WHAT IS REUSABLE ACROSS LORA RUNS? THE BASE-CONDITIONED CORE AS A RELATIONAL UNIT

Anonymous authors
Paper under double-blind review

ABSTRACT

PEFT makes task adaptation modular: a small LoRA update can be trained once, stored in an adapter library, and transferred or composed without retraining the full model. This workflow raises a practical question: when independently trained updates move to a new run, what preserves their function? We answer that reuse is carried by a base-conditioned core–base relation jointly determined by the donor–target frame relation, the learned core, and the target base. We introduce Bilateral Target-Frame Functional Registration (BTF-FR), which expresses a donor core in the target frame and evaluates the resulting dense intervention in the target model. Functional controls show that registered interventions recover same-task behavior, while raw splicing and spectrum- or norm-matched controls reveal the specific contribution of registration; related task and interface conditions produce an ordered response. BTF-FR therefore provides a functional and auditable criterion for organizing cross-run adapter reuse around donor–target relations and target bases.

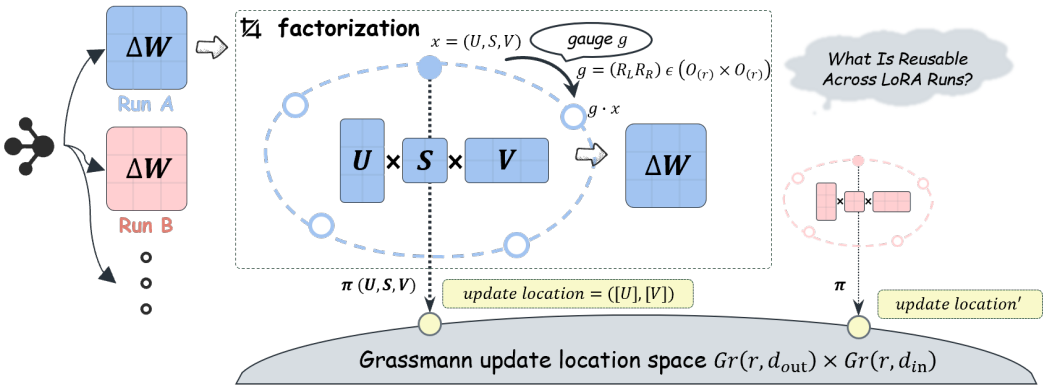


Figure 1: Problem definition across LoRA runs. Each run produces a dense update and can be represented by multiple equivalent (U, S, V) triples under an internal orthogonal gauge. Its left and right column spaces define an update location $b = ([U], [V])$ in a Grassmann space. Conditioning S on this update location makes cross-run comparison explicit; the central question is whether the reusable object across runs is a raw adapter, a registered core, or a target-base-conditioned core–base functional relation.

1 INTRODUCTION

Parameter-efficient fine-tuning (PEFT) makes adaptation modular: most pretrained weights remain fixed while a small task-specific module is learned and can be stored, transferred, or composed (Houlsby et al., 2019; Pfeiffer et al., 2020; Li & Liang, 2021; Lester et al., 2021). Frameworks such as AdapterHub and LoRA Hub operationalize this modularity (Pfeiffer et al., 2020; Huang et al., 2024), while Low-Rank Adaptation (LoRA) provides a compact way to package task

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.

adaptation as a low-rank weight update (Hu et al., 2022). The practical promise is substantial: a capability learned once could be moved across runs and tasks, reducing retraining cost and enabling modular reuse and composition. We test this promise directly. Raw splicing can fail to recover stable task selectivity even when donor and target adapters were trained on the same task, suggesting that functional reuse cannot be reduced to copying an isolated update and instead depends on its relation to the target model.

This leads to a direct research question: *What is reusable?* Specifically, we ask whether the reusable object across runs should be defined as a raw adapter, a normalized core, or a functional relation conditioned on the target base. This question yields a falsifiable prediction: after target-frame registration, same-task relations should recover the most target functionality; relations with the same output interface but different tasks should retain an intermediate response; and relations with different tasks and interfaces should approach Zero-Core. The prediction turns "what is reusable?" into a measurable donor–target compatibility relation rather than a similarity between isolated parameters.

We formulate cross-run reuse as a gauge-aware functional registration problem. Given a donor and a target run, Bilateral Target-Frame Functional Registration (BTF-FR) estimates orthogonal transports between their left and right frames, expresses the donor core in target coordinates, and reconstructs the intervention in the target base. The functional object is the pair $(W_A, \mathcal{R}_{A\leftarrow B})$: the relation records the registration, while W_A supplies the target context for its functional readout. This yields a testable prediction: unilateral transport should recover most same-task functionality, while bilateral registration should add gauge invariance, left–right symmetry, and auditability. Functional experiments support this prediction: registered transport recovers same-task functionality, raw splicing and spectrum- or norm-matched controls remain nonselective, and geometry and base-swap controls show that functional recovery depends on more than proximity alone. BTF-FR therefore provides a gauge-invariant measurement under the stated bilateral orthogonal transformations and its dependence on the donor–target relation.

Our contributions are threefold. **First, testing the modularity promise of PEFT.** We turn the practical assumption that adapters can be directly reused into a functional cross-run test and show that the reusable object is not an isolated matrix detached from context, but a relational object determined by the update location, internal mapping, and target base. **Second, establishing a functional measurement protocol.** We construct bilateral target-frame functional registration (BTF-FR) as a measurement method that expresses a donor core in the target frame and intervenes in the target base. The intervention controls distinguish coordinate mismatch, geometric compatibility, target-base effects, and learned coupling. **Third, characterizing conditional functional reuse.** We show that unilateral transport already recovers most functionality while bilateral registration adds gauge invariance and auditability. Functional recovery forms a hierarchy across core–base matching relations and is validated across a second model family.

2 Related Work

Parameter-efficient fine-tuning, LoRA, and geometric representations. Parameter-efficient fine-tuning reduces adaptation and deployment costs by freezing most pretrained parameters and learning small task-specific modules. Inserted adapters and unified management frameworks established the practice of storing and reusing task capabilities as modules (Houlsby et al., 2019; Pfeiffer et al., 2020); prefix/prompt tuning and other lightweight-parameter methods extended this paradigm (Li & Liang, 2021; Lester et al., 2021; Liu et al., 2022). LoRA implements parameter-efficient adaptation through low-rank updates $W_0 + BA$ (Hu et al., 2022), and its expressive power, low-dimensional structure, SVD decompositions, and orthogonal representations have been widely studied (Aghajanyan et al., 2021; Ding et al., 2023; Meng et al., 2024; Liu et al., 2024). These studies clarify how low-rank updates are represented and optimized, leaving the cross-run coordinate relation and the functional content retained after registration as open questions.

Cross-run alignment and model merging. Independent training runs motivate explicit coordinate alignment before parameter combination. Git Re-Basin aligns models through permutation-symmetry matching before merging, while ZipIt! and task-parameter-subspace matching address partial feature sharing and cross-task correspondences (Ainsworth et al., 2023; Stoica et al., 2024; Tam et al., 2024). For LoRA, KnOTS uses SVD to transform multiple updates into a shared repre-

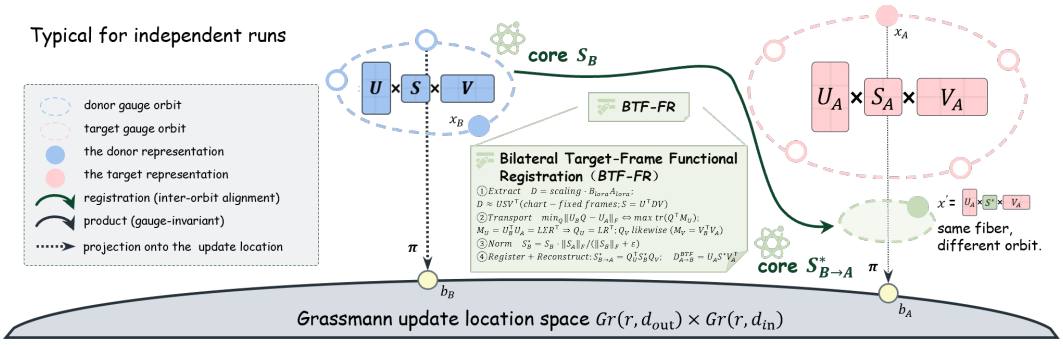


Figure 2: Overview of Bilateral Target-Frame Functional Registration (BTF-FR). The donor core S_B is extracted in a fixed frame, transported to the target frame by left and right Procrustes maps Q_U, Q_V, norm-adjusted, and reconstructed at the target update location $b_A = ([U_A], [V_A])$ as $D^{\text{BTF}}_{A\leftarrow B} = U_A S^*_{B\to A} V_A^\top$, where $S^*_{B\to A} = Q_U^\top S^*_B Q_V$. Dashed lines denote equivalent representations of the same update, green paths denote registration between frames, and dotted projections show the update locations in the Grassmann space.

sentation before applying existing merge rules, while GAM performs bilateral orthogonal Procrustes alignment of left and right singular bases (Stoica et al., 2025; Sun et al., 2026). This line of work highlights the importance of coordinate relations and primarily evaluates aligned weights or the task performance of merged models. We extend this geometric perspective by treating the registration relation itself as an auditable object and measuring the dense intervention reconstructed in the target frame.

Adapter composition, retrieval, and routing. AdapterFusion, LoRAHub, and LoraRetriever study adapter composition, storage, retrieval, and dynamic selection (Pfeiffer et al., 2021; Huang et al., 2024; Zhao et al., 2024); LoRA-LEGO treats rank-wise components as finer-grained modular units (Zhao et al., 2025). Task-vector and parameter-composition methods likewise reuse fine-tuned capabilities through vector operations and conflict handling (Ilharco et al., 2023; Yadav et al., 2023). These methods primarily address how to combine or select existing modules. Before composition, however, a basic question remains: *What is reusable?* We turn it into a functional criterion: after target-frame registration, can a donor core recover same-task functionality on a particular target base? This criterion separates the reusable relation jointly determined by internal coordinates, the target base, and learned coupling.

3 Base-conditioned Cores and Functional Measurement

For one target LoRA module, let A denote the target run, B the donor run, and W_A the target base. The functional object is the pair $(W_A, \mathcal{R}_{A\leftarrow B})$, where the auditable registration record is

$$\mathcal{R}_{A\leftarrow B} = \big(U_A, V_A, Q_U, Q_V, S^*_B\big), \qquad \Phi(\mathcal{R}_{A\leftarrow B}) = U_A Q_U^\top S^*_B Q_V V_A^\top. \tag{1}$$

Here $\Phi(\mathcal{R}_{A\leftarrow B})$ is the dense intervention written into W_A, and $M_A(D)$ denotes target-A Macro-F1 after injecting D into that base. Thus the record and its functional readout are distinct but linked. We call the resulting measurement method Bilateral Target-Frame Functional Registration (BTF-FR). Its validity is guaranteed by Proposition 1, while its measurement target is separated by the controls in the experimental design.

Figure 2 summarizes the registration procedure: we first extract the core from the donor update in a fixed frame, solve orthogonal donor-to-target transports on the output and input sides, apply a prespecified norm rule to obtain S^*_B, and reconstruct and write the dense intervention in the target base. The gauge orbit represents equivalent internal representations; the two points in the Grassmann space record the donor and target update locations; and the green path denotes registration across representations rather than arbitrary replacement of a raw core.

3.1 Base–core Representation and Gauge Freedom

For layer l, task t, and training run s, the effective PEFT update is

$$\Delta W_{l,t,s} = \lambda_{l,t,s} B_{l,t,s} A_{l,t,s}, \qquad \lambda_{l,t,s} = \frac{\alpha}{r}, \tag{2}$$

where λ can also be obtained from the runtime scaling used by the implementation. For each LoRA target module, we retain its numerical rank-r reconstruction and explicitly separate the original effective update from reconstruction error:

$$\Delta W_i = D_i^{(r)} + E_{\mathrm{svd},i}, \qquad D_i^{(r)} = U_i S_i V_i^\top, \qquad U_i^\top U_i = V_i^\top V_i = I_r. \tag{3}$$

$D_i^{(r)}$ is the rank-r donor/target reconstruction used for registration, and $E_{\mathrm{svd},i}$ is the factorization error between the original effective LoRA update and this reconstruction. All registration formulas below are applied independently to each module, so module indices are omitted for readability.

The following five quantities must be distinguished: the donor's left and right frames (U_B, V_B), the donor core S_B, the target frame (U_A, V_A), the bilateral transport matrices (Q_U, Q_V), and the norm-adjusted core S_B^*. They represent, respectively, the update location, the internal mapping, the target coordinates, the coordinate transformation, and scale calibration. The column spaces of U_i and V_i define the subspace location

$$\mathcal{B}_i = \big(\mathrm{Col}(U_i), \mathrm{Col}(V_i)\big), \tag{4}$$

while $S_i \in \mathbb{R}^{r\times r}$ describes the full linear mapping between the two frames. S_i is not restricted to be diagonal; recomputing $S_i = U_i^\top \Delta W_i V_i$ preserves all left–right directional couplings.

With the subspace location $b = ([U], [V])$ fixed, the updates generated by different cores form the local fiber

$$\mathcal{F}_b = \{USV^\top : S \in \mathbb{R}^{r\times r},\ U \text{ spans } [U],\ V \text{ spans } [V]\}. \tag{5}$$

After selecting a particular frame (U, V), S is the matrix coordinate within this local fiber. For $R_L, R_R \in \mathrm{O}(r)$,

$$(U, S, V) \mapsto (UR_L, R_L^\top S R_R, V R_R) \tag{6}$$

leaves $USV^\top$ unchanged. This is the internal gauge freedom: individual core entries depend on the frame, while the complete update and its functional output do not. A formal Stiefel/Grassmann treatment and the gauge-equivalence class are given in Appendix A.

3.2 Bilateral Target-Frame Transport

Let A denote the target run and B the donor run. The left orthogonal transport is defined by

$$Q_U = \underset{Q^\top Q = I_r}{\arg\min} \|U_B Q - U_A\|_F^2 . \tag{7}$$

If $U_B^\top U_A = L_U \Sigma_U R_U^\top$, its closed-form solution is $Q_U = L_U R_U^\top$. Similarly, $V_B^\top V_A = L_V \Sigma_V R_V^\top$ gives $Q_V = L_V R_V^\top$. The two transports describe output-side and input-side frame relations and are generally different.

The donor rank-r core expressed in target internal coordinates is

$$S_{B\to A} = Q_U^\top S_B Q_V. \tag{8}$$

After the prescribed norm matching, write $S^*_{B\to A} = Q_U^\top S_B^* Q_V$. The main intervention fixes the target base and injects the transported donor core:

$$D^{\mathrm{BTF}}_{A\leftarrow B} = U_A Q_U^\top S_B^* Q_V V_A^\top, \tag{9}$$

where S_B^* follows a prespecified norm rule applied independently to each LoRA module. Under target-norm matching,

$$S^*_{B,l} = c_{A,B,l} S_{B,l}, \qquad c_{A,B,l} = \frac{\|S_{A,l}\|_F}{\|S_{B,l}\|_F + \varepsilon}, \qquad \varepsilon = 10^{-12}.$$

When the left and right subspaces coincide and no norm scaling is applied, $S_B^* = S_B$ and Eq. (9) exactly equals the donor rank-r reconstruction $D_B^{(r)}$; with $E_{\mathrm{svd},B}$, this equality holds relative to the original effective update up to factorization error. Under the main target-norm rule, BTF is a per-module rescaled version of that rank-r reconstruction. When the subspaces only partially overlap, BTF is an intervention that transports and injects the core into the target base rather than an exact recovery of the original dense donor update; the Procrustes residual records the remaining base mismatch.

Proposition 1 (Gauge Invariance of BTF)**.** *Fix the target frame, a gauge-invariant norm rule, and the write-back protocol. After any bilateral orthogonal gauge transformation of the donor and an equivariant recomputation of the Procrustes transports, the dense matrix in Eq.* (9) *and its functional output remain unchanged.*

This proposition is the **validity guarantee** for BTF-FR as a measurement method: if the measurement changed with the gauge choice, it could not support comparison. Proposition 1 guarantees that the registered object's measurement does not depend on the donor's representation. The mathematical role of Q_U, Q_V is to establish comparable coordinates; functional interpretation comes from the transformed dense intervention. Full proofs, degenerate-singular-value cases, and a two-dimensional rotation example appear in Appendix A.5.

4 Experimental Design

The experiments address one central question: after coordinate registration, can independently trained LoRA updates recover same-task functionality in a target model, and how does this recovery depend on the target base? The main protocol fixes the model interface, evaluation data, write-back precision, and statistical unit, while varying the donor-core representation, coordinate transport, and target-base condition.

4.1 Goals and Protocol

The main matrix uses Qwen2.5-1.5B-Instruct and eight heterogeneous SuperNI tasks with a shared four-class single-token output interface. Construction runs provide donor representations, determine the reference frame, and generate fixed controls; confirmatory runs serve as independent targets. Each target uses three same-task donors and seven different-task donors, yielding 400 directed target–donor pairs. All conditions use the same target carrier, evaluation examples, label mapping, write-back rank, and FP32 precision.

To test the generality of the main relation, Qwen2.5-3B, Qwen2.5-7B, and Llama-3.2-1B use the same task interface, LoRA rank, seed split, C0–C8 conditions, and target-level aggregation protocol. The three Qwen2.5 models provide scale extensions, while Llama provides cross-family validation. The downstream classification and recommendation matrix follows a separate protocol described in Appendix C.4.

4.2 Conditions, Functional Measurement, and Statistical Units

Table 1 summarizes the functional conditions. The artifact labels map as $C0 = \mathrm{FT}$, $C1 = \mathrm{Raw\text{-}L}$, $C2 = \mathrm{Raw\text{-}R}$, $C3 = \mathrm{Tr\text{-}L}$, $C4 = \mathrm{Tr\text{-}R}$, $C5 = \mathrm{BTF}$, $C6 = \mathrm{Rand\text{-}Q}$, $C7 = \mathrm{SMRC}$, and $C8 = \mathrm{Zero}$. Full update formulas, norm rules, geometry matching, and base-swap details are given in Appendix B; the main text retains only the role of each condition.

The primary metric is target-level Macro-F1. For each target, we first average same-task and different-task donors separately, then compute the same–wrong difference; task-level summaries use equal task weights and report mean $\pm$ SD. We also report gain relative to Zero-Core. NLL and ECE are direction-reversed so that positive values always favor same-task transfer. Donor rows are not treated as independent repetitions. Bootstrap procedures, geometric covariates, numerical tolerances, and leakage controls are given in Appendix B.

Beyond the main matrix, geometry matching pairs donors using precomputed subspace geometry without reading functional scores; exact-core base-swap fixes the donor core and changes only the

Table 1: Cross-run functional conditions and main results. Same/wrong are target-level Macro-F1 means ± SD; Δ is same–wrong. Full Target is a target-native reference, not a cross-run donor condition.

Condition	Experimental role	Same/wrong F1	Δ
FT	Target-native adaptation reference	–	–
Raw-L	Left raw splice; tests direct splicing	$0.293 \pm 0.137/0.292 \pm 0.140$	0.001 ± 0.035
Raw-R	Right raw splice; tests direct splicing	$0.290 \pm 0.136/0.285 \pm 0.139$	0.006 ± 0.045
Rand-Q	Random orthogonal transport; rules out arbitrary rotations	$0.289 \pm 0.140/0.291 \pm 0.135$	-0.002 ± 0.028
SMRC	Spectrum/norm-matched random core; rules out spectral and magnitude explanations	$0.295 \pm 0.145/0.287 \pm 0.135$	0.007 ± 0.027
Zero	Zero-Core; no-core reference	$0.300 \pm 0.143/0.300 \pm 0.143$	0.000 ± 0.000
Tr-L	Unilateral coordinate transport; isolates left-side alignment	$0.705 \pm 0.166/0.306 \pm 0.158$	0.400 ± 0.197
Tr-R	Unilateral coordinate transport; isolates right-side alignment	$0.720 \pm 0.142/0.322 \pm 0.168$	0.399 ± 0.195
BTF	Bilateral target-frame registration; main condition	$\mathbf{0.712 \pm 0.155/0.321 \pm 0.172}$	$\mathbf{0.391 \pm 0.196}$

target base and corresponding transport. These controls test the contributions of geometric proximity and target-base choice to functional recovery. Downstream classification and recommendation extensions follow their own protocols and are summarized separately.

5 Results

5.1 Registered Relations Carry Functional Selectivity

The functional matrix compares coordinate registration, spectral structure, and task selectivity. Raw splicing, random orthogonal transport, and spectrum/norm-matched random cores remain near zero in same–wrong selectivity; in particular, SMRC preserves the donor's full singular-value spectrum and Frobenius norm but does not recover task selectivity. Functional selectivity therefore appears with the registration relation rather than being determined by the spectrum or overall magnitude of an isolated matrix.

Table 1 jointly reports the role and main result of each condition. Unregistered raw splicing, random orthogonal transport, and spectrum/norm-matched random cores do not produce stable task selectivity, whereas selectivity appears in the frame-registered transport conditions. This comparison positions BTF as a functional measurement of the registration relation. Full Target supplies a target-native reference for the functionality of the same target under its own adaptation. The main analysis uses the raw BTF–Zero-Core gain and same–wrong selectivity, while the normalized recovery ratio ρ is reported as an auxiliary diagnostic in Appendix B.

5.2 Same-task Recovery and Cross-model Validation

As a numerical self-check, the BTF matrix after a gauge rotation has relative matrix error below 10^{-6}, prediction agreement of 1.0, and cumulative maximum logit drift of 3.5×10^{-4}, within the FP32 registration tolerance of 10^{-3}. This numerically confirms the validity guarantee in Proposition 1 and checks the transport direction, orthogonality, and write-back implementation. Table 1 shows that Tr-L, Tr-R, and BTF recover high same-task functionality while wrong-task functionality is substantially lower, with a positive BTF difference for every target. The BTF gain is concentrated on the same-task path: unilateral transport already recovers most functionality, while bilateral BTF expresses both sides in the target frame and provides a symmetric, gauge-invariant, auditable intervention. Accuracy, NLL, and ECE follow the same direction as Macro-F1, with larger variation in NLL (Table 2). Full task-level results are given in Appendix C.1.

Table 2: Task selectivity under standard metrics. Each cell reports same-task/wrong-task mean ± SD on the first line and the directional Δ mean ± SD on the second; positive values favor same-task, with NLL and ECE direction-reversed.

Condition	Accuracy	NLL	ECE
Raw-L	$0.353 \pm 0.113/0.352 \pm 0.112$ $\Delta = 0.001 \pm 0.026$	$1.533 \pm 0.222/1.526 \pm 0.226$ $\Delta = -0.008 \pm 0.093$	$0.192 \pm 0.067/0.189 \pm 0.075$ $\Delta = -0.003 \pm 0.038$
Raw-R	$0.356 \pm 0.103/0.349 \pm 0.109$ $\Delta = 0.007 \pm 0.032$	$1.640 \pm 0.362/1.565 \pm 0.235$ $\Delta = -0.075 \pm 0.234$	$0.218 \pm 0.103/0.203 \pm 0.077$ $\Delta = -0.015 \pm 0.067$
Rand-Q	$0.351 \pm 0.109/0.351 \pm 0.108$ $\Delta = 0.000 \pm 0.021$	$1.555 \pm 0.256/1.537 \pm 0.239$ $\Delta = -0.018 \pm 0.073$	$0.196 \pm 0.089/0.193 \pm 0.080$ $\Delta = -0.003 \pm 0.030$
SMRC	$0.355 \pm 0.115/0.350 \pm 0.107$ $\Delta = 0.005 \pm 0.019$	$1.520 \pm 0.240/1.541 \pm 0.229$ $\Delta = 0.022 \pm 0.094$	$0.187 \pm 0.079/0.190 \pm 0.074$ $\Delta = 0.003 \pm 0.039$
Zero	$0.360 \pm 0.118/0.360 \pm 0.118$ $\Delta = 0.000 \pm 0.000$	$1.510 \pm 0.226/1.510 \pm 0.226$ $\Delta = 0.000 \pm 0.000$	$0.173 \pm 0.079/0.173 \pm 0.080$ $\Delta = 0.000 \pm 0.000$
Tr-L	$0.712 \pm 0.151/0.361 \pm 0.125$ $\Delta = 0.350 \pm 0.180$	$1.709 \pm 0.912/2.160 \pm 0.494$ $\Delta = 0.451 \pm 1.114$	$0.201 \pm 0.094/0.338 \pm 0.083$ $\Delta = 0.137 \pm 0.133$
Tr-R	$0.723 \pm 0.138/0.383 \pm 0.131$ $\Delta = 0.340 \pm 0.174$	$1.544 \pm 0.741/1.893 \pm 0.356$ $\Delta = 0.350 \pm 0.901$	$0.178 \pm 0.082/0.298 \pm 0.071$ $\Delta = 0.120 \pm 0.123$
BTF	$\mathbf{0.717 \pm 0.146/0.385 \pm 0.133}$ $\mathbf{\Delta = 0.331 \pm 0.173}$	$\mathbf{1.494 \pm 0.678/1.859 \pm 0.482}$ $\mathbf{\Delta = 0.365 \pm 0.939}$	$\mathbf{0.183 \pm 0.081/0.293 \pm 0.094}$ $\mathbf{\Delta = 0.110 \pm 0.134}$

Table 3: BTF functional metrics across Qwen2.5 model scales. S/W denote same-task/wrong-task; Adv. is the directional advantage (mean ± SD), with NLL and ECE reversed so positive values favor same-task.

	Accuracy			NLL			ECE		
Model	S	W	Adv.	S	W	Adv.	S	W	Adv.
1.5B	0.717 ± 0.146	0.385 ± 0.133	0.331 ± 0.173	1.494 ± 0.678	1.859 ± 0.482	0.365 ± 0.939	0.183 ± 0.081	0.293 ± 0.094	0.110 ± 0.134
3B	0.752 ± 0.128	0.446 ± 0.198	0.306 ± 0.195	1.365 ± 0.558	2.247 ± 0.614	0.882 ± 0.782	0.177 ± 0.063	0.344 ± 0.098	0.167 ± 0.119
7B	0.770 ± 0.144	0.490 ± 0.236	0.280 ± 0.214	1.312 ± 0.630	2.176 ± 0.720	0.864 ± 0.787	0.176 ± 0.086	0.335 ± 0.122	0.160 ± 0.121

Model-scale extension. Under the same construction/confirmatory protocol, the complete Qwen2.5-3B-Instruct and Qwen2.5-7B-Instruct matrices are available. Figure 3(f) shows that BTF's same-task recovery and different-task suppression persist across all three Qwen2.5 scales. Raw-L, Raw-R, Rand-Q, SMRC, and Zero remain close to the nonselective reference level, while Tr-L and Tr-R continue to recover substantial functionality. The classical metrics have the same direction (Table 3), indicating that scale does not change the main conclusion.

Cross-family validation: Llama-3.2-1B. Llama-3.2-1B reproduces the relation ordering under the matched protocol: Zero-Core gains are 0.4543, 0.1433, and 0.0239 for same-task, same-interface/different-task, and cross-interface relations, respectively. Unilateral and bilateral transports are both positive and similar to Qwen2.5-1.5B; detailed task, seed, interval, and run-identity records appear in Appendix C.2. **Seed and rank robustness.** We further test sensitivity to construction seeds and transport rank using the same seed split and target-level aggregation protocol. The donor-seed balance check and rank-2/rank-4 truncations preserve positive functional selectivity across construction seeds and ranks, confirming robustness beyond the rank-8 representation. Per-seed and per-task records are provided in Appendix B.

5.3 Why Reuse Depends on the Core–Base Combination

After establishing same-task recovery through registration, we next ask why the recovery depends jointly on the core and the base. Rand-Q and SMRC provide coordinate, spectrum, and norm controls, while the paired BTF–SMRC comparison shows that learned coupling carries the task-specific signal along the same-task path. Geometry matching leaves BTF positively selective while SMRC remains near zero, and exact-core base-swap shows that the same donor core produces a larger gain on the task-matched base than on other task bases (Appendix C). Figure 3(d)–(e) displays these target-level distributions. Together, these controls show that functional recovery combines subspace geometry, update location, target-base context, and learned coupling, with task-specific information carried beyond spectrum and scale. The reusable object is therefore jointly determined by the

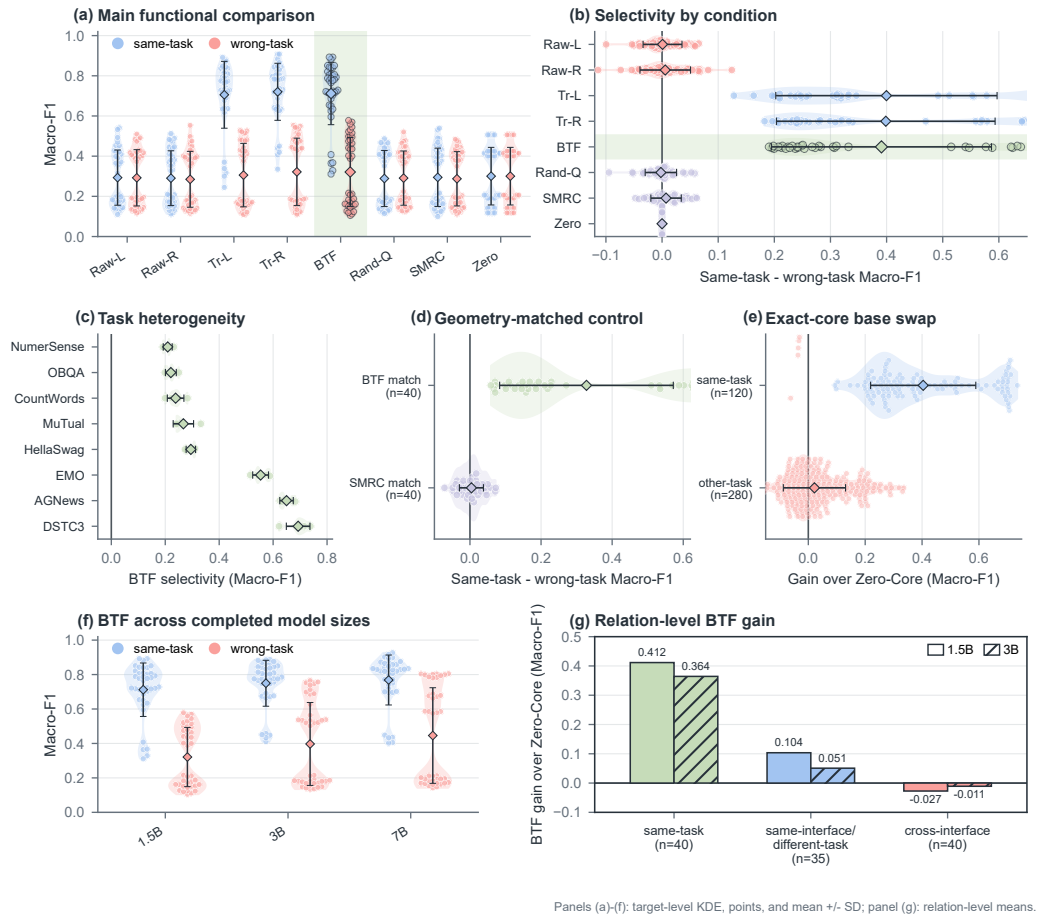


Figure 3: Main experimental results. The composite summarizes functional selectivity, task heterogeneity, geometry and base controls, model-scale extension, and relation-level gains. Points retain target-level observations where applicable; panel (g) compares relation-level gains for Qwen2.5-1.5B and 3B. Detailed counts, intervals, and positive-target fractions are reported in Table 14.

base-conditioned core and the target base. The higher wrong-task scores on multiple-choice tasks further indicate shared formatting and decision behavior across tasks; the complete task and interface breakdown is in Appendix C.1.

5.4 Functional Reuse Forms an Ordered Core–Base Compatibility Structure

The main matrix establishes the overall difference between same-task and different-task relations. We further group target–donor pairs into same-task, same-interface/different-task, and cross-interface relations. Donors are averaged within each target and targets are then weighted equally, so the statistical unit remains the target rather than an unaggregated pair row. In addition to raw Macro-F1, we report the functional gain relative to Zero-Core,

$$g_{A \leftarrow B} = M_A\left(D^{\mathrm{BTF}}_{A \leftarrow B}\right) - M_A\left(D^{\mathrm{Zero}}_{A \leftarrow B}\right). \tag{10}$$

Figure 3(g) shows that the point estimates for both Qwen2.5-1.5B and 3B follow the order same-task $>$ same-interface/different-task $>$ cross-interface, which is also supported by the Llama cross-family validation described above. This makes "conditional" an observable compatibility structure: reuse is a joint response of the donor core and target base. Group-level geometry follows the same ordering and provides a descriptive geometric view alongside the functional readout. Complete counts, raw Macro-F1 values, clustered bootstrap intervals, and positive-target counts are provided in Table 14.

6 DISCUSSION

6.1 REGISTRATION ESTABLISHES COMPARABLE COORDINATES

Q_U and Q_V connect the donor's internal frames to the target frame, making cross-run core comparison possible in a common coordinate system. The gauge-rotation positive control confirms representation invariance, while raw splicing and random transport quantify the functional contribution of registration. BTF-FR therefore converts coordinate differences in parameter representations into a computable registration relation, evaluated through the dense intervention written into the target model and the resulting model outputs.

6.2 THE BASE-CONDITIONED CORE DETERMINES REUSE BEHAVIOR

Geometry matching and exact-core base swaps show that functional recovery combines geometric compatibility with target-base context, and that the same core can produce different responses on different target bases. SMRC shows that learned directional mapping carries task-specific information beyond singular-value spectrum and norm. The core and base therefore form a joint object: the core maps directions, while the base determines how those directions act in the target model.

The relation stratification gives the functional form of this joint dependence: same-task relations recover the most, same-interface/different-task relations retain an intermediate response, and cross-interface relations approach Zero-Core. The same ordered structure appears in Qwen2.5 and Llama. Group-level geometry follows the same direction and provides a descriptive geometric view alongside the intervention-based functional readout.

This relation also suggests a downstream interface for adapter libraries. A library can record donor cores, target frames, transport residuals, and functional responses, and use these records to evaluate compatibility before reuse on a target model. The higher wrong-task scores on multiple-choice tasks further show that registered relations capture both task-specific selectivity and shared cross-task behavior: common label interfaces, instruction-following patterns, and general decision structure can persist across tasks, while task-specific coupling produces the same–wrong difference. The base-conditioned core thus captures both task-selective transfer and shared functionality.

6.3 RELATION CONDITIONS SHAPE FUNCTIONAL REUSE

Across multiple Qwen2.5 scales, a Llama model family, eight heterogeneous tasks, and independent downstream tasks, registered donor cores exhibit the same central relation: their functional response is organized by the donor–target relation and the target base. Downstream results extend this picture to recommendation ranking and show that task interfaces and model conditions modulate the amount of reuse: registered cores remain selective on AGNews and MovieLens-1M, while SST-5 produces model-dependent responses. The full relation and downstream analyses are given in Appendices C.2, C.4, and E. A preliminary Fiber Memory experiment found that reference-frame aggregation can provide an effective warm start, motivating evaluation of broader downstream uses.

7 CONCLUSION

Our central finding is that LoRA reusability is carried by a base-conditioned core: a learned core enters the target base through a cross-run frame relation and produces a measurable functional response at that update location, shifting adapter reuse from an isolated parameter block to a core–base functional relation. Across Qwen2.5-1.5B, Qwen2.5-3B, and Llama-3.2-1B, functional responses follow a donor–target stratification, with strongest recovery for same-task relations, an intermediate response for same-interface/different-task relations, and the weakest response for cross-interface relations. BTF-FR establishes comparable coordinates and, together with functional interventions, random controls, geometry matching, and exact-core base swaps, separates coordinate, geometry, base, and learned-coupling effects. A preliminary Fiber Memory experiment also found warm-start gains from reference-frame aggregation, motivating future evaluation across broader downstream tasks.

MANDATORY STATEMENTS

AI USE STATEMENT

Generative AI tools assisted this work in five areas: (i) editing the manuscript for readability, including polishing selected paragraphs and condensing verbose passages without changing their information; (ii) proposing an organizational structure for the paper; (iii) locating and identifying literature relevant to the study. The authors assessed differences from prior work and novelty through their own reading of the retrieved literature; (iv) drafting and refining plotting code to visualize experimental results generated by the authors; and (v) drafting, debugging, and refactoring selected code fragments in the experiment runners and reproducibility utilities. The authors wrote and controlled the core implementation and experimental logic. Before use, each AI-assisted fragment was checked against the intended protocol and tested through the executable experiment or unit-test pipeline. AI tools did not make final decisions about the theoretical model, methodology, experimental protocol, reported results, or scientific interpretation. All experimental data and figures derive from the authors' executed runs and recorded outputs. The authors read and verified all cited references, confirmed the behavior of AI-assisted code and its outputs, and take responsibility for the final manuscript, including its text, claims, and artifacts produced with AI assistance.

REPRODUCIBILITY STATEMENT

The method and experimental protocol are specified in Sections 3–4. Appendix A gives the mathematical definitions, derivations, and gauge-invariance proof; Appendix B gives the run splits, statistical units, bootstrap procedures, and protocol checks; Appendix C gives the full task-, relation-, model-, and downstream-level results; and Appendix D records the complementary shared-envelope evidence. Appendix E provides the pair-level algorithm, pseudocode, artifact schema, run identities, integrity checks, and resource fields. The anonymous supplementary materials contain the anonymized source code, configuration and protocol files, task and model manifests, analysis scripts, and machine-path-free result summaries used to generate the reported analyses. Experiments are determined by fixed model and data manifests, protocol files, seed lists, code hashes, and norm rules; pretrained model weights are referenced by their model identities and checksums rather than redistributed.

REFERENCES

Armen Aghajanyan, Sonal Gupta, and Luke Zettlemoyer. Intrinsic dimensionality explains the effectiveness of language model fine-tuning. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 7319–7328. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.568.

Samuel K. Ainsworth, Jonathan Hayase, and Siddhartha Srinivasa. Git Re-Basin: Merging models modulo permutation symmetries. In *International Conference on Learning Representations*, 2023. arXiv:2209.04836.

Ning Ding, Yujia Qin, Guang Yang, Fuchao Wei, Zonghan Yang, Yusheng Su, Shengding Hu, Yulin Chen, Chi-Min Chan, Weize Chen, Jing Yi, Weilin Zhao, Xiaozhi Wang, Zhiyuan Liu, Hai-Tao Zheng, Jianfei Chen, Yang Liu, Jie Tang, Juanzi Li, and Maosong Sun. Parameter-efficient fine-tuning of large-scale pre-trained language models. *Nature Machine Intelligence*, 5(3):220–235, 2023. doi: 10.1038/s42256-023-00626-4.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin de Laroussilhe, Andréa Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2790–2799. PMLR, 2019.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. arXiv:2106.09685.

Chengsong Huang, Qian Liu, Bill Yuchen Lin, Tianyu Pang, Chao Du, and Min Lin. LoraHub: Efficient cross-task generalization via dynamic LoRA composition. In *Conference on Language Modeling*, 2024. arXiv:2307.13269.

Gabriel Ilharco, Marco Tulio Ribeiro, Mitchell Wortsman, Suchin Gururangan, Ludwig Schmidt, Hannaneh Hajishirzi, and Ali Farhadi. Editing models with task arithmetic. In *International Conference on Learning Representations*, 2023. arXiv:2212.04089.

Brian Lester, Rami Al-Rfou, and Noah Constant. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 3045–3059. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.emnlp-main.243.

Xiang Lisa Li and Percy Liang. Prefix-tuning: Optimizing continuous prompts for generation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 4582–4597. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.acl-long.353.

Haokun Liu, Derek Tam, Mohammed Muqeeth, Jay Mohta, Tenghao Huang, Mohit Bansal, and Colin Raffel. Few-shot parameter-efficient fine-tuning is better and cheaper than in-context learning. In *Advances in Neural Information Processing Systems*, volume 35, pp. 1950–1965, 2022. doi: 10.52202/068431-0142.

Weiyang Liu, Zeju Qiu, Yao Feng, Yuliang Xiu, Yuxuan Xue, Longhui Yu, Haiwen Feng, Zhen Liu, Juyeon Heo, Songyou Peng, Yandong Wen, Michael J. Black, Adrian Weller, and Bernhard Schölkopf. Parameter-efficient orthogonal finetuning via butterfly factorization. In *International Conference on Learning Representations*, 2024. arXiv:2311.06243.

Fanxu Meng, Zhaohui Wang, and Muhan Zhang. PiSSA: Principal singular values and singular vectors adaptation of large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 121038–121072, 2024. doi: 10.52202/079017-3846.

Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. AdapterHub: A framework for adapting transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pp. 46–54. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.emnlp-demos.7.

Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. AdapterFusion: Non-destructive task composition for transfer learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 487–503. Association for Computational Linguistics, 2021. doi: 10.18653/v1/2021.eacl-main.39.

George Stoica, Daniel Bolya, Jakob Bjorner, Pratik Ramesh, Taylor Hearn, and Judy Hoffman. ZipIt! merging models from different tasks without training. In *International Conference on Learning Representations*, 2024. arXiv:2305.03053.

George Stoica, Pratik Ramesh, Boglárka Ecsedi, Leshem Choshen, and Judy Hoffman. Model merging with SVD to tie the knots. In *International Conference on Learning Representations*, 2025. arXiv:2410.19735.

Qi Sun, Siyue Zhang, Yulin Chen, Yuxiang Xue, Ru Peng, and Chen Zhao. From "weak" signals to strong models: Preference delta aggregation with LoRA merging. *arXiv preprint arXiv:2606.00357*, 2026.

Derek Tam, Mohit Bansal, and Colin Raffel. Merging by matching models in task parameter subspaces. *Transactions on Machine Learning Research*, 2024. arXiv:2312.04339.

Prateek Yadav, Derek Tam, Leshem Choshen, Colin Raffel, and Mohit Bansal. TIES-Merging: Resolving interference when merging models. In *Advances in Neural Information Processing Systems*, volume 36, pp. 7093–7115, 2023. doi: 10.52202/075280-0310.

Ziyu Zhao, Leilei Gan, Guoyin Wang, Wangchunshu Zhou, Hongxia Yang, Kun Kuang, and Fei Wu. LoraRetriever: Input-aware LoRA retrieval and composition for mixed tasks in the wild. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 4447–4462. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.findings-acl.263.

Ziyu Zhao, Tao Shen, Didi Zhu, Zexi Li, Jing Su, Xuwu Wang, and Fei Wu. Merging LoRAs like playing LEGO: Pushing the modularity of LoRA to extremes through rank-wise clustering. In *International Conference on Learning Representations*, 2025. arXiv:2409.16167.

A Mathematical Definitions and Proofs

A.1 Subspace Locations, Frames, and Gauge Equivalence

Let

$$\mathcal{X}_{d,r} = \mathrm{St}(d, r) = \{U \in \mathbb{R}^{d\times r} : U^\top U = I_r\} \tag{11}$$

be the Stiefel manifold of r-frames, and let $[U]$ denote the column space of U, a point on the Grassmann manifold. The paired left/right subspace location is

$$b = ([U], [V]) \in \mathrm{Gr}(r, d_{\mathrm{out}}) \times \mathrm{Gr}(r, d_{\mathrm{in}}). \tag{12}$$

All low-rank updates at a fixed location b form the local fiber

$$\mathcal{F}_b = \{D : D = USV^\top,\ S \in \mathbb{R}^{r\times r},\ U \text{ spans } [U],\ V \text{ spans } [V]\}. \tag{13}$$

Once a particular frame (U, V) is selected, S is the matrix coordinate within this fiber. Each S defines an effective update at a fixed subspace location, whose effect is measured through the chosen functional interface.

The group $\mathrm{O}(r) \times \mathrm{O}(r)$ acts on the three-factor representation as

$$(U, S, V) \mapsto (UR_L, R_L^\top S R_R, V R_R), \qquad R_L, R_R \in \mathrm{O}(r). \tag{14}$$

Multiplication gives

$$(UR_L)(R_L^\top S R_R)(VR_R)^\top = U(R_L R_L^\top) S (R_R R_R^\top) V^\top \tag{15}$$

$$= USV^\top. \tag{16}$$

All three-factor representations of the same dense update form the gauge-equivalence class

$$[U, S, V] = \{(UR_L, R_L^\top S R_R, VR_R) : R_L, R_R \in \mathrm{O}(r)\}. \tag{17}$$

The equivalence class with its frame is therefore gauge-invariant, whereas raw S depends on the selected frame. When two runs occupy different subspace locations, cross-run comparison additionally requires an explicit coordinate transformation; Procrustes transport selects the closest orthogonal relation between frames, while the residual retains the actual subspace difference.

A.2 Effective Updates and a Deterministic Chart

For each PEFT module, we read the runtime scaling and form the original effective update

$$D = \Delta W = \mathrm{scaling} \cdot B_{\mathrm{lora}} A_{\mathrm{lora}}. \tag{18}$$

We then compute a numerical rank-r decomposition and distinguish the reconstruction from the original effective update:

$$\Delta W = \widetilde{D}^{(r)} + E_{\mathrm{svd}}, \qquad \widetilde{D}^{(r)} = \widetilde{U}\widetilde{\Sigma}\widetilde{V}^\top, \qquad \widetilde{U}^\top\widetilde{U} = \widetilde{V}^\top\widetilde{V} = I_r. \tag{19}$$

Here $E_{\mathrm{svd}} = \Delta W - \widetilde{D}^{(r)}$. The implementation first obtains a numerical rank-r subspace and then recomputes $S = U^\top \Delta W V$ in a deterministic chart. The rank-r reconstruction used below is therefore $D^{(r)} := USV^\top$, kept distinct from the original effective update ΔW. Randomized SVD determines the numerical low-rank range for large matrices; its oversampling width, power-iteration count, and random seed are included in the run identity.

To reduce the effect of sign choices and nearly repeated singular values, we form projection matrices

$$\Pi_U = \widetilde{U}\widetilde{U}^\top, \qquad \Pi_V = \widetilde{V}\widetilde{V}^\top. \tag{20}$$

The implementation scans projection-matrix columns in environmental coordinate order, applies Gram–Schmidt to candidate vectors, and fixes the largest-magnitude coordinate of each basis vector to be positive. The resulting U, V are deterministic frames for the corresponding Grassmann points. The core is recomputed as

$$S = U^\top \Delta W V, \tag{21}$$

which defines $D^{(r)} = USV^\top$ and preserves the complete left–right coupling in the selected chart. The diagonal singular-value matrix describes the spectrum, while S additionally retains the mapping between left and right directions.

A.3 Procrustes Transport, Principal Angles, and Compatibility Error

The left-side transport solves

$$Q_U = \operatorname*{arg\,min}_{Q^\top Q = I_r} \|U_B Q - U_A\|_F^2 . \tag{22}$$

Expanding the objective gives

$$\begin{aligned} \|U_B Q - U_A\|_F^2 &= \operatorname{tr}((U_B Q - U_A)^\top (U_B Q - U_A)) && (23)\\ &= \operatorname{tr}(Q^\top U_B^\top U_B Q) + \operatorname{tr}(U_A^\top U_A) - 2\operatorname{tr}(Q^\top U_B^\top U_A) && (24)\\ &= 2r - 2\operatorname{tr}(Q^\top M_U), && (25) \end{aligned}$$

where $M_U = U_B^\top U_A$. Minimizing is equivalent to maximizing $\operatorname{tr}(Q^\top M_U)$. Let

$$M_U = L_U \Sigma_U R_U^\top, \qquad \Sigma_U = \operatorname{diag}(\sigma_{U,1}, \ldots, \sigma_{U,r}). \tag{26}$$

With $Z = L_U^\top Q R_U$,

$$\operatorname{tr}(Q^\top M_U) = \operatorname{tr}(Z^\top \Sigma_U) = \sum_{j=1}^{r} \sigma_{U,j} Z_{jj} \le \sum_{j=1}^{r} \sigma_{U,j}. \tag{27}$$

The upper bound is attained at $Z = I$, so

$$Q_U = L_U R_U^\top, \qquad Q_U^\top Q_U = I_r. \tag{28}$$

The same argument applied to $M_V = V_B^\top V_A = L_V \Sigma_V R_V^\top$ gives $Q_V = L_V R_V^\top$.

The singular values of M_U give the principal-angle cosines:

$$\theta_{U,j} = \arccos(\sigma_{U,j}). \tag{29}$$

The normalized Procrustes residual and squared overlap are

$$\delta_U^2 = \min_{Q^\top Q = I} \frac{\|U_B Q - U_A\|_F^2}{r} = \frac{2}{r} \sum_{j=1}^{r} (1 - \cos\theta_{U,j}), \tag{30}$$

$$o_U = \frac{\left\|U_A^\top U_B\right\|_F^2}{r} = \frac{1}{r} \sum_{j=1}^{r} \cos^2\theta_{U,j}, \tag{31}$$

with δ_V, o_V defined analogously. Principal angles describe the subspace relation, while the Procrustes residual describes frame differences after the best internal coordinate transport.

Let

$$D_B^{(r)} = U_B S_B V_B^\top, \quad R_U = U_B Q_U - U_A, \quad R_V = V_B Q_V - V_A, \quad S' = Q_U^\top S_B Q_V. \tag{32}$$

Then $U_B = (U_A + R_U)Q_U^\top$, $V_B = (V_A + R_V)Q_V^\top$, and

$$\begin{aligned} D_B^{(r)} &= (U_A + R_U) S' (V_A + R_V)^\top && (33)\\ &= U_A S' V_A^\top + R_U S' V_A^\top + U_A S' R_V^\top + R_U S' R_V^\top. && (34) \end{aligned}$$

This is the no-norm expansion: the first term is the unscaled BTF, while the remaining terms arise from subspace mismatch. The corresponding bound is

$$\left\| D_B^{(r)} - U_A S' V_A^\top \right\|_F \le \|S'\|_2 \left(\|R_U\|_F + \|R_V\|_F + \|R_U\|_F \, \|R_V\|_2 \right). \tag{35}$$

Under target-norm matching, $S'^* = c_{A,B,l} S'$, and

$$D_{A \leftarrow B}^{\mathrm{BTF}} = c_{A,B,l} U_A S' V_A^\top.$$

Relative to $D_B^{(r)}$, an additional scale term $(1 - c_{A,B,l}) U_A S' V_A^\top$ appears. Thus Q handles frame coordinates, R_U, R_V record base compatibility, and target-norm matching separately standardizes each module's core scale.

A.4 Two-dimensional Rotation Example

Consider coincident left and right subspaces with rotated internal frames:

$$U_B = U_A R_L^\top, \qquad V_B = V_A R_R^\top, \qquad R_L, R_R \in \mathrm{O}(2). \tag{36}$$

The donor update in the target frame is

$$D_B^{(r)} = U_A R_L^\top S_B R_R V_A^\top. \tag{37}$$

The Procrustes solutions are $Q_U = R_L$ and $Q_V = R_R$, so the aligned core is $R_L^\top S_B R_R$. For

$$R_L = R_R = R = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \qquad S_B = \begin{bmatrix} 2 & 0 \\ 0 & 1 \end{bmatrix}, \tag{38}$$

$$R^\top S_B R = \begin{bmatrix} 1 & 0 \\ 0 & 2 \end{bmatrix}. \tag{39}$$

Raw splicing uses $\mathrm{diag}(2, 1)$ in the target frame, whereas gauge-consistent coordinates use $\mathrm{diag}(1, 2)$. BTF exactly recovers the donor rank-r reconstruction in this coincident-subspace, no-norm example; target-norm matching gives a per-module rescaled reconstruction. Tr-L and Tr-R retain one donor frame each to localize unilateral coordinate effects.

A.5 Norm, Spectrum, and Gauge Controls

The main analysis applies target-norm matching independently to each LoRA module:

$$S_{B,l}^* = c_{A,B,l} S_{B,l}, \qquad c_{A,B,l} = \frac{\|S_{A,l}\|_F}{\|S_{B,l}\|_F + \varepsilon}, \qquad \varepsilon = 10^{-12}. \tag{40}$$

This standardizes donor scale within each target module. Because of ε, the scaled norm is numerically close to, but not mathematically identical to, the target norm. If

$$S_B^* = L_B \Sigma_B R_B^\top, \tag{41}$$

the spectrum-matched random core is

$$S_{\mathrm{spec-rand}} = H_L \Sigma_B H_R^\top, \tag{42}$$

where H_L, H_R are independent Haar orthogonal matrices. Its Frobenius norm and singular values are unchanged, so the control randomizes only the coupling between left and right directions.

Under the gauge transformation

$$U_B' = U_B R_L, \quad S_B' = R_L^\top S_B R_R, \quad V_B' = V_B R_R, \tag{43}$$

choose $Q_U' = R_L^\top Q_U$ and $Q_V' = R_R^\top Q_V$. Then $U_B' Q_U' = U_B Q_U$, $V_B' Q_V' = V_B Q_V$, and

$$(Q_U')^\top S_B' Q_V' = (Q_U^\top R_L)(R_L^\top S_B R_R)(R_R^\top Q_V) \tag{44}$$

$$= Q_U^\top S_B Q_V. \tag{45}$$

The dense target-frame matrix is therefore unchanged. When repeated singular values make the Procrustes solution non-unique, the implementation uses equivariant tie-breaking and checks dense matrices, logits, and predictions rather than individual Q entries.

Table 4: Main experiment configuration.

Item	Configuration
Model and tasks	Qwen2.5-1.5B-Instruct; eight heterogeneous SuperNI tasks
Output interface	Four-class single-token output; task names hidden
LoRA	Rank 8, $\alpha = 16$, dropout 0; q/v projections
Run split	Construction seeds 11/22/33; confirmatory seeds 101–105
Pairing	Three same-task and seven different-task donors per target
Precision	FP32 write-in and evaluation

A.6 Orthogonal Projection and Target-frame Injection

The two-sided target-subspace projection is

$$D_{B|A}^{\mathrm{proj}} = U_A(U_A^\top \Delta W_B V_A)V_A^\top. \tag{46}$$

BTF instead preserves the transported donor coupling and injects it into the target base. For any orthonormal (U, V), $P_{U,V}(D) = UU^\top DVV^\top$ is the Frobenius projection:

$$\|D\|_F^2 = \|P_{U,V}(D)\|_F^2 + \|D - P_{U,V}(D)\|_F^2. \tag{47}$$

Model outputs are additionally affected by input covariance, nonlinearities, and cross-layer paths; functional importance is therefore determined by the intervention results.

A.7 Reference-frame Core Aggregation and Gauge Consistency

For a reference run $s_0 = 11$, let $\mathcal{C}_t$ denote the construction seeds for task t.

$$U_t^{\mathrm{ref}} = U_{t,s_0}, \qquad V_t^{\mathrm{ref}} = V_{t,s_0}. \tag{48}$$

Each construction core is transported to the reference frame:

$$\widetilde{S}_{t,s}^{\mathrm{ref}} = \gamma_{t,s}(Q_{U,t,s\to\mathrm{ref}})^\top S_{t,s} Q_{V,t,s\to\mathrm{ref}}, \tag{49}$$

and averaged:

$$\bar{S}_t^{\mathrm{ref}} = \frac{1}{|\mathcal{C}_t|}\sum_{s\in\mathcal{C}_t} \widetilde{S}_{t,s}^{\mathrm{ref}}, \tag{50}$$

with dense initialization

$$\Delta W_{l,t}^{(0)} = U_{l,t}^{\mathrm{ref}} \bar{S}_{l,t}^{\mathrm{ref}} (V_{l,t}^{\mathrm{ref}})^\top. \tag{51}$$

The coefficient $\gamma_{t,s}$ is the per-module reference-norm coefficient. The artifact stores reference frames, transports, norm coefficients, and residuals so this initialization can be reconstructed module by module. A gauge rewrite of the reference frame changes the aggregate core equivariantly and leaves the dense initialization unchanged.

B Protocol, Statistics, and Numerical Implementation

B.1 Run Splits and Leakage Prevention

The construction set $\mathcal{C}$ extracts U, S, V, determines reference frames, generates fixed random controls, and forms donor lists. The confirmatory set $\mathcal{H}$ produces target–donor scores. The adaptation split is disjoint and is reserved for new Fiber Memory runs. Construction seeds are 11/22/33 and confirmatory seeds are 101–105. Each target has three same-task and seven different-task donors, yielding 400 directed pairs.

The nine functional conditions share the target carrier, evaluation examples, label mapping, batch partition, carrier rank, and numerical precision. Full Target and Zero-Core are target-level references. Donors are aggregated within target before task-level aggregation.

Table 5: Task-level BTF results for Qwen2.5-7B. Values are means ± SD over five target-level observations.

Task	Targets	Same	Wrong	Difference	Positive
AGNews	5	.8235	.1929	.6307	5/5
CountWords	5	.4248	.1547	.2701	5/5
DSTC3	5	.8610	.1950	.6661	5/5
EMO	5	.7597	.1767	.5829	5/5
HellaSwag	5	.9077	.8022	.1055	5/5
MuTual	5	.8215	.6754	.1460	5/5
NumerSense	5	.7019	.5820	.1199	5/5
OBQA	5	.8472	.7889	.0583	5/5

Table 6: Same-task Memory paired effects. Values are means ± SD over 20 task–seed pairs.

Comparison	Final F1	0–100 AULC	10–100 AULC
Same-task vs Standard	$.0749 \pm .0470$	17.5230 ± 9.4909	13.4590 ± 8.1131
Same-task vs Wrong-task	$.0655 \pm .0318$	15.8243 ± 7.1129	11.5408 ± 5.8689
Same-task vs Random-spectrum	$.0180 \pm .0180$	8.2010 ± 4.6148	4.7021 ± 3.2600

B.2 Main Implementation Configuration

B.3 Complete Fiber Memory Pairing Records

Same-task Memory is positive relative to Standard, Wrong-task Memory, and Random-spectrum Memory across the paired metrics. The direct comparison targets the initialization state and short continued training under a shared adaptation budget.

B.4 Artifact Execution Schema

```
for target in confirmatory_runs:
    A = load_factorization(target)
    for donor in registered_donors(target):
        B = load_factorization(donor)
        for module in target_modules:
            qu = procrustes(B.U[module], A.U[module])
            qv = procrustes(B.V[module], A.V[module])
            s_star = norm_match(B.S[module], A.S[module])
            dense = (A.U[module] @ qu.T @ s_star
                     @ qv @ A.V[module].T)
        evaluate_and_write(all_registered_conditions, atomic=True)
aggregate_by_target_then_task()
bootstrap_task_clustered()
```

Each target–donor pair/module stores run identity, target and donor task/seed, module name, rank, carrier rank, norm rule, evaluation dtype, frame shapes and hashes, Q_U, Q_V, orthogonality residuals, principal angles, overlaps, Procrustes residuals, norm coefficients, dense condition hashes, predictions, logits, Macro-F1, accuracy, NLL, ECE, and completion state. The minimum status record contains code hash, data hash, protocol hash, and `complete=true`.

B.5 Execution Recovery and Result Integrity

All results are determined jointly by fixed model manifests, data manifests, protocol files, code, and norm rules. Pair results enter aggregates only after registered modules, functional conditions, and Full-reinjected checks complete. The main matrix has 400/400 expected/completed pairs, 40 targets, three same-task donors and seven wrong-task donors per target; C1–C8 each contain 400 pair rows, while C0/Full Target is a target-native reference repeated only for storage and contributes 40 target-level observations. The scale extensions and Llama validation pass the same pair-key and gauge-invariance checks.

The main run identity is

Table 7: Paired Same-task Memory effects. Each entry is a mean ± SD over 20 task–seed pairs.

Comparison	Final F1	0–100 AULC	10–100 AULC
vs Standard	$.0749 \pm .0470$	17.5230 ± 9.4909	13.4590 ± 8.1131
vs Wrong-task	$.0655 \pm .0318$	15.8243 ± 7.1129	11.5408 ± 5.8689
vs Random-spectrum	$.0180 \pm .0180$	8.2010 ± 4.6148	4.7021 ± 3.2600

Table 8: Mean geometric variables by relation group.

Model	Relation	Angle	Overlap	Residual
1.5B	same-task	71.11	.1754	1.1726
1.5B	same-interface/different-task	81.46	.0423	1.3055
1.5B	cross-interface	82.50	.0316	1.3188
3B	same-task	72.00	.1661	1.1832
3B	same-interface/different-task	82.11	.0371	1.3137
3B	cross-interface	82.99	.0286	1.3251
Llama-3.2-1B	same-task	72.73	.1564	1.1934
Llama-3.2-1B	same-interface/different-task	82.58	.0329	1.3199
Llama-3.2-1B	cross-interface	83.42	.0249	1.3307

`39b8cc3dcbec2d8a93b98306c59e8b0dbce28195b5c6e875c898a022f20f26b6`.

The Qwen2.5-3B, Qwen2.5-7B, and Llama-3.2-1B identities are recorded in the anonymous artifact together with model configuration, tokenizer, and safetensors checksums.

B.6 Memory, Resource, and Audit Records

The Memory artifact stores task IDs, construction run IDs, reference frames, per-run Q^U, Q^V, reference-norm coefficients, core means, transport residuals, hashes, and dense initializations. The downstream protocol records states, metrics, predictions, and audit fields at steps 0/10/25/50/100 for four tasks, four initializations, and five seeds. Resource records include parameter count, hidden size, layer count, LoRA target modules, carrier rank, loading memory, peak training memory, peak write-back memory, time per pair, total pairs, and disk usage. FP16, BF16, and FP32 are recorded separately.

B.7 Detailed Error Decomposition

For module l, let the original effective update be $\Delta W_{B,l}$ and the rank-r reconstruction be $D^{(r)}_{B,l}$. The factorization error is

$$\Delta W_{B,l} = D^{(r)}_{B,l} + E_{\mathrm{svd},B,l}, \qquad \eta_{\mathrm{svd},B,l} = \frac{\|E_{\mathrm{svd},B,l}\|_F}{\|\Delta W_{B,l}\|_F + \varepsilon}. \tag{52}$$

The base-mismatch terms are the R_U, R_V terms in the expansion in Appendix A. The carrier write-back error is

$$\eta^{(c)}_{\mathrm{write}} = \frac{\left\|\widetilde{D}^{(c)} - D^{(c)}\right\|_F}{\left\|D^{(c)}\right\|_F + \varepsilon}, \tag{53}$$

where $D^{(c)}$ is the ideal dense intervention and $\widetilde{D}^{(c)}$ is reread from the model. A condition enters the functional aggregate only after its write-back and Full-reinjected checks pass.

B.8 Control-specific Functional Quantities

In the following formulas, $D^{(c)}_{A\leftarrow B}$ denotes the dense intervention for condition c in Table 1; specifically, $D^{\mathrm{RawL}}_{A\leftarrow B}$, $D^{\mathrm{RawR}}_{A\leftarrow B}$, $D^{\mathrm{RandQ}}_{A\leftarrow B}$, $D^{\mathrm{SMRC}}_{A\leftarrow B}$, and $D^{\mathrm{Zero}}_{A\leftarrow B}$ correspond to Raw-L, Raw-R, Rand-Q, SMRC, and Zero, with $D^{\mathrm{Zero}}_{A\leftarrow B} = 0$. The target-native reference is denoted by D^{FT}_A.

$$G_{\text{transport}}(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{RawL}}_{A\leftarrow B}), \tag{54}$$

$$G_{\text{same}}(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{Zero}}_{A\leftarrow B}), \tag{55}$$

$$G_{\text{learned}}(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{SMRC}}_{A\leftarrow B}), \tag{56}$$

$$G_{Q}(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{RandQ}}_{A\leftarrow B}). \tag{57}$$

Same–wrong measures task selectivity, whereas BTF–SMRC measures learned coupling beyond a random core with the same spectrum and norm.

B.9 Numerical and Precision Checks

The main matrix uses FP32 write-in and evaluation so condition differences primarily reflect the intervention. BF16 sensitivity records matrix relative error, maximum logit drift, and prediction agreement. The gauge smoke audit reports factorization error below 10^{-6}, prediction agreement 1.0, and maximum FP32 logit drift 3.5×10^{-4}. The artifact records the corresponding values for every model and condition.

B.10 Expanded Procrustes Derivation

Writing $M_U = U_B^\top U_A$, the Procrustes objective expands as

$$\|U_B Q - U_A\|_F^2 = 2r - 2\,\mathrm{tr}(Q^\top M_U). \tag{58}$$

If $M_U = L_U \Sigma_U R_U^\top$ and $Z = L_U^\top Q R_U$, then

$$\mathrm{tr}(Q^\top M_U) = \mathrm{tr}(Z^\top \Sigma_U) \le \sum_j \sigma_{U,j}, \tag{59}$$

with equality at $Z = I$. Hence $Q_U = L_U R_U^\top$; the right-side result follows analogously. This derivation is used for all model sizes.

B.11 Expanded Gauge-invariance Proof

Under

$$U'_B = U_B R_L, \quad S'_B = R_L^\top S_B R_R, \quad V'_B = V_B R_R, \tag{60}$$

choose $Q'_U = R_L^\top Q_U$ and $Q'_V = R_R^\top Q_V$. Then

$$U'_B Q'_U = U_B Q_U, \qquad V'_B Q'_V = V_B Q_V, \tag{61}$$

$$(Q'_U)^\top S'_B Q'_V = (Q_U^\top R_L)(R_L^\top S_B R_R)(R_R^\top Q_V) \tag{62}$$

$$= Q_U^\top S_B Q_V. \tag{63}$$

The dense target-frame matrix and functional output are unchanged. Repeated singular values are handled by equivariant tie-breaking and dense-matrix, logit, and prediction checks.

B.12 Expanded Projection Identity

For $\mathcal{L}(U,V) = \{UXV^\top : X \in \mathbb{R}^{r\times r}\}$ and $S = U^\top D V$, $USV^\top = UU^\top DVV^\top$. For any X,

$$\left\langle D - USV^\top, UXV^\top \right\rangle_F = \mathrm{tr}((U^\top DV)^\top X) - \mathrm{tr}(S^\top X) = 0. \tag{64}$$

Therefore the two-sided reconstruction is the Frobenius projection and

$$\|D\|_F^2 = \left\|USV^\top\right\|_F^2 + \left\|D - USV^\top\right\|_F^2. \tag{65}$$

B.13 Expanded Hierarchical Statistics

For condition c, let $m_{A,B}^{(c)}$ denote the Macro-F1 for target A and donor B. Let $\mathcal{S}(A)$ and $\mathcal{W}(A)$ be the same-task and different-task donor sets for target A, let $\mathcal{H}_t$ be the confirmatory targets for task t, and let $\mathcal{T}$ be the tested task set. The target-level difference is

$$d_A^{(c)} = \frac{1}{|\mathcal{S}(A)|} \sum_{B \in \mathcal{S}(A)} m_{A,B}^{(c)} - \frac{1}{|\mathcal{W}(A)|} \sum_{B \in \mathcal{W}(A)} m_{A,B}^{(c)}. \tag{66}$$

For task t,

$$\mu_t^{(c)} = \frac{1}{|\mathcal{H}_t|} \sum_{A \in \mathcal{H}_t} d_A^{(c)}, \qquad \mu_{\text{task}}^{(c)} = \frac{1}{|\mathcal{T}|} \sum_{t \in \mathcal{T}} \mu_t^{(c)}. \tag{67}$$

The target remains the statistical unit; pair rows support reproducibility and paired diagnostics.

B.14 Additional Artifact and Execution Records

The computation for one target–donor pair proceeds in the following order:

1. Read the actual PEFT scaling and form $D_l = \text{scaling} \cdot B_l A_l$.
2. Compute a fixed-rank numerical factorization and check singular values, finiteness, and reconstruction error.
3. Construct deterministic left/right charts and record $U_l^\top U_l - I$ and $V_l^\top V_l - I$.
4. Recompute $S_l = U_l^\top D_l V_l$ and solve donor-to-target Q_U, Q_V.
5. Save principal angles, overlaps, Procrustes residuals, norm coefficients, dense hashes, predictions, logits, and metrics.
6. Aggregate same/wrong donors within target, then apply task-clustered hierarchical bootstrap.

The minimum pair/module record contains:

```
pair_id
  target_run_id, donor_run_id, target_task, donor_task
  target_seed, donor_seed, same_or_wrong
  module_name, rank, carrier_rank, norm_rule, eval_dtype
geometry
  q_u, q_v, q_u_orth_error, q_v_orth_error
  principal_angles_u, principal_angles_v
  overlap_u, overlap_v, procrustes_residual_u/v
core
  target_s_norm, donor_s_norm, norm_coefficient
  target_s_hash, donor_s_hash, aligned_core_hash
condition
  condition_dense_hashes, writeback_error
  macro_f1, accuracy, nll, ece, predictions, logits
status
  code_hash, data_hash, plan_hash, complete
```

The main matrix contains 400/400 expected and completed pairs; all 40 targets contain three same-task and seven different-task donors; all nine conditions contain 400 rows; the target-norm rule is fixed; and the gauge-invariance flag is passed. The Qwen2.5-3B, Qwen2.5-7B, and Llama-3.2-1B extensions pass the same completeness checks.

B.15 Run Identities and Model Records

The main run identity is

`39b8cc3dcbec2d8a93b98306c59e8b0dbce28195b5c6e875c898a022f20f26b6`.

Table 9: Mean geometric variables by relation group. Angles are in degrees.

Model	Relation	Angle	Overlap	Transport residual
1.5B	same-task	71.11	.1754	1.1726
1.5B	same-interface/different-task	81.46	.0423	1.3055
1.5B	cross-interface	82.50	.0316	1.3188
3B	same-task	72.00	.1661	1.1832
3B	same-interface/different-task	82.11	.0371	1.3137
3B	cross-interface	82.99	.0286	1.3251
Llama-3.2-1B	same-task	72.73	.1564	1.1934
Llama-3.2-1B	same-interface/different-task	82.58	.0329	1.3199
Llama-3.2-1B	cross-interface	83.42	.0249	1.3307

The Qwen2.5-3B and Qwen2.5-7B identities are

`2dc2e93c386961eb408e2cc702082123a2dc70d89050bb53d3767826c8a506be c52a5d512a6161370b4c9914407a85bcd74372ebbceb463307719a2674e964d2.`

The Llama-3.2-1B identity is

`866cd1540272bbb368095e44cc602fd9ca8974f3cc5725568c2763b590fd71f7.`

Model-identity files record model configuration, tokenizer, and safetensors SHA-256 values. The anonymous artifact contains machine-path-free pair-level results, target and relation summaries, seed stratification, configurations, and analysis scripts; it does not redistribute model weights, logits, or per-example predictions.

B.16 Independent Downstream Artifact Records

The downstream artifact is organized by model directory and stores source manifests, environment and protocol files, training and factorization indices, adapter states, pair-level `result.json`, and model-level `summary.json`. Qwen2.5-1.5B-Instruct, Qwen2.5-3B-Instruct, Qwen2.5-7B, and Qwen3-1.7B each complete 9/9 adapters and 72/72 pairs, for 36 adapters and 288 directed pairs in total. Classification retains Macro-F1; recommendation retains NDCG@2 together with Recall@1, Recall@2, MRR, NLL, and ECE.

B.17 Resource Fields

Resource records include parameter count, hidden size, layer count, LoRA target modules, carrier rank, loading memory, peak training memory, peak write-back memory, time per pair, total pair count, and disk usage. FP16, BF16, and FP32 are recorded separately because training precision and dense-intervention evaluation precision serve different roles.

B.18 Relation Geometry and Evidence Map

These group means show descriptive co-movement between geometry and function. Target-aware regressions are exploratory association checks and are not used as a causal adjustment or a universal continuous law. The available summaries contain final C0–C8 values rather than relation-group trajectories at intermediate steps.

B.19 Reference-frame Initialization Results

The warm-start experiment uses AGNews, OBQA, HellaSwag, and EMO with adaptation seeds 701–705. Standard, Same-task Memory, Wrong-task Memory, and Random-spectrum Memory share the same split, data order, optimizer, learning rate, and 100-step budget.

All units record steps 0/10/25/50/100. Using 90% of paired Standard final performance as a threshold, Same-task Memory reaches the threshold 56.7500 ± 37.2147 steps earlier on average.

Table 10: Evidence map for core mechanisms and downstream uses.

Claim	Direct evidence	Interpretation
Registered coordinates	Gauge rotation leaves BTF unchanged; raw controls cross zero	Raw values depend on frame
Same-task recovery	BTF difference $.3908 \pm .1960$	Target-conditioned recovery
Geometry is insufficient	Matched BTF $.3278 \pm .2442$	Angle, overlap, residual matched
Base changes one core	Matched-base $.4039 \pm .1849$; other-base $.0216 \pm .1094$	Donor core fixed
Learned coupling	BTF–SMRC same $.4174 \pm .1860$; wrong $.0337 \pm .0513$	Spectrum and norm controlled
Cross-family relation structure	Llama gains $.454/.143/.024$	400-pair family validation
Reference-frame warm start	Same-task Memory improves final F1 and AULC	Historical core aggregation

Table 11: Task–seed paired effects of Same-task Memory.

Comparison	Final F1	0–100 AULC	10–100 AULC
vs Standard	$.0749 \pm .0470$	17.5230 ± 9.4909	13.4590 ± 8.1131
vs Wrong-task	$.0655 \pm .0318$	15.8243 ± 7.1129	11.5408 ± 5.8689
vs Random-spectrum	$.0180 \pm .0180$	8.2010 ± 4.6148	4.7021 ± 3.2600

B.20 Raw, Unilateral Transport, and Random-core Interpretation

Raw-L and Raw-R mix the donor frame, target carrier, and an unconverted coordinate relation. Their low scores locate the failure of raw-coordinate splicing. Tr-L, Tr-R, and BTF recover same-task function through frame registration. Tr-R has same-task F1 $.720 \pm .142$, slightly above BTF's $.712 \pm .155$, while BTF provides the symmetric target-frame construction. SMRC has near-zero same–wrong selectivity; its BTF–SMRC same-task difference is $.4174 \pm .1860$ and the wrong-task difference is $.0337 \pm .0513$.

B.21 Numerical-precision Sensitivity

The main matrix uses FP32 so condition differences primarily reflect the intervention matrix. BF16 sensitivity records matrix error, maximum logit drift, and prediction agreement. The FP32 gauge audit has factorization error below 10^{-6}, prediction agreement 1.0, and maximum logit drift 3.5×10^{-4}.

C Full Results, Robustness, and Failure Conditions

C.1 Task-level Breakdown

C.2 Cross-task Compatibility Structure

All three models follow the point-estimate ordering same-task $>$ same-interface/different-task $>$ cross-interface. Llama adjacent paired differences are positive; the intermediate Qwen2.5-3B interval includes zero. Donor rows are grouped within target before aggregation.

C.3 Interface-level and Mechanism Controls

C.4 Independent Downstream Extension: Classification and Recommendation

The downstream extension covers four models with SST-5, SuperNI-AGNews, and MovieLens-1M. Classification uses Macro-F1 and recommendation uses NDCG@2.

Table 12: Fiber Memory Macro-F1 after 100 steps.

Task	Standard	Same-task	Wrong-task	Random-spectrum
AGNews	$.7839 \pm .0159$	$.8508 \pm .0060$	$.8076 \pm .0137$	$.8204 \pm .0244$
OBQA	$.7199 \pm .0084$	$.7673 \pm .0058$	$.7090 \pm .0160$	$.7597 \pm .0095$
HellaSwag	$.7507 \pm .0142$	$.7885 \pm .0109$	$.7364 \pm .0129$	$.7653 \pm .0055$
EMO	$.6135 \pm .0097$	$.7611 \pm .0111$	$.6526 \pm .0212$	$.7502 \pm .0154$

Table 13: Task-level BTF results for Qwen2.5-1.5B. Values are means $\pm$ SD over five target-level observations.

Task	Targets	Same	Wrong	Difference	Positive
AGNews	5	.8391	.1893	.6497	5/5
CountWords	5	.3558	.1173	.2384	5/5
DSTC3	5	.8624	.1696	.6928	5/5
EMO	5	.7075	.1539	.5536	5/5
HellaSwag	5	.8018	.5069	.2949	5/5
MuTual	5	.7198	.4526	.2672	5/5
NumerSense	5	.6317	.4225	.2091	5/5
OBQA	5	.7775	.5567	.2208	5/5

C.5 Protocol Audit Details

The four paired quantities used in the audit are

$$G_{\text{transport}}(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{RawL}}_{A\leftarrow B}), \tag{68}$$

$$G_{\text{same}}(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{Zero}}_{A\leftarrow B}), \tag{69}$$

$$G_{\text{learned}}(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{SMRC}}_{A\leftarrow B}), \tag{70}$$

$$G_Q(A,B) = M_A(D^{\text{BTF}}_{A\leftarrow B}) - M_A(D^{\text{RandQ}}_{A\leftarrow B}). \tag{71}$$

Same–wrong measures task selectivity, whereas BTF–SMRC measures learned coupling beyond a random core with the same spectrum and norm.

For condition c, the target-level difference is

$$d_A^{(c)} = \frac{1}{|\mathcal{S}(A)|}\sum_{B\in\mathcal{S}(A)} m_{A,B}^{(c)} - \frac{1}{|\mathcal{W}(A)|}\sum_{B\in\mathcal{W}(A)} m_{A,B}^{(c)}. \tag{72}$$

For task t,

$$\mu_t^{(c)} = \frac{1}{|\mathcal{H}_t|}\sum_{A\in\mathcal{H}_t} d_A^{(c)}, \qquad \mu_{\text{task}}^{(c)} = \frac{1}{|\mathcal{T}|}\sum_{t\in\mathcal{T}} \mu_t^{(c)}. \tag{73}$$

The target-level statistic is the primary unit; pair rows are retained for replication, paired diagnostics, and anomaly detection. Task summaries use equal task weights.

C.6 Three Numerical Error Layers

For module l, let the original effective update be $\Delta W_{B,l}$ and its rank-r reconstruction be $D_{B,l}^{(r)}$:

$$\Delta W_{B,l} = D_{B,l}^{(r)} + E_{\text{svd},B,l}, \qquad \eta_{\text{svd},B,l} = \frac{\|E_{\text{svd},B,l}\|_F}{\|\Delta W_{B,l}\|_F + \varepsilon}. \tag{74}$$

The second layer is subspace-mismatch error, measured by the R_U, R_V expansion in Appendix A. The third layer is carrier write-back error:

$$\eta_{\text{write}}^{(c)} = \frac{\left\|\widetilde{D}^{(c)} - D^{(c)}\right\|_F}{\left\|D^{(c)}\right\|_F + \varepsilon}, \tag{75}$$

where $D^{(c)}$ is the ideal intervention and $\widetilde{D}^{(c)}$ is the reread matrix. Functional aggregation requires write-back and Full-reinjected logit checks to pass.

Table 14: Detailed relation-stratification results. Raw scores and gains are target-level means; intervals use task-clustered then target-level bootstrap resampling.

Model	Relation	Target/donor	Raw F1	Gain [95% CI]
1.5B	same-task	40/120	.7119	.4115 [.2935,.5424]
1.5B	same-interface/different-task	35/90	.4196	.1036 [.0295,.1795]
1.5B	cross-interface	40/190	.2732	-.0272 [-.0474,-.0049]
3B	same-task	40/120	.7494	.3643 [.2243,.5053]
3B	same-interface/different-task	35/90	.4599	.0508 [-.0055,.1124]
3B	cross-interface	40/190	.3741	-.0110 [-.0341,.0113]
Llama-3.2-1B	same-task	40/120	.6425	.4543 [.3510,.5637]
Llama-3.2-1B	same-interface/different-task	35/90	.3339	.1433 [.0906,.1914]
Llama-3.2-1B	cross-interface	40/190	.2121	.0239 [-.0075,.0504]

Table 15: Llama-3.2-1B-Instruct task-level results.

Task	Targets	Same C5	Wrong C5	Same–wrong
AGNews	5	.8159	.2105	.6054
CountWords	5	.3609	.1234	.2375
DSTC3	5	.8688	.2237	.6451
EMO	5	.7610	.2375	.5235
HellaSwag	5	.5732	.3408	.2324
MuTual	5	.5565	.2774	.2791
NumerSense	5	.5589	.1869	.3720
OBQA	5	.6451	.4365	.2086

C.7 Carrier Factorization

For intervention D,

$$D = U_D \Sigma_D V_D^\top, \qquad B_D = U_D \Sigma_D^{1/2}, \qquad A_D = \Sigma_D^{1/2} V_D^\top / \text{scaling}. \tag{76}$$

With carrier rank $r_c = 32$, unused rows and columns are zero-padded when the effective rank is smaller. After write-back the model is reread, and Full versus Full-reinjected probabilities, logits, and predictions are compared. FP32 is used in the main matrix; BF16 sensitivity is retained as an implementation audit.

C.8 Detailed Shared-envelope Effects

Full-deletion effects are 9.7 ± 1.2, 25.3 ± 5.4, 29.9 ± 1.6, and 6.4 ± 2.8 percentage points for MRPC, QQP, QNLI, and RTE. All five confirmatory seeds are positive for each task. After excluding seed 101, the deletion-effect mean is $.1793 \pm .1053$ with interval $[.0825, .2805]$; own-over-random recovery is $.1284 \pm .1050$ with interval $[.0401, .2213]$; and learned-over-matched-random deletion is $.1440 \pm .1008$ with interval $[.0559, .2359]$. Full reinjection preserves predictions with agreement 1.0 across all 20 runs.

For QQP→MRPC, Related–Standard gains at 25/50/100 steps are $+.029 \pm .018$, $+.048 \pm .008$, and $+.055 \pm .012$. For QNLI→RTE they are $+.031 \pm .006$, $+.029 \pm .019$, and $+.055 \pm .076$. The source analysis uses 36 crossed runs; initialization and task effects are stronger and more stable than data-order effects.

C.9 Execution Recovery and Result Integrity

All results are determined jointly by fixed model manifests, data manifests, protocol files, code, and norm rules. A pair enters an aggregate only after all registered modules, conditions, and Full-reinjected checks pass. The main matrix has 400/400 expected/completed pairs, 40 targets, three same-task and seven wrong-task donors per target, and 400 rows for each of nine conditions. The scale and Llama extensions pass the same pair-key and gauge-invariance checks.

The main run identity is

`39b8cc3dcbec2d8a93b98306c59e8b0dbce28195b5c6e875c898a022f20f26b6`.

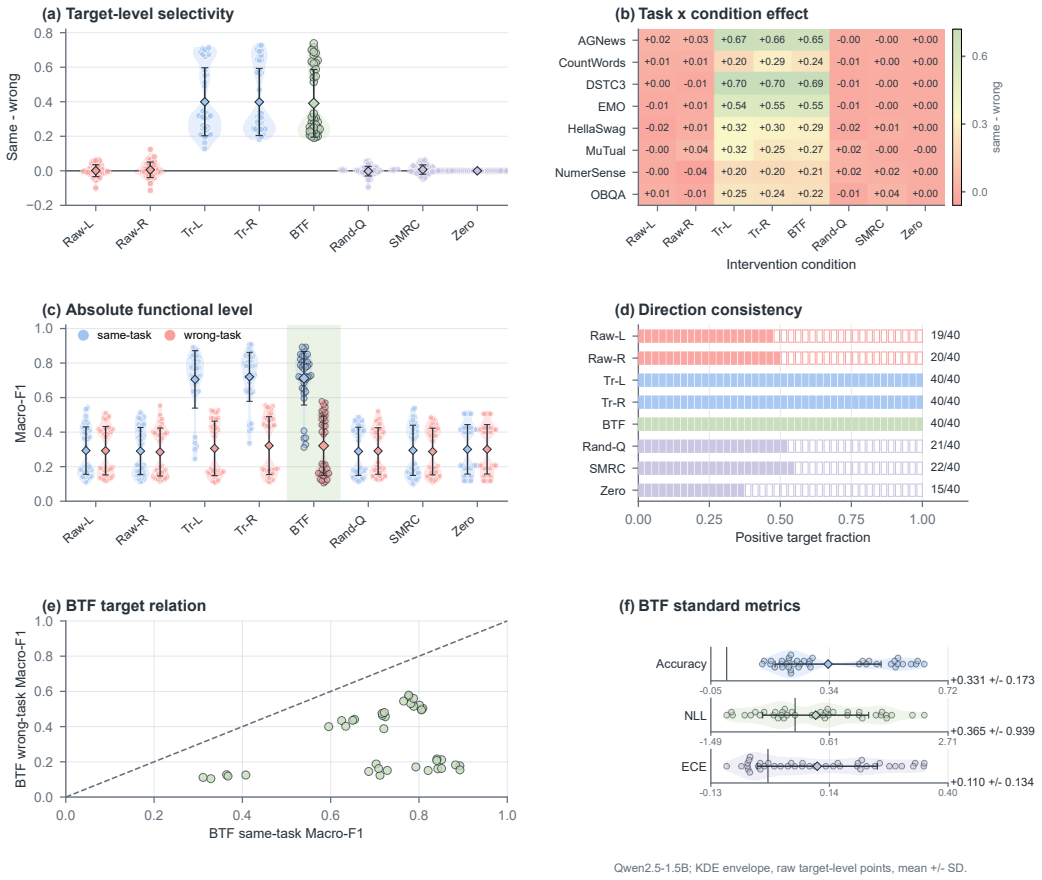


Figure 4: Detailed Qwen2.5-1.5B results: selectivity, task-by-condition effects, directional consistency, target relations, and standard-metric advantages.

Table 16: BTF effects grouped by task interface for Qwen2.5-1.5B.

Interface	Tasks	Targets	Same–wrong mean ± SD
Classification	3	15	.4806 ± .1837
Dialogue state	1	5	.6928 ± .0438
Multiple choice	4	20	.2480 ± .0421

Model-identity files record configuration, tokenizer, and safetensors SHA-256 values. The anonymous artifact stores machine-path-free pair-level results, target/relation summaries, seed stratification, configurations, and analysis scripts.

C.10 Artifact Resource Fields

Resource records include parameter count, hidden size, layer count, LoRA target modules, carrier rank, loading and peak memory, write-back memory, time per pair, total pairs, and disk usage. FP16, BF16, and FP32 are recorded separately. The Memory artifact stores reference frames, per-run transports, norm coefficients, core means, residuals, dense initializations, and short training trajectories.

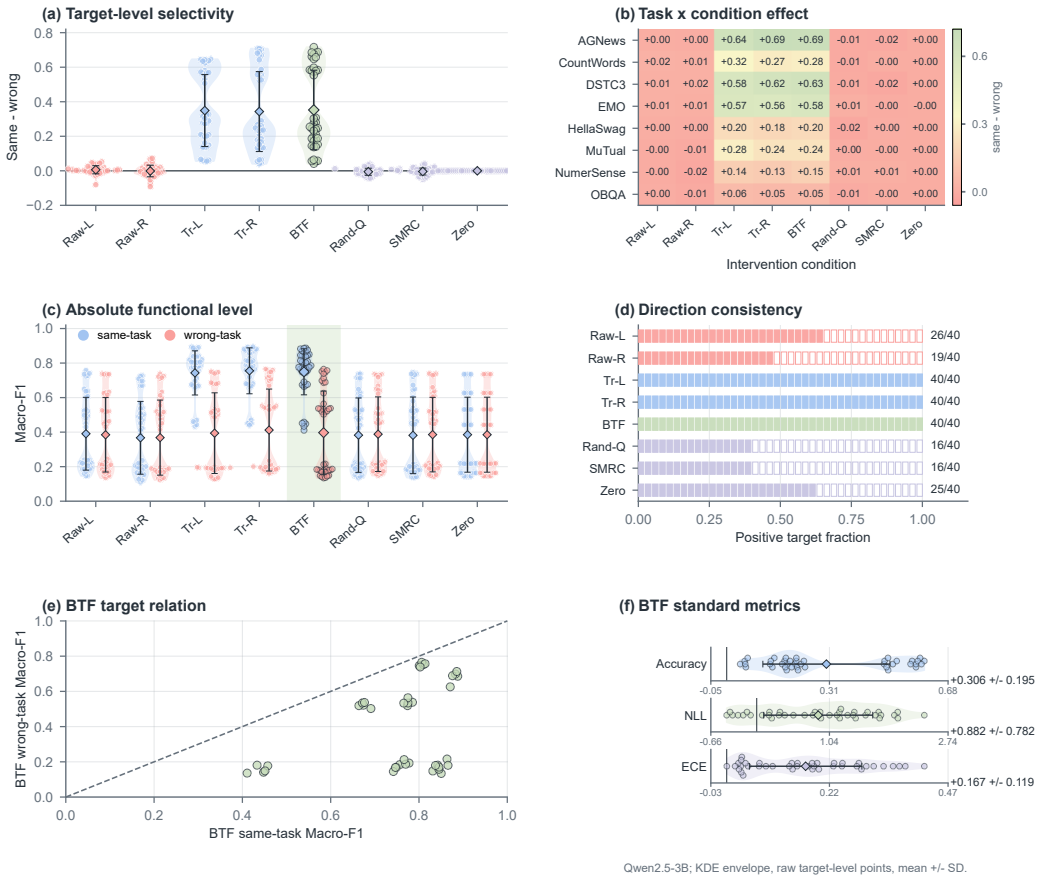


Figure 5: Detailed Qwen2.5-3B results using the same panel definitions as Fig. 4.

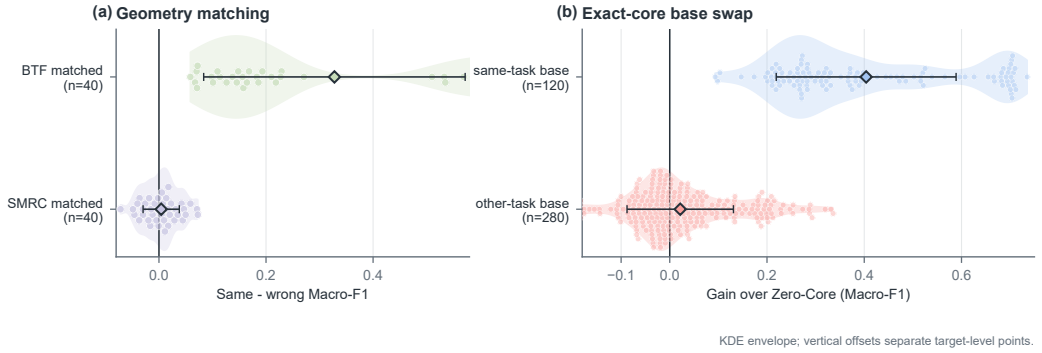


Figure 6: Geometry-matching and exact-core base-swap mechanism controls for Qwen2.5-1.5B.

D Shared Envelopes and Complementary Functional Evidence

Shared-envelope experiments project multiple runs into a common compressed space to identify compact structure, deletion effects, and trajectory initialization. BTF retains each run's local frame to analyze gauge and base conditioning.

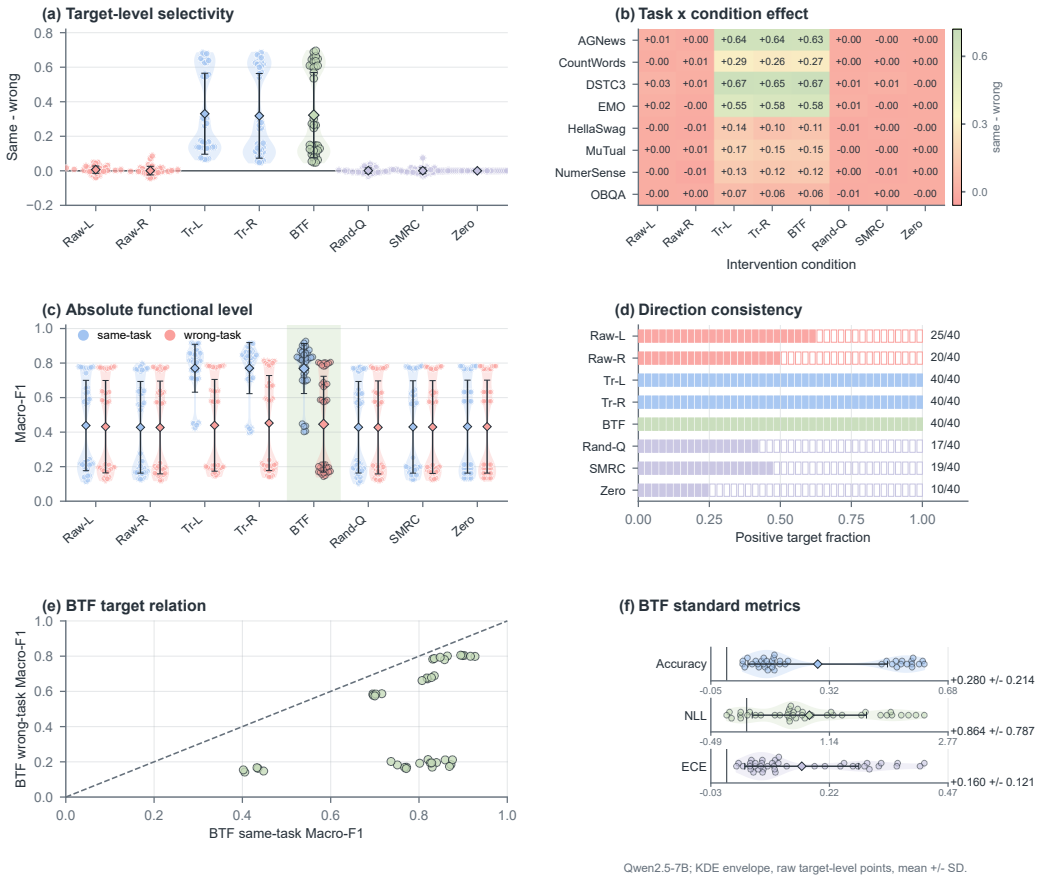


Figure 7: Detailed Qwen2.5-7B results using the same panel definitions as Fig. 4.

Table 17: Unilateral transport and bilateral registration on Llama-3.2-1B.

Condition	Same/wrong F1	Same–wrong [95% CI]	Positive targets
Tr-L (C3)	.6379/.2303	.4076 [.3041, .5222]	40/40
Tr-R (C4)	.6473/.2620	.3853 [.2781, .4978]	40/40
BTF (C5)	.6425/.2546	.3879 [.2775, .5089]	40/40

D.1 Functional Effects of Deletion and Reinjection

Full-deletion effects are 9.7 ± 1.2, 25.3 ± 5.4, 29.9 ± 1.6, and 6.4 ± 2.8 percentage points for MRPC, QQP, QNLI, and RTE. All five confirmatory seeds are positive for each task. Full reinjection preserves predictions with agreement 1.0 across all 20 runs.

D.2 Task-Trajectory Initialization

For QQP→MRPC, Related–Standard gains at 25/50/100 steps are $+.029 \pm .018$, $+.048 \pm .008$, and $+.055 \pm .012$. For QNLI→RTE they are $+.031 \pm .006$, $+.029 \pm .019$, and $+.055 \pm .076$. These trajectories provide an early-adaptation signal dependent on task pair and budget.

D.3 SuperNI Task Matrix and Shared-Envelope Construction

After Frobenius normalization, the shared rank-q bilateral envelope solves

$$\max_{U^\top U=I, V^\top V=I} \sum_{t,s} \left\| U^\top \widehat{\Delta W}_{t,s} V \right\|_F^2 . \tag{77}$$

Table 18: Independent downstream BTF-FR results.

Model	Task	Metric	BTF same	BTF wrong	BTF Δ
Qwen2.5-1.5B	SST-5	Macro-F1	.1746	.1428	+.0318
Qwen2.5-1.5B	AGNews	Macro-F1	.8045	.1994	+.6052
Qwen2.5-1.5B	MovieLens-1M	NDCG@2	.8155	.5862	+.2293
Qwen2.5-3B	SST-5	Macro-F1	.4658	.2523	+.2135
Qwen2.5-3B	AGNews	Macro-F1	.8399	.3430	+.4969
Qwen2.5-3B	MovieLens-1M	NDCG@2	.8510	.6802	+.1708
Qwen2.5-7B	SST-5	Macro-F1	.4733	.1974	+.2759
Qwen2.5-7B	AGNews	Macro-F1	.8385	.2463	+.5922
Qwen2.5-7B	MovieLens-1M	NDCG@2	.8942	.7499	+.1443
Qwen3-1.7B	SST-5	Macro-F1	.1156	.1174	-.0018
Qwen3-1.7B	AGNews	Macro-F1	.6876	.2224	+.4652
Qwen3-1.7B	MovieLens-1M	NDCG@2	.8132	.6487	+.1645

Table 19: Energy coverage of the shared envelope (mean ± SD across runs).

Model	Rank	Runs	Module mean	Global weighted
RoBERTa-base	2	20	$.0338 \pm .0418$	$.0532 \pm .0185$
RoBERTa-base	16	28	$.1102 \pm .0898$	$.1520 \pm .0468$
BERT-base	16	8	$.1368 \pm .1074$	$.2054 \pm .0366$

For one update, $P = U(U^\top \Delta W V)V^\top$ and $E = \Delta W - P$, with $\|\Delta W\|_F^2 = \|P\|_F^2 + \|E\|_F^2$.

E Artifact, Algorithm, and Reproducibility Information

E.1 Operational Algorithm for One Pair

1. Read PEFT scaling and form $D_l = \text{scaling} \cdot B_l A_l$.
2. Factorize D_l at fixed rank and record reconstruction error.
3. Construct deterministic charts and record orthogonality residuals.
4. Recompute $S_l = U_l^\top D_l V_l$ and solve Q_U, Q_V.
5. Save angles, overlap, residuals, norm coefficients, hashes, logits, metrics, and completion status.
6. Aggregate donors within target and apply the task-clustered bootstrap.

E.2 Core Pseudocode

```
for target in confirmatory_runs:
    A = load_factorization(target)
    for donor in registered_donors(target):
        B = load_factorization(donor)
        for module in target_modules:
            qu = procrustes(B.U[module], A.U[module])
            qv = procrustes(B.V[module], A.V[module])
            s_star = norm_match(B.S[module], A.S[module])
            dense = (A.U[module] @ qu.T @ s_star
                     @ qv @ A.V[module].T)
        evaluate_and_write(all_conditions, atomic=True)
aggregate_by_target_then_task()
bootstrap_task_clustered()
```

E.3 Artifact Schema and Integrity

Pair/module records store run identities, task and seed identities, ranks, frame shapes and hashes, transports, angles, overlaps, residuals, norm coefficients, dense hashes, predictions, logits, metrics, code hash, data hash, protocol hash, and completion status. The main matrix contains 400/400

Table 20: Rank-2 functional interventions (mean ± SD across five confirmatory seeds).

Task	Base	Full	P-only	E-only	50% deletion	Matched random	E+random	E+own
MRPC	.684 ± .000	.796 ± .012	.702 ± .010	.699 ± .006	.744 ± .015	.758 ± .019	.723 ± .025	.791 ± .015
QQP	.368 ± .015	.767 ± .016	.671 ± .029	.513 ± .065	.671 ± .046	.723 ± .037	.583 ± .104	.767 ± .017
QNLI	.500 ± .008	.811 ± .012	.564 ± .029	.512 ± .020	.668 ± .061	.759 ± .056	.548 ± .037	.812 ± .010
RTE	.527 ± .000	.607 ± .026	.532 ± .007	.544 ± .015	.570 ± .024	.595 ± .029	.561 ± .032	.586 ± .021

Table 21: Early adaptation using task trajectories as initialization signals.

Transfer	Steps	Standard	Related	Unrelated	Random spectrum	Oracle
QQP→MRPC	25	.683 ± .001	.712 ± .016	.691 ± .009	.681 ± .004	.732 ± .010
QQP→MRPC	50	.681 ± .003	.729 ± .005	.719 ± .006	.688 ± .005	.755 ± .017
QQP→MRPC	100	.683 ± .001	.738 ± .011	.725 ± .002	.695 ± .018	.769 ± .010
QNLI→RTE	25	.503 ± .012	.534 ± .006	.548 ± .014	.502 ± .019	.569 ± .011
QNLI→RTE	50	.508 ± .017	.537 ± .008	.544 ± .015	.515 ± .015	.586 ± .027
QNLI→RTE	100	.514 ± .025	.569 ± .054	.551 ± .020	.540 ± .037	.614 ± .029

completed pairs and complete C0–C8 records; C0 is target-native while C1–C8 are pair-level donor conditions. The scale and Llama extensions pass the same completeness checks.

E.4 Fiber Memory and Downstream Artifacts

The Fiber Memory artifact stores reference frames, transports, norm coefficients, core means, residuals, dense initializations, and short adaptation trajectories. The downstream artifact stores model manifests, protocols, adapter states, pair-level results, and model-level summaries. Machine-specific paths are removed from the anonymous release; model identities are recorded by configuration and checksum.

E.5 Resource Fields

Resource records include parameter count, hidden size, layer count, LoRA target modules, carrier rank, loading and peak memory, write-back memory, pair time, total pairs, and disk usage. FP16, BF16, and FP32 are recorded separately.

Qwen2.5-3B uses 72 q/v matrices, Qwen2.5-7B uses 56, and Llama-3.2-1B uses 32. The scale and family extensions complete 400 directed pairs each. The Llama construction gate passes all eight tasks with at least a 0.05 Macro-F1 gain over base.

E.6 Geometric Variables and Matching Rules

For $L \in \{32, 56, 72\}$, the principal-angle summary is

$$\bar{\theta}_{A,B} = \frac{1}{2L} \sum_{l=1}^{L} \left(\frac{1}{r} \sum_{j=1}^{r} \theta_{U,l,j} + \frac{1}{r} \sum_{j=1}^{r} \theta_{V,l,j} \right). \tag{78}$$

The artifact stores within-module medians, maximum angles, overlaps, and residuals. Geometry-matched controls use only precomputed geometry and complete matching before functional evaluation. Exact-core base-swap fixes the donor core and changes only the target base and transport.

E.7 Functional Effects and Auxiliary Recovery

$$G_{A \leftarrow B} = M_A(D^{\mathrm{BTF}}_{A \leftarrow B}) - M_A(D^{\mathrm{Zero}}_{A \leftarrow B}). \tag{79}$$

When stable Full Target and Zero-Core references exist,

$$\rho_{A \leftarrow B} = \frac{M_A(D^{\mathrm{BTF}}_{A \leftarrow B}) - M_A(D^{\mathrm{Zero}}_{A \leftarrow B})}{M_A(D^{\mathrm{FT}}_{A}) - M_A(D^{\mathrm{Zero}}_{A \leftarrow B})}. \tag{80}$$

The main decisions use raw gain, target-level same–wrong, and paired BTF–SMRC; ρ is auxiliary because its denominator can be small.

Table 22: Heterogeneous SuperNI task matrix.

Task	SuperNI ID	Type	Original labels
AGNews	task379	News-topic classification	World/Sports/Business/Sci-Tech
OBQA	task1286	Science-knowledge multiple choice	A/B/C/D
HellaSwag	task1389	Commonsense event completion	A/B/C/D
EMO	task517	Dialogue emotion recognition	angry/happy/other/sad
MuTual	task611	Multi-turn dialogue choice	A/B/C/D
NumerSense	task1360	Numerical commonsense	a/b/c/d
CountWords	task163	Symbol counting	1/2/3/4
DSTC3	task1500	Dialogue state tracking	Cheap/Don't Care/Expensive/Moderate

E.8 Hierarchical Statistics and Numerical Error

$$d_A^{(c)} = \frac{1}{|\mathcal{S}(A)|} \sum_{B \in \mathcal{S}(A)} m_{A,B}^{(c)} - \frac{1}{|\mathcal{W}(A)|} \sum_{B \in \mathcal{W}(A)} m_{A,B}^{(c)}. \tag{81}$$

Standard deviations use target-level observations. Donor rows are not independent. Bootstrap resamples task clusters and then targets; factorization, base-compatibility, and write-back errors are recorded in the artifact.

E.9 Carrier Write-back

$$D = U_D \Sigma_D V_D^\top, \qquad B_D = U_D \Sigma_D^{1/2}, \qquad A_D = \Sigma_D^{1/2} V_D^\top / \text{scaling}. \tag{82}$$

The written model is reread and compared with the ideal intervention using matrix, probability, logit, and prediction checks.