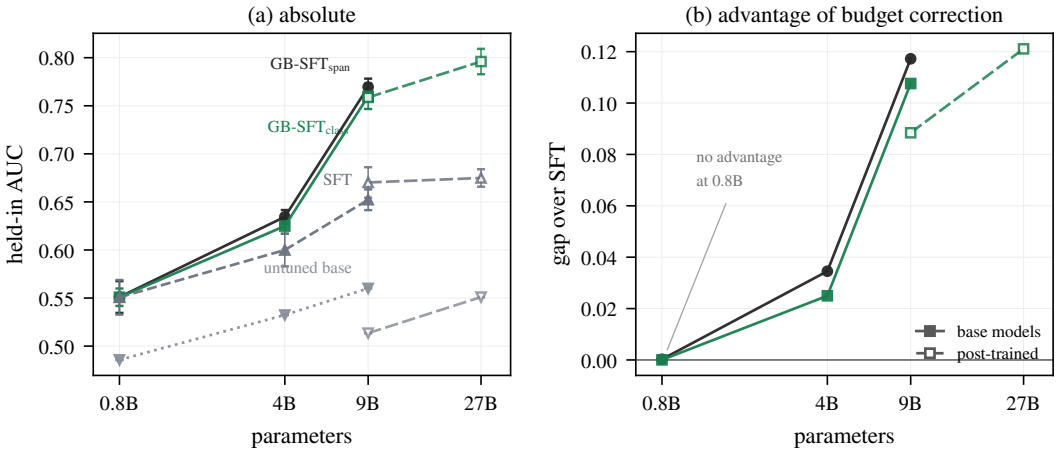


Figure 12: **The advantage of group balancing grows with model capacity.** HOUSING-QA within-group AUC across the Qwen family, three seeds per arm at every size. *Left:* absolute AUC-in. *Right:* the quantity the claim rests on, each group-balanced arm minus SFT at the same size. The advantage is monotone in scale: virtually zero at 0.8B ($+0.000$ for GB-SFT$_{\text{class}}$, $+0.000$ for GB-SFT$_{\text{span}}$), small at 4B ($+0.025$, $+0.035$), and large at 9B ($+0.108$, $+0.117$).

GB-SFT$_{\text{class}}$ over SFT under each phrasing remains the same: GB-SFT$_{\text{class}}$ leads SFT by $+0.1056$ on the original wording and $+0.1058$ on wording it has never seen. Thus, improvements of group balancing are robust to different phrasings and not just due to memorization of the phrasing.

D.3 Effect of Model Capacity

We also apply our methods to LLMs in the Qwen3.5 dense base models, which range in size from 0.8B-9B, to show that our results hold across scales. To study results on a larger scale, we also consider the Qwen3.5-27B model. Only the instruction-tuned variant is available for this model size, so to have another comparable point in the plot, we also apply our method to the instruction-tuned Qwen3.5-9B model. In Fig. 12, we see that not only do larger models do better after fine-tuning (as is expected), but the relative advantage of group balancing compared to ungrouped fine-tuning also grows with model size. Group balancing has limited effect at a small scale (0.8B) but its effects have monotone growth with model size.

D.4 Effect of LoRA Rank

In addition to using a large base model, an additional way to perform supervised fine-tuning with additional capacity when using LoRA is to simply increase the rank of the update. In Fig. 13 we vary the LoRA rank and study its effects on relative performance. Note that neither the performance of standard SFT with LoRA nor the relative advantage of budget correction is monotone as a function of rank (in general, the higher rank capacity helps up to a point, but performance can saturate or even decrease as the model starts to overfit). However, we observe a consistent improvement from budget correction across LoRA ranks.

D.5 Effect of Learning Rate

Every arm in this paper trains at the same nominal learning rate of 2×10^{-5}, but a shared rate is not a shared step size. The objectives differ in loss scale: SFT averages over all tokens with weight $1/N$, while GB-SFT$_{\text{class}}$ averages per-class means, so if the classes differ in mean cross-entropy, its value and its gradient magnitude differ from SFT's. This may mean that the effective learning rates for grouped and ungrouped SFT are different. In Fig. 14, we compare their performance at several different learning rates.

SFT peaks at 5×10^{-5} with 0.6850, away from the shared rate, while both grouped arms peak at the shared rate. Comparing each arm at its own best rate leaves GB-SFT$_{\text{class}}$ ahead by $+0.0904$

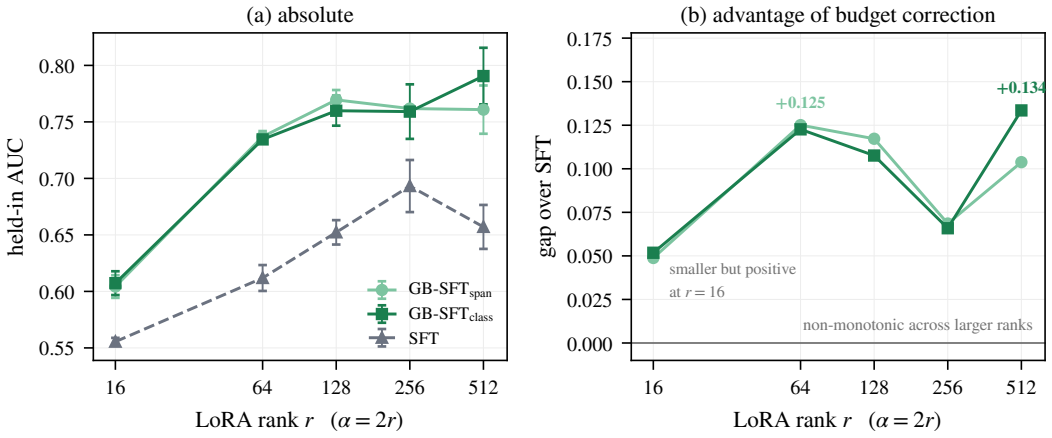


Figure 13: Fine-tuned Qwen3.5-9B-Base performance on HOUSING-QA as a function of LoRA rank.

Method	ρ	lr	AUC-in	AUC-out	cite-in	cite-out
shared learning rate						
SFT	1.000	2.0×10^{-5}	0.6523 $\pm$0.0108	0.5761 $\pm$0.0212	0.988 $\pm$0.003	0.583 $\pm$0.016
GB-SFT$_{class}$	2.597	2.0×10^{-5}	**0.7599** $\pm$0.0132	0.6025 $\pm$0.0025	0.983 $\pm$0.002	0.600 $\pm$0.017
GB-SFT$_{span}$	2.515	2.0×10^{-5}	**0.7695** $\pm$0.0087	0.6029 $\pm$0.0096	0.972 $\pm$0.003	0.624 $\pm$0.006
reduced learning rate ($2\times10^{-5}/\rho$)						
SFT	1.000	2.0×10^{-5}	0.6553 $\pm$0.0074	0.5901 $\pm$0.0082	0.988 $\pm$0.003	0.579 $\pm$0.041
GB-SFT$_{class}$	2.597	7.70×10^{-6}	0.6820 $\pm$0.0038	0.5865 $\pm$0.0188	0.971 $\pm$0.002	**0.672** $\pm$0.009
GB-SFT$_{span}$	2.515	7.95×10^{-6}	0.6901 $\pm$0.0065	0.5972 $\pm$0.0078	0.972 $\pm$0.000	**0.689** $\pm$0.002
advantage over SFT *in the same block*						
GB-SFT$_{class}$, shared			+0.1076	+0.0264	−0.005	+0.017
GB-SFT$_{class}$, reduced			+0.0268	−0.0036	−0.017	**+0.093**
GB-SFT$_{span}$, shared			+0.1172	+0.0268	−0.016	+0.041
GB-SFT$_{span}$, reduced			+0.0348	+0.0071	−0.016	**+0.111**

Table 10: HOUSING-QA, Qwen3.5-9B-Base. ρ is the measured ratio of an arm's gradient norm to SFT's, over the LoRA parameters at *initialization*, where the adapter is still zero and the objective is the only thing that differs. We use this to compute a reduced learning rate.

and GB-SFT$_{span}$ by $+0.1000$, against $+0.1076$ and $+0.1172$ at the shared rate: independent tuning recovers roughly 30% of the advantage. Notably, all arms remain well above SFT at every rate tried, including the 1×10^{-4} column where every arm degrades. Thus, the grouping and not just the choice of learning rate is responsible for the improvements we show in our main results.

In Tab. 10, we report results with a stronger control: a reduced learning rate for our group-balanced methods, computed by dividing each learning rate by the initial gradient norm ratio relative to SFT to equalize first step size. (Note that this is a strong correction, but does not guarantee that the trajectory will be equalized; the standard practice control (Kim et al., 2025) is the per-arm tuning of Fig. 14.) We see that this gradient-norm-matched control makes the two advantages of GB-SFT move in opposite directions. Lowering the rate removes roughly three quarters of the injection advantage ($+0.1076 \rightarrow +0.0268$ for GB-SFT$_{class}$) and leaves the AUC-out advantage indistinguishable from zero (-0.0036 and $+0.0071$, against seed spreads of ±0.019 and ±0.008). Meanwhile, it multiplies the held-out citation advantage about fivefold ($+0.017 \rightarrow +0.093$). With a reduced learning rate that approximately matches the first step size of SFT, GB-SFT achieves a smaller but solid injection gain and a *larger* preservation gain.

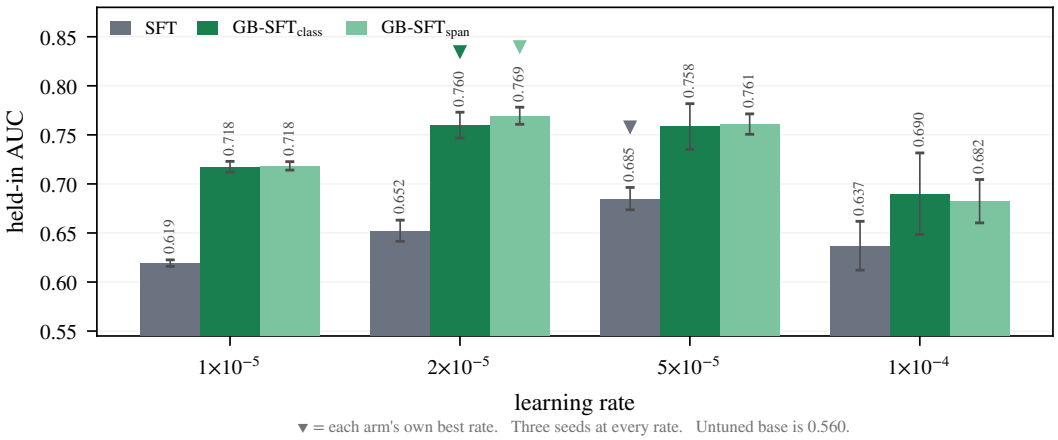


Figure 14: **Held-in AUC across learning rates**, HOUSING-QA, Qwen3.5-9B-Base. ▼ marks each arm's own best rate.

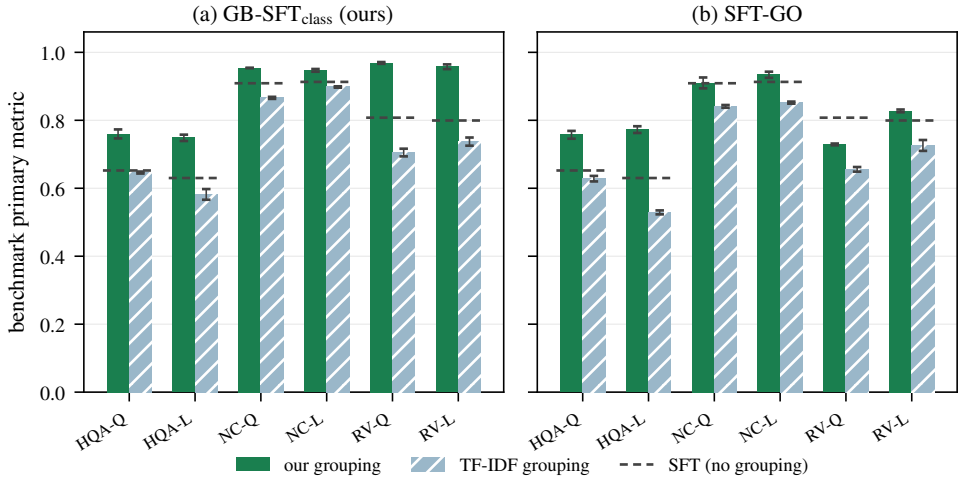


Figure 15: GB-SFT$_{class}$ and SFT-GO with TF-IDF grouping vs our grouping for HOUSING-QA (HQA), NC-REAL (NC), and REALVAL (RV) with Qwen3.5-9B-Base (Q) and Llama 3.1-8B (L) base models.

D.6 EFFECT OF GROUPING STRATEGIES

Both our methods and SFT-GO apply a token-level group-aware loss. Here we explore the effect of the grouping for both our balanced method and SFT-GO. We try our proposed role-based grouping and SFT-GO's TF-IDF grouping method Kim et al. (2025). In Fig. 15 we see that our role-based grouping substantially improves both GB-SFT and SFT-GO. Without our grouping, neither formulation can beat ungrouped supervised fine-tuning. Thus, we see that good grouping quality is instrumental to making a group-aware objective (whether GB-SFT or SFT-GO) improve on standard SFT, and for knowledge-intensive domains our grouping supplies this whereas the TF-IDF heuristic of prior work does not. We also found this to be true for the Qwen base model even on TULU, which does not have a natural payload-scaffold decomposition and where the overall improvement from group balancing is overall smaller than on other corpora.

D.7 LLM GROUPER

To group tokens without human-defined labels, each supervised target is shown to a frontier LLM (we use Claude Opus 5 in our experiments), which is asked to mark the answer span and the citation

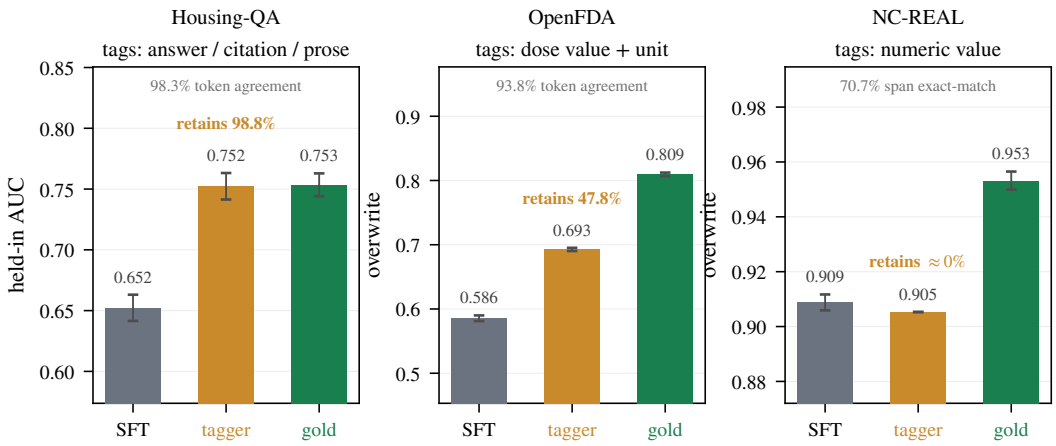


Figure 16: LLM grouper results: successful on HOUSING-QA, where the classes are lexically distinctive, and less successful on NC-REAL, where the LLM can make plausible but wrong mistakes that mislead a model.

span for HOUSING-QA or the value for NC-REAL and OPENFDA; everything it leaves unmarked is prose. Results are shown in Fig. 16.

On HOUSING-QA, because the citations are lexically distinctive near-verbatim strings, the grouper is able to recover them almost perfectly, reaching token-level F1 of .9997 and exact-string agreement on 1166 of 1176 datums (its residual error is over-inclusion, precision .756 and recall 1.000 on answer spans, rather than misdirection). On this dataset, the held-in AUC using LLM classes retains 98.8% of the benefit over SFT that we get from grouping with the role-based classes, indicating that LLMs are able to detect meaningful classes for this benchmark.

On NC-REAL and OPENFDA, the grouper's task is the payload itself, which is a numerical value (statute quantities, drug doses), and it is easy for an LLM to find other plausible but wrong numeric values that mislead the model. OpenFDA's values empirically have enough distinctiveness than an LLM grouper is able to find good enough groups to retain a large share of the improvement over ungrouped SFT. However, on the hardest-to-group dataset NC-REAL, the LLM grouper errors at a rate $p = .293$ and reaches an overwrite of .905 (in line with ungrouped SFT). We tried corrupting the gold class labels with 30% random noise, matching the LLM grouper's error rate, and found that the same amount of random noise hardly affects the overwrite levels a model is able to achieve. This is evidence that the issue is not the labeling error rate but the specific plausible-but-wrong misleading mistakes an LLM can make. Automatic labeling is viable, then, when the classes are lexically distinctive enough that a wrong guess for a group is not also a plausible one.

D.8 GENERATIVE EVALUATION

In Tab. 11, we compare a generative setting to the teacher forcing we use throughout the paper. GB-SFT$_{\text{class}}$ and GB-SFT$_{\text{span}}$ emit the taught value about $4\times$ as often as SFT, with clean separation at three seeds, and *orig.* falls monotonically as *contains* rises (.114 at base to .042). Thus, we see that better methods really do displace the original rather than merely outranking it. However, ranking reports .957 whereas generation reports .196; in absolute terms, the model does $5\times$ better at recognizing the correct answer when supplied than generating it from scratch. While group balancing improves the generative abilities of the fine-tuned model relative to existing methods, further research is encouraged to increase its absolute abilities.

D.9 COMPARISONS AGAINST ADDITIONAL MECHANISMS

We study additional interventions based on mechanisms modifying what data the update sees, how large the update is, or what parameters it affects, to see if other mechanisms besides loss-level weighting can achieve similar knowledge injection performance.

Method	ranking	free generation				
	ovwrite	starts	contains	×SFT	orig.	neither
untuned base	–	.000	.000	–	.114	.886
group balancing (ours)						
GB-SFT$_{\text{class}}$	.9544 ±.0000	.173 ±.000	.190 ±.006	4.1×	.046 ±.012	.766 ±.014
GB-SFT$_{\text{span}}$	.9544 ±.0029	.183 ±.011	.196 ±.008	4.2×	.042 ±.001	.766 ±.008
anchored variants (ours)						
GATED	.9286 ±.0066	.060 ±.009	.062 ±.010	1.3×	.088 ±.006	.850 ±.006
group-based aggregation						
SFT-GO$_{\text{oracle}}$	.9099 ±.0163	.062 ±.013	.067 ±.013	1.4×	.064 ±.007	.869 ±.014
loss shaping						
SFT	.9088 ±.0029	.044 ±.004	.047 ±.004	1.0×	.095 ±.007	.859 ±.005

Table 11: **Ranking versus free generation**, NC-REAL, Qwen3.5-9B-Base. *ovwrite* is the teacher-forced measure used throughout the paper: the fraction of held-in items where the taught value outranks the original. The remaining columns greedily decode a continuation from the *same* prompts and ask what the model actually said: *starts* requires the continuation to begin with the taught value, *contains* to include it within sixty characters, and *orig.* reports the original value by the same rule (the latter two cases overlap a small number of times if a generation contains both values). *neither* is the fraction of continuations containing neither value. Matching is done after mapping number words to digits.

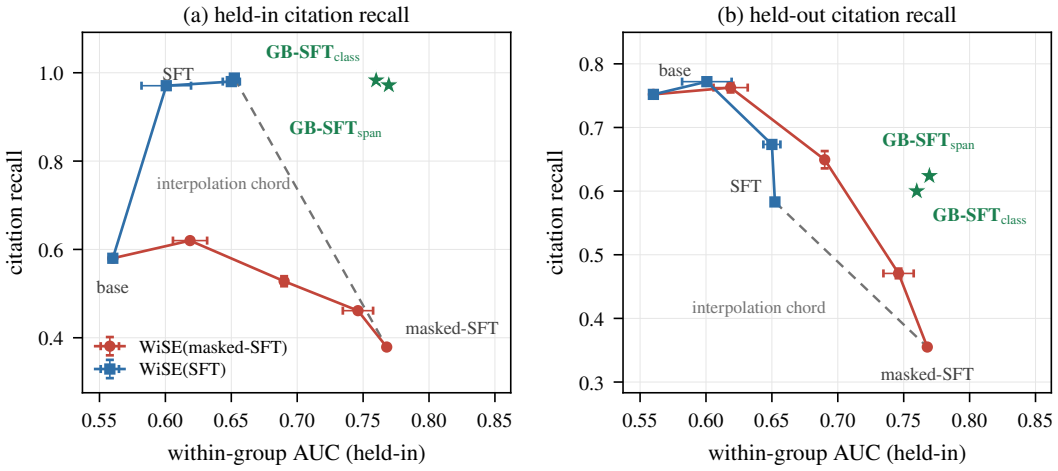


Figure 17: Weight-space interpolation for Qwen3.5-9B-Base on HOUSING-QA. Both panels share the same horizontal axis, within-group AUC on *held-in* groups, and differ only in which citation-recall split is on the vertical axis: (a) held-in recall, which overlaps training and measures fitting, and (b) held-out recall on jurisdictions no arm trained, which is the preservation measurement. The chord representing an AUC-citation recall tradeoff is defined only between the two endpoints SFT ↔MASKED-SFT chord (dashed); points left of SFT's AUC are off-segment and are given no gap. Endpoints and the two starred arms are the three-seed values of Table 1.

D.9.1 FORGETTING MITIGATION

REPLAY Chaudhry et al. (2019); Kotha & Liang (2026) empirically varies the class emphasis by changing volume instead of objective weight. We run two variants of REPLAY on HousingQA that hold that share fixed and vary only how it is spread. REPLAY$_{\text{single}}$ draws one sample from the unmasked corpus to mix back into the masked training set, dropping the citation share of supervised tokens from 81% to 30.7%, and trains on it for all three epochs. Thus, only 59 of 1,176 citations are ever supervised, and the same 59 three times. REPLAY$_{\text{cover}}$ keeps the aggregate share at 30.3% but draws a *disjoint* fold each epoch, which increases coverage at identical intensity and identical step

Method	AUC-in	AUC-out	cite-in	cite-out
untuned base	0.5601	0.5411	0.580	0.752
references, from Table 1				
SFT	0.6523 $\pm$0.0108	0.5761 $\pm$0.0212	0.988 $\pm$0.003	0.583 $\pm$0.016
MASKED-SFT	0.7679 $\pm$0.0070	0.5784 $\pm$0.0118	0.379 $\pm$0.005	0.355 $\pm$0.010
GB-SFT$_{\text{class}}$	**0.7599** $\pm$0.0132	0.6025 $\pm$0.0025	0.983 $\pm$0.002	0.600 $\pm$0.017
GB-SFT$_{\text{span}}$	**0.7695** $\pm$0.0087	0.6029 $\pm$0.0096	0.972 $\pm$0.003	0.624 $\pm$0.006
forgetting mitigation				
REPLAY$_{\text{single}}$	0.6899 $\pm$0.0156	0.5790 $\pm$0.0088	0.797 $\pm$0.011	**0.783** $\pm$0.010
REPLAY$_{\text{cover}}$	0.5706 $\pm$0.0142	0.5294 $\pm$0.0114	0.871 $\pm$0.005	**0.783** $\pm$0.013

Table 12: Forgetting-mitigation controls, HOUSING-QA, Qwen3.5-9B-Base. Both replay arms carry a citation share of supervised tokens (30.7% and 30.3%) close to GB-SFT$_{\text{class}}$'s 33.3%, so the comparison holds the budget roughly fixed and varies only whether it is delivered by data volume or by loss weight.

count. It is implemented by tripling the corpus and running a single epoch, each copy carrying its own mask, so an example's citation is supervised in at most one of its three appearances.

In Table 12, we see that volume does not reproduce weighting: at a matched share REPLAY$_{\text{single}}$ reaches 0.6899 held-in AUC against GB-SFT$_{\text{class}}$'s 0.7599. The shortfall is not coverage: raising the number of distinct citations supervised sixfold, at identical intensity and step count, moves held-in AUC *down* to 0.5706, barely above the untuned base. The missing ingredient is rather structure, which is exactly what our group-balanced objective supplies. However, while both REPLAY arms also lag GB-SFT in terms of held-in citation recall, they hold the best held-out citation recall in the table (0.783, above the base's 0.752), so we can see them achieving high preservation (but they pay for with a substantial injection cost).

Meanwhile, WISE-FT Wortsman et al. (2022) simply creates a weighted combination of the fine-tuned and base models: $\theta_\alpha = (1-\alpha)\,\theta_0 + \alpha\,\theta_{\text{ft}}$. Under LoRA $\theta_{\text{ft}} = \theta_0 + \Delta W$, so we can implement WISE-FT and sweep different values for it simply by a rescaling of the weight update that costs no training at all.

In Fig. 17, we see that WISE-FT reaches a different part of the frontier, not a better one. Scaling down an aggressive masked-SFT update sits *on or below* the SFT↔masked-SFT chord on held-in recall and only modestly above it on held-out ($+0.073$ to $+0.141$), where budget correction stands two to three times further above the same line ($+0.229$, $+0.272$). Scaling down SFT never exceeds SFT's own AUC at all, so only the masked endpoint was ever a candidate to reach the high-AUC regions.

As a specificity control (not plotted), we measure held-out AUC, which we find stays flat across the entire sweep: 0.570–0.593 for every WISE arm against an untuned base of 0.541, in contrast to held-in AUC which climbs from 0.619 to 0.746. The change in within-group AUC therefore measures injection rather than a general shift in discriminability. Note that GB-SFT$_{\text{class}}$ and GB-SFT$_{\text{span}}$ reach 0.6025 and 0.6029, the highest in the comparison.

The takeaway is that both anti-forgetting techniques buy real held-out citation recall at the cost of knowledge injection ability. Neither REPLAY nor WISE-FT, however, reaches GB-SFT's injection at a comparable citation budget. They are points on the injection-preservation Pareto frontier that trade injection away for preservation, which is the opposite of what we are asking the method to do; they remain so at matched budget, at sixfold REPLAY coverage, and across the whole WISE-FT interpolation path.

D.9.2 MEMORY EDITING

FT-L (Meng et al., 2022) restricts a factual edit to one layer's MLP output projection, on the view that a fact can be written to a small, identifiable part of the network. We implement its defining restriction of adaptation confined to a single layer ℓ's down-projection with every other adapter parameter frozen; our implementation does not reproduce its L_∞ clamp on the weight delta, but captures the

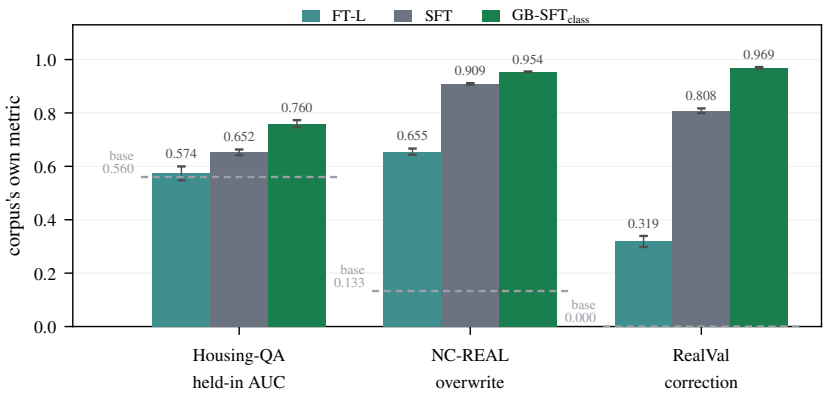


Figure 18: **Single-layer editing does not recover what reweighting does.** FT-L-style adaptation confined to layer 8's MLP down projection at 5×10^{-4} against ungrouped SFT and GB-SFT$_{\text{class}}$ on all three legal fact-overwrite corpora. Each corpus carries its own measure (held-in AUC, overwrite, correction) so a bar is comparable only to the others in its group.

primary single-layer editing mechanism of FT-L. Because a single layer trains under very different conditions from a full-network update, we swept both free settings rather than inheriting ours: layers $\{2, 4, 6, 8, 16, 24\}$ of 32 against learning rates $\{10^{-4}, 5\times10^{-4}, 2\times10^{-3}\}$, on HOUSING-QA at one seed, then carried the winning cell to all three corpora at three seeds.

FT-L clearly learns, clearing the untuned base on all three corpora particularly its own best setting of knowledge editing (NC-REAL and REALVAL). However, it is still below ungrouped SFT on all three corpora and far below GB-SFT$_{\text{class}}$ (Figure 18). Confining the update to one layer therefore does not recover what reweighting it does.