Mechanisms are Transferable: Data-Efficient Low-Resource Adaptation via Circuit-Targeted Supervised Fine-Tuning

Anonymous ACL submission

Abstract

Adapting LLMs to low-resource languages is difficult: labeled data is scarce, full-model fine-tuning is unstable, and continued cross-lingual tuning can cause catastrophic forgetting. We propose Circuit-Targeted Supervised Fine-Tuning (CT-SFT): a counterfactual-free adaptation of CD-T (Contextual Decomposition Transformer) that uses a label-balanced mean baseline and task-directional relevance scoring to identify a sparse set of task-relevant attention heads in a proxy-language checkpoint, then transfer learns to a target language by updating only those heads (plus LayerNorm) via head-level gradient masking. Across NusaX-Senti and XNLI, CT-SFT improves cross-lingual accuracy over continued full fine-tuning while updating only a small subset of model parameters. We find an editing-preserving trade-off: harder transfers favor editing circuit heads, while easier transfers often favor near-zero (i.e., low-relevance heads) updates, preserving the source mechanism. CT-SFT also substantially reduces catastrophic forgetting, preserving proxy/source-language competence during transfer.

1 Introduction

Most of the world's languages sit in the long tail: they have limited digitized text, limited labeled data, and limited tooling compared to a handful of high-resource languages. This imbalance matters because many standard NLP recipes implicitly assume abundant supervision and stable evaluation, leaving low-resource adaptation underserved in practice (Joshi et al., 2020; Hu et al., 2020; Aji et al., 2022).

In this regime, full-model supervised fine-tuning is often fragile: model capacity vastly exceeds what small datasets can constrain, making training sensitive to sample choice and hyperparameters and prone to measurable instability in small-data settings (Somayajula et al., 2024; Du et al., 2023; Mosbach et al., 2021; Dodge et al., 2020). Cross-lingual continued tuning adds another failure mode, catastrophic forgetting of the source competence, hindering multilingual deployment where we want to add capability without destroying what was already learned (Li et al., 2024; Vu et al., 2022; Pfeiffer et al., 2020).

A natural response is to restrict learning capacity via parameter-efficient or sparse updates (Ding et al., 2023). The lottery-ticket view suggests large networks contain many trainable subnetworks (Frankle and Carbin, 2019; Mehta et al., 2019; Chen et al., 2020), and composable sparse tuning can reduce interference in cross-lingual transfer (Ansell et al., 2022). However, sparsity alone does not answer the central question: which parameters should move, and which should be preserved. This motivates mechanism-aware selection. Mechanistic interpretability tools can localize computation in transformers (Rai et al., 2024), often identifying small sets of causally important heads in realistic behaviors (Wang et al., 2023), while also highlighting open challenges in extending methods beyond controlled settings (Sharkey et al., 2025).

We employ mechanistic interpretability for diagnosis and as a training method for low-resource adaptation. We introduce Circuit-Targeted Supervised Fine-Tuning (CT-SFT), a two-stage process where (i) a task-competent checkpoint is learned in a higher-resource proxy/source language and (ii) adaptation to a low-resource target occurs by updating a small, relevant circuit found in that checkpoint. Our discovery procedure enhances Contextual Decomposition for Transformers (CD-T) (Hsu et al., 2025) by using methods compatible with unstructured text.

We evaluate CT-SFT on NusaX-Senti (Winata et al., 2023) and XNLI (Conneau et al., 2018). Using competence-tuning as a proxy for transfer difficulty, we find an editing–preserving trade-off: larger source–target mismatches benefit from

editing decision-aligned heads, while smaller mismatches favor preserving the source mechanism. Overall, CT-SFT improves data efficiency and reduces destructive drift, complementing evidence of component reuse (Merullo et al., 2024).

Our contributions in this work are as follows:
(i) We propose the adaptation of CD-T (Hsu et al., 2025) to unstructured low-resource text using a label-balanced statistical baseline and a task-directional relevance score, avoiding handcrafted counterfactuals.
(ii) Our proposal: Circuit-Targeted Supervised Fine-Tuning (CT-SFT) is a mechanism-guided fine-tuning method that restricts updates to a discovered set of task-relevant attention heads.
(iii) We identify an editing–preserving trade-off in cross-lingual transfer that tracks source–target mismatch measured by competence tuning.
(iv) We show that CT-SFT substantially reduces catastrophic forgetting compared to continued full fine-tuning.

2 Related Work

Mechanistic interpretability (MI) has led to an expanding set of tools for localizing computation in transformers (Rai et al., 2024; Sharkey et al., 2025). Techniques like path patching and causal interventions have been used to identify the components responsible for specific behaviors in real-world tasks, such as the IOI task (Wang et al., 2023). Research has shown that only a small subset of heads can be causally important in these contexts (Wang et al., 2023). Automated methods such as ACDC recover sparse causal graphs via activation patching (Conmy et al., 2023), while attribution-based approaches, EAP, and variants like integrated gradients and GradPath, scale circuit identification through edge-level scoring (Nanda, 2023; Syed et al., 2024; Hanna et al., 2024; Zhang et al., 2025b). Contextual Decomposition for Transformers (CD-T) complements these with a mathematically grounded decomposition into "irrelevant" and "relevant" streams that can efficiently recover circuits (Hsu et al., 2025).

Many methods are validated on template-based tasks (e.g., IOI, greater-than, docstring) where clear counterfactuals and controlled reference states are available. However, circuit discovery is more challenging with unstructured, diverse data lacking faithful counterfactuals. Recent reviews highlight that evaluations on real-world tasks are still uncommon, raising questions about the transferability of current interpretability methods beyond controlled settings (Sharkey et al., 2025).

Most research on mechanistic interpretability has primarily focused on English (Yu et al., 2025). However, recent studies have started to explore multilingual behaviors and the internal structures that exist across different languages (Resck et al., 2025; Zhang et al., 2025c). Investigations into cross-lingual circuit similarity suggest that models maintain structurally similar pathways across languages, while also demonstrating language-specific specializations when necessary (Zhang et al., 2025c).

Several lines of research connect mechanistic structure to adaptation. Circuits and components can be reused across tasks (Merullo et al., 2024), and circuit analyses of fine-tuning suggest that substantial changes are concentrated in a small subset of heads, motivating mechanism-aware adaptation strategies such as circuit-informed LoRA (Wang et al., 2025).

Relatedly, Li et al. cast fine-tuning as subgraph search and propose *circuit-tuning*, which iteratively prunes an EAP-based circuit and updates only the associated parameters (Li et al., 2025). Mechanism-guided tuning has also been explored for translation, where selectively tuning identified components can outperform full or random tuning, though these approaches typically rely on large-scale datasets and carefully designed interventions (Zhang et al., 2025a).

In contrast, we modify CD-T for unstructured, low-resource datasets by using counterfactual-free baseline construction and a task-relevance score to aid component selection. Additionally, while multilingual MI primarily serves a diagnostic purpose, we leverage the discovered mechanisms to guide low-resource adaptation by reusing these circuits to fine-tune models for lower-resource languages.

3 Contextual Decomposition Transformer (CD-T)

Contextual Decomposition Transformer (CD-T) (Hsu et al., 2025) links model behavior to specific internal components by decomposing the activation of h of every component into two additive terms: an irrelevant component (γ) representing the baseline processing, and a relevant component (β) representing the deviation induced by the input. Let $a_x(s)$ denote the activation produced by the attention head s on input x. Given the pre-computed

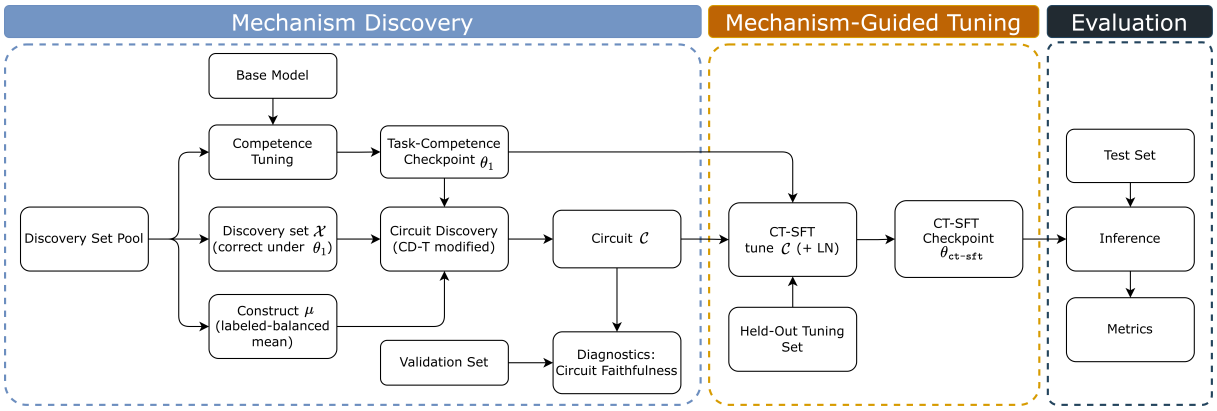


Figure 1: **CT-SFT overview.** Our method has two phases. **Mechanism Discovery** (left): starting from a base model, we perform competence tuning on a higher-resource proxy language to obtain a task-competent checkpoint θ_1. From a discovery pool, we form (i) a correctly-predicted discovery set $\mathcal{X}$ under θ_1 and (ii) a label-balanced statistical baseline μ for CD-T. We then run a modified CD-T procedure to score components and select a task-relevant circuit $\mathcal{C}$ (a subset of attention heads). We optionally compute circuit-faithfulness diagnostics on a validation set. **Mechanism-Guided Tuning** (middle): for a low-resource target language, we fine-tune only the parameters corresponding to heads in $\mathcal{C}$ (and LayerNorm), using a held-out tuning pool, producing the CT-SFT checkpoint $\theta_{\text{ct-sft}}$. **Evaluation** (right): we evaluate $\theta_{\text{ct-sft}}$ on the test set and report task metrics.

mean μ for head s, i.e., $\mu(s)$, CD-T defines:

$$\gamma_s = \mu(s); \qquad \beta_s = a_x(s) - \mu(s). \tag{1}$$

where the precomputed mean μ is constructed from faithful counterfactual examples, such as swapping names or years, when applied to templated tasks like IOI, greater-than, and docstring (Hsu et al., 2025). These streams are then propagated through the network using contextual decomposition rules to produce contribution $\beta_{s\to t}(x)$ from head s to a downstream target t, including contributions at the logit readout or at the intermediated residual-stream sites.

CD-T rank components using an *unsigned* magnitude ratio between relevant and irrelevant streams (Hsu et al., 2025):

$$R(s,t) = \frac{\|\beta_{s\to t}\|_1}{\|\gamma_{s\to t}\|_1}. \tag{2}$$

This relevance score R favors heads whose "relevant" signal (β) is large relative to the background mean (γ).

Circuit Selection. Let $\mathcal{T}^{(d)}$ denote the frontier targets at depth d. At d=0, $\mathcal{T}^{(0)}$ is the logit readout, and we select the top K_0 heads based on relevance score (averaged over discovery inputs) to form $\mathcal{C}^{(0)}$. For $d\geq 1$, we set $\mathcal{T}^{(d)} = \mathcal{C}^{(d-1)}$ and select the top K_d heads by mean directional relevance to $\mathcal{T}^{(d)}$ (averaged over targets and inputs), forming $\mathcal{C}^{(d)}$. We stop the iteration at layer 0 and output the selected heads $\mathcal{C} = \bigcup_d \mathcal{C}^{(d)}$.

4 Adapting CD-T for Unstructured Text in Low-Resource Languages

Our approach consists of two distinct phases (Figure 1): (1) *Mechanism Discovery*, where we identify a task-relevant circuit in a higher-resource proxy language, and (2) *Mechanism-Guided Tuning*, where we adapt the model to low-resource languages by training only the parameters corresponding to the discovery circuit. In our setting, a component is an attention head (l, h) at layer l and head index h. The circuit is a subset of heads selected using a relevance score aggregated across the discovery set.

4.1 Phase 1: Mechanism Discovery

We utilize Contextual Decomposition for Transformers (CD-T) (Hsu et al., 2025) to identify the task-relevant circuit.

4.1.1 Precomputed-mean Construction: Establishing a "Neutral State" (Label-Balanced Mean)

A critical dependency in this formulation is the choice of μ: the precomputed-mean determines what is treated as "background" (γ) versus "signal" (β). In templated tasks (e.g., IOI, greater-than, docstring), μ can be defined using faithful counterfactuals (e.g., swapping names or years) (Hsu et al., 2025). However, for unstructured real-world text, generating such counterfactuals is impractical without expensive human annotation.

Accordingly, we approximate μ from in-distribution samples rather than explicit counterfactuals, and enforce label balance in the mean-estimation set $\mathcal{M}$ (i.e., using equal number of examples per class) to reduce leakage of class-directional signal into the baseline. Specifically, for each head s, we compute:

$$\mu(s) = \mathbb{E}_{x \sim \mathcal{M}}[a_x(s)]. \tag{3}$$

Balancing ensures that class-directional information cancels in expectation, and the baseline approximates a neutral operating point.

4.1.2 Scoring Protocol: Directional Relevance over Magnitude

Original CD-T scores components using an unsigned magnitude ratio (Eq. 2) (Hsu et al., 2025), but high magnitude does not necessarily mean task support: a head can be non-discriminative or even anti-aligned with the gold label. This can cause magnitude-based selection to include components that amplify spurious features or push probability mass away from the correct label.

To address this limitation, we *propose* a directional relevance scoring protocol that explicitly measures whether a head's contribution supports the correct prediction direction (i.e., increases the logit of the gold label relative to alternatives), rather than merely having large activation magnitude. We use this directional score for circuit selection throughout our experiments.

Directional relevance at the logit readout (classification margin). Let y_c be the correct label and $\mathcal{Y}_{\text{other}} = \mathcal{Y} \setminus \{y_c\}$. When the target t is a logit readout, we score head s using the relevant contribution to the correct logit relative to the average contribution to incorrect logits:

$$R(s,t) = \beta_{s \to t}(y_c) - \frac{1}{|\mathcal{Y}_{\text{other}}|} \sum_{y \in \mathcal{Y}_{\text{other}}} \beta_{s \to t}(y). \tag{4}$$

This measures discriminative contribution at the decision boundary: positive scores indicate that the head increases the correct class logit more than the competing classes.

Directional relevance beyond the logit layer (task-direction projection). Intermediate layers do not write directly to logits; instead, they update the residual stream, and the final logits are obtained by a linear readout via the unembedding matrix W_U (Belrose et al., 2023; Nanda, 2024). This makes W_U a natural label-specific reference direction for judging whether an intermediate residual update increases the gold-label logit relative to competing labels. To evaluate whether an intermediate contribution supports the correct class, we define a task direction using the unembedding matrix W_U:

$$\mathbf{v}_{\text{task}} = W_U(y_c) - \frac{1}{|\mathcal{Y}_{\text{other}}|} \sum_{y \in \mathcal{Y}_{\text{other}}} W_U(y). \tag{5}$$

This vector approximates the readout direction that separates the correct label from alternatives at the output layer. For an intermediate contribution vector $\beta_{s \to t}(x)$ we define:

$$R(s,t) = \frac{\beta_{s \to t}(x) \cdot \mathbf{v}_{\text{task}}}{\|\mathbf{v}_{\text{task}}\|_2}. \tag{6}$$

This score measures alignment with the task direction: positive values indicate alignment with the correct label, and negative values indicate opposition. We empirically validate this scoring choice via circuit faithfulness in Section 6.4 and use this scoring mechanism to conduct our circuit selection.

4.2 Phase 2: Mechanism-Guided Tuning

Given the selected attention-head circuit $\mathcal{C}$, we conduct Circuit-Targeted Supervised Fine-Tuning (CT-SFT) by restricting optimization to the heads in $\mathcal{C}$. This focuses adaptation on task-relevant components identified in Phase 1, rather than on globally distributed weight updates.

In our surgical setting, we freeze all model parameters except LayerNorm and apply head-level gradient masking. This ensures that only the parameters of heads in $\mathcal{C}$ are updated while keeping MLPs, token embeddings, the unembedding matrix, and all unselected heads fixed. This approach localizes learning to the discovered circuit, clarifying the connection between behavioral changes and the mechanism.

5 Experimental Setup

5.1 Base Model, Dataset, and Task

All experiments use Qwen2.5-0.5B as the base model. We evaluate sentiment classification on low-resource NusaX-Senti (Winata et al., 2023) and XNLI (Conneau et al., 2018) under the same low-resource language settings (Appendix A).

Source language and targets. We discover circuits on INDONESIAN for NusaX and transfer to {ace, bug, jav, min}; and discover circuits on ENGLISH for XNLI and transfer to {th, vi, es, zh}.

Two-phase protocol and data pools. For each dataset, we split training data into two disjoint pools (Appendix A): a **discovery pool** and a **held-out tuning pool**. The discovery pool is used to (i) obtain a task-competent checkpoint via source adaptation and (ii) run circuit discovery. Transfer performance is reported on the held-out *test* set, while the validation split is used only for diagnostics (e.g., circuit faithfulness), not for optimization.

Mechanism-discovery budget. To study the effect of source competence, we vary the number of source-language examples used in the mechanism-discovery phase, $n_{\text{src}} = 50$ for NusaX and $n_{\text{src}} \in \{50, 250\}$ for XNLI. Unless stated otherwise, circuit discovery and CT-SFT start from the resulting task-competent checkpoints.

5.2 CD-T Discovery Set and Baseline Construction

Discovery inputs (x). We compute head relevance on 50 correctly predicted samples to reduce noise from mispredicted instances when estimating task-supportive contributions.

Balanced-mean baseline (μ). We compute the baseline mean using 50 examples because label-balanced sampling is limited by the available label distribution in NusaX for some settings.

5.3 Circuit Construction and Depth

We construct the attention-head circuit by iteratively expanding backward from the logit readout (Section 3). We report results for multiple circuit expansion depths (max_depth $\in \{0, 1, 2\}$), where depth 0 selects heads that most directly support the output and larger depths progressively include upstream contributors. To control circuit growth, at each iteration, we retain only the top 2% of heads (i.e., 6 heads) by aggregated relevance score. Appendix D.2 compares selection ratios $p \in \{2\%, 5\%, 10\%\}$ and shows that larger ratios substantially increase circuit size with depth while yielding diminishing faithfulness gains.

5.4 Evaluation Protocol

All results (accuracy and faithfulness) are averaged across 4 seeds. We evaluate CT-SFT under the surgical protocol (Section 4.2) and compare against: (i) **Full fine-tuning** (all parameters), (ii) **CT-SFT** (update heads in the circuit: top-K by aggregated relevance), (iii) **Random head tuning** (uniformly sampled K heads), (iv) **Least-relevant tuning** (bottom-K heads), and (v) **Near-zero tuning** (K heads with relevance scores closest to zero). We report downstream task performance in both within-language tuning and cross-lingual transfer settings.

	n	Acc.	Δ
Comp. Full FT	–	**0.757**	–
Cont. Full FT	25	0.706	-0.051
	50	0.741	-0.016
	75	0.771	+0.014
	100	0.758	+0.001

Table 1: Within-language sanity check on Indonesian (Ind→Ind). We start from a 50-sample competence-tuned checkpoint (Comp. Full FT), then continue *full-model* fine-tuning on held-out Indonesian data of size n (Cont. Full FT). Δ is measured relative to the competence-tuning baseline.

6 Results

6.1 When Continued Full Fine-Tuning Can Hurt: A Within-Language Sanity Check (Ind→Ind)

In our within-language Indonesia→Indonesia sanity check, we first fine-tuned Qwen on 50 Indonesian samples, achieving a mean test accuracy of 0.757 across four seeds (Table 1). We then continued *full-model* fine-tuning from this Stage-1 checkpoint using additional Indonesian data with $n \in 25, 50, 75, 100$. Table 1 shows a non-monotonic pattern: adding 25 or 50 samples degrades performance relative to the competence-tuning baseline, whereas 75–100 samples recover or slightly improve accuracy. This behavior is consistent with prior findings that fully fine-tuning large pretrained models on small downstream datasets can be unstable and prone to overfitting, with generalization highly sensitive to data size (Somayajula et al., 2024; Ding et al., 2023).

This instability leads to restricted update regimes that limit learning to a small set of attention heads. Figure 2 compares these regimes across second-stage tuning sizes n and circuit expansion depths (max_depth $= 0, 1, 2$). Random sparse updates consistently outperform full-model tuning, indicating that sparsity serves as a strong regularizer in few-shot training. This supports the "lottery ticket" hypothesis that large networks contain many trainable subnetworks, allowing even randomly selected sparse subsets to enable functional adaptation (Frankle and Carbin, 2019; Mehta et al., 2019; Ansell et al., 2022).

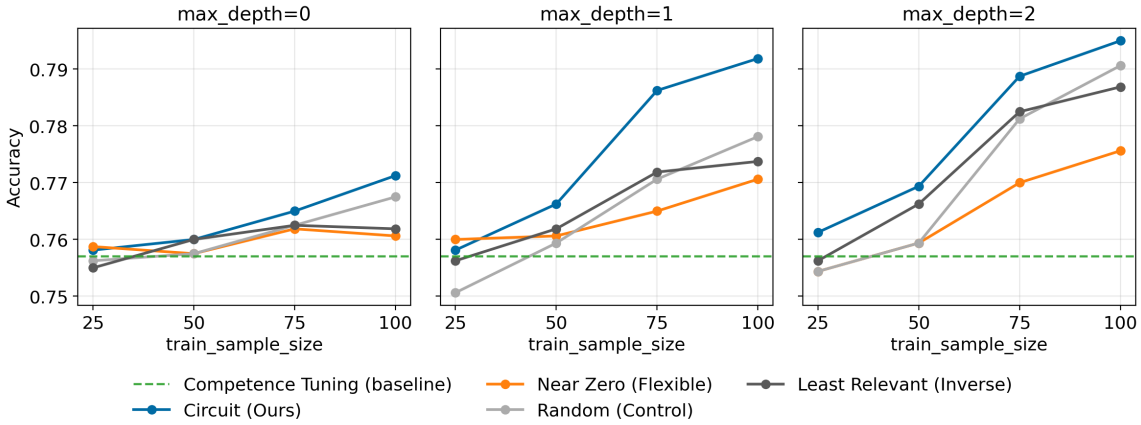


Figure 2: Indonesian → Indonesian: test accuracy versus training sample size under different head-selection strategies, shown for circuit expansion depths (max_depth = 0, 1, 2).

Target lang	n	Full FT (100%)	Depth 0 (6H, 0.23%)		Depth 1 (12H, 0.44%)		Depth 2 (18H, 0.66%)	
			Circuit	NearZero	Circuit	NearZero	Circuit	NearZero
ace	25	0.346	**0.482**	0.47	**0.491**	0.468	**0.486**	0.473
bug	25	0.376	0.374	0.373	**0.398**	0.392	0.394	**0.402**
jav	25	0.384	**0.572**	0.569	**0.609**	0.572	**0.592**	0.571
min	25	0.394	**0.558**	0.555	**0.583**	0.562	**0.568**	**0.568**
ace	100	0.428	**0.512**	0.478	**0.532**	0.499	**0.547**	0.537
bug	100	0.454	0.428	0.408	**0.488**	0.449	**0.493**	0.486
jav	100	0.475	**0.607**	0.584	**0.640**	0.611	0.623	**0.636**
min	100	0.512	**0.597**	0.580	**0.604**	0.586	0.601	**0.612**

Table 2: **Cross-lingual NusaX transfer:** test accuracy (%) across circuit depths. Full FT continues training from 50 samples Indonesian-tuned checkpoint (transfer setting) and is depth-independent. Underline marks the best among Circuit vs NearZero within each depth; **bold** indicates the underlined method also outperforms Full FT. Percentages in the depth headers denote the trainable-parameter budget for CT-SFT (LayerNorms + selected heads) as a fraction of the full model (630.5M params): depth 0/1/2 use 6/12/18 heads, corresponding to 0.23%/0.44%/0.66% of full model parameters. Full results across all tuning sizes and methods are reported in Appendix C.1.

Crucially, circuit-targeted updates are both stronger and more reliable than random: the circuit curves improve more consistently with n, while random exhibits higher variability and weaker gains, indicating that *where* we update matters beyond the benefit of *sparsity* alone. Overall, this within-language result supports the paper's central claim: when full fine-tuning is unstable in low-resource settings, interpretability-guided head selection provides a practical and mechanistically grounded update locus.

For readability, we omit the random-head baseline from the subsequent cross-lingual transfer plots in the paper (adding it would substantially increase visual clutter across languages/depths).

6.2 Cross-Lingual Transfer on NusaX: Editing vs Preserving the Mechanism

Table 2 shows that CT-SFT consistently improves cross-lingual accuracy over continued Full FT (competence-tuning on 50 Indonesian data) across languages and tuning sizes, with the best strategy depending on both circuit depth and target language. Updating the discovered circuit heads ("mechanism-editing") is particularly effective for Acehnese (ace) and Buginese (bug): for example, at $n = 100$ the Circuit strategy improves from Full FT 0.428→0.547 (ace) and 0.454→0.493 (bug) at depth 2, and similar gains appear at $n = 25$. This pattern is consistent with these transfers requiring larger representation shifts, where directly modifying decision-relevant computation is beneficial. In contrast, for Javanese (jav) and Minangkabau (min) the mechanism-preserving variant ("NearZero") becomes more competitive as depth increases: at depth 2, NearZero matches or surpasses Circuit (jav: 0.636 vs 0.623; min: 0.612 vs 0.601 at n=100), suggesting that preserving the Indonesian-aligned mechanism while adapting through weakly task-aligned components can reduce unnecessary drift when transfer is already relatively favorable. This is consistent

Target (tune)	Competence Tuning on IND	n=25 (eval on IND)			n=100 (eval on IND)		
		Direct FT	Full FT	**CT-SFT**	Direct FT	Full FT	**CT-SFT**
ace	0.757	0.432	0.341	**0.769**	0.614	0.435	**0.757**
bug	0.757	0.432	0.358	**0.771**	0.501	0.426	**0.758**
jav	0.757	0.393	0.386	**0.769**	0.632	0.451	**0.756**
min	0.757	0.470	0.424	**0.769**	**0.767**	0.380	0.758

Table 3: **Indonesian retention after continued fine-tuning (catastrophic forgetting diagnostic).** Test accuracies (%) after continuing fine-tuning (Full-FT and CT-SFT) on the target language (row). We evaluate trained models on IND to measure source-language retention. CT-SFT uses the *Circuit* head set at discovery depth d=2, taking the better of mechanism-preserving/editing scopes. *Direct FT* fine-tunes directly on the target language from the base model (no competence tuning).

with the competence-tuning baseline (Appendix Table 4), where the Indonesian-tuned checkpoint starts lower on Acehnese/Buginese than on Javanese/Minangkabau.

Circuit expansion depth modulates this trade-off. Depth 0 is often weaker or less reliable, consistent with selecting only readout-adjacent contributors: e.g., bug at n=25 slightly underperforms Full FT (0.374 vs 0.376), while deeper expansions recover improvements. Moving to depth 1 typically yields the strongest and most stable gains across languages (e.g., ace 0.532; jav 0.640 at n=100), while depth 2 can help further on harder transfers (ace/bug) but may start to dilute the advantage on easier transfers (jav/min), likely because deeper circuits include more upstream heads and therefore expand the set of updated parameters. We include the full accuracy@n curves for each depth in Appendix C.1 Figure 4, which show these trends persist across the intermediate tuning sizes and are not an artifact of the n=25/100 snapshot.

Catastrophic forgetting: CT-SFT preserves Indonesian competence. Table 3 demonstrates a stark difference in source-language retention after continued fine-tuning. Continued Full FT substantially degrades Indonesian accuracy (e.g., down to 0.341–0.451 at n=25/100 depending on target), indicating strong catastrophic forgetting when the full parameter set is updated for the target language. In contrast, CT-SFT preserves Indonesian competence near the competence tuning baseline (0.757) across all targets and tuning sizes, and can even slightly improve it (e.g., 0.769–0.771 at n=25), consistent with CT-SFT acting as a mechanism-constrained update that limits destructive drift in decision-relevant components. The "Direct FT" baseline, fine-tuning from the base model without Indonesian competence tuning, yields markedly lower Indonesian accuracy after tuning on the target language, reinforcing that the retained Indonesian performance under CT-SFT is not incidental but arises from starting with an Indonesian-competent checkpoint and constraining updates to preserve (or selectively edit) its mechanism.

6.3 Beyond Sentiment: Cross-Task Transfer on XNLI

We include XNLI (Conneau et al., 2018) as a secondary cross-task benchmark to test whether the NusaX (Winata et al., 2023) (sentiment) transfer pattern generalizes to a harder, non-sentiment task (natural language inference). Since our main story is still NusaX, we summarize only the key takeaways on XNLI here and defer the full per-language/per-depth results to the Appendix C.2.

The first result is that CT-SFT requires minimal task competence from the competence-tuning phase. With only 50 English samples for competence tuning, the checkpoint is not yet a reliable NLI model, and both surgical variants become unstable, often failing to improve and sometimes degrading transfer, especially at larger circuit depths (Appendix C.2; Table 6). We treat this setting mainly as a diagnostic that CT-SFT is a post-competence method rather than a substitute for learning the task.

Under 250-sample English competence tuning, CT-SFT becomes stable and exhibits the same editing–preserving trade-off as NusaX. Thai is the clearest hard transfer case: the competence-tuned checkpoint starts much lower on Thai (0.362) than on Vietnamese/Spanish/Chinese (>0.50; 0.511/0.510/0.544). Correspondingly, Thai benefits most from mechanism-editing (Circuit) updates, while Vietnamese/Chinese/Spanish more often favor mechanism-preserving NearZero updates, particularly at deeper circuit expansions, suggest-

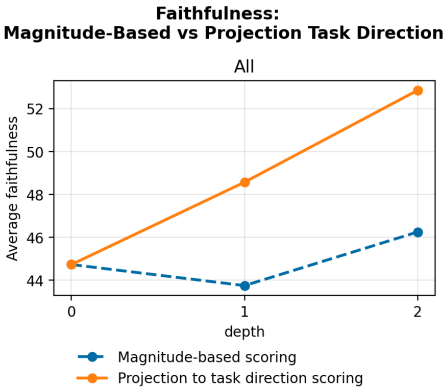


Figure 3: Accuracy-based faithfulness across circuit expansion depths comparing magnitude-based (norm) scoring and our task-direction projection scoring.

ing that when transfer is already favorable, constrained low-drift adaptation is preferable to directly editing decision-relevant computation (Appendix C.2; Figure 5).

6.4 Diagnostics: Why Task-Direction Scoring Works

Unless otherwise stated, all diagnostic results in this section are reported on NusaX.

Circuit Faithfulness. We evaluate the circuit's *faithfulness* across various depths of the CD-T model by comparing two scoring methods: (i) magnitude-based scoring and (ii) projection onto the task direction. We compute faithfulness using a mean-ablation protocol following CD-T: heads outside the circuit are replaced by their mean activation (the γ / "irrelevant" baseline) (Hsu et al., 2025), and we measure how well the circuit-only model preserves the full model behavior (Appendix D.1). As shown in Figure 3, projection-based scoring yields steadily increasing faithfulness with depth, whereas magnitude-based scoring is non-monotonic and lags behind at larger depths. This pattern suggests that deeper CD-T expansions benefit from selecting components aligned with the task direction. Magnitude-based ranking can prioritize high-norm but weakly task-aligned components, which becomes increasingly likely as the candidate set grows with depth. In contrast, projection-based scoring explicitly favors task-aligned contributions, producing circuits that more reliably preserve the full model's computation under mean ablation.

Circuit Topology. Figure 8 (Appendix D.4) shows how selected heads distribute across layers as circuit depth increases. Consistent with prior work on layerwise structure in transformers (Tenney et al., 2019; Jawahar et al., 2019; Clark et al., 2019; Ethayarajh, 2019), magnitude-based scoring concentrates selections in early layers (L0–2), while task-direction projection shifts selection toward mid layers (roughly L3–15). This suggests projection scoring is less dominated by shallow lexical/positional variation and more likely to capture decision-relevant computation.

7 Discussion and Conclusion

Our results support a mechanism-level explanation for why CT-SFT works in the few-shot transfer regime: it constrains where plasticity is allowed. Continued full fine-tuning provides too much freedom and can drift away from a previously learned mechanism, leading to unstable improvements and catastrophic forgetting of the source language. In contrast, CT-SFT updates only a tiny, causally motivated subset of attention heads (plus LayerNorm), acting as a strong regularizer: it remains plastic enough to adapt when the source–target gap is large, but stable enough to preserve competence when the gap is small. This stability–plasticity trade-off is visible in both in transfer accuracy and retention. In NusaX, continued Full FT substantially degrades Indonesian performance after tuning on each target language (Table 3), whereas CT-SFT preserves Indonesian accuracy near the competence-tuning baseline (and can even slightly improve it), consistent with a mechanism-constrained update that reduces destructive drift.

The same logic explains the editing–preserving pattern across languages. Harder transfers benefit from mechanism-editing (updating the discovered circuit heads), while easier transfers often benefit from mechanism-preserving updates that adapt with minimal disruption to the source mechanism. XNLI reproduces this behavior once the model reaches minimal NLI competence: after 250-sample English competence tuning, Thai (with lower starting transfer) benefits more from circuit editing, while languages with stronger starting transfer more often favor mechanism-preserving. Together, these results support a modular view of adaptation: competence tuning learns a reusable task mechanism, and CT-SFT either edits it when transfer is difficult or preserves it when transfer is favorable, while consistently reducing forgetting relative to continued full-model tuning.

8 Limitations

First, CT-SFT depends on the quality of mechanistic discovery. Although we modify CD-T to avoid counterfactuals via a labeled-balanced statistical baseline and directional relevance scoring, circuit quality can still be sensitive to the competence-tuned checkpoint and to the limited mean/discovery pool size (50 examples in our setting due to memory constraints). When competence is weak (e.g., the 50-sample English XNLI checkpoint), restricted update can become unstable, reinforcing that CT-SFT is intended as a post-competence adaptation method rather than a substitute for learning the task.

Second, we evaluate catastrophic forgetting primarily through source-language retention (Indonesian accuracy after continued-tuning) and show strong gains for CT-SFT over continued Full FT. However, forgetting can also be measured in other ways (e.g., distributional shifts within the source language, robustness metrics, or multilingual retention across multiple sources), which we do not exhaustively explore here.

Third, our experiments focus on one base model family (Qwen2.5-0.5B), a surgical update regime (heads + LayerNorm), and two benchmarks under a standardized single-token label protocol. Whether the same retention and editing-preserving behavior holds for larger models, other architectures, multi-token generation tasks, or alternative parameter-efficient methods (e.g., LoRA variants) under identical low-resource constraints remains to be validated.

Potential Risks. Our method is a training procedure and does not introduce new capabilities on its own, but it can change the *cost* of adapting models to new languages. This may lower the barrier to deploying adapted models without adequate evaluation in the target language(s), thereby amplifying existing harms such as biased predictions, offensive outputs, or uneven performance across dialects and demographic groups. In addition, circuit-based explanations may be over-interpreted: the discovered heads are evidence of task-relevant computation under our specific model, datasets, and prompts, but they should not be treated as definitive or universal mechanisms. We therefore recommend target-language evaluation and basic safety checks before deployment, and we frame mechanistic claims as empirical findings within the studied setting.

References

Alham Fikri Aji, Genta Indra Winata, Fajri Koto, Samuel Cahyawijaya, Ade Romadhony, Rahmad Mahendra, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Timothy Baldwin, Jey Han Lau, and Sebastian Ruder. 2022. One country, 700+ languages: NLP challenges for underrepresented languages and dialects in Indonesia. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7226–7249, Dublin, Ireland. Association for Computational Linguistics.

Alan Ansell and 1 others. 2022. Composable sparse fine-tuning for cross-lingual transfer. In *Proceedings of ACL*.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Tianlong Chen, Jonathan Frankle, Shiyu Chang, Sijia Liu, Yang Zhang, Zhangyang Wang, and Michael Carbin. 2020. The lottery ticket hypothesis for pre-trained BERT networks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems (NeurIPS)*.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of BERT's attention. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 276–286, Florence, Italy. Association for Computational Linguistics.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. In *Advances in Neural Information Processing Systems*, volume 36.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Ning Ding and 1 others. 2023. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*.

Wei Du, Thien Huu Nguyen, David Rosenberg, David Grangier, Andrew McCallum, Hongyu Lin, and

Alexander M. Rush. 2023. Measuring the instability of fine-tuning in neural language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1728–1747, Toronto, Canada. Association for Computational Linguistics.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Jonathan Frankle and Michael Carbin. 2019. The lottery ticket hypothesis: Finding sparse, trainable neural networks. In *International Conference on Learning Representations (ICLR)*.

Michael Hanna, Sandro Pezzelle, and Yonatan Belinkov. 2024. Have faith in faithfulness: Going beyond circuit overlap when finding model mechanisms. In *ICML 2024 Workshop on Mechanistic Interpretability*. Spotlight.

Aliyah R. Hsu, Georgia Zhou, Yeshwanth Cherapanamjeri, Yaxuan Huang, Anobel Odisho, Peter R. Carroll, and Bin Yu. 2025. Efficient automated circuit discovery in transformers using contextual decomposition. In *International Conference on Learning Representations (ICLR)*. Poster.

Junjie Hu, Sebastian Ruder, Aditya Siddhant, Graham Neubig, Orhan Firat, and Melvin Johnson. 2020. XTREME: A massively multilingual multi-task benchmark for evaluating cross-lingual generalisation. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 4411–4421. PMLR. Also available via ACM DL: https://dl.acm.org/doi/10.5555/3524938.3525348.

Ganesh Jawahar, Benoît Sagot, and Djamé Seddah. 2019. What does BERT learn about the structure of language? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3651–3657, Florence, Italy. Association for Computational Linguistics.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Kun Li, Huan Li, Dong Shen, Xinyang Jiang, and Hao Ji. 2024. Mitigating catastrophic forgetting in large language model tuning with sharpness-aware minimization. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 4308–4322, Miami, Florida, USA. Association for Computational Linguistics.

Yueyan Li, Wenhao Gao, Caixia Yuan, and Xiaojie Wang. 2025. Fine-tuning is subgraph search: A new lens on learning dynamics. *arXiv preprint*, arXiv:2502.06106.

Ronak Mehta and 1 others. 2019. Sparse transfer learning via winning lottery tickets. *arXiv preprint arXiv:1905.07785*.

Jack Merullo, Carsten Eickhoff, and Ellie Pavlick. 2024. Circuit component reuse across tasks in transformer language models. In *The Twelfth International Conference on Learning Representations (ICLR)*. Spotlight (OpenReview).

Marius Mosbach, Maksym Andriushchenko, and Dietrich Klakow. 2021. On the stability of fine-tuning BERT: Misconceptions, explanations, and strong baselines. In *Proceedings of the International Conference on Learning Representations (ICLR)*.

Neel Nanda. 2023. Attribution patching: Activation patching at industrial scale. Blog post.

Neel Nanda. 2024. Mechanistic interpretability explainer & glossary. `https://www.neelnanda.io/mechanistic-interpretability/glossary`. Accessed: 2025-12-10.

Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An adapter-based framework for multi-task cross-lingual transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 7654–7673. Association for Computational Linguistics.

Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. 2024. A practical review of mechanistic interpretability for transformer-based language models. *arXiv preprint*.

Lucas Resck, Isabelle Augenstein, and Anna Korhonen. 2025. Explainability and interpretability of multilingual large language models: A survey. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 20465–20497, Suzhou, China. Association for Computational Linguistics.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeffrey Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, and 1 others. 2025. Open problems in mechanistic interpretability. *Transactions on Machine Learning Research*. Published in TMLR (09/2025). Reviewed on OpenReview.

Sai A. Somayajula and 1 others. 2024. Generalizable and stable finetuning of pretrained language models. In *Proceedings of NAACL*.

Aaquib Syed, Can Rager, and Arthur Conmy. 2024. Attribution patching outperforms automated circuit discovery. In *Proceedings of the 7th BlackboxNLP Workshop: Analyzing and Interpreting Neural Networks for NLP*, pages 407–416, Miami, Florida, US. Association for Computational Linguistics.

Ian Tenney, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy. Association for Computational Linguistics.

Tu Vu, Aditya Barua, Brian Lester, Daniel Cer, Mohit Iyyer, and Noah Constant. 2022. Overcoming catastrophic forgetting in zero-shot cross-lingual generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 9279–9300. Association for Computational Linguistics.

Kevin Ro Wang, Alexandre Variengien, Arthur Conmy, Buck Shlegeris, and Jacob Steinhardt. 2023. Interpretability in the wild: a circuit for indirect object identification in gpt-2 small. In *International Conference on Learning Representations (ICLR)*. Poster.

Xu Wang, Yan Hu, Wenyu Du, Reynold Cheng, Benyou Wang, and Difan Zou. 2025. Towards understanding fine-tuning mechanisms of llms via circuit analysis. In *ICLR 2025 Workshop on Building Trust in Language Models and Applications*. Workshop paper (OpenReview).

Genta Indra Winata, Alham Fikri Aji, Samuel Cahyawijaya, Rahmad Mahendra, Fajri Koto, Ade Romadhony, Kemal Kurniawan, David Moeljadi, Radityo Eko Prasojo, Pascale Fung, Timothy Baldwin, Jey Han Lau, Rico Sennrich, and Sebastian Ruder. 2023. NusaX: Multilingual parallel sentiment dataset for 10 Indonesian local languages. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 815–834, Dubrovnik, Croatia. Association for Computational Linguistics.

Haeun Yu, Seogyeong Jeong, Siddhesh Pawar, Jisu Shin, Jiho Jin, Junho Myung, Alice Oh, and Isabelle Augenstein. 2025. Entangled in representations: Mechanistic investigation of cultural biases in large language models. *arXiv preprint arXiv:2508.08879*.

Hongbin Zhang, Kehai Chen, Xuefeng Bai, Xiucheng Li, Yang Xiang, and Min Zhang. 2025a. Exploring the translation mechanism of large language models. In *NeurIPS 2025*. Poster (OpenReview).

Lin Zhang, Wenshuo Dong, Zhuoran Zhang, Shu Yang, Lijie Hu, Ninghao Liu, Pan Zhou, and Di Wang. 2025b. Eap-gp: Mitigating saturation effect in gradient-based automated circuit identification. In *Advances in Neural Information Processing Systems (NeurIPS)*. OpenReview preprint / NeurIPS 2025.

Ruochen Zhang, Qinan Yu, Matianyu Zang, Carsten Eickhoff, and Ellie Pavlick. 2025c. The same but different: Structural similarities and differences in multilingual language modeling. In *International Conference on Learning Representations (ICLR)*. Poster (OpenReview).

A Dataset Construction

We evaluate two tasks under matched low-resource settings: sentiment classification on NusaX-Senti (Winata et al., 2023) and natural language inference on XNLI (Conneau et al., 2018). To make the comparison fair, we standardize (i) the data scale, (ii) the input formatting, (iii) the maximum prompt length, and (iv) the label prediction protocol across both datasets. Unless stated otherwise, all fine-tuning runs use a fixed training recipe: 5 epochs, max sequence length 128 tokens, learning rate 5×10^{-5}, and batch size 16.

A.1 NusaX-Senti

We follow the NusaX-Senti (Winata et al., 2023) setup for multilingual sentiment classification. Each sample example consist of a single input sentence, and we cast the task into an autoregressive single-token label prediction by appending a label field at the end of the prompt: `<sentence> | sentiment: <label>`, where the label is one of negative, neutral, positive. We use the model tokenizer and truncation policy across languages with `max_length = 128`, padding to a fixed length.

A.2 XNLI (Natural Language Inference)

Split. To match the NusaX-Senti low-resource scale, we create an XNLI subset with `train=500`, `val=100`, and `test=400` examples *per language*.

Parallel pools. XNLI's parallel structure is available mainly in the official validation and test splits (Conneau et al., 2018). Since there is no dedicated multilingual parallel training split, we construct low-resource train/val by disjointly sampling from the XNLI validation pool, and use the official XNLI test split for final evaluation.

Prompt-length filtering. To ensure prompt consistency across languages, we filter candidate examples to those with tokenized prompt length ≤ 128 (under our template), then sample train/val/test from the filtered pools.

Autoregressive formatting. We cast XNLI as an autoregressive single-token label prediction task using: `Premise: <premise> Hypothesis: <hypothesis> Label: <label>`, where `<label>` is one of {`entail`, `neutral`, `contradiction`}.

A.3 Single-token label prediction (applies to both datasets)

We evaluate both tasks as *one-token prediction* at the label position. Concretely, we choose label strings that tokenize into a single token under the model tokenizer and use a leading space (e.g., `" entail"`, `" negative"`).

A.4 Holdout Protocol and Pool Usage.

- **Discovery pool (competence tuning + discovery).** We allocate **400** examples from the training split to the *discovery pool*. Competence tuning is performed on this pool (with $n_{\text{src}} \in \{50, 250\}$), and we run baseline-mean construction and circuit discovery on the *same* discovery pool to characterize the task mechanism learned during competence tuning.

- **Held-out tuning pool (second-phase holdout).** We allocate the remaining **100** training examples to a disjoint *held-out tuning pool*. Second-phase tuning (e.g., CT-SFT or continued full-model tuning) is performed exclusively on this pool to preserve a strict holdout between mechanism discovery and subsequent tuning.

- **Validation (no optimization).** The validation split (**100** examples) is used only for diagnostics (e.g., circuit faithfulness).

- **Test (final evaluation).** The test split (**400** examples) is used only for final reporting and is never used for optimization or selection.

B Circuit Selection

We follow the original CD-T (Hsu et al., 2025) algorithm to build the circuit by expanding backward from the model output. We maintain a *frontier* set of target heads, denoted $\mathcal{T}^{(d)}$. At iteration $d = 0$, the target is the output readout (logit layer), and we score every attention head by how strongly its CD-T relevant contribution supports the correct label direction. We average each head's score across discovery inputs x and select the top K_0 heads to form the first circuit set $\mathcal{C}^{(0)}$.

For iterations $d \geq 1$, the frontier targets become the heads selected in the previous iteration, i.e., $\mathcal{T}^{(d)} = \mathcal{C}^{(d-1)}$. For each candidate head, we compute its directional relevance to each target head in $\mathcal{T}^{(d)}$, average across targets, and also average across discovery inputs x. We then select the top

Target	Comp. Tuning (50 IND)	$n = 25$			$n = 50$			$n = 75$			$n = 100$		
		Direct FT	Full FT	CT-SFT	Direct FT	Full FT	CT-SFT	Direct FT	Full FT	CT-SFT	Direct FT	Full FT	CT-SFT
ace	0.471	0.441	0.346	**0.486**	0.453	0.368	**0.482**	**0.535**	0.470	0.530	0.537	0.428	**0.547**
bug	0.366	**0.417**	0.376	0.402	0.372	0.351	**0.421**	**0.492**	0.464	0.478	**0.516**	0.454	0.493
jav	0.581	0.377	0.384	**0.592**	0.405	0.372	**0.591**	0.562	0.413	**0.629**	0.550	0.475	**0.636**
min	0.554	0.440	0.394	**0.568**	0.448	0.533	**0.582**	**0.637**	0.505	0.601	**0.693**	0.510	0.612

Table 4: **Cross-lingual transfer baselines and CT-SFT summary (test accuracy). Competence tuning** evaluates the Indonesian-tuned checkpoint directly on each target language with no continued fine-tuning. **Direct FT** fine-tunes the base model directly on the target language using n samples. **Full FT** continues full-model tuning from the Indonesian competence-tuned checkpoint using n target-language samples. **CT-SFT (d=2)** reports circuit-targeted tuning from the same competence-tuned checkpoint using discovery depth d=2, taking the better of the mechanism-editing and mechanism-preserving update scopes. All numbers are mean test accuracy over 4 seeds. **Bold** indicates the best method among {Direct FT, Full FT, CT-SFT} for each n and target language.

Seed	Depth 0	Depth 1	Depth 2	Depth 3
31	23–23	0–21	0–2	0–1
777	23–23	0–20	0–1	0–0
12345	23–23	0–20	0–1	0–0
2025	16–23	3–16	0–4	0–2

Table 5: Layer range (min–max) covered by the accumulated circuit after each expansion depth.

K_d heads to form $\mathcal{C}^{(d)}$, add them to the circuit, and continue expanding backward. The final circuit is the union of selected heads across iterations, $\mathcal{C} = \bigcup_d \mathcal{C}^{(d)}$.

We stop when the expansion reaches shallow layers. This typically happens within three iterations (Table 5), so we cap expansion at three iterations in all experiments.

C Cross-Lingual Transfer Results

C.1 Full cross-lingual transfer results (NusaX Dataset)

We report the full cross-lingual transfer curves by varying the Stage-2 tuning size $n \in \{25, 50, 75, 100\}$ and circuit expansion depth (`max_depth` $= 0, 1, 2$). Figure 4 shows that the relative ordering of head-selection strategies is consistent across n, complementing the $n = 100$ summary reported in the main paper.

We report absolute baseline performance (Stage-1 evaluation, Direct FT from the base model, and continued Full FT from the Indonesian Stage-1 checkpoint) alongside CT-SFT in a single anchor table (Table 4). Full FT does not depend on circuit depth, so including it directly in depth-wise transfer plots would compress the y-axis and obscure differences among the head-restricted methods. We therefore keep the plots focused on CT-SFT variants and controls, while Table 4 provides the corresponding baseline reference values for each target language and tuning size.

Defining "easy" vs. "hard" transfer. We operationalize transfer difficulty using the *competence-tuning* baseline $A_0(\ell)$: the accuracy of the Indonesian-tuned checkpoint evaluated directly on target language ℓ with no continued tuning (Table 4). Lower $A_0(\ell)$ indicates a larger mismatch between the Indonesian-adapted mechanism and the target-language distribution, and thus a harder transfer regime. Under this definition, Acehnese (A_0=0.471) and Buginese (A_0=0.366) are harder transfers than Javanese (A_0=0.581) and Minangkabau (A_0=0.554). This stratification is consistent with the editing–preserving trade-off observed across tuning sizes: targets with lower $A_0(\ell)$ tend to benefit more from *mechanism-editing* updates to decision-aligned heads (Circuit), whereas targets with higher $A_0(\ell)$ more often remain competitive under *mechanism-preserving* updates that keep the Indonesian circuit largely intact while adapting through weakly aligned components (NearZero).

C.2 Full cross-lingual transfer results (XNLI Dataset)

We discover the NLI circuit on an English competence-tuned checkpoint and transfer it to four target languages: Thai (th), Vietnamese (vi), Spanish (es), and Chinese (zh). Unless stated otherwise, the plots in Figure 5 report the main setting with 250 English samples for competence tuning, where the checkpoint is task-competent and CT-SFT yields stable transfer curves across depths.

To highlight the competence requirement, Table 6 additionally summarizes the same transfer protocol under a weaker 50-sample English competence-tuning checkpoint. This setting is in-

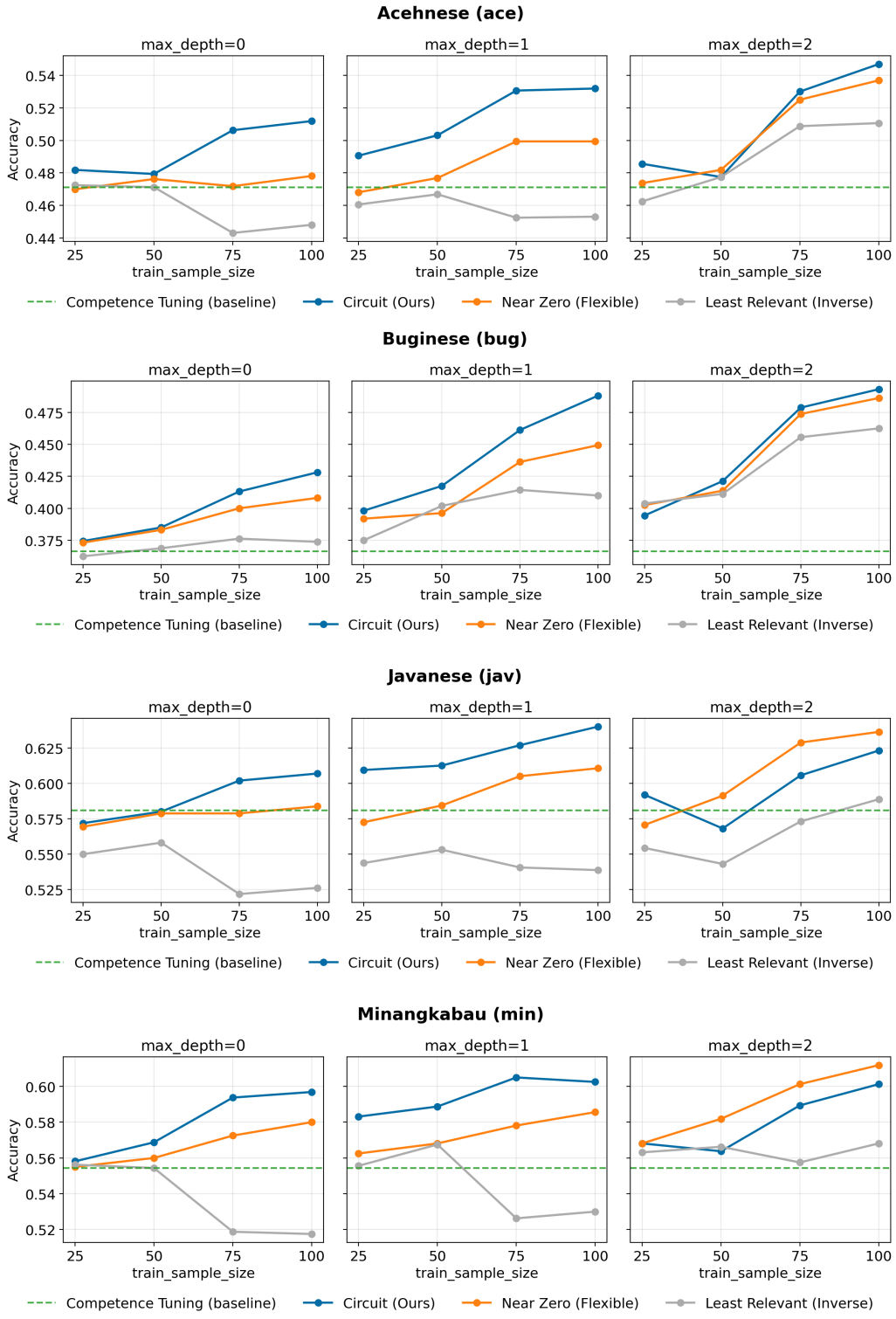


Figure 4: Cross-lingual NusaX transfer (Indonesian $\rightarrow$ target): test accuracy vs. target-language tuning size n, shown per target language. Each panel contains three subplots corresponding to circuit expansion depth (`max_depth` $= 0, 1, 2$) under the surgical setting, comparing Circuit CT-SFT against control head selections. Dashed horizontal lines indicate the competence-tuning baseline (Indonesian-tuned checkpoint evaluated directly on the target language, with no Stage-2 tuning).

Target	Comp. Tuning (50 ENG)	$n = 25$			$n = 50$			$n = 75$			$n = 100$		
		Full FT	Circuit	NearZero	Full FT	Circuit	NearZero	Full FT	Circuit	NearZero	Full FT	Circuit	NearZero
th	0.284	**0.346**	0.298	0.288	**0.310**	0.291	0.282	**0.344**	0.276	0.276	**0.373**	0.271	0.278
vi	0.412	0.338	0.407	**0.436**	0.356	0.384	**0.416**	0.371	0.392	0.402	0.402	0.386	**0.412**
es	0.441	0.359	0.426	**0.450**	0.333	0.434	**0.456**	0.362	0.417	**0.468**	0.409	0.423	**0.479**
zh	0.465	0.357	0.42	0.427	0.364	0.398	0.438	0.439	0.386	0.436	0.434	0.394	0.441

Table 6: **XNLI cross-lingual transfer under weak source competence (50-sample English competence tuning). Competence tuning** evaluates the 50-sample English-tuned checkpoint directly on each target language with no continued fine-tuning. **Full FT** continues full-model tuning from the English competence-tuned checkpoint using n target-language samples. **Circuit** updates only the discovered circuit heads (plus LayerNorm) under the surgical setting. **NearZero** updates the same number of heads selected by near-zero relevance score (plus LayerNorm) under the surgical setting. All mechanism settings using discovery depth d=2. All numbers are mean test accuracy over 4 seeds.

cluded primarily as a diagnostic: when the starting NLI competence is low, head-restricted updates become substantially less reliable and can degrade performance, especially at larger circuit depths (cf. the main-paper discussion in Section 6.3).

Catastrophic forgetting on the source language (ENG). To quantify how much continued cross-lingual tuning drifts away from the source NLI mechanism, we measure *English retention*: after Stage-2 tuning on each target language, we re-evaluate the resulting checkpoint on ENG. Table 7 reports this forgetting diagnostic for the main setting (250-sample English competence tuning), directly mirroring the Indonesian retention analysis for NusaX (Table 3).

D Additional Diagnostic Analysis

D.1 Faithfulness Evaluation via Mean Ablation

We evaluate circuit *faithfulness* using a mean-ablation protocol consistent with CD-T (Hsu et al., 2025). The goal is to quantify how well a discovered circuit preserves the full model's behavior when computation outside the circuit is replaced by a neutral baseline.

Mean baseline (γ). For each attention head h, we define a baseline activation μ_h as the mean head output computed over a small label-balanced set of correctly predicted examples. Following the CD-T decomposition viewpoint, this mean corresponds to the γ ("irrelevant") baseline stream, while deviations from it represent input-specific signal (Hsu et al., 2025). In our experiments, this baseline set contains 50 examples due to label-balance and memory constraints.

Circuit-only forward pass (mean ablation). Given a circuit $\mathcal{C}$, we construct a circuit-only model by preserving activations for heads in $\mathcal{C}$ and mean-ablating all other heads: heads outside $\mathcal{C}$ have their output activations replaced by μ_h. This removes input-dependent signal outside the circuit while keeping non-circuit components at a neutral reference level.

Faithfulness metrics. We compute faithfulness on the validation split to avoid using the test set for diagnostic selection. We report **accuracy-based faithfulness** (main paper) and additionally provide a **margin-based faithfulness** variant (appendix) that measures preservation of the logit margin between the correct label and the alternatives. Both metrics are reported as percentages and aggregated across random seeds for each circuit expansion depth.

Aggregation across seeds and depth. For each scoring method (magnitude-based vs. task-direction projection), we compute faithfulness at each circuit expansion depth and average the resulting values across seeds under matched experimental settings. This produces the faithfulness-versus-depth curves shown in Figure 3 (accuracy) and Figure 6 (margin).

D.2 Selection-ratio sweep for circuit construction (2%, 5%, 10%)

We study the effect of the per-iteration selection ratio used during circuit expansion. Recall that at each CD-T iteration we retain the top-p fraction of heads by aggregated relevance score, where the main paper uses p=2% to enforce sparsity. We compare $p \in \{2\%, 5\%, 10\%\}$ across circuit expansion depths $\texttt{max\_depth} \in \{0, 1, 2\}$.

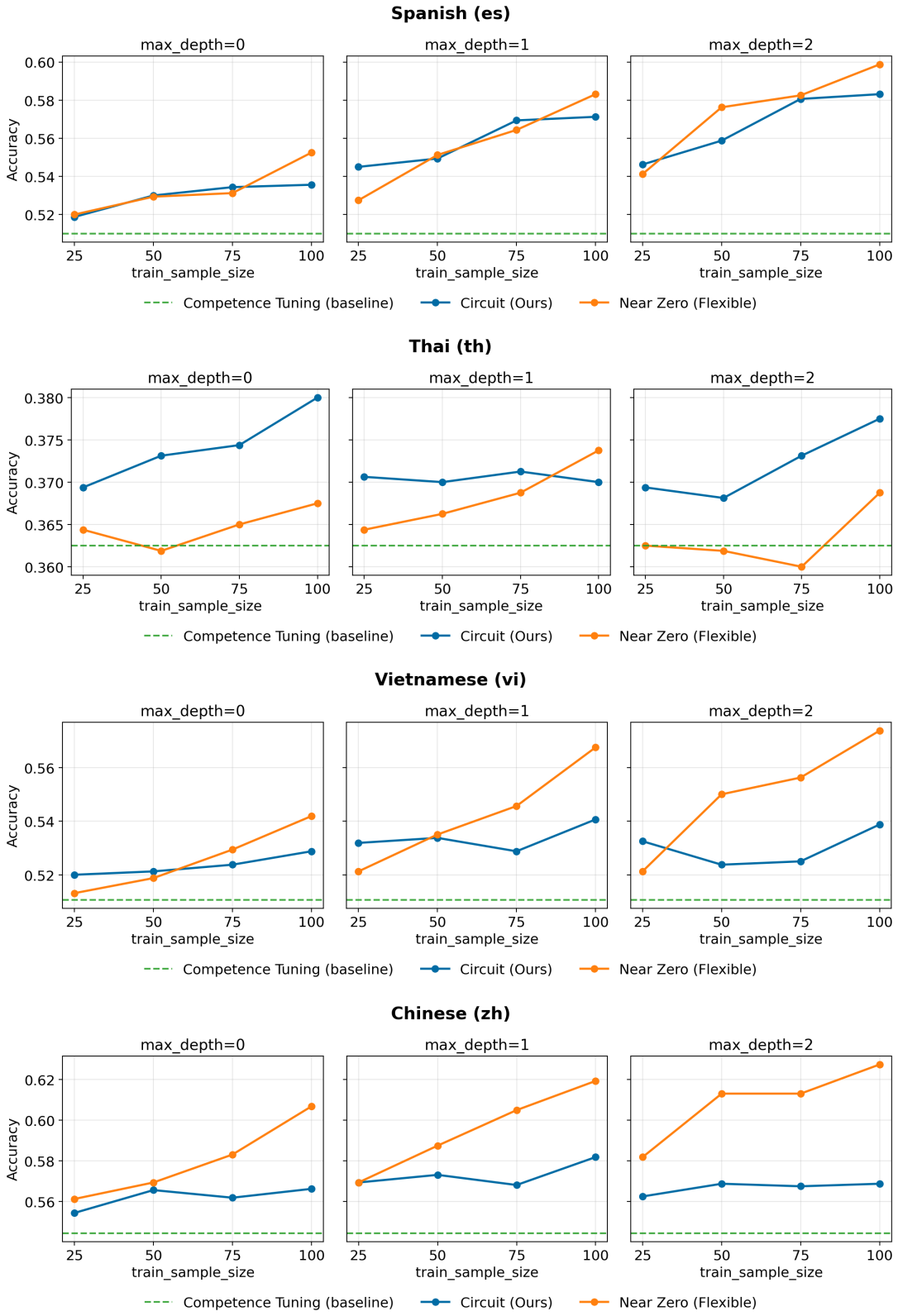


Figure 5: XNLI cross-lingual transfer results (surgical setting; 250-sample English competence tuning): test accuracy vs. tuning set size n across circuit depths. We compare circuit-targeted updates to near-zero head updates for multiple target languages.

Target (tune)	Comp. Tuning on ENG (250 samples)	n=25 (eval on ENG)			n=100 (eval on ENG)		
		Direct FT	Full FT	**CT-SFT**	Direct FT	Full FT	**CT-SFT**
th	0.574	0.291	0.354	**0.597**	0.411	0.412	**0.611**
vi	0.574	0.400	0.344	**0.583**	**0.607**	0.357	0.595
es	0.574	0.316	0.364	**0.591**	0.589	0.496	**0.614**
zh	0.574	0.507	0.331	**0.591**	0.494	0.494	**0.607**

Table 7: **English retention after continued fine-tuning (forgetting diagnostic).** Starting from the 250-sample English competence-tuned checkpoint, we continue tuning on each target language (row) and evaluate on ENG to measure source-language retention. *Direct FT* fine-tunes from the base model directly on the target language (no English competence tuning), then evaluates on ENG. *Full FT* continues full-model tuning from the English competence-tuned checkpoint. *CT-SFT* performs circuit-targeted tuning from the same checkpoint using discovery depth d=2 (surgical setting) of mechanism-editing (Circuit). All numbers are mean test accuracy over 4 seeds.

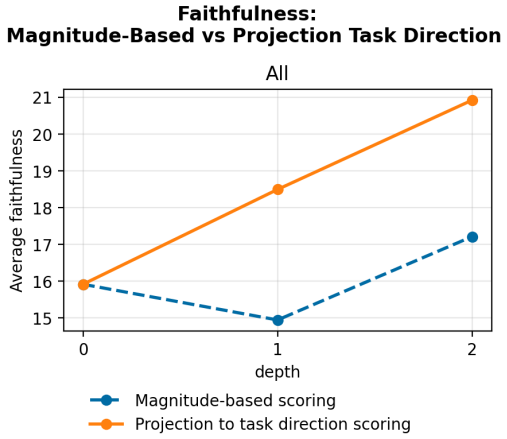


Figure 6: Faithfulness (margin-based). We report the same comparison as Figure 3, but using logit-margin preservation. The qualitative trend matches the accuracy-based metric.

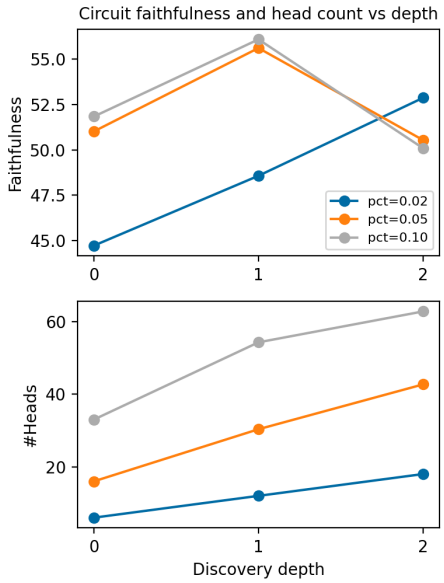


Figure 7: **Circuit faithfulness–size trade-off under different selection ratios.** We vary the per-iteration selection ratio $p \in \{2\%, 5\%, 10\%\}$ and circuit expansion depth `max_depth` $\in \{0, 1, 2\}$. Top: circuit faithfulness. Bottom: circuit size (#heads). Larger p increases circuit size rapidly, but faithfulness shows diminishing returns at higher depth.

Circuit size grows rapidly with larger p. Figure 7 shows that increasing p substantially enlarges the circuit at every depth, with the gap widening as depth increases. This is expected: larger ratios admit a broader set of upstream heads during iterative expansion, leading to fast accumulation of components.

Faithfulness exhibits diminishing returns at larger p. While increasing p can improve faithfulness at intermediate depth (e.g., depth 1), the gains do not persist at depth 2: faithfulness can stagnate or decrease even as the circuit becomes much larger. This pattern is consistent with larger p admitting low-signal heads that dilute the task-relevant computation, making the circuit less faithful despite higher capacity.

D.3 Shared vs. Held-out Stage-1 Pool for Circuit Discovery (Iteration 0 Stability)

A critical dependency in CD-T is the precompute-mean μ, which determines what is treated as baseline (γ) versus input-induced deviation (β). In our low-resource, unstructured setting, μ must be estimated from a small label-balanced set, so the discovery protocol can substantially affect both mean stability and early (iteration-0) head ranking.

Rather than constructing the baseline mean μ and the circuit-discovery inputs from fully disjoint data sources, we intentionally draw (i) the Stage-1

	Shared pool (S1 + CD)		Held-out CD subset	
Seed	Top-1 (L,H)	Score	Top-1 (L,H)	Score
31	(23,6)	3.165	(0,11)	0.235
777	(23,7)	0.696	(23,11)	0.033
2025	(23,7)	0.179	(23,7)	0.059
12345	(23,1)	1.951	(23,7)	0.368

Table 8: Iteration-0 (depth=0) top-ranked head under two discovery protocols using Stage-1 finetuning size $= 50$.

fine-tuning examples, (ii) the balanced-mean pool used to estimate μ, and (iii) the circuit-discovery set from the same underlying training pool. This design choice is primarily practical and stability-driven: label-balanced mean estimation is constrained by the class distribution in low-resource splits, and using a separate, disjoint pool can reduce the availability of balanced samples and increase variance in μ.

Importantly, this shared-pool choice is not merely about allowing overlap between the mean pool and discovery set; it is about ensuring that μ and the head-relevance estimates are computed from the same in-distribution pool as the checkpoint's Stage-1 adaptation. This yields more stable circuits (especially at early iterations) and avoids degenerate selections that can arise when the baseline and discovery inputs are drawn from different pools. Concretely, 'shared-pool' refers to using Pool A for Stage-1 fine-tuning, μ estimation, and CD-T discovery; Pool B remains disjoint and is used only for Stage-2 tuning.

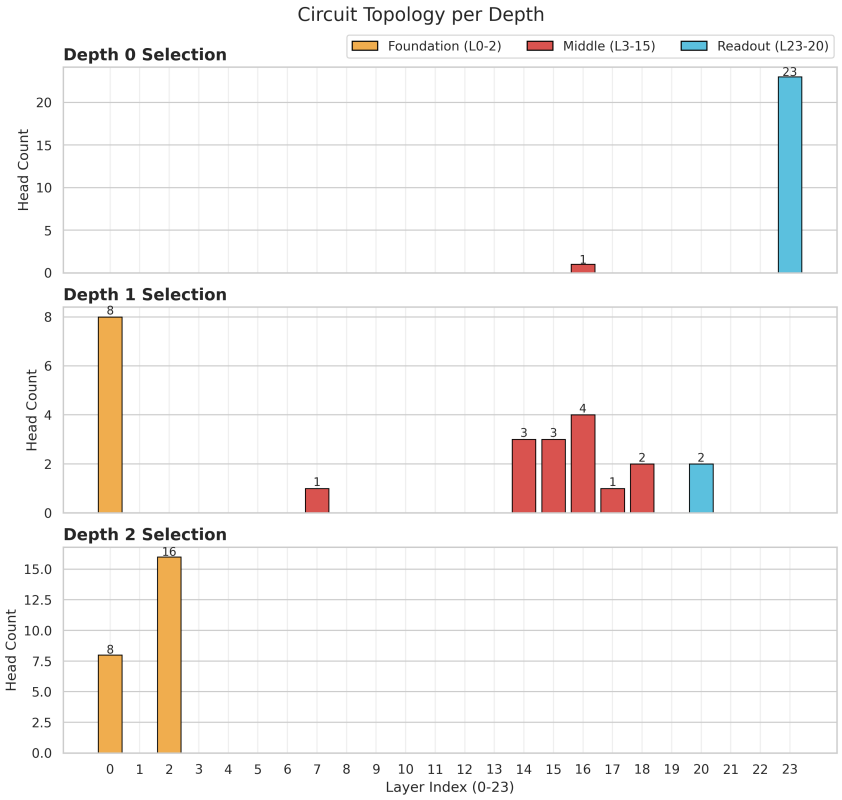
We observe in our low-resource setting that CD-T's *iteration-0* selection is more stable when circuit discovery is performed on the same pool used for Stage-1 fine-tuning, rather than on a held-out subset from that pool. We therefore compare two protocols:

- **Shared Stage-1 pool.** Stage-1 fine-tuning, μ estimation, and CD-T discovery inputs are all drawn from the same Stage-1 training pool.
- **Held-out discovery subset.** The Stage-1 training pool is partitioned into disjoint subsets: one subset is used for Stage-1 fine-tuning, while the other is used for μ estimation and CD-T discovery inputs.

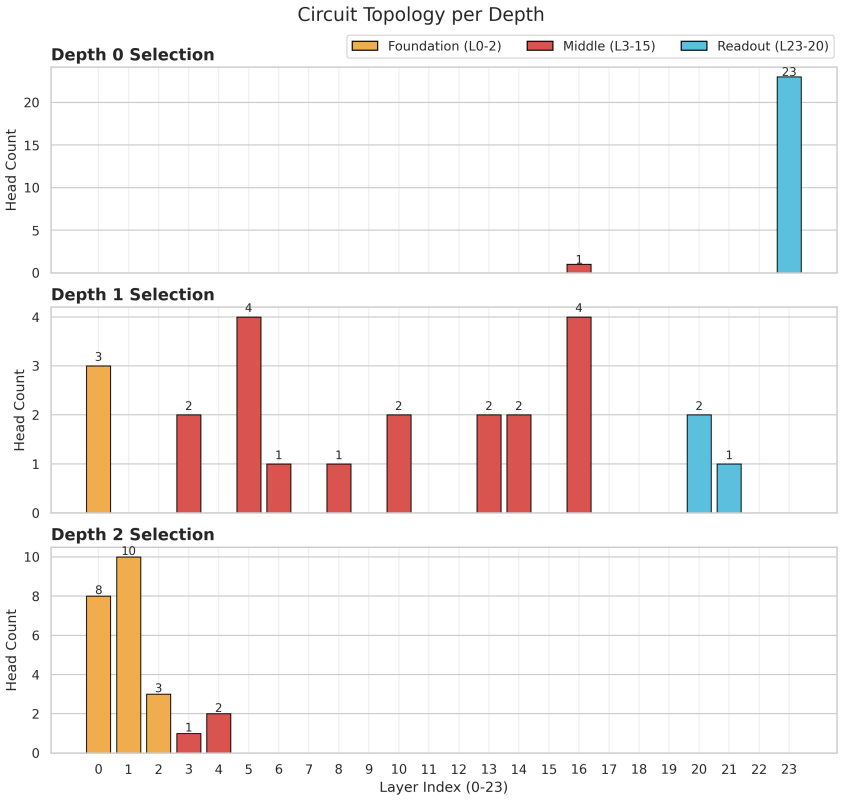
Across random seeds on NusaX, CD-T's *iteration-0* (depth $= 0$) ranking is noticeably more stable under the shared-pool protocol. Under the held-out protocol, the top-ranked component at iteration 0 can intermittently collapse to shallow (including layer-0) heads, and the relevance scores are markedly smaller. We summarize the layer index of the top-ranked head per seed under each protocol in Table 8. This suggests that, in low-resource settings, small subset-level shifts (e.g., lexical composition, label-frequency fluctuations, or difficulty/confidence differences) can weaken the task-aligned deviation signal used at iteration 0, leading to less reliable circuit identification.

D.4 Circuit Topology

We analyze where CD-T-selected heads are located across layers as the circuit expansion depth increases. This provides a coarse view of circuit topology, helping us understand whether a scoring rule tends to emphasize early "foundation" components (often tied to lexical/positional variation) versus deeper, more decision-adjacent computation. Figure 8 compares the layer distribution of selected heads under magnitude-based scoring and task-direction projection scoring.



(a) Magnitude-based scoring.



(b) Task-direction projection scoring.

Figure 8: Circuit topology per depth under magnitude-based scoring (top) and task-direction projection scoring (bottom). Magnitude-based scoring concentrates selected heads in early foundation layers (L0–2), whereas task-direction projection selects more heads from mid layers (L3–15) across depths, indicating reduced reliance on shallow lexical/embedding variation and greater emphasis on task-relevant computation.