SUPPORT BEFORE SIGN: IDENTIFYING DEFERRAL GAIN IN LLM CASCADES

Anonymous authors
Paper under double-blind review

ABSTRACT

A cascade must decide whether replacing a small language model's answer with a larger model's would improve the graded outcome. Under answer-factored grading, this signed deferral gain factorizes into the probability of answer disagreement and the expected benefit conditional on disagreement. The factors require different supervision. Unlabeled paired outputs point-identify disagreement support but leave the signed margin undetermined. Under canonical exact-match grading and an unrestricted conditional gold-answer law, we characterize the sharp gain interval and show how observationally equivalent outcome laws induce opposite routing orders. Selective outcome labeling identifies the missing margin under conditional independence and positivity. This information structure motivates *Support Before Sign*: learn support from paired outputs without gold labels, learn the margin from labeled disagreements, and route on their product. On a fixed model pair, signed supervision at the main labeled budget improves integrated accuracy over call rate under both the original and equal-weight task mixtures. Its matched-token-cost benefit depends on the mixture, while observation-matched direct estimators perform better at some low-deferral operating points. The results distinguish three questions: what paired outputs identify, what signed labels add, and how the estimator converts that information into a routing policy.

1 INTRODUCTION

A two-model cascade answers a query with a small model and escalates some fraction of traffic to a large one (Chen et al., 2024; Gupta et al., 2024). The router must commit before the large model runs, so it uses the query and the small model's trace. The signals usually available at that point describe how reliable the small model's answer is: token entropy and sequence likelihood, agreement across sampled reasoning paths (Wang et al., 2023), a self-verification pass (Weng et al., 2023), learned self-assessment (Kadavath et al., 2022), or semantic uncertainty (Kuhn et al., 2023).

The value of escalation depends on the change in graded outcome when the large model replaces the small model's answer. A query on which the small model looks unreliable but the large model is equally wrong is a wasted call; a query on which the small model is confidently wrong and the large model is right is the call worth making. The decision target is therefore the *signed deferral gain*

$$g(z) \;=\; \mathbb{E}[Y_\ell - Y_s \mid Z = z]\,, \tag{1}$$

where Y_s, Y_ℓ are the graded outcomes of the small and large model and Z is the router-visible information. For binary correctness, this is the established Bayes-optimal target for a two-model classification cascade (Jitkrittum et al., 2023); our focus is the information needed to learn it.

Learning signed gain requires graded outcomes. Paired model outputs carry a different kind of information: after generation, their answers can be compared without additional grading. If grading depends on a response only through the answer it commits to, identical answers receive identical scores, so the gain vanishes wherever the two models agree. Writing D for the indicator that the answers differ, $(Y_\ell - Y_s)(1 - D) = 0$ pointwise and

$$g(z) \;=\; \underbrace{\delta(z)}_{\text{support}} \cdot \underbrace{m(z)}_{\text{sign}}, \qquad \delta(z) = \mathbb{P}(D = 1 \mid Z = z), \quad m(z) = \mathbb{E}[Y_\ell - Y_s \mid D = 1, Z = z]. \tag{2}$$

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.

The two factors require different sources of supervision. Running both models offline on unlabeled traffic reveals D, so unlabeled paired outputs point-identify δ. For canonical exact-match grading with an unrestricted gold-answer law, the same unlabeled law permits the large model to win every disagreement or lose every one. The sharp identified set for $g(z)$ is therefore $[-\delta(z), \delta(z)]$. These endpoint laws induce opposite gain orderings; their routing regrets sum to the spread between the largest and smallest support mass selectable at the given deferral rate. Outcome labels collected on disagreements identify the missing factor.

> *Unlabeled pairs reveal where substitution can matter; outcome labels determine whether it helps.*

This division of supervision motivates **Support Before Sign**. It learns $\hat{\delta}$ from unlabeled paired outputs, then learns $\hat{m}$ from outcome labels on a sample of disagreements, and routes on $\hat{g}(z) = \hat{\delta}(z)\ \mathrm{clip}(\hat{m}(z), -1, 1)$. The first factor also provides a routing policy before signed labels are collected.

We evaluate the two stages, their product, and direct gain estimators given the same observations. On the original task mixture, support improves integrated accuracy over call rate relative to reliability signals and reduces serving token cost at matched accuracy. At the main labeled budget, signed supervision further improves integrated routing AUC under both the original and equal-weight mixtures. Translating that gain into token savings depends on the mixture and operating point. An interaction direct estimator outperforms the factorized estimator at some low-deferral and cost operating points. The experiments therefore separate the contribution of supervision from the performance of the estimator using it.

Contributions. (i) We characterize the supervision structure of signed deferral gain: a conditional support–margin decomposition, a sharp identified set under canonical exact-match grading, and a complementary-regret identity for two observationally equivalent endpoint laws. (ii) We identify the margin from selectively labeled disagreements and construct a two-stage estimator that combines it with support learned from unlabeled paired outputs. (iii) We evaluate this construction with observation-matched direct controls, paired query resampling, and task-mixture and token-cost analyses. These comparisons quantify the signed-stage increment and distinguish call-rate gains from token-cost gains across task mixtures and operating points.

2 Related Work

Routing, cascades and the deferral target. Cascades answer cheaply and escalate selectively (Chen et al., 2024); routers select a model per query (Ong et al., 2025; Shnitzer et al., 2023); recent work unifies the two (Dekoninck et al., 2025), and benchmarks report cost–quality frontiers over model pools (Hu et al., 2024). Related inference strategies improve output quality by ensembling model responses (Jiang et al., 2023) or reduce generation cost by verifying draft tokens (Leviathan et al., 2023).

The signed benefit of substitution is an established decision target. Jitkrittum et al. (2023) prove that the Bayes-optimal two-model cascade rule thresholds the difference in the two models' correctness probabilities, and fit that signed gap directly by least squares on $\mathbf{1}[y = h_\ell] - \mathbf{1}[y = h_s]$. What differs across this literature is the estimand actually fitted and the supervision it assumes. Gupta et al. (2024) train a post-hoc LM-cascade router on the *binary event* that the large model is correct while the small model is not; their translation experiments instead regress a signed quality difference. Ding et al. (2024) route each query to one model. Their transformed probabilistic router estimates $\mathbb{P}[q(A_s) - q(A_\ell) \geq -t]$, a tail probability of the quality gap at a tuned threshold. This decision precedes model generation, whereas our cascade can use the small model's trace. Reliability signals for generated answers include token-level confidence, self-consistency across sampled reasoning paths (Wang et al., 2023), self-verification (Weng et al., 2023), learned self-assessment (Kadavath et al., 2022; Lin et al., 2022a), and semantic uncertainty (Kuhn et al., 2023). Chain-of-thought prompting elicits intermediate reasoning (Wei et al., 2022), while supervised process verifiers assess individual steps (Lightman et al., 2024). AutoMix uses self-verification to guide escalation (Aggarwal et al., 2024). Label-free methods use several distinct targets. C3PO optimizes agreement with a strongest-model reference (Valkanas et al., 2025); Smoothie estimates latent output quality from a

graphical model of multiple models' output embeddings (Guha et al., 2024). Gorbett & Jana (2026) score a generated answer using a verifier model's perplexity and entropy, while Hamidieh et al. (2026) combine within-model and cross-model semantic variation to estimate uncertainty. Agreement-based cascades use consensus within a model ensemble to decide whether to escalate (Kolawole et al., 2025), including semantic agreement for open-ended generation (Soiffer et al., 2025).

In C3PO's two-model reduction, the reference-agreement objective is equivalent to support routing (Appendix F). Our focus is the information structure connecting these supervision regimes: answer-factored grading makes realized gain zero on agreement, so paired outputs identify one factor of $g = \delta m$ without outcome labels, and grading sampled disagreements identifies the other.

Deferral, abstention, and learning from disagreement. Classification with a reject option trades coverage for accuracy (Chow, 1970; El-Yaniv & Wiener, 2010; Geifman & El-Yaniv, 2017; Cortes et al., 2016), and learning to defer trains a predictor jointly with a rule that passes hard cases to a costly expert (Madras et al., 2018; Mozannar & Sontag, 2020; Verma & Nalisnick, 2022; Verma et al., 2023). This literature models predictor–expert complementarity and also studies limited expert predictions with gold targets available (Charusaie et al., 2022). Here, unlabeled paired outputs identify support, and grading only sampled disagreements identifies the signed margin. Disagreement among hypotheses has long been used as a label-free signal: Query by Committee selects points on which hypotheses disagree (Seung et al., 1992; Freund et al., 1997; Hanneke, 2014), co-validation compares classifiers through disagreement on unlabeled data (Madani et al., 2004), and domain-adaptation bounds control target risk through a disagreement discrepancy (Ben-David et al., 2010). We characterize disagreement bounds relative to router-visible information and identify the sharp set of signed gains compatible with unlabeled observations. This follows the partial-identification approach of characterizing the parameter values consistent with an observed law (Manski, 1990; Imbens & Manski, 2004); our complementary-regret identity then compares the two endpoint laws.

3 Deferral Gain and Its Information Structure

This section derives the estimator: what the router may see, why the decision target factorizes, what each factor costs in supervision, and the result that makes the second factor recoverable. Proofs are in Appendix A; $(\mathcal{Z}, \mathcal{A})$ is standard Borel, so regular conditional distributions exist and conditional statements hold for P_Z-almost every z.

3.1 Decision target and router information

A query X is answered by a small model s and, if the router defers, by a large model ℓ. Model j returns a response with extracted answer A_j and graded outcome $Y_j \in [0, 1]$. Write $H = Y_\ell - Y_s$ for the realized gain and $D = \mathbf{1}\{A_s \neq A_\ell\}$ for disagreement, with answer equality decided by a normalizer shared with grading.

The router acts after the small model has responded and before the large model runs. It sees $Z = \phi(X, \text{small-model trace})$: query-text features, the small model's confidence summaries, and its reduced hidden states. It does not see A_ℓ, any large-model state, or any gold outcome. Appendix B specifies the input features.

A routing policy is a measurable $\pi : \mathcal{Z} \to [0, 1]$ giving the probability of deferring, randomized so exact rates stay attainable when the score has atoms. For a budget B let $\Pi_B = \{\pi : \mathbb{E}[\pi(Z)] = B\}$. Expected graded performance is $\mathbb{E}[Y_s + \pi(Z)H]$, so within Π_B a policy's value is determined by $\mathbb{E}[\pi(Z)g(Z)]$. Ranking queries by an estimate of g is therefore the object of interest.

3.2 The structural zero and the support–sign factorization

Assumption 1 (Answer-factored grading)**.** *For every gold answer y the grade of a response depends on the response only through the answer it commits to: there is a function ρ with $Y_j = \rho(A_j, y)$ for $j \in \{s, \ell\}$.*

Assumption 1 holds for exact-match, canonical-numeric and multiple-choice grading under a shared normalizer, the regime our evaluation uses. It is one-way: identical answers must receive identical

grades, while distinct answers may receive equal grades. Disagreement is therefore an *outer* support for $\{H \neq 0\}$; Appendix A.6 bounds the effect of an imperfect equivalence rule.

Proposition 1 (Support identity and factorization)**.** *Under Assumption 1, $H(1-D) = 0$ almost surely, and therefore*

$$g(z) = \delta(z)\, m(z), \qquad \delta(z) = \mathbb{P}(D = 1 \mid Z = z), \qquad m(z) = \mathbb{E}[H \mid D = 1, Z = z]$$

at every z with $\delta(z) > 0$; $g(z) = 0$ wherever $\delta(z) = 0$, where we set $m(z) = 0$.

The support target D is obtained by comparing answers; the margin target H on disagreements requires graded outcomes. The factors therefore use different sources of supervision.

3.3 What unlabeled paired outputs identify

D is a deterministic function of (A_s, A_ℓ), so the unlabeled law of (Z, A_s, A_ℓ) — obtained by running both models once, offline, and comparing answers — point-identifies δ. The signed gain satisfies the following bound.

Theorem 1 (Conditional partial identification)**.** *Under Assumption 1 and grades in $[0,1]$, every outcome law compatible with the unlabeled law of (Z, A_s, A_ℓ) satisfies $g(z) \in [-\delta(z), \delta(z)]$. For exact-match grading $\rho(a,y) = \mathbf{1}\{a = y\}$ on a common canonical answer space, with arbitrary conditional gold-answer kernels $P(dy \mid Z, A_s, A_\ell)$, this interval is the identified set. This model permits the gold answer to equal either observed answer; it imposes neither a fixed gold mapping nor additional conditional-independence restrictions.*

The bound follows from the structural zero and bounded gain. In the exact-match model, either endpoint is attainable: taking the gold answer to be A_ℓ on disagreements gives $g = \delta$, while taking it to be A_s gives $g = -\delta$. These laws leave the observed answers unchanged, and their mixtures attain the intervening gains. Thus additional unlabeled data cannot determine the sign wherever $\delta > 0$ in this model. Outcome restrictions can narrow the set: knowing $m(z) \geq 0$ gives $[0, \delta(z)]$ (Appendix A.1).

The two endpoint extensions also give an exact relation between routing regrets.

Theorem 2 (Routing ambiguity between endpoint laws)**.** *Fix $B \in (0,1)$. Under the exact-match model with unrestricted conditional gold-answer law in Theorem 1, there are two outcome laws P^+, P^- inducing the same unlabeled law of (Z, A_s, A_ℓ) and satisfying Assumption 1, with $g_+ = \delta$ and $g_- = -\delta$, such that every $\pi \in \Pi_B$ satisfies the identity*

$$R_B^+(\pi) \;+\; R_B^-(\pi) \;=\; U_B - L_B,$$

Here $R_B^\pm$ is regret against the exact-rate oracle in each law, with $U_B = \sup_{\pi \in \Pi_B} \mathbb{E}[\pi(Z)\delta(Z)]$ and $L_B = \inf_{\pi \in \Pi_B} \mathbb{E}[\pi(Z)\delta(Z)]$. Consequently any policy $\tilde{\pi}_B$ built from the unlabeled law alone has $\max\{R_B^+(\tilde{\pi}_B), R_B^-(\tilde{\pi}_B)\} \geq \frac{1}{2}(U_B - L_B)$.

The identity compares the two endpoint laws: improving against one increases regret against the other by the same amount. When $U_B > L_B$, every common policy incurs at least half this spread in one of them. If $U_B = L_B$, both endpoint regrets vanish for every exact-rate policy. In particular, constant δ gives constant endpoint gains and zero spread. With constant $\delta > 0$ and nondegenerate Z, the full identified set also permits gains that vary across Z.

Unlabeled paired outputs leave both endpoint laws compatible with the observations. Graded disagreement outcomes distinguish them; under the sampling conditions below, their population law identifies the conditional signed margin.

3.4 Recovering the sign from labeled disagreements

Outcome labels reveal gain only on disagreements: on agreement, $H = 0$ is already known. We therefore sample outcome labels within disagreements. The following conditions identify the margin under this selective design.

Proposition 2 (Identification under selective labeling)**.** *Let R indicate that a query is selected for outcome labeling, and suppose $R \perp (Y_s, Y_\ell) \mid (Z, D)$. Let $e(z) = \mathbb{P}(R = 1 \mid Z = z, D = 1) > 0$*

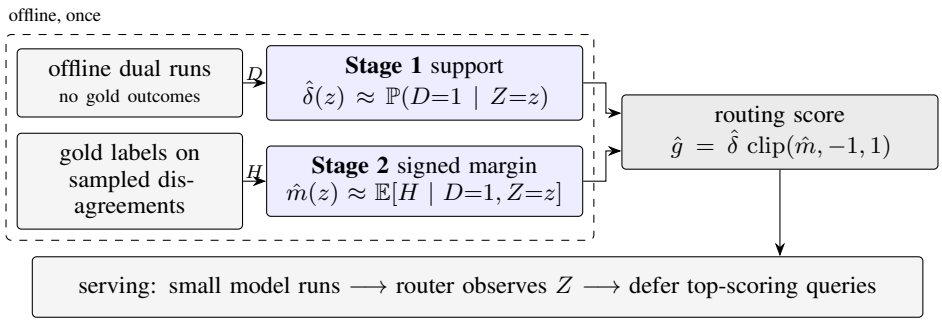

Figure 1: Support Before Sign. The two stages are fitted offline on the supervision that identifies each factor, and multiplied at serving time. The router never sees large-model information before it decides.

for P_Z-almost every z with $\delta(z) > 0$. On this support, the observed population law identifies

$$m(z) = \mathbb{E}\left[\frac{R}{e(Z)}\, H \;\middle|\; D = 1, Z = z\right],$$

Unlabeled paired outputs identify δ; disagreement labels identify m under positivity. Under Assumption 1, these determine $g = \delta m$. Finite-label estimation additionally depends on the representation and fitted model. At zero label budget, m remains defined where $\delta > 0$ but is not point-identified in the unrestricted exact-match model, and the supervised signed head has no labels to fit. Agreement labels reveal no gain beyond the known value $H = 0$ there and do not identify m on disagreements (Corollary 1). Appendix A.4 shows that plug-in estimation error separates into a support term and a signed-margin term, one per factor. Appendix A additionally covers when ranking by a score is exact-rate optimal, a threshold-crossing identity for the regret, a label-free cost certificate, imperfect equivalence and partial credit, and sketches an extension to multiple downstream options.

4 Support-Before-Sign Routing

The estimator is the factorization of Proposition 1 with each factor fitted on the supervision that identifies it: δ from unlabeled paired outputs (Theorem 1), m from outcome labels on sampled disagreements (Proposition 2). Figure 1 shows the procedure end to end.

4.1 Stage 1: support from unlabeled paired outputs

Run both models once, offline, on a pool of queries and record only whether their extracted answers agree. This yields (Z_i, D_i) with no gold outcomes. Fit $\hat{\delta}(z) \approx \mathbb{P}(D = 1 \mid Z = z)$ by cross-fitted regularized logistic regression on a reduced representation of Z. The representation — its dimension, a shrinkage multiplier and the regularization strength — is selected by nested cross-validation *within the pool*, with D as the selection target, so neither gold outcomes nor test rows enter the choice. Because the target is D and not H, this stage consumes no outcome labels.

4.2 Stage 2: signed margin from labeled disagreements

Sample disagreement cases uniformly from the pool and obtain gold outcomes for them. Fit $\hat{m}(z) \approx \mathbb{E}[H \mid D = 1, Z = z]$ on those rows by ridge regression and clip it to $[-1, 1]$, which the plug-in bound of Appendix A.4 requires. Sampling uniformly within the disagreement stratum gives a constant propensity e, so Proposition 2 applies with no reweighting inside the stage. Agreement rows carry the known value $H = 0$ without additional grading; they identify nothing about m, but they are available to any estimator of g, which is what makes the comparison of Section 5.3 observation-matched.

4.3 SERVING

After the small model responds, the router ranks queries by $\hat{g}(z) = \hat{\delta}(z)\ \mathrm{clip}(\hat{m}(z), -1, 1)$ and defers the highest-scoring queries at the chosen budget, using the finite-sample rate and tie conventions in Appendix B. With no signed head it uses $\hat{\delta}$ alone. At the population level, ranking by δ is optimal among δ-measurable policies at every exact rate if $s\,\mathbb{E}[m(Z) \mid \delta(Z) = s]$ is nondecreasing (Corollary 2). This condition depends on outcome information and need not hold across tasks (Section 5.5).

The zero-gold support stage uses offline paired generation in place of outcome labels. Amortizing this computation depends on per-query serving costs and the chosen operating point (Section 5.1).

5 EXPERIMENTS

We ask what routing performance support provides, what signed labels add, and how factorized and direct estimators use the same observations. Task weighting and token cost determine how these comparisons translate to different routing budgets.

Setup. We route 15,361 queries from Qwen2.5-1.5B-Instruct to Qwen2.5-7B-Instruct (Qwen Team, 2024), drawn from GSM8K (Cobbe et al., 2021) and MMLU (Hendrycks et al., 2021). The collection contains 14,042 MMLU and 1,319 GSM8K queries, so MMLU is 91.41% of the collection (original mixture A). Splits are task-stratified; paired answers are fixed. Self-consistency and self-verification use extra smaller-model generations (Table 1). Each of the seven fixed splits partitions the collection into a pool of 9,216 and a test set of 6,145. The summary metric integrates accuracy against deferral rate on a 51-point grid; a displayed $\pm$ is the split standard deviation, and split/label-draw intervals summarize resampling stability; query intervals condition on fitted heads. Appendix B gives the implementation and per-split settings.

Grading and outcomes. A shared answer normalizer enforces Assumption 1: agreement implies equal grades. The outcome partition in Table 4 is an implementation check of this identity. Within disagreement, the large model wins 3,553 cases, the small model 986, and both are wrong on 1,296; observing disagreement alone leaves the signed outcome unresolved.

5.1 STAGE 1: LEARNING SUPPORT WITHOUT GOLD OUTCOMES

Table 1 compares $\hat{\delta}$ with reliability-based routers. In integrated routing AUC, support leads token entropy by 0.01363 ± 0.00144, five-sample self-consistency by 0.01414 ± 0.00171, and self-verification by 0.02045 ± 0.00155; every paired difference is positive on all seven splits. Support uses one smaller-model generation, compared with one greedy plus five sampled completions for self-consistency, and two completions for self-verification.

We also report the restricted area $\mathrm{AUC}_{0.2} = 5 \int_0^{0.2} \mathrm{acc}(B)\, dB$. There the support router leads token entropy by 0.00794 ± 0.00206; the difference from self-consistency is -0.00091 $(3/7)$, with an interval spanning zero. At 10% deferral, self-consistency leads by 0.00249, with a paired interval excluding zero. Appendix G gives the additional controls.

A learned correctness probe uses the same fixed representation and classifier family, with gold labels drawn uniformly over the pool. Support has higher mean performance at every evaluated budget and call-rate metric; the 2,000-label AUC difference is 0.00346 ($[0.00287, 0.00402]$). The representation was selected using D (Appendix G).

At a fixed deferral rate two routers issue the same number of large-model calls. For the matched-accuracy comparison we sweep all sorted prefixes and interpolate adjacent policies. At the accuracy token entropy reaches with a 20% deferral rate, the support router reaches the same accuracy at a rate of 0.156992 ± 0.005863, that is with $21.50 \pm 2.93\%$ fewer large-model calls on all seven splits.

Call counts and token costs differ because policies select different queries. At the accuracy of entropy's 20% policy, support reduces token cost by 8.85% on average across seven splits, with a saving in each split (Table 19). This is a serving token proxy; the offline pool and router execution incur additional costs (Section 5.4).

Table 1: Zero-gold routers on the two-task collection, original proportions. Generations count smaller-model completions before routing. Differences are support minus baseline within split; ± is the sample SD across seven splits. The realized-gain reference ranks by test-outcome differences (clairvoyant).

Router	Generations	Routing AUC	Support − baseline
support only	1	0.66771 ± 0.00455	—
token entropy	1	0.65408 ± 0.00456	$+0.01363 \pm 0.00144$
maximum probability	1	0.65294 ± 0.00439	$+0.01477 \pm 0.00168$
self-consistency	6	0.65357 ± 0.00482	$+0.01414 \pm 0.00171$
self-verification	2	0.64726 ± 0.00395	$+0.02045 \pm 0.00155$
random	1	0.63925 ± 0.00417	$+0.02846 \pm 0.00218$
realized-gain reference (clairvoyant)	—	0.75782 ± 0.00443	reference

Table 2: Task-specific signed-stage performance at 2,000 labels. The mixed-task routers are ranked within each task at separate call budgets. Entries average five label fits within each of seven splits. Differences are factorized minus support; the final column counts positive paired split differences.

Task	Metric	Support-only	Factorized	Difference	Positive
MMLU	Full AUC	0.65168	0.65402	+0.00234	7/7
MMLU	Accuracy at 20%	0.59595	0.60348	+0.00753	7/7
GSM8K	Full AUC	0.84347	0.84300	−0.00047	2/7
GSM8K	Accuracy at 20%	0.78923	0.79026	+0.00103	4/7

5.2 Stage 2 and the product

Labels on disagreements identify the margin left undetermined by paired outputs (Theorem 1 and Proposition 2). With 2,000 labeled disagreements (56.4–57.7% of pool disagreements across splits), the factorized score reaches 0.67049 ± 0.00455, improving on Stage 1 by 0.00278 ± 0.00057 on all seven splits.

In the matched 2,000-label factor ablation, support alone scores 0.66771, the signed margin alone scores 0.64859, and the product scores 0.67049: the product exceeds margin-only by +0.02190 on all seven splits and support-only by 0.00278 ± 0.00057. Table 2 separates the within-task increments: full AUC increases on MMLU in every split, while GSM8K has a small negative mean. At 20% deferral, both task means increase. These evaluate the mixed-training router with separate within-task budgets. Tables 7–8 give the full label-budget grid.

5.3 Observation-matched comparisons

Direct estimators share the unlabeled pool, disagreement indicators, agreement zeros, representation and labeled subsets with the factorized estimator. The linear D-aware, interaction and neural direct estimators additionally receive $\hat{\delta}$, computed out of fold for training queries. Controls comprise linear ridge on $[Z, \hat{\delta}]$, interaction ridge on $[Z, \hat{\delta}, \hat{\delta} \odot Z]$, and a width-64 neural network, with fixed tuning grids (Appendix D.1).

Factorized-minus-direct integrated-AUC means are positive at every evaluated budget in the seven-split, five-label-draw analysis (Figure 2). Against the interaction estimator, the mean difference decreases from +0.01607 at 25 labels to +0.00140 at 2,000. At low deferral rates the pattern changes: factorized-minus-interaction $\mathrm{AUC}_{0.2}$ intervals favor factorized routing through 250 labels, span zero at 500, and favor interaction at 1,000 and 2,000. Tables in Appendices D.2 and G report the complete grid.

Those intervals resample splits and label draws. Paired within-task query resampling instead holds seed-0 heads fixed and recomputes budget boundaries (Appendix H). At 2,000 labels, support-minus-entropy and factorized-minus-support have positive intervals on all five call-rate metrics under the original mixture. Against interaction, the integrated-AUC interval spans zero, while the $\mathrm{AUC}_{0.2}$ and

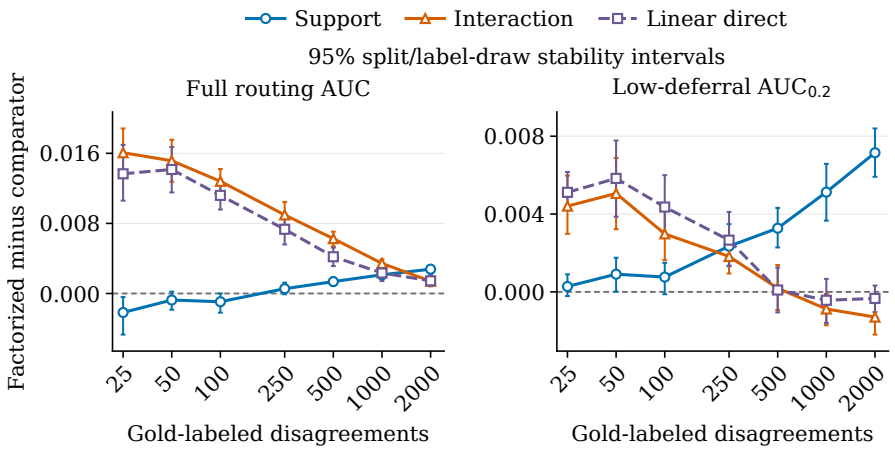

Figure 2: Factorized minus support-only and direct controls, in integrated AUC (left) and normalized $\mathrm{AUC}_{0.2}$ (right). Points are paired means over seven splits and five label replicates; bars show the 95% split/label-draw resampling stability intervals.

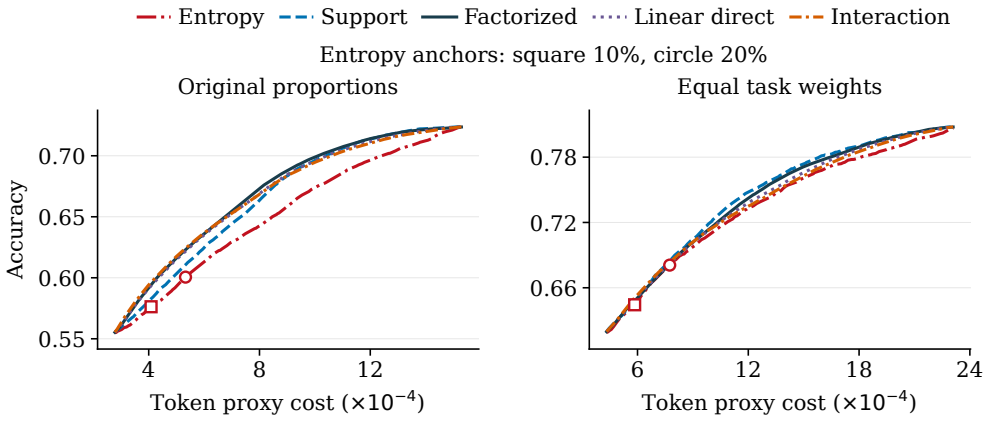

Figure 3: Serving token proxy cost and accuracy at 2,000 labels for supervised routers, with original task proportions (left) and equal task weights (right). Curves average five label fits within each split and then seven split curves on a common cost grid. Markers average entropy's per-split 10% and 20% anchor coordinates; no uncertainty band is shown. Support and entropy use no gold labels.

20% accuracy intervals favor interaction. Against linear D-aware direct, all five intervals span zero, including integrated AUC ($+0.00121$, $[-0.00018, +0.00260]$).

At zero labels, the margin is unidentified. At 25–100 labels the factorized estimator's mean integrated AUC is below Stage 1. At 500, 1,000 and 2,000 labels, all five split/label-draw stability intervals lie above zero (Table 9). The budget curves describe the transition on this collection.

5.4 Serving token cost

The serving proxy charges each query's realized tokens: the small model for every query and the large model only for deferrals. Realized completion lengths determine the cost assigned to each routed query. Entropy's 10% and 20% policies supply anchors; we compare accuracy at their cost and minimum cost at their accuracy over each method's full prefix curve. Adjacent-prefix randomization supplies intermediate points on the empirical test-set frontiers. Appendix I gives the protocol and all intervals.

At 2,000 labels and the 20% anchor, factorized-minus-support accuracy at equal cost is $+0.01216$ ($[0.00737, 0.01718]$) under the original mixture and -0.00595 ($[-0.01457, 0.00292]$) under equal task weights. These seed-0 query intervals average metrics over five fixed label fits. At the 10% anchor the original-mixture increment is also positive, while the equal-weight interval spans zero. All four fixed-cost comparisons with linear D-aware span zero. Interaction performs better at the equal-weight 10% anchor, and costs less at matched accuracy at both original-mixture anchors and the equal-weight 10% anchor. Factorized improves on entropy at both original-mixture anchors; both equal-weight intervals span zero. Figure 3 shows seven-split means separately from these query intervals.

5.5 Task mixture and budget allocation

MMLU dominates the collection. We distinguish task slices of a global policy, equal call-rate budgets within tasks, and equal task weights with a recomputed global boundary. Support-minus-entropy accuracy at 20% is $+0.00634$ under separate within-task budgets and $+0.00982$ under an equal-weight global budget (mixture B). At 2,000 labels, factorized-minus-support full AUC under equal task weights is $+0.00310$, with split/label-draw stability interval $[0.00245, 0.00378]$. Thus the integrated call-rate gain persists under equal weights; its matched-token-cost counterpart is unresolved (Section 5.4). Appendix H reports the equal-weight comparisons, task slices and realized deferral rates. The separate per-task sweep in Appendix E fits each router within its own benchmark and tests the breadth of the support signal under that different fitting protocol.

6 Discussion and Limitations

The decomposition separates two sources of information about substitution. Paired outputs locate its support, and grading disagreements reveals its direction and magnitude. This makes support useful both as a routing score before labels are collected and as a component of a signed gain estimator. The identified set and endpoint regret identity describe what can be learned from each supervision source; the fitted routing comparisons measure how effectively particular estimators use that information.

At the main labeled budget, signed supervision improves integrated call-rate performance under both task mixtures. Its within-task full-AUC increment is concentrated on MMLU, and its serving-token benefit depends on task weighting. The direct controls further separate supervision from parameterization: interaction regression performs better at several low-call and cost operating points, while query intervals leave differences from linear D-aware unresolved. The factorized form provides an explicit allocation of supervision to support and margin; finite-sample performance also depends on the fitted representation and head.

Scope and limitations. The primary evidence concerns one model pair and fixed benchmark generations. Split/label-draw intervals describe variation across partitions and labeled subsets; paired query intervals describe sampling variation conditional on the fitted heads and generations. All matched arms share a representation selected on disagreement, so their comparisons concern head parameterizations within that representation. The support head uses small-model hidden states and requires access to model internals. Support depends on the model pair and query population; the primary experiment does not evaluate transfer. The grading identity holds by construction under the shared normalizer, and sharpness additionally uses an unrestricted conditional gold-answer law. Serving-token accounting uses a fixed model-rate ratio and excludes grading, fitting and router execution; Appendix I separately reports paired-pool amortization. The collection selection and reuse of test queries are described in Appendix B.

7 Conclusion

Answer-factored grading decomposes signed deferral gain into disagreement support and a conditional margin. Paired outputs identify support; selective labels identify the margin under conditional independence and positivity. Support Before Sign fits these factors from their respective supervision sources. On the evaluated model pair, signed supervision improves integrated call-rate performance under both task mixtures. Token savings depend on task weighting, and direct estimators are stronger at some operating points.

AI USE STATEMENT

AI tools assisted with manuscript editing and computational workflows, including experiment automation and result analysis. The authors take responsibility for the final manuscript and results.

REPRODUCIBILITY STATEMENT

Section 3.1 states the information available to the router, and Appendix B specifies its features. Assumptions accompany each result in Section 3; proofs are in Appendix A. Appendix B specifies model and dataset versions, prompts and decoding, answer extraction, split construction, feature extraction and reduction, head architectures, the optimization and validation protocol, the per-split representation settings, the BLAS thread count, the deferral grid, the tie rule and our statistical conventions. Appendix D gives the observation-matched protocol, including how shared labeled subsets are drawn and how the inverse-propensity weights and out-of-fold support feature are constructed. Appendices H and I distinguish query resampling from split and label-draw variation, and specify task weighting and token-cost matching. Reproducing the empirical tables requires the experiment code and per-split predictions in addition to these specifications.

ETHICS STATEMENT

This work studies routing between two language models on public benchmark questions. It introduces no new data collection, involves no human subjects, and releases no model weights. A router changes which model answers a query, so one fitted on a given population can allocate the large model unevenly across question types. Appendix E reports task-level results, and Appendix B specifies the evaluation collection and its selection.

REFERENCES

Pranjal Aggarwal, Aman Madaan, Ankit Anand, Srividya Pranavi Potharaju, Swaroop Mishra, Pei Zhou, Aditya Gupta, Dheeraj Rajagopal, Karthik Kappaganthu, Yiming Yang, Shyam Upadhyay, Manaal Faruqui, and Mausam. AutoMix: Automatically mixing language models. In *Advances in Neural Information Processing Systems*, 2024.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79:151–175, 2010.

Mohammad-Amin Charusaie, Hussein Mozannar, David Sontag, and Samira Samadi. Sample efficient learning of predictors that complement humans. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2972–3005, 2022. URL https://proceedings.mlr.press/v162/charusaie22a.html.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*, 2024.

C. K. Chow. On optimum recognition error and reject tradeoff. *IEEE Transactions on Information Theory*, 16(1):41–46, 1970.

Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. BoolQ: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pp. 2924–2936, 2019.

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

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Corinna Cortes, Giulia DeSalvo, and Mehryar Mohri. Learning with rejection. In *Proceedings of the 27th International Conference on Algorithmic Learning Theory*, pp. 67–82, 2016.

Jasper Dekoninck, Maximilian Baader, and Martin Vechev. A unified approach to routing and cascading for LLMs. In *Proceedings of the 42nd International Conference on Machine Learning*, 2025.

Dujian Ding, Ankur Mallick, Chi Wang, Robert Sim, Subhabrata Mukherjee, Victor Rühle, Laks V. S. Lakshmanan, and Ahmed Hassan Awadallah. Hybrid LLM: Cost-efficient and quality-aware query routing. In *International Conference on Learning Representations*, 2024.

Ran El-Yaniv and Yair Wiener. On the foundations of noise-free selective classification. *Journal of Machine Learning Research*, 11:1605–1641, 2010.

Yoav Freund, H. Sebastian Seung, Eli Shamir, and Naftali Tishby. Selective sampling using the query by committee algorithm. *Machine Learning*, 28:133–168, 1997.

Yonatan Geifman and Ran El-Yaniv. Selective classification for deep neural networks. In *Advances in Neural Information Processing Systems*, pp. 4878–4887, 2017.

Matt Gorbett and Suman Jana. Cross-model disagreement as a label-free correctness signal. *arXiv preprint arXiv:2603.25450*, 2026.

Neel Guha, Mayee F. Chen, Trevor Chow, Ishan S. Khare, and Christopher Ré. Smoothie: Label free language model routing. In *Advances in Neural Information Processing Systems*, volume 37, 2024.

Neha Gupta, Harikrishna Narasimhan, Wittawat Jitkrittum, Ankit Singh Rawat, Aditya Krishna Menon, and Sanjiv Kumar. Language model cascades: Token-level uncertainty and beyond. In *International Conference on Learning Representations*, 2024.

Kimia Hamidieh, Veronika Thost, Walter Gerych, Mikhail Yurochkin, and Marzyeh Ghassemi. Complementing self-consistency with cross-model disagreement for uncertainty quantification. In *International Conference on Learning Representations*, 2026.

Steve Hanneke. Theory of disagreement-based active learning. *Foundations and Trends in Machine Learning*, 7(2-3):131–309, 2014.

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

Qitian Jason Hu, Jacob Bieker, Xiuyu Li, Nan Jiang, Benjamin Keigwin, Gaurav Ranganath, Kurt Keutzer, and Shriyash Kaustubh Upadhyay. RouterBench: A benchmark for multi-LLM routing system. In *ICML 2024 Workshop on Agentic Markets*, 2024.

Guido W. Imbens and Charles F. Manski. Confidence intervals for partially identified parameters. *Econometrica*, 72(6):1845–1857, 2004.

Dongfu Jiang, Xiang Ren, and Bill Yuchen Lin. LLM-Blender: Ensembling large language models with pairwise ranking and generative fusion. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics*, pp. 14165–14178, 2023.

Wittawat Jitkrittum, Neha Gupta, Aditya Krishna Menon, Harikrishna Narasimhan, Ankit Singh Rawat, and Sanjiv Kumar. When does confidence-based cascade deferral suffice? In *Advances in Neural Information Processing Systems*, 2023.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Steven Kolawole, Don Dennis, Ameet Talwalkar, and Virginia Smith. Agreement-based cascading for efficient inference. *Transactions on Machine Learning Research*, 2025. URL `https://openreview.net/forum?id=jn9B7LMlzk`.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *International Conference on Learning Representations*, 2023.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*, pp. 19274–19286, 2023.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. Let's verify step by step. In *International Conference on Learning Representations*, 2024.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *Transactions on Machine Learning Research*, 2022a.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*, pp. 3214–3252, 2022b.

Omid Madani, David M. Pennock, and Gary W. Flake. Co-validation: Using model disagreement on unlabeled data to validate classification algorithms. In *Advances in Neural Information Processing Systems*, 2004.

David Madras, Toniann Pitassi, and Richard Zemel. Predict responsibly: Improving fairness and accuracy by learning to defer. In *Advances in Neural Information Processing Systems*, 2018.

Charles F. Manski. Nonparametric bounds on treatment effects. *The American Economic Review*, 80 (2):319–323, 1990.

Todor Mihaylov, Peter Clark, Tushar Khot, and Ashish Sabharwal. Can a suit of armor conduct electricity? a new dataset for open book question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018.

Hussein Mozannar and David Sontag. Consistent estimators for learning to defer to an expert. In *Proceedings of the 37th International Conference on Machine Learning*, 2020.

Isaac Ong, Amjad Almahairi, Vincent Wu, Wei-Lin Chiang, Tianhao Wu, Joseph E. Gonzalez, M. Waleed Kadous, and Ion Stoica. RouteLLM: Learning to route LLMs with preference data. In *International Conference on Learning Representations*, 2025.

Qwen Team. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*, 2024.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

H. Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual ACM Workshop on Computational Learning Theory*, 1992.

Tal Shnitzer, Anthony Ou, Mírian Silva, Kate Soule, Yuekai Sun, Justin Solomon, Neil Thompson, and Mikhail Yurochkin. Large language model routing with benchmark datasets. *arXiv preprint arXiv:2309.15789*, 2023.

Duncan Soiffer, Steven Kolawole, and Virginia Smith. Semantic agreement enables efficient open-ended LLM cascades. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pp. 2499–2537. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.emnlp-industry.171. URL https://aclanthology.org/2025.emnlp-industry.171/.

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

Antonios Valkanas, Soumyasundar Pal, Pavel Rumiantsev, Yingxue Zhang, and Mark Coates. C3PO: Optimized large language model cascades with probabilistic cost constraints for reasoning. In *Advances in Neural Information Processing Systems*, 2025.

Rajeev Verma and Eric Nalisnick. Calibrated learning to defer with one-vs-all classifiers. In *Proceedings of the 39th International Conference on Machine Learning*, 2022.

Rajeev Verma, Daniel Barrejón, and Eric Nalisnick. Learning to defer to multiple experts: Consistent surrogate losses, confidence calibration, and conformal ensembles. In *Proceedings of the 26th International Conference on Artificial Intelligence and Statistics*, 2023.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V. Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models. In *International Conference on Learning Representations*, 2023.

Yubo Wang, Xueguang Ma, Ge Zhang, Yuansheng Ni, Abhranil Chandra, Shiguang Guo, Weiming Ren, Aaran Arulraj, Xuan He, Ziyan Jiang, Tianle Li, Max Ku, Kai Wang, Alex Zhuang, Rongqi Fan, Xiang Yue, and Wenhu Chen. MMLU-Pro: A more robust and challenging multi-task language understanding benchmark. In *Advances in Neural Information Processing Systems, Datasets and Benchmarks Track*, 2024.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc V. Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, 2022.

Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. Large language models are better reasoners with self-verification. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, 2023.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pp. 4791–4800, 2019.

A Additional Theory and Proofs

Throughout, $(\mathcal{Z}, \mathcal{A})$ is standard Borel so that a regular conditional distribution of (Y_s, Y_ℓ) given Z exists, and all conditional statements hold for P_Z-almost every z. Appendix A.1 proves the results of Section 3; Appendices A.2–A.6 state and prove the further results on ordering, selective labeling, plug-in regret, cost certification and imperfect equivalence.

A.1 Proofs of the main-text results

Proof of Proposition 1. Under Assumption 1, on $\{D = 0\}$ we have $A_s = A_\ell$ and hence $Y_s = \rho(A_s, y) = \rho(A_\ell, y) = Y_\ell$, so $H(1 - D) = 0$ pointwise. Taking conditional expectations, $g(z) = \mathbb{E}[H \mid Z = z] = \mathbb{E}[HD \mid Z = z] = \mathbb{P}(D = 1 \mid Z = z)\,\mathbb{E}[H \mid D = 1, Z = z] = \delta(z)m(z)$ whenever $\delta(z) > 0$, and $g(z) = 0$ otherwise. □

Proof of Theorem 1. Since $H = HD$ and $|H| \le 1$, $|g(z)| \le \mathbb{E}[|H|D \mid Z = z] \le \delta(z)$. For sharpness under the stated exact-match model with arbitrary kernels $P(dy \mid Z, A_s, A_\ell)$, keep the unlabeled law fixed and set $y = A_\ell$ in one extension and $y = A_s$ in another. On agreement both have $H = 0$; on disagreement they have $H = 1$ and $H = -1$, respectively. Their conditional gains are therefore δ and $-\delta$. For any measurable $t : \mathcal{Z} \to [-1, 1]$, choose the first extension with conditional probability $(1 + t(Z))/2$ and the second otherwise. This preserves the unlabeled law and gives $g(z) = \delta(z)t(z)$, realizing every measurable selection of the interval. Where $\delta(z) = 0$ it collapses to $\{0\}$. □

The exact-match construction uses a common canonical space for model and gold answers. For a fixed general answer-factored grader the interval is an outer bound and need not be sharp. Under the exact-match model, the additional restriction $m(z) \ge 0$ gives the identified set $[0, \delta(z)]$: bounded gain gives inclusion, and the construction with $t(z) \in [0, 1]$ gives attainability.

Proof of Theorem 2. Use the two exact-match extensions from the proof of Theorem 1, with $y = A_\ell$ under P^+ and $y = A_s$ under P^-. They have the same unlabeled law and gains $g_+ = \delta$ and $g_- = -\delta$. The oracle values are U_B and $-L_B$, respectively. Fix $\pi \in \Pi_B$ and write $v = \mathbb{E}[\pi(Z)\delta(Z)] \in [L_B, U_B]$. Then

$$R_B^+(\pi) = U_B - v, \qquad R_B^-(\pi) = v - L_B,$$

and adding them gives the identity. Because $\tilde{\pi}_B$ is a function of the unlabeled law alone, it is the same policy under both laws, so the identity applies to it and the corollary follows from $\max\{a, b\} \geq (a + b)/2$. □

A.2 When ranking by a score is optimal

The estimator ranks queries by a scalar score. The following gives a sufficient population ordering condition within the class of policies that use that score.

Proposition 3 (Score-stratum ordering)**.** *Let S be a router score and $r(s) = \mathbb{E}[g(Z) \mid S = s]$. If r has a nondecreasing version, then deferring the largest values of S maximizes expected gain among all S-measurable policies at every exact deferral rate. Conversely, if S has an atomless distribution and top-S routing is optimal at every exact rate, then r has a nondecreasing version.*

By Proposition 1, $r(s) = \mathbb{E}[\delta(Z)m(Z) \mid S = s]$. For $S = \delta$ this reduces to $r(s) = s\,\mathbb{E}[m(Z) \mid \delta(Z) = s]$, the criterion in Corollary 2. Whether it holds depends on population outcomes and is not determined by unlabeled paired outputs alone.

Proof of Proposition 3. Forward direction: among S-measurable policies at an exact rate, expected gain is $\mathbb{E}[\pi(S)r(S)]$ subject to $\mathbb{E}[\pi(S)] = B$ and $0 \leq \pi \leq 1$. This is a fractional knapsack in S; if r is nondecreasing, putting all mass on the largest S is optimal, with randomization used to hit the rate exactly when S has atoms.

For the converse, write $\mu = P_S$. If the inversion set $\{(s, u) : s < u,\ r(s) > r(u)\}$ has zero $\mu \otimes \mu$ measure, each rational superlevel set $\{r > a\}$ agrees μ-almost everywhere with an upper interval. These countably many nested intervals define a nondecreasing version of r. Otherwise, rational cuts of a positive-measure inversion set give t and $a > b$ such that $E = \{S < t, r(S) \geq a\}$ and $F = \{S > t, r(S) \leq b\}$ both have positive probability. At rate $B = \mathbb{P}(S > t)$, choose $0 < \eta \leq \min\{\mathbb{P}(E), \mathbb{P}(F)\}$ and set

$$\pi_\eta(S) = \mathbf{1}\{S > t\} + \frac{\eta}{\mathbb{P}(E)}\mathbf{1}_E - \frac{\eta}{\mathbb{P}(F)}\mathbf{1}_F.$$

This is an S-measurable policy in $[0, 1]$ with expected mass B. Its gain exceeds that of top-S routing by $\eta(\mathbb{E}[r(S) \mid E] - \mathbb{E}[r(S) \mid F]) \geq \eta(a - b) > 0$. Atomlessness removes threshold ties. Optimality at every rate therefore rules out a positive-measure inversion set and gives the required almost-everywhere monotone version. □

A.3 Identifying the sign from selectively labeled disagreements

Outcome labels reveal gain on disagreements. Proposition 2, proved below, identifies the conditional margin under selective labeling. Positivity is essential for its IPW expression: at $e \equiv 0$ the ratio is undefined. The estimand m still exists wherever $\delta > 0$, but unlabeled observations do not point-identify it in the unrestricted exact-match model. We therefore report the supervised signed head as unavailable at zero labels (Appendix D.2). Agreement labels do not resolve this ambiguity.

Corollary 1 (Agreement labels do not shrink the identified set)**.** *Under Assumption 1, $H = 0$ is already known on $\{D = 0\}$. In the model with arbitrary gold-answer kernels $P(dy \mid Z, A_s, A_\ell)$ of Theorem 1, outcome labels collected only on agreement cases do not shrink the identified set for m on disagreement strata.*

Agreement rows provide known-zero gain targets for a finite-sample estimator of g that pools both regions. The observation-matched comparison in Section 5.3 gives every competing estimator those known-zero targets without additional grading.

Proof of Proposition 2. For P_Z-almost every z with $\delta(z) > 0$, positivity makes the ratio well-defined. On $\{D = 1\}$, $\mathbb{E}[RH/e(Z) \mid Z, D = 1] = \mathbb{E}[\mathbb{E}[R \mid Z, D = 1, H]\, H/e(Z) \mid Z, D = 1]$. Under $R \perp (Y_s, Y_\ell) \mid (Z, D)$ the inner conditional expectation is $e(Z)$, which cancels, leaving $\mathbb{E}[H \mid Z, D = 1] = m(Z)$. The observed labeling indicators identify e, and the labeled outcomes identify RH, with $RH = 0$ when $R = 0$. This establishes population identification; estimation from finitely many labels also incurs sampling error. □

Proof of Corollary 1. On agreement the common grade $Y_s = Y_\ell$ may be unknown, but their difference is already zero. Fix the observed law, including all agreement labels. The model leaves the conditional gold law on $\{D = 1\}$ unrestricted, so any compatible extension on disagreement can be combined with that fixed agreement law. In particular, under exact-match grading the endpoint extensions and their mixtures from Theorem 1 remain available. Thus agreement labels do not reduce the identified set for m. □

A.4 What the plug-in costs

Theorem 3 (Factorized plug-in regret)**.** *Let $\hat{g} = \hat{\delta}\,\hat{m}$ with $\hat{m}$ taking values in $[-1, 1]$. At every exact deferral rate B, the routing regret of ranking by $\hat{g}$ satisfies*

$$R_B(\hat{g}) \;\le\; 2\,\mathbb{E}\big|\, g(Z) - \hat{g}(Z) \,\big| \;\le\; 2\Big(\underbrace{\mathbb{E}\big[\delta(Z)\,|m(Z) - \hat{m}(Z)|\big]}_{\textit{sign error}} \;+\; \underbrace{\mathbb{E}\big|\delta(Z) - \hat{\delta}(Z)\big|}_{\textit{support error}} \Big).$$

Clipping $\hat{m}$ to $[-1, 1]$ removes its magnitude from the support-error term. The two terms correspond to the two stages and their different supervision sources.

Corollary 2 (Support-only special case)**.** *Write $\bar{m}(s) = \mathbb{E}[m(Z) \mid \delta(Z) = s]$. If $s \mapsto s\bar{m}(s)$ has a nondecreasing version, ranking by δ, with a constant deferral probability within each tie, maximizes gain among all δ-measurable policies at every exact rate. Sufficient conditions are that m is a positive constant on $\{\delta > 0\}$, or that $\bar{m}$ is nonnegative and nondecreasing.*

Proof of Corollary 2. Since $g = \delta m$, $\mathbb{E}[g(Z) \mid \delta(Z) = s] = s\bar{m}(s)$, so Proposition 3 applies with $S = \delta$. For nonnegative, nondecreasing $\bar{m}$ and $0 \le s_1 \le s_2$, $s_1\bar{m}(s_1) \le s_2\bar{m}(s_1) \le s_2\bar{m}(s_2)$; the constant-margin case follows directly. □

A nondecreasing margin alone does not ensure product ordering. With $\delta = (0.1, 0.9)$ and $\bar{m} = (-1, -0.5)$, the conditional margin is nondecreasing, yet $s\bar{m}(s) = (-0.1, -0.45)$ decreases: the support ordering reverses the gain ordering. Corollary 2 therefore places monotonicity on the product $s\bar{m}(s)$.

The bound in Theorem 3 holds uniformly over exact rates. The realized regret has an exact expression that localizes it to where a policy differs from the oracle.

Proposition 4 (Threshold-crossing identity)**.** *Fix $B \in (0, 1)$ and an oracle threshold τ_B satisfying $\mathbb{P}(g(Z) > \tau_B) \le B \le \mathbb{P}(g(Z) \ge \tau_B)$. Let π_B^* equal one above this threshold, zero below it, and randomize on ties to have expectation B. For every $\pi \in \Pi_B$,*

$$R_B(\pi) = \mathbb{E}\big[|g(Z) - \tau_B|\,|\pi_B^*(Z) - \pi(Z)|\big].$$

The threshold is that of the true gain. Regret is concentrated where the candidate policy differs from the oracle, weighted by distance from this threshold. Differences at the threshold carry zero regret.

Proof of Theorem 3. Let π_B^* and $\hat{\pi}_B$ maximize expected true and estimated gain over Π_B, respectively. Optimality for $\hat{g}$ gives $\mathbb{E}[(\pi_B^* - \hat{\pi}_B)\hat{g}] \le 0$, and hence

$$R_B(\hat{g}) \le \mathbb{E}[\pi_B^*|g - \hat{g}|] + \mathbb{E}[\hat{\pi}_B|g - \hat{g}|] \le 2\mathbb{E}|g - \hat{g}|.$$

Adding and subtracting $\delta\hat{m}$,

$$|\delta m - \hat{\delta}\hat{m}| \le \delta|m - \hat{m}| + |\hat{m}|\,|\delta - \hat{\delta}| \le \delta|m - \hat{m}| + |\delta - \hat{\delta}|,$$

where the last step uses $|\hat{m}| \le 1$, which the clipping enforces. Taking expectations gives the claim. □

Proof of Proposition 4. Both policies have expected mass B, so $R_B(\pi) = \mathbb{E}[(\pi^*_B - \pi)g] = \mathbb{E}[(\pi^*_B - \pi)(g - \tau_B)]$. Above the threshold $\pi^*_B = 1$, below it $\pi^*_B = 0$, and at the threshold the second factor vanishes. Thus the integrand equals $|\pi^*_B - \pi|\,|g - \tau_B|$ everywhere. □

A.5 Declining to defer, with a guarantee

An exact-rate policy must spend its budget even where the expected gain does not repay the call. The support factor alone certifies part of the traffic as not worth deferring, without any outcome labels.

Proposition 5 (One-sided cost certificate)**.** *Under Assumption 1 and grades in* $[0, 1]$*, let* c *be the per-deferral cost in grade units. Since* $g(z) \le \delta(z)$*, every stratum with* $\delta(z) \le c$ *has non-positive cost-penalized utility, so the mass* $\mathbb{P}(\delta(Z) \le c)$ *is provably not worth deferring.*

This is a population statement. A finite-sample version replaces δ by a valid upper confidence bound. Thresholding a point estimate alone does not supply a finite-sample coverage guarantee.

Proof of Proposition 5. Cost-penalized utility of deferring at z is $g(z) - c \le \delta(z) - c$, which is non-positive whenever $\delta(z) \le c$. □

A.6 When the equivalence rule is imperfect

Proposition 6 (Equivalence-leakage bound)**.** *Drop Assumption 1 and let* $\alpha(z) = \mathbb{P}(H \neq 0,\ D = 0 \mid Z = z)$ *be the false-agreement probability. If grades lie in an interval of width* Δ*, then* $|g(z) - \delta(z)m(z)| \le \Delta\,\alpha(z)$.

The factorization degrades linearly in the false-agreement rate and the grading range. False *disagreement*, $\mathbb{P}(H = 0, D = 1 \mid Z = z)$, leaves the identity intact by including zero-gain cases within the disagreement stratum. On our primary evaluation the false-agreement count is 0, so the structural zero holds for all observed agreement rows.

Proof of Proposition 6. Decompose $g(z) = \mathbb{E}[H\mathbf{1}\{D = 1\} \mid Z = z] + \mathbb{E}[H\mathbf{1}\{D = 0\} \mid Z = z]$. The first term is $\delta(z)m(z)$. The second is supported on the false-agreement event and is bounded in absolute value by $\Delta\,\alpha(z)$, since $|H| \le \Delta$. □

A.7 Grading ranges and partial credit

Under Assumption 1, grades in an interval of width Δ preserve Proposition 1, and bounded gain gives $g(z) \in [-\Delta\delta(z), \Delta\delta(z)]$. This is an outer bound for a general partial-credit grader. For scaled exact-match grading, either endpoint gain is attainable on every disagreement, giving sharpness under arbitrary gold-answer kernels $P(dy \mid Z, A_s, A_\ell)$ on the common canonical answer space. Theorem 3 holds with $\hat{m}$ clipped to $[-\Delta, \Delta]$ and the support term scaled by Δ, by the same proof.

A.8 More than one downstream option: a sketch

With several downstream models one may factor each pairwise gain in the same way. Joint routing uses the pairwise conditional gains and per-call costs to allocate traffic under the shared budget. We record the pairwise factorization as a sketch.

B Experimental Implementation

Models and data. Smaller model Qwen2.5-1.5B-Instruct, larger model Qwen2.5-7B-Instruct (Qwen Team, 2024). The primary collection is 15,361 queries: GSM8K test (Cobbe et al., 2021) and MMLU (Hendrycks et al., 2021). The wider sweep of Appendix E adds eight further families; they are named in Table 11. Resolved revisions, as used for generation:

artifact	revision
`Qwen/Qwen2.5-1.5B-Instruct`	`989aa7980e4cf806f80c7fef2b1adb7bc71aa306`
`Qwen/Qwen2.5-7B-Instruct`	`a09a35458c702b33eeacc393d103063234e8bc28`
`openai/gsm8k`	`740312add88f781978c0658806c59bc2815b9866`
`cais/mmlu`	`c30699e8356da336a370243923dbaf21066bb9fe`

Generation ran in `bfloat16` with the `sdpa` attention implementation on one NVIDIA GB10, torch 2.13.0+cu130, Python 3.12.13.

Generation and grading. Primary paired generations use greedy decoding (`num_beams`$= 1$, temperature and `top_p` unused), with maximum new tokens 512 for GSM8K and 8 for MMLU. MMLU is prompted as a lettered multiple-choice question whose answer is a single option letter; GSM8K uses a fixed chain-of-thought template terminated by a final answer line. Answers are extracted by a per-family rule — the option letter for MMLU, the final numeric answer for GSM8K — and compared under a normalizer shared with grading, so that answer equality and grade equality are decided by the same function. Thus Assumption 1 holds by construction. The outcome partition in Table 4 checks implementation consistency and records the positive, negative, and zero gains on disagreements.

Reliability-baseline generations. Self-consistency draws five additional smaller-model completions per query with sampling ($T = 0.7$, top-$p = 0.95$, one beam), batch size 12, and the same task-specific token caps as the primary generations (512 and 8 for GSM8K and MMLU). Source shards contain 512 queries; the sampling seed is zero plus the shard index, with queries sorted by family, prompt length and sample identifier before batching. The score is minus the modal frequency of normalized sampled answers. Modal ties use the lexicographically first normalized key for recording; the frequency is unchanged. The five samples provide the routing score, while the original greedy answer remains the smaller model's served answer. The complete protocol therefore uses one greedy and five sampled completions.

Self-verification adds a greedy smaller-model completion (one beam, at most four new tokens, batch size 64) using the model's chat template. The user message contains the dataset prompt and full smaller-model response. Its instruction is "Determine whether the proposed answer correctly answers the question. Respond with exactly Yes or No." It then contains the following blocks, separated by blank lines:

```
Question:
What is 2 + 2?
Proposed answer:
4
Is the proposed answer correct?
Answer: Yes

Question:
What is the capital of France?
Proposed answer:
London
Is the proposed answer correct?
Answer: No

Question:
{dataset prompt}
Proposed answer:
{small-model response}
Is the proposed answer correct?
Answer:
```

The confidence is first-verdict-token mass $P(\text{yes})/[P(\text{yes}) + P(\text{no})]$, and routing orders queries by its negative. Affirmative strings are `Yes`, `␣yes`, `␣Yes`, `YES`; negative strings are `No`, `␣no`, `␣No`, `NO`. Only strings encoding to one token are retained, with duplicate token identifiers removed. The score comes from those token masses rather than the parsed verdict. Both baselines use the same paired answers; their additional generations affect the routing score only.

Table 3: Full-pool support-representation selections for the two-benchmark experiment. Each row is one of the seven pool/test splits. The representation is selected for disagreement prediction and then shared by the observation-matched arms.

Seed	SVD dimension	Multiplier	Regularization C
0	64	0.3	1
1	64	1	1
2	128	0.03	10
3	128	0.1	1
4	64	0.3	1
5	128	0.1	1
6	64	1	1

Splits. seven fixed partitions into pool 9,216 and test 6,145; the same partitions are reused by every method and every label budget. The primary evaluation uses MMLU and GSM8K, selected after task-mixture and token-cost analyses of an original three-benchmark collection. The per-task analysis in Table 11 covers ten of the eleven originally evaluated families. Queries keep their original pool/test assignments; preprocessing, representation selection and heads are fitted on the selected pools, and labeled subsets are drawn deterministically at each budget. Evaluation therefore reuses the test queries from the earlier analysis.

Router-visible features. Z contains surface features of the query text, smaller-model confidence summaries (mean token entropy, mean top-1 probability, sequence log-probability, generated length), and reduced hidden states from the smaller model. Hidden states are taken from the causal-LM backbone's `last_hidden_state` after the final layer normalization, recorded at three positions — the last prompt token, the last generated token, and the mean over generated tokens.

Representation and heads. Hidden states are reduced by truncated SVD; the dimension, a shrinkage multiplier and the regularization strength are selected by nested cross-validation *within the pool* with D as the target, so neither gold outcomes nor test rows enter the choice. The grids are dimension $\in \{64, 128, 256, 512, 1024\}$, multiplier $\in \{0.03, 0.1, 0.3, 1, 3\}$, and ten log-spaced regularization values, $\alpha \in [10^{-3}, 10^{6}]$ for ridge and the reciprocal grid $C = 1/\alpha$ for logistic heads. The support head is a cross-fitted regularized logistic regression; the margin head is a ridge regression on the labeled disagreements, clipped to $[-1, 1]$. The two-head comparator adds an equality penalty $\lambda \in \{0.01, 0.1, 1, 10, 100\}$.

Selection and out-of-fold predictions. The support stage uses five outer stratified folds on D and three inner stratified folds for selection by held-out D log loss. Within each inner-training fold the hidden block is standardized, reduced by randomized SVD with $1{,}024$ components and five iterations, then standardized again. The dimension grid uses prefixes of these components, followed by the chosen multiplier. The selected hidden transformations and classifier are refitted on the outer-training rows to predict the excluded rows. Base scalar features are standardized once on the unlabeled pool. A separate threefold selection and refit on the full pool supplies the test predictor and the fixed representation shared by downstream heads; their training support feature uses the outer out-of-fold probabilities. The shared representation is selected using D once per split and reused across label budgets.

Selected configuration. Representation selection is performed separately within each training pool using the grids above. Table 3 gives the full-pool selections.

Numerical environment. The BLAS thread count is set to 8. Model generations are fixed across the seven splits, so the reported split dispersion does not include decoding variability.

Evaluation. Routing AUC integrates accuracy against nominal deferral rate on the 51-point grid $0.00, 0.02, \ldots, 1.00$ by the trapezoid rule. Under the original mixture, a nominal rate B selects the first $\lfloor Bn \rfloor$ of n queries; the realized rate is $\lfloor Bn \rfloor / n$. This same convention is used for within-task comparisons and the nominal 10% and 20% accuracies. Queries are ordered by descending score

and, within an exact tie, by ascending row index. The population theory permits randomized policies at exact rates. The equal-task weighted analysis and token-cost matching implement adjacent-prefix randomization to reach their target weighted rates or costs exactly. Supervised gain estimators are evaluated at positive label budgets; the zero-gold policy uses the support score.

Cost proxy. Completion cost is the prompt-plus-completion token count multiplied by a model-specific rate: 1.5×10^{-6} for the smaller model and 7×10^{-6} for the larger one, proportional to nominal parameter counts. Costs and savings are expressed in these normalized compute units using realized completion lengths rather than the generation cap. Hardware latency and dollar billing require separate measurement. The ratio of unconditional mean model costs is $c_\ell / c_s = 4.557343$. To account for query selection, serving-cost comparisons compute $\mathbb{E}[c_s(X) + \pi(X)c_\ell(X)]$ over the queries each policy actually routes. This preserves the dependence between query cost and a router's selection.

Statistical conventions. A displayed $\pm$ in a primary table is the sample standard deviation across the seven splits with $\mathrm{ddof} = 1$; "$7/7$" counts splits on which a paired difference has the stated sign. All splits partition the same collection and share queries. For the primary budget analysis, a *resampling stability interval* is the central 95% of paired means obtained by sampling seven split indices with replacement, then five label fits within each sampled split, over 10,000 resamples. It summarizes stability across these partitions and label draws. The query analysis instead resamples seed-0 test queries within task while holding heads and generations fixed (Appendix H); its percentile intervals describe conditional query-sampling uncertainty. The separate task sweep reports descriptive split dispersion for independently fitted per-task routers.

C Additional Primary Results

Outcome partition. The shared answer normalizer makes gain identically zero on agreement. Table 4 records that consistency check and the distribution of gains on disagreements.

C.1 Fixed-rate accuracies and reference gaps

Table 5 gives the nominal-rate accuracies behind Figure 4.

Fitted and clairvoyant references. Accuracy gaps depend on the reference used. The *conditional-gain reference* is a finite-sample ridge estimator trained with outer-fold cross-fitting on pool gold outcomes and applied to test queries. Against it, at a 20% deferral rate, the support router's accuracy gap is 0.01139 ± 0.00210 and token entropy's is 0.02288 ± 0.00128. Their difference equals the accuracy difference between the two routers at that rate (Table 5).

The *realized-gain reference* in Table 1 ranks test queries by their realized outcome differences. It is a clairvoyant benchmark using test outcomes, whereas the conditional-gain reference predicts gains from router-visible features.

C.2 The signed stage

At $2{,}000$ labels, the factorized score reaches 0.67049 routing AUC, compared with 0.66771 for support alone and 0.64859 for the signed margin alone. Tables 7–8 report the complete budget grid under the observation-matched protocol, which averages five label replicates within each split. The operational increment over support-only is reported with paired intervals in Table 9.

Table 4: Realized-gain decomposition over the full primary collection. Agreement has zero realized gain under the shared grading rule. Shares use all queries as denominator.

Region	Outcome	Queries	Share
agreement	both correct	7,513	48.9%
agreement	both incorrect	2,013	13.1%
agreement	different correctness	0	0.0%
disagreement	larger model wins	3,553	23.1%
disagreement	smaller model wins	986	6.4%
disagreement	both incorrect	1,296	8.4%
total		15,361	100.0%

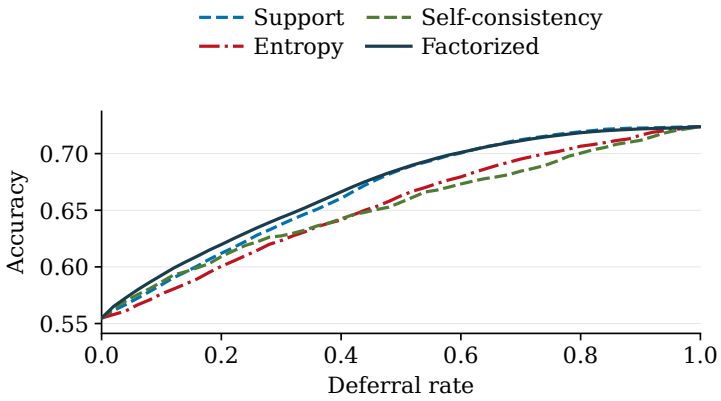

Figure 4: Accuracy against deferral rate, averaged over the seven fixed splits. The factorized router uses 2,000 labeled disagreements; support-only uses no gold outcome labels. Integrated and low-deferral comparisons are reported in Table 1 and Appendix G.

Table 5: Accuracy at fixed nominal deferral rates under original proportions. Each split defers $\lfloor Bn \rfloor$ queries. Values and paired differences are means $\pm$ sample SD across seven splits.

Deferral rate	Support	Token entropy	Paired difference
0.10	0.584168 ± 0.003989	0.576194 ± 0.002985	$+0.007974 \pm 0.002964$
0.20	0.612019 ± 0.003715	0.600535 ± 0.003347	$+0.011484 \pm 0.001335$
0.30	0.637545 ± 0.003324	0.623248 ± 0.004234	$+0.014297 \pm 0.002336$
0.50	0.685947 ± 0.005970	0.662769 ± 0.004940	$+0.023178 \pm 0.003730$

D Observation-Matched Label-Efficiency Details

D.1 Protocol

Shared subsets. For each split, each label budget at zero or on the grid 25, 50, 100, 250, 500, $1{,}000$, $2{,}000$, and each of five replicates, one labeled subset of disagreements is drawn before the supervised gain estimators are fitted, using a random seed determined by (split, budget, replicate), and is shared by those estimators. The support stage has already been fitted without gold labels; the correctness probe has a separate all-pool sampling protocol. The deterministic zero-label setting is evaluated once per split. Every gain estimator also receives the same unlabeled pool with its disagreement indicators, the same agreement rows carrying $H = 0$, and the same representation. Split/label-draw intervals are the descriptive resampling stability intervals defined in Appendix B, using 10,000 resamples.

Competitors. The competitors are a direct regression of H on Z with inverse-propensity weights 1 on agreement rows and $1/e$ on labeled disagreements, as Proposition 2 prescribes; the same regression additionally given the cross-fitted out-of-fold support estimate as an input feature; an equality-constrained two-head estimator in which agreement rows contribute only a penalty on the difference of the two heads, encoding $Y_s = Y_\ell$ without asserting a value; an unweighted direct regression, retained as a diagnostic; and margin-only and random ablations. The second competitor is the linear D-aware direct baseline: it receives the support estimate as an input feature. The product-capable direct baselines below additionally test the role of the estimator's function class.

Inverse-propensity weights. The labeling propensity is $e = |\text{labeled}|/|\text{pool disagreements}|$, computed separately for each split, budget and replicate. The pool disagreement counts for seeds 0–6 are respectively $3{,}543$, $3{,}539$, $3{,}467$, $3{,}521$, $3{,}537$, $3{,}530$, and $3{,}480$. Each budget b is sampled uniformly without replacement, so $e_s = b/N_{D,s}$ is identical across the five label draws within a split. Table 6 reports its range across splits at every budget. Training rows are all agreement rows with weight 1 and the labeled disagreements with weight $1/e$; mean-normalizing the weights preserves their ratio.

Out-of-fold support feature. The linear D-aware direct baseline receives $\hat{\delta}$ as an input feature. For pool rows this is the out-of-fold value, so no training row sees a support estimate fitted using itself. Test-query support estimates come from a head refitted on the full pool.

Two-head estimator. Heads are parameterized by a common and a difference component. Agreement rows contribute only a penalty λd^2 on the difference, encoding $Y_s = Y_\ell$ without asserting a value; there is no symmetric correctness target. The penalty λ is chosen from the fixed grid $\{0.01, 0.1, 1, 10, 100\}$ on validation loss, with no access to test rows.

Capacity and search effort. All learned arms use the same reduced representation. The factorized arm selects the margin ridge regularization on its labeled disagreements; direct ridge arms also select their regularization from the fixed grid. The two-head arm additionally searches a five-point λ grid, and the network searches the nine configurations specified in Appendix G. These controls share observations and a representation, but their architectures and search spaces differ.

D.2 Full results

Table 7 gives the levels and Table 8 the paired differences. Support-only scores are constant across label budgets; random-routing scores vary with sampling noise.

Increment over the zero-gold stage. Table 9 reports factorized minus support-only on all five metrics. At 250 labels, the full-AUC stability interval spans zero while all four operational intervals are positive. At the evaluated budgets of 500, $1{,}000$ and $2{,}000$ labels, every reported interval is positive. Crossings refer to the evaluated budget grid.

Zero-label cells. At a zero label budget the labeling propensity is zero, so Proposition 2 does not identify the signed margin. A constant supervised score would give an AUC determined by the tie rule. Such zero-label rankings are reported as undefined and omitted from the positive-budget tables. The support-only policy remains available without outcome labels.

Table 6: Labeling fractions within the pool disagreement stratum. All nominal budgets are filled. Entries are minimum and maximum propensities across seven splits, in percent; at zero labels the propensity is zero.

Label budget	25	50	100	250	500	1,000	2,000
Minimum	0.71	1.41	2.82	7.06	14.11	28.22	56.45
Maximum	0.72	1.44	2.88	7.21	14.42	28.84	57.69

Table 7: Observation-matched routing AUC under original proportions. Each cell first averages label-fit metrics within split, then the seven splits. Supervised zero-label scores have no defined ranking and are omitted; the support-only router uses no gold labels at every displayed budget.

Method	25	50	100	250	500	1,000	2,000
support only	0.66771	0.66771	0.66771	0.66771	0.66771	0.66771	0.66771
factorized	0.66554	0.66697	0.66677	0.66826	0.66907	0.66985	0.67049
linear D-aware	0.65188	0.65281	0.65557	0.66093	0.66487	0.66750	0.66904
interaction direct	0.64947	0.65181	0.65397	0.65929	0.66281	0.66641	0.66908
direct MLP	0.64298	0.64493	0.64631	0.64912	0.65070	0.65356	0.65407
direct, IPW	0.65147	0.65200	0.65445	0.65916	0.66338	0.66647	0.66833
two-head equality	0.65107	0.65269	0.65439	0.66015	0.66140	0.66281	0.66839
direct, unweighted	0.65350	0.65457	0.65773	0.66255	0.66531	0.66794	0.66896
margin only	0.64364	0.64142	0.64057	0.64299	0.64378	0.64691	0.64859
random	0.63961	0.63992	0.63929	0.63896	0.63901	0.63880	0.63925

Operating points. Table 10 reports the same paired comparison at the two nominal rates the paper otherwise uses. At both 10% and 20%, the intervals favor the factorized estimator through 250 labels and span zero from 500 onward. Full routing AUC summarizes the entire deferral-rate grid, while these comparisons isolate the two operating points.

Direct-regression diagnostics. Unweighted direct regression has higher mean full AUC than its IPW version throughout the evaluated grid. It first exceeds support-only at $1{,}000$ labels, whereas IPW direct does so at $2{,}000$. Adding the support feature raises the mean AUC of IPW direct at each budget. These comparisons quantify the contribution of weighting and the support feature to integrated performance.

Table 8: Factorized minus comparator full routing AUC, paired on the same split and label subset. Cells are paired means over seven splits and five label fits; Figure 2 displays split/label-draw resampling stability.

Comparator	25	50	100	250	500	1,000	2,000
linear D-aware	+0.01366	+0.01416	+0.01120	+0.00733	+0.00419	+0.00234	+0.00144
interaction direct	+0.01607	+0.01515	+0.01281	+0.00897	+0.00626	+0.00344	+0.00140
direct MLP	+0.02256	+0.02203	+0.02046	+0.01913	+0.01837	+0.01629	+0.01642
direct, IPW	+0.01407	+0.01497	+0.01232	+0.00910	+0.00569	+0.00338	+0.00216
two-head equality	+0.01447	+0.01428	+0.01238	+0.00811	+0.00766	+0.00703	+0.00210
direct, unweighted	+0.01204	+0.01239	+0.00904	+0.00571	+0.00375	+0.00190	+0.00153
margin only	+0.02190	+0.02554	+0.02620	+0.02527	+0.02528	+0.02294	+0.02190
support only	−0.00217	−0.00075	−0.00094	+0.00055	+0.00135	+0.00213	+0.00278

Table 9: Factorized minus support-only performance in percentage points. Cells show paired means and 95% resampling stability intervals across seven splits and five label fits. The secondary operational metric is $\mathrm{AUC}_{0.5}$.

Metric	25	50	100	250	500	1,000	2,000
Full AUC	−0.217 [−0.469, −0.041]	−0.075 [−0.186, 0.021]	−0.094 [−0.219, 0.000]	+0.055 [−0.009, 0.124]	+0.135 [0.088, 0.184]	+0.213 [0.164, 0.274]	+0.278 [0.238, 0.324]
$\mathrm{AUC}_{0.2}$	+0.028 [−0.021, 0.090]	+0.091 [0.002, 0.175]	+0.076 [−0.012, 0.149]	+0.235 [0.132, 0.348]	+0.327 [0.228, 0.431]	+0.513 [0.366, 0.658]	+0.715 [0.591, 0.840]
$\mathrm{AUC}_{0.5}$	−0.058 [−0.178, 0.010]	+0.034 [−0.036, 0.103]	+0.023 [−0.068, 0.104]	+0.180 [0.097, 0.267]	+0.294 [0.216, 0.372]	+0.444 [0.361, 0.533]	+0.597 [0.533, 0.655]
Accuracy at 10%	+0.051 [−0.022, 0.150]	+0.133 [0.006, 0.264]	+0.104 [0.003, 0.194]	+0.293 [0.151, 0.447]	+0.413 [0.270, 0.567]	+0.634 [0.426, 0.835]	+0.842 [0.665, 1.015]
Accuracy at 20%	−0.038 [−0.147, 0.047]	+0.011 [−0.133, 0.124]	+0.065 [−0.080, 0.204]	+0.230 [0.136, 0.339]	+0.341 [0.228, 0.466]	+0.558 [0.423, 0.701]	+0.771 [0.613, 0.931]

Table 10: Factorized minus linear D-aware accuracy at the two nominal deferral rates under original proportions. Each cell shows the paired mean and 95% resampling stability interval.

Labels	Accuracy at 10%	Accuracy at 20%
25	+0.00550 [0.00425, 0.00682]	+0.01028 [0.00828, 0.01217]
50	+0.00647 [0.00412, 0.00884]	+0.01108 [0.00830, 0.01397]
100	+0.00441 [0.00228, 0.00645]	+0.00880 [0.00642, 0.01169]
250	+0.00308 [0.00133, 0.00504]	+0.00483 [0.00305, 0.00661]
500	+0.00038 [−0.00138, 0.00204]	+0.00104 [−0.00062, 0.00270]
1,000	−0.00024 [−0.00178, 0.00123]	−0.00109 [−0.00272, 0.00042]
2,000	−0.00028 [−0.00121, 0.00060]	−0.00108 [−0.00233, 0.00010]

E TASK-LEVEL ROBUSTNESS

Independent per-task fits. Table 11 reports a separate fixed-dimension sweep with independently fitted per-task representations on ten task families: ARC-Challenge (Clark et al., 2018), HellaSwag (Zellers et al., 2019), WinoGrande (Sakaguchi et al., 2020), OpenBookQA (Mihaylov et al., 2018), CommonsenseQA (Talmor et al., 2019), MMLU-Pro (Wang et al., 2024), TruthfulQA (Lin et al., 2022b) and BoolQ (Clark et al., 2019), alongside the two primary families. Each benchmark has its own pool/test partition, fitted reduction and support head, with representation dimension fixed at 128. HellaSwag, MMLU-Pro and BoolQ each contain $3{,}000$ queries, of which $1{,}200$ are held out; Table 11 gives test sizes for all families. This per-task fitting protocol differs from the joint training used in the primary experiment.

Support improves mean integrated routing AUC on eight of the ten task families; BoolQ and WinoGrande have negative mean differences.

Table 11: Task-level robustness on the ten benchmark families, sorted by the support-minus-entropy routing AUC. Each benchmark has its own training split, fitted reducer and support head, with representation dimension fixed at 128. Each benchmark is fitted separately from the primary joint-collection analysis. Test counts are per split; ± is the sample standard deviation over seven fixed splits. Positive-split counts summarize consistency across the overlapping splits. The last two rows average task differences within each split before summarizing across splits.

benchmark	test n	support − entropy	positive splits
TruthfulQA-MC	327	$+0.06393 \pm 0.00680$	7/7
CommonsenseQA	489	$+0.03987 \pm 0.00698$	7/7
OpenBookQA	200	$+0.02273 \pm 0.00691$	7/7
HellaSwag	1,200	$+0.01740 \pm 0.00369$	7/7
ARC-Challenge	469	$+0.01475 \pm 0.00227$	7/7
MMLU	5,617	$+0.01425 \pm 0.00114$	7/7
GSM8K	528	$+0.00541 \pm 0.00464$	6/7
MMLU-Pro	1,200	$+0.00253 \pm 0.00265$	5/7
BoolQ	1,200	-0.00142 ± 0.00158	2/7
WinoGrande	507	-0.00683 ± 0.00593	2/7
Task macro	—	$+0.01726 \pm 0.00115$	7/7
Query weighted	—	$+0.01308 \pm 0.00055$	7/7

F Relationship to Reference-Agreement Routing

For the two-model reduction of reference-agreement routing (Valkanas et al., 2025), the zero-one loss of the served answer against the larger model is $\mathbf{1}\{A_s \neq A_\ell\} = D$ when the small answer is served, and zero after deferral. Consequently the expected reference loss of a policy is

$$L_{\mathrm{ref}}(\pi) = \mathbb{E}[(1 - \pi(Z))D] = \mathbb{E}[D] - \mathbb{E}[\pi(Z)\delta(Z)].$$

At a fixed deferral rate, minimizing this loss therefore ranks by disagreement support. Thresholding a confidence $1 - \hat{\delta}$ at t defers the same rows as thresholding $\hat{\delta}$ at $1 - t$, provided the boundary and tie conventions agree. The equivalence includes tied scores when both implementations use the same randomization.

The support stage thus shares the reference-agreement objective. The representation specifies the hypothesis class for this shared prediction target. Reference loss does not determine the sign of the outcome-conditioned margin: both endpoint laws of Theorem 1 have the same reference-agreement data. The signed stage supplies that missing outcome information. The equivalence is between the two-model objectives; C3PO additionally addresses multi-model optimization and calibration.

G Capacity Controls

We test whether the observation-matched advantage persists when direct estimators can represent interactions with the support estimate, and compare support with a fitted correctness head on the same fixed, D-selected representation. The controls reuse the seven splits, unlabeled pool, agreement zeros and test rows of Appendix D.1. Direct arms also share identical labeled subsets within each split, budget and replicate.

Product-capable direct estimators. The interaction arm regresses H on $[Z, \hat{\delta}, \hat{\delta} \odot Z]$ with training-row $\hat{\delta}$ out-of-fold, Z scaled on the current training rows only before the interactions are formed, and the same IPW weights and ridge grid as the linear support-aware arm. The network arm is `MLPRegressor` with one ReLU hidden layer of width 64, Adam, and predictions clipped to $[-1, 1]$ after fitting. Its nine configurations combine initial learning rates $\{10^{-3}, 3\times10^{-3}, 10^{-2}\}$ with L2 penalties $\alpha \in \{10^{-4}, 10^{-3}, 10^{-2}\}$. Fits allow at most 300 iterations with internal early stopping and a patience of 10. Configuration selection uses the first split of a shuffled fivefold partition of the current training rows: each candidate fits the same bootstrap of the selection-training rows, sampled with replacement in proportion to IPW weights and with the same number of rows. The selected configuration minimizes IPW-weighted MSE on the held-out selection rows and is refitted using a

Table 12: Factorized minus direct-estimator performance with observation-matched labels and representations. Cells are paired means over seven splits and five label fits. Positive differences favor factorized routing.

Comparator	25	50	100	250	500	1,000	2,000
Full AUC							
interaction direct	+0.01607	+0.01515	+0.01281	+0.00897	+0.00626	+0.00344	+0.00140
direct MLP	+0.02256	+0.02203	+0.02046	+0.01913	+0.01837	+0.01629	+0.01642
linear D-aware	+0.01366	+0.01416	+0.01120	+0.00733	+0.00419	+0.00234	+0.00144
$\mathrm{AUC}_{0.2}$							
interaction direct	+0.00441	+0.00506	+0.00298	+0.00182	+0.00019	−0.00088	−0.00129
direct MLP	+0.00827	+0.00734	+0.00604	+0.00698	+0.00707	+0.00776	+0.00943
linear D-aware	+0.00511	+0.00583	+0.00436	+0.00266	+0.00010	−0.00043	−0.00034
Accuracy at 20%							
interaction direct	+0.01001	+0.00992	+0.00799	+0.00443	+0.00166	−0.00117	−0.00236
direct MLP	+0.01683	+0.01519	+0.01393	+0.01338	+0.01240	+0.01277	+0.01458
linear D-aware	+0.01028	+0.01108	+0.00880	+0.00483	+0.00104	−0.00109	−0.00108

Table 13: Learned correctness probe minus support-only, within the same fixed representation and classifier family. Probe labels are drawn uniformly over the training pool. Cells are paired means over seven splits and five label fits. Negative differences favor support.

Metric	25	50	100	250	500	1,000	2,000
Full AUC	−0.01943	−0.01732	−0.01318	−0.00969	−0.00742	−0.00516	−0.00346
$\mathrm{AUC}_{0.2}$	−0.00766	−0.00678	−0.00570	−0.00476	−0.00443	−0.00403	−0.00380
$\mathrm{AUC}_{0.5}$	−0.01672	−0.01477	−0.01170	−0.00870	−0.00748	−0.00583	−0.00462
Accuracy at 10%	−0.00795	−0.00685	−0.00607	−0.00506	−0.00459	−0.00403	−0.00388
Accuracy at 20%	−0.01490	−0.01305	−0.01012	−0.00808	−0.00753	−0.00758	−0.00690

fresh weighted bootstrap of all current training rows. The current rows include known agreement zeros and labeled disagreements. Before this head selection, Z is scaled without mean-centering on those rows and the raw out-of-fold support feature is appended. Seeds depend only on split, label budget and replicate; no test rows enter fitting or selection. Remaining options use the defaults of scikit-learn 1.9.1 release. Table 12 reports the paired differences.

The factorized estimator has higher seven-split mean integrated routing AUC than all three direct arms at every evaluated positive budget. Against interaction, its $\mathrm{AUC}_{0.2}$ intervals are positive through 250 labels, span zero at 500, and are negative at 1,000 and 2,000. At 20% deferral they are positive through 500, span zero at 1,000, and are negative at 2,000. The conditional seed-0 query intervals at 2,000 likewise favor interaction on $\mathrm{AUC}_{0.2}$ and 20% accuracy, while both direct estimators' full-AUC intervals span zero (Table 16).

Learned reliability on the fixed representation. The probe predicts the smaller model's correctness from the same fixed representation with the same classifier family and a comparable regularization grid, and routes by $1 - \hat{c}(z)$. Its gold labels are drawn uniformly over all pool queries rather than over disagreements, because small-model correctness is defined everywhere; one gold answer counts as one label, and the arm is bootstrapped within its own sampling protocol. This probe is a supervised reliability control.

Table 13 shows lower probe performance at every evaluated budget on all five metrics, with each split/label-draw stability interval below zero. At 2,000 gold labels it remains 0.00346 behind in routing AUC. Within this fixed, D-selected representation and classifier family, support therefore retains its advantage over a fitted correctness head at the tested label budgets.

G.1 Call-rate label-budget comparisons

Figure 2 reports the split/label-draw analysis of the two-benchmark experiment. The query-level intervals of Appendix H and token-cost comparisons of Appendix I condition on the fitted heads and

Table 14: Distinct task-allocation analyses, with means ± sample split SDs; difference SDs are computed after within-seed pairing. E1 uses the original global call budget. E2 slices its 20% policy by task. E3 gives each task its own call-rate budget, then reports within-task AUC or macro accuracy. E4 recomputes the single global boundary after assigning each task weight $1/2$.

Analysis / quantity	Support	Entropy	Support − entropy
E1: global full AUC	0.66771 ± 0.00455	0.65408 ± 0.00456	$+0.01363 \pm 0.00144$
E3: within-task macro accuracy at 20%	0.69259 ± 0.00522	0.68625 ± 0.00435	$+0.00634 \pm 0.00289$
E4: equal-weight global accuracy at 20%	0.69063 ± 0.00455	0.68081 ± 0.00660	$+0.00982 \pm 0.00485$
E2: MMLU accuracy	0.59834 ± 0.00424	0.58880 ± 0.00421	$+0.00954 \pm 0.00101$
E2: MMLU deferral (%)	20.88 ± 0.11	21.41 ± 0.07	-0.53 ± 0.08
E2: GSM8K accuracy	0.75758 ± 0.00638	0.72538 ± 0.01285	$+0.03220 \pm 0.00796$
E2: GSM8K deferral (%)	10.63 ± 1.16	4.95 ± 0.79	$+5.68 \pm 0.85$
E3: MMLU within-task AUC	0.65168 ± 0.00489	0.63724 ± 0.00522	$+0.01445 \pm 0.00148$
E3: GSM8K within-task AUC	0.84347 ± 0.00924	0.84203 ± 0.00823	$+0.00144 \pm 0.00357$

address different sources of variation. These crossings summarize the evaluated budget grid on this collection.

H Task Mixture and Query-Level Resampling

Task-mixture evaluation and query resampling use the same fitted preprocessing, representations, heads and model generations.

Collection composition. The primary collection contains 14,042 MMLU and 1,319 GSM8K queries, total 15,361, so MMLU is 91.41%. Each test set has 5,617 MMLU and 528 GSM8K queries.

Four routing analyses. Table 14 distinguishes four quantities. E1 evaluates the global router on the original mixture. E2 keeps this global policy and evaluates the selected rows within each task; its task-specific call rates can differ. E3 sets the same call-rate budget separately inside each task and averages the resulting accuracies with equal task weight. E4 gives each task half the evaluation weight and recomputes one global boundary. Adjacent-prefix randomization attains the weighted budget exactly in E4. The latter two analyses change the allocation policy and describe sensitivity to task weighting.

Signed supervision under equal weights. Table 15 reports the same call-rate metrics after giving each task weight $1/2$ and recomputing the global routing boundary. At 2,000 labels, the factorized-minus-support differences are positive across all three displayed metrics and their split/label-draw stability intervals. This global allocation differs from the separate within-task budgets in Table 2. It also differs from matching token cost, where queries carry unequal costs (Appendix I).

Query-level resampling. The split-level stability intervals elsewhere in the paper resample partitions and label draws on one fixed collection with fixed generations. To resample queries instead we fix the seed-0 held-out test set, chosen by index before any interval was computed, and draw 10,000 stratified paired bootstrap samples within task. This analysis compares entropy, support-only, and the factorized, linear D-aware and interaction heads at 2,000 labels. Inside a resample these methods and their label replicates share the sampling weights; a query drawn twice keeps its generation, label and score; budget boundaries and routing are recomputed rather than reused; and metrics are computed per replicate and then averaged, never by averaging scores. Because the task sample sizes are held at their seed-0 values, these intervals do not include uncertainty in the task mixture.

Seven-split point estimates summarize partition and label-draw variation separately. Repeated queries, the thirty-five fits and the seven overlapping test sets are not pooled as independent observations.

Table 15: Factorized minus comparator performance under equal task weights (mixture B), at 2,000 labels, in percentage points. Each task has weight $1/2$ and the global call-budget boundary is recomputed. Cells show the paired point estimate and 95% resampling stability interval from 10,000 hierarchical resamples of seven overlapping splits and then their five label fits. Positive differences favor factorized routing. Support-only uses no gold labels.

Comparator	Full AUC	$\mathrm{AUC}_{0.2}$	Accuracy at 20%
Support-only	$+0.310$ [0.245, 0.378]	$+0.694$ [0.602, 0.793]	$+0.747$ [0.517, 0.967]
Linear D-aware	$+0.156$ [0.081, 0.233]	-0.001 [-0.074, 0.086]	-0.073 [-0.227, 0.073]
Interaction direct	$+0.208$ [0.142, 0.281]	$+0.037$ [-0.059, 0.134]	-0.107 [-0.349, 0.121]

Table 16: Conditional query-level comparisons on seed 0, original proportions, at 2,000 labels. Cells show points and pointwise paired 95% intervals from within-task query resampling, holding fitted heads and generations fixed and recomputing routing boundaries. Metrics are computed for each label fit and then averaged.

Comparison	Full AUC	$\mathrm{AUC}_{0.2}$	Accuracy at 20%
support only − token entropy	$+0.01385$ [0.01053, 0.01709]	$+0.01011$ [0.00661, 0.01389]	$+0.01090$ [0.00504, 0.01741]
factorized − support only	$+0.00258$ [0.00106, 0.00401]	$+0.00502$ [0.00239, 0.00762]	$+0.00609$ [0.00172, 0.01009]
factorized − interaction direct	$+0.00095$ [-0.00074, 0.00264]	-0.00282 [-0.00452, -0.00099]	-0.00482 [-0.00859, -0.00133]
factorized − linear D-aware	$+0.00121$ [-0.00018, 0.00260]	-0.00103 [-0.00249, 0.00062]	-0.00316 [-0.00641, 0.00023]

These query intervals quantify sampling variation conditional on the fitted heads, model pair and generations.

I SERVING TOKEN COST AND TASK-MIXTURE SENSITIVITY

We compare token entropy, support-only, factorized, linear D-aware direct, and interaction direct on MMLU and GSM8K using per-query token counts. Preprocessing and heads are fitted on the training pools and held fixed during cost evaluation and resampling. The analysis protocol was specified before computing these results, following the scope selection described in Appendix B. Model generations, representation and hyperparameter grids, and score directions are held constant. Supervised methods share label budgets and five label replicates. The tables report the 2,000-label setting; evaluations cover seven splits and budgets $25, 50, 100, 250, 500, 1{,}000, 2{,}000$.

Per-query accounting and matching. For evaluation weights w_i summing to one, a routing policy has

$$\mathrm{Acc}(\pi) = \sum_i w_i \big[Y_{s,i} + \pi_i (Y_{\ell,i} - Y_{s,i})\big],$$

$$\mathrm{Cost}(\pi) = \sum_i w_i \big[c_{s,i} + \pi_i c_{\ell,i}\big].$$

Here $c_{s,i} = 1.5 \times 10^{-6} T_{s,i}$ and $c_{\ell,i} = 7.0 \times 10^{-6} T_{\ell,i}$, where T includes prompt and realized completion tokens. Every query incurs the small-model cost, and a deferred query also incurs its own large-model cost. Realized large-model token counts enter cost accounting only. Queries are

Table 17: Factorized minus comparator accuracy at equal serving token proxy cost, in percentage points. Positive differences favor factorized routing. Rows use entropy's 20% or 10% anchors, under original proportions (A) or equal task weights (B). Cells show seed-0 points and pointwise paired query-bootstrap 95% intervals, at 2,000 labels, averaging metrics over five fixed fits.

Mixture / anchor	Support-only	Token entropy	Linear D-aware	Interaction
A / 20%	$+1.22$ $[0.74, 1.72]$	$+2.26$ $[1.53, 3.00]$	$+0.06$ $[-0.26, 0.44]$	-0.25 $[-0.65, 0.12]$
A / 10%	$+0.96$ $[0.50, 1.43]$	$+1.94$ $[1.35, 2.57]$	$+0.21$ $[-0.07, 0.48]$	-0.28 $[-0.61, 0.00]$
B / 20%	-0.59 $[-1.46, 0.29]$	$+0.26$ $[-0.80, 1.36]$	$+0.01$ $[-0.55, 0.64]$	-0.34 $[-0.86, 0.31]$
B / 10%	-0.24 $[-0.83, 0.27]$	$+0.64$ $[-0.01, 1.26]$	-0.14 $[-0.41, 0.19]$	-0.32 $[-0.63, -0.01]$

Table 18: Factorized minus comparator minimum serving token proxy cost at equal accuracy, in units of 10^{-4} per query. Negative differences favor factorized routing; cost differences are absolute. Rows use entropy's 20% or 10% anchors, under original proportions (A) or equal task weights (B). Cells show seed-0 points and pointwise paired query-bootstrap 95% intervals, at 2,000 labels, averaging metrics over five fixed fits.

Mixture / anchor	Support-only	Token entropy	Linear D-aware	Interaction
A / 20%	-0.469 $[-0.687, -0.215]$	-0.996 $[-1.267, -0.687]$	-0.093 $[-0.217, 0.045]$	$+0.117$ $[0.003, 0.270]$
A / 10%	-0.215 $[-0.390, -0.059]$	-0.649 $[-0.836, -0.442]$	$+0.041$ $[-0.020, 0.107]$	$+0.139$ $[0.070, 0.213]$
B / 20%	$+0.389$ $[-0.091, 0.786]$	-0.110 $[-0.659, 0.395]$	$+0.054$ $[-0.275, 0.313]$	$+0.282$ $[-0.056, 0.551]$
B / 10%	$+0.101$ $[-0.100, 0.339]$	-0.319 $[-0.562, 0.017]$	$+0.074$ $[-0.068, 0.195]$	$+0.185$ $[0.043, 0.333]$

ordered by each router's score: descending score, with ties broken by ascending row index. We sweep every sorted prefix and use one shared mixing weight to interpolate cost, accuracy, and deferral rate between adjacent prefixes. Each head is evaluated using its original routing scores.

Mixture A retains each split's test-set task proportions. Mixture B gives MMLU and GSM8K weight $1/2$ each, without changing any head or prediction. Within each split and mixture, entropy at 20% deferral defines the primary cost and accuracy anchors; 10% is secondary. Deferral rates under B are weighted rates. Table 17 matches each router to entropy's anchor cost. Table 18 instead finds the minimum cost reaching entropy's anchor accuracy over all prefixes and adjacent-prefix crossings, allowing for non-monotone accuracy. All target accuracies are attainable on the evaluated curves. Because an earlier prefix can already attain the target, entropy's minimum cost at its target accuracy need not equal its anchor cost. Matching thresholds are computed on the test-set frontiers for this comparison.

Uncertainty and aggregation. The two tables report the preselected seed-0 held-out test set at 2,000 labels. Each fitted head is evaluated separately and the five label-replicate metrics are averaged. The intervals are the 2.5th and 97.5th percentiles from 10,000 paired query-bootstrap resamples, stratified within task, with bootstrap seed 20260916. A resample applies identical query multiplicities to every method and label replicate, then recomputes the weighted curves, entropy anchors, and matched points. This conditions on the fitted heads, fixed generations, and task sample sizes; it does not cover retraining or uncertainty in the task mixture. Intervals are pointwise, with no simultaneous frontier or multiplicity guarantee. No query-bootstrap intervals were computed at the other label budgets.

Table 19: Support-only token savings and formal break-even counts. Savings are relative to entropy's actual 20% policy in mixture A at matched accuracy; the seven-split mean is 8.85%. Break-even counts cover all four anchors, in thousands of future queries (rounded to 0.1 thousand). The full paired-pool cost is divided by the per-query serving saving relative to entropy's actual anchor policy, matching its accuracy. A uses original task proportions; B gives each task weight $1/2$. An em dash denotes nonpositive serving saving and hence no finite positive break-even. The calculation covers paired-pool generation for support-only under the assumptions in Appendix I.

Split seed	Saving (A / 20%)	A / 20%	A / 10%	B / 20%	B / 10%
0	10.08%	262.9	325.6	276.9	312.3
1	10.14%	261.7	568.5	700.2	690.6
2	10.02%	264.8	442.2	304.7	1,048.4
3	7.24%	361.1	872.6	—	7,188.2
4	8.11%	324.2	568.9	421.9	932.6
5	9.97%	266.2	291.5	—	695.5
6	6.36%	422.0	1,413.8	492.8	2,395.8

Seven-split descriptive means are a separate summary, not the centers of these seed-0 intervals. At the 20% anchor the seven-split factorized-minus-support accuracy differences are $+1.24$ percentage points in A and -0.23 in B, positive in seven and two splits respectively. The corresponding mean relative cost changes at equal accuracy are -11.00% and $+2.20\%$, with the same direction in every split of each mixture. These average paired per-split changes and ratios; the overlapping splits are not independent samples.

What the comparisons establish. At 2,000 labels, both equal-cost and equal-accuracy intervals favor the signed stage over support-only at both A anchors. At both B anchors its equal-cost point estimates are negative and its matched-accuracy costs are higher, with all four intervals spanning zero. Comparisons with linear D-aware direct remain unresolved. Interaction direct is more accurate at equal cost at B's 10% anchor, and cheaper at matched accuracy at both A anchors and B's 10% anchor. Factorized routing improves on entropy at both A anchors; its comparisons with entropy at both B anchors span zero.

Equal-cost and equal-accuracy comparisons evaluate different points on each frontier. For example, against interaction at A's 20% anchor the accuracy difference is -0.00251 ($[-0.00654, 0.00121]$), while the relative cost increase at equal accuracy is 2.78% ($[0.08\%, 6.41\%]$). Accuracy can vary non-monotonically along these empirical frontiers. Task weighting and the operating point therefore matter even with identical predictions.

Offline amortization. For support-only, let C_{pool} be the cost of all 9,216 paired pool runs and s the per-query serving saving at entropy's anchor accuracy, relative to entropy's actual anchor policy. The formal break-even is C_{pool}/s when $s > 0$; otherwise there is no finite positive break-even. Table 19 reports every split and anchor. The original-mixture 20% anchor gives the smallest range, about 262,000 to 422,000 queries; other anchors require more traffic, and two equal-weight 20% cases have negative savings. These calculations amortize paired-pool generation for support-only. They assume the same future distribution and token proxy, and exclude fitting, features and router execution.