CERTIFIED STOPPING FOR ACTIVE MODEL SELECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Active model selection reduces labeling cost by querying items on which candidate models disagree. For large language models with cached predictions on a fixed pool, one gold label evaluates every candidate. A preset budget, however, does not determine whether enough labels have been collected to identify the winner. We characterize completion certificates: observed labels that guarantee the same unique winner for every assignment of the remaining labels. For categorical predictions and shared nonnegative item weights fixed before labeling, a candidate is certified exactly when its observed lead over each rival exceeds their remaining disagreement weight. This gives the earliest safe stop along any fixed query path and identifies which candidates can still win. Extending the bounds to model choices across deployment cost caps gives an integer covering formulation over shared labels. We use the surviving candidates to guide two acquisition rules: FRONTIERCUT selects items separating the most surviving pairs, and CERTSELECT restricts and renormalizes SELECT-style weights over survivors. Under the same exact stopping rule on 353 LLM organization panels, their respective panel-median paired label savings are 9.39% and 10.50% over adapted SELECT, and 26.66% and 30.19% over a deployable Model Selector. On 19 binary and multiclass panels, FRONTIERCUT saves a run-level median paired 18.18% over its static disagreement-ranking ablation. For JudgeTuning's full deployment map, hindsight certificate bounds span 57.29%–62.46% of the pool across one- and two-percentage-point accuracy tolerances. On the LLM organization panels, the panel-median ratios of labels used to the hindsight lower bound are 1.63 for CERTSELECT and 1.66 for FRONTIERCUT, compared with 1.85 for adapted SELECT. Thus, tracking unresolved disagreements reduces labeling cost while certifying the fixed-pool winner.

1 INTRODUCTION

Selecting a large language model (LLM) or judge configuration often requires comparing cached predictions on a fixed evaluation pool. A gold label for one item reveals which candidates are correct on that item. Efficient model selection can therefore reuse each annotation across all candidates. The central decision is when the acquired labels are sufficient to identify the full-pool winner.

Existing methods reduce labeling cost by choosing informative queries. Active testing and prediction-powered inference use labels to improve performance estimates (Kossen et al., 2021; Berrada et al., 2025; Angelopoulos et al., 2023). ModelPicker and Active Comparison focus on distinguishing pretrained models (Reza Karimi et al., 2021; Sawade et al., 2012), while DiffUse selects preference queries using semantic differences between generated outputs (Ashury Tahan et al., 2024). Model Selector targets information about the best classifier, and SELECT-LLM uses accumulated evidence to weight response similarities (Okanovic et al., 2025; Durmazkeser et al., 2026). For the latter two methods, a preset label budget determines when querying ends.

A common budget leaves different amounts of decision uncertainty across tasks. With 200 uniformly acquired labels, exact-winner recovery is 97.5% on IFEval and 40.0% on MMLU-Pro (Figure 1). A budget can therefore expire while unresolved disagreements still permit another winner, or continue after the observed labels have already fixed the choice. Related sequential evaluation methods provide statistical stopping and identification guarantees (Wang et al., 2025; Tolochinsky et al., 2026; Arviv et al., 2026). We study an exact finite-pool criterion: which observed labels force the same winner under every assignment of the remaining labels?

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

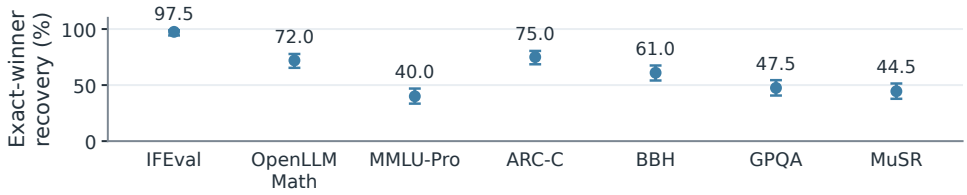

Figure 1: A fixed label budget leaves task-dependent decision uncertainty. Points show exact-winner recovery with 200 uniformly acquired labels over 200 trials per task. Error bars are 95% Wilson intervals.

We call such observed labels a *completion certificate*. For categorical predictions with shared non-negative item weights fixed before labeling, agreement terms cancel from pairwise score differences. Each remaining disagreement can change a pairwise lead by at most its item weight. A candidate is therefore certified as the unique winner exactly when its observed lead over every rival exceeds their remaining disagreement weight. The same bounds identify the *possible winners*: candidates that can attain a best score under some completion of the unknown labels. For any one candidate, a favorable completion attains its pairwise maxima simultaneously, making this survivor set exact (Section 3).

Stopping when evidence settles a decision has precedents in decision-driven active learning and preference elicitation (Golovin et al., 2010; Javdani et al., 2014; Walsh, 2007). Concurrent work by Kuboyama (2026) derives the same exact stopping test for binary labels under candidate-specific rank weights. Our analysis shows how shared weights support exact candidate elimination for categorical predictions and connects that survivor set to query selection. FRONTIERCUT queries items separating the most surviving candidate pairs. CERTSELECT restricts SELECT-style weights to the survivors and renormalizes them before choosing the next item. Both rules use the same exact stopping condition (Section 5).

The certificate also extends to deployment decisions at different inference cost caps (Salinas et al., 2025). Each cap defines a feasible candidate set, and a *cost map* specifies a choice for each such set. One shared collection of labels can certify all these choices within an accuracy tolerance. Given the realized labels, finding the smallest certificate becomes an integer covering problem because one label can support several comparisons at once. We establish its computational hardness and use a linear relaxation and a greedy construction to bound the minimum certificate size (Section 4).

Our contributions are:

- We characterize exact completion certificates and possible-winner sets for categorical predictions with shared weights, giving the earliest safe stopping point along a fixed query path.
- We formulate shared-label certification across deployment cost caps as integer covering, prove minimum certification is NP-hard, and bound certificate requirements on deployment workloads (Section 6.5).
- We develop FRONTIERCUT and CERTSELECT to target unresolved candidates. Under a common exact stopping rule on 353 LLM organization panels, they save panel-median paired 9.39% and 10.50% of labels over adapted SELECT, respectively (Section 6.3).

2 RELATED WORK

Active testing and limited-label model selection. Active testing (Kossen et al., 2021; Berrada et al., 2025) and prediction-powered inference (Angelopoulos et al., 2023) reduce the labels needed to estimate performance, while Active Comparison targets differences between candidates (Sawade et al., 2012). Model Selector prioritizes information about the best classifier (Okanovic et al., 2025), while SELECT-LLM weights response similarities using accumulated evidence (Durmazkeser et al., 2026). Racing and successive elimination use statistical bounds to discard candidates (Maron & Moore, 1994; Even-Dar et al., 2006), alongside fixed-budget (Audibert et al., 2010), fixed-confidence (Garivier & Kaufmann, 2016) and multiple-answer identification (Degenne & Koolen, 2019). We compare acquisition rules under an exact completion certificate for a fixed-pool winner.

Efficient LLM evaluation. Efficient LLM evaluation predicts full-test performance from representative subsets, adaptive item allocation or reference instances (Vivek et al., 2023; Maia Polo et al., 2024; Perlitz et al., 2023; Pacchiardi et al., 2024). Item response theory models candidate ability and item difficulty, while SparseEval learns a sparse weighted reconstruction of benchmark scores (Maia Polo et al., 2024; Zhang et al., 2026). Their target is score or ranking recovery from partial evaluation. We instead certify the winner from shared gold labels after candidate responses have been cached.

Completion guarantees and covering. Partial identification bounds unresolved quantities (Manski, 1990). Decision-driven active learning seeks decisions valid across consistent hypotheses (Golovin et al., 2010; Javdani et al., 2014). Adaptive submodularity supplies greedy acquisition guarantees under diminishing expected marginal gains (Golovin & Krause, 2011). Concurrent selective-prediction certificates cover binary labels with candidate-specific rank weights and leave differing categorical predictions open (Kuboyama, 2026). Shared weights make that pairwise elimination exact for every candidate that can still win rather than merely necessary (Appendix B.17), and they carry it to items whose gold answer matches neither candidate. Our binary stopping test and item order coincide with theirs, while the multi-region integer cover and its hardness lie outside that single-decision setting. Classical set and submodular cover provide logarithmic greedy bounds for selecting sufficient evidence (Chvátal, 1979; Wolsey, 1982). Possible and necessary winners quantify over completions of partial preferences (Walsh, 2007; Xia & Conitzer, 2011; Shachnai et al., 2025), where deciding possible winners is NP-complete for many scoring rules. Shared-weight categorical scores reduce the same question to pairwise inequalities.

Cost-aware configuration selection. Cost-aware selection searches for model or judge choices that balance prediction quality against inference expense (Salinas et al., 2025; Monteiro et al., 2026). FrugalGPT allocates increasingly costly calls through cascades, RouteLLM learns routing from preference data and IRT-Router relates candidate ability to query attributes (Chen et al., 2023; Ong et al., 2024; Song et al., 2025). These objectives optimize the quality–cost tradeoff and Appendix A compares them in detail. We take candidate costs as given and reuse shared labels to certify deployment choices across eligible candidate sets.

3 Problem Setting and Exact Completion Certificate

We seek observed labels that force a unique model choice under every completion of a shared pool. This section defines the shared-label scores and derives attainable pairwise bounds that give safe stopping and exact candidate elimination. Section 4 applies the same bounds to the labels that certify choices across deployment costs, and Section 5 uses the remaining possible winners to select queries.

3.1 Cached Predictions and Shared Labels

One acquired label scores all candidates. Let $\mathcal{A} = \{1, \dots, M\}$ be the candidates and $\mathcal{Q} = \{1, \dots, N\}$ the items. Candidate a has cached prediction $p_{ia} \in \mathcal{Y}$ on item i, where $\mathcal{Y}$ is a finite alphabet and y_i is the unknown gold label. With shared nonnegative weights w_i fixed before labeling, we define

$$S_a(y) = \sum_{i \in \mathcal{Q}} w_i \mathbb{I}[p_{ia} = y_i]. \tag{1}$$

Here $\mathbb{I}[\cdot]$ is the indicator function. S_a is a weighted correct count; dividing all scores by the same positive total weight leaves the winner unchanged. Write $P = (p_{ia})$ and $y = (y_i)_{i \in \mathcal{Q}}$. The history contains the revealed labels of $\mathcal{O}$, while $\mathcal{U} = \mathcal{Q} \setminus \mathcal{O}$ remains unlabeled. A completion $z \in \mathcal{Y}^{\mathcal{U}}$ produces a full vector $y^{(z)}$ agreeing with the observed labels.

Definition 1 (Robust strict winner)**.** Candidate a is a robust strict winner at history $(\mathcal{O}, y_{\mathcal{O}})$ if $S_a(y^{(z)}) > S_b(y^{(z)})$ for every $b \neq a$ and every completion z.

The observed labels then form a *completion certificate*. A policy uses cached predictions and acquired labels to choose queries. It is *completion-safe* if every output is a robust strict winner. Its label cost counts distinct acquired items. The online evaluation assumes a unique full-pool winner, ensuring that a strict certificate is eventually attainable. If the full-pool scores tie, completion safety still defines valid outputs, but it does not ensure termination with a strict winner.

3.2 Attainable Bounds and Exact Decisions

Only unobserved disagreements can reverse an observed lead. For candidates a and b, define their observed margin d_{ab} and remaining disagreement weight r_{ab} by

$$d_{ab} = \sum_{i \in \mathcal{O}} w_i \big(\mathbb{I}[p_{ia} = y_i] - \mathbb{I}[p_{ib} = y_i] \big), \quad r_{ab} = \sum_{i \in \mathcal{U}} w_i \mathbb{I}[p_{ia} \neq p_{ib}]. \tag{2}$$

Agreement terms cancel under every label. The remaining disagreement weight bounds how far each score difference can move from its observed margin.

Lemma 2 (Pairwise extremes). *For every ordered pair (a, b), the maximum and minimum of $S_a(y^{(z)}) - S_b(y^{(z)})$ over completions are $d_{ab} + r_{ab}$ and $d_{ab} - r_{ab}$. Assigning $y_i = p_{ia}$ on $\mathcal{U}$ attains all maxima for a simultaneously. Assigning p_{ib} on its disagreements with a attains the pairwise minimum.*

For a fixed candidate, one favorable completion attains all pairwise maxima, deciding whether it can be best. Strict certification instead excludes every rival's own adverse completion by checking the corresponding pairwise minimum. The itemwise arguments and stopping proofs are in Appendix B.13.

Theorem 3 (Exact robust-winner certificate). *Candidate a is a robust strict winner if and only if $d_{ab} > r_{ab}$ for all $b \neq a$.*

Sufficiency follows from the pairwise minimum in Lemma 2, and necessity follows because the pair-specific adverse completion realizes that minimum. The test is determined exactly by the attainable score differences at the current history. For binary labels this is the test of Kuboyama (2026) with a strict tie rule. The same attainable bounds identify every *possible winner*, meaning a candidate attaining a best score under some completion including ties.

Corollary 4 (Exact elimination). *Candidate a is a possible winner if and only if $d_{ab} + r_{ab} \geq 0$ for every $b \neq a$. Thus $d_{ab} + r_{ab} < 0$ for some rival excludes a.*

Further observations only remove completions, so eliminated candidates cannot return. A singleton possible-winner set is equivalent to a robust strict winner. Sharing the weights across candidates is what makes these $M - 1$ inequalities decisive.

Proposition 5 (Candidate-specific weights can prevent exact elimination). *If each candidate carries its own item weights, the pairwise attainable tests can hold against every rival although no completion makes that candidate best.*

Appendix B.17 gives a three-candidate instance. We call possible winners *survivors*. Shared weights make this set exact, while analogous attainable pairwise tests give only an outer approximation under candidate-specific weights in general. Initializing all pairwise disagreements takes $O(M^2 N)$ time. Each revealed label updates the state and tests elimination in $O(M^2)$ time. Query ranking is a separate cost, given in Appendix C.

Corollary 6 (Earliest exact stopping). *Along any fixed acquisition path, the first history satisfying Theorem 3 is the earliest history at which any completion-safe policy can output a strict winner.*

This result fixes the stopping point for a given query order. Acquisition still determines that order and how many labels are needed to reach it. The next section measures hindsight certificate size, before Section 5 returns to online query selection.

4 Certificates for Deployment Cost Maps

A deployment cost cap can change the selected candidate even when the evaluation pool stays fixed. We define these linked choices as a cost map and express its supporting labels as an integer cover. A linear relaxation and a greedy construction then bound the minimum evidence required.

4.1 From a Winner to a Cost Map

A deployment cost cap determines which candidates can be used. For fixed inference costs c_a, cap B_r defines a nonempty feasible set $\mathcal{A}_r = \{a : c_a \leq B_r\}$, which we call region $r \in \mathcal{R}$. A map g

chooses $g(r) \in \mathcal{A}_r$. A *fixed-map certificate* validates a specified g under every completion, whereas an *any-output certificate* may choose its map from the evidence.

This section permits choices within a specified accuracy shortfall and uses unit item weights, so S_a counts correct items. An accuracy tolerance $\epsilon \geq 0$ in percentage points corresponds to $\epsilon_N = N\epsilon/100$ correct items. Lemma 2 certifies $a = g(r)$ within tolerance exactly when $r_{ab} - d_{ab} \leq \epsilon_N$ for every feasible rival b. Here $r_{ab} - d_{ab}$ is the largest correct-count advantage the rival can attain. This criterion permits ties, and ϵ is a deployment accuracy tolerance rather than a statistical error probability.

4.2 Integer Cover and Certificate-Size Bounds

We now assume that the complete realized label vector is available and seek its smallest sufficient subset for a fixed map. This hindsight optimization measures certificate requirements; online acquisition observes only queried labels (Section 5). Let $D_{ab} = |\{j : p_{ja} \neq p_{jb}\}|$. Revealing y_j reduces rival b's attainable advantage over a by

$$C_{ab,j} = \mathbb{I}[p_{ja} \neq p_{jb}]\big(1 + \mathbb{I}[p_{ja} = y_j] - \mathbb{I}[p_{jb} = y_j]\big) \in \{0, 1, 2\}. \tag{3}$$

On a disagreement, labels matching the target, neither prediction, or the rival supply 2, 1, or 0 units respectively. Agreement items supply zero. The evidence matrix has one row for each region–rival comparison and one column per item. Its row demand is $\eta_{ab} = \max\{0, \lceil D_{ab} - \epsilon_N \rceil\}$, rounding up because the accumulated evidence is integral.

Theorem 7 (Fixed-map integer cover)**.** *For a unit-weight finite pool with realized labels and fixed map g, the minimum fixed-map certificate size at tolerance ϵ_N equals*

$$\min_{x \in \{0,1\}^N} \sum_j x_j \quad \textit{subject to} \quad \sum_j C_{g(r)b,j} x_j \geq \eta_{g(r)b} \quad \forall r \in \mathcal{R},\ b \in \mathcal{A}_r \setminus \{g(r)\}. \tag{4}$$

Here x_j selects item j's label. If no feasible certificate exists, the optimum is $+\infty$.

Each label is counted once and can satisfy several rows, making joint certification an integer covering problem. For binary labels it reduces to multicover (Appendix B.14). A map admits a certificate exactly when it meets the tolerance under the full realized labels. Exact minimization remains computationally difficult.

Theorem 8 (Hardness)**.** *Minimum certification with one region and binary labels is NP-hard for a fixed target map and, on the same instances, in the any-output form.*

The reduction is from Set Cover with padding items that equalize the pairwise demands, and its unique admissible output makes the same instances hard for both targets. For a certifiable map, relaxing x_j to $[0, 1]$ enlarges the feasible set and gives a linear programming (LP) lower bound. A feasible greedy certificate supplies an upper bound. Greedy cover selects the item with the largest total reduction in unmet demand after capping each contribution at its comparison's residual demand (Chvátal, 1979; Wolsey, 1982). The covering and hardness proofs with the inherited logarithmic integer-demand bound appear in Appendix I. Appendix B.14 specifies residual-demand updates and termination. These bounds concern hindsight evidence for fixed deployment choices.

5 Certificate-Aware Acquisition

Online acquisition knows P and the labels already queried, so the complete evidence matrix of Section 4 is unavailable. We instead use the current possible-winner set to guide each query. FrontierCut ranks items by the survivor pairs they separate. CertSELECT restricts SELECT's correctness-based weights to those survivors. Both rules update the exact certificate after each label and stop when only one candidate survives. We evaluate pools with a unique full-pool winner. Let $\mathcal{I} = \{i : \exists a, b,\ p_{ia} \neq p_{ib}\}$ be the informative query pool. Agreement items remain in the evaluation pool and in N, but need no labels for pairwise decisions.

5.1 FrontierCut

Let $A_t \subseteq \mathcal{A}$ contain the possible winners after t labels. These *survivors* include candidates that can only tie. Static disagreement scores also count eliminated candidates, although subsequent queries

need only distinguish survivors. For each available item $i \in \mathcal{I} \cap \mathcal{U}$, let $n_{i\ell}^{(t)} = |\{a \in A_t : p_{ia} = \ell\}|$ count the current survivors predicting label ℓ. The number of separated survivor pairs is

$$q_t(i) = \frac{1}{2}\left(|A_t|^2 - \sum_{\ell \in \mathcal{Y}} (n_{i\ell}^{(t)})^2\right). \tag{5}$$

The factor $1/2$ counts each unordered disagreeing pair once. FRONTIERCUT maximizes q_t over $\mathcal{I} \cap \mathcal{U}$, using a seeded random stream for exact ties. It updates d_{ab} and r_{ab} after each label and applies Corollary 4. Holding $A_t = \mathcal{A}$ recovers static all-pair cut. Each elimination removes that candidate's pairs from subsequent item rankings. Recomputing the score therefore directs queries toward remaining possible winners. For binary predictions, this ranking agrees with the adaptive pair-sum rule of Kuboyama (2026). Appendix C.6 gives the complete procedure.

5.2 SELECT ACQUISITION AND CERTSELECT

Accumulated correctness can distinguish survivors with different observed support. We adapt SELECT-LLM's acquisition objective (Durmazkeser et al., 2026) to binary unit-weight predictions and call this baseline *SELECT*. Candidate a receives mass $\pi_a^{(t)} \propto \exp(E_a^{(t)}/\tau)$, where $E_a^{(0)} = 0$, each query adds $\mathbb{I}[p_{ia} = y_i]$ to E_a, and $\tau > 0$ is the temperature. These acquisition masses guide queries. For the mass $u_i = \sum_{a:p_{ia}=1} \pi_a^{(t)}$ predicting label 1, SELECT minimizes $h_t(i) = u_i^2 + (1-u_i)^2$. This favors balanced answer masses among unqueried informative items. The binary form is used in the organization-panel comparison. The categorical extension $h_t(i) = \sum_{\ell \in \mathcal{Y}} (\sum_{a:p_{ia}=\ell} \pi_a^{(t)})^2$ is evaluated separately on the 19-panel study (Table 28).

CERTSELECT removes the mass assigned to eliminated candidates by defining

$$\widetilde{\pi}_a^{(t)} = \frac{\exp(E_a^{(t)}/\tau)}{\sum_{b \in A_t} \exp(E_b^{(t)}/\tau)} \quad (a \in A_t), \qquad \widetilde{\pi}_a^{(t)} = 0 \quad (a \notin A_t). \tag{6}$$

We substitute these masses into u_i and minimize the same objective. Both SELECT variants use $\tau = 0.7$, chosen by replacing gold correctness with an unlabeled response-agreement proxy. Appendix C.5 specifies the calibration cohort, temperature grid and selection rule, and Algorithm 1 includes the per-query mass update. Using the same exact stopper makes their label counts a comparison of how acquisition responds to survivor information.

6 EXPERIMENTS

We first test whether certificate feedback reduces the labels acquired online to identify a fixed-pool winner. We then use complete labels to bound the smallest certificate for a deployment map, providing an offline reference for its evidence requirements. A separate uniform-sampling study measures recovery of joint deployment decisions.

6.1 SHARED EVALUATION SETUP

Workloads. Our primary comparison uses 353 organization panels from the processed Open LLM Leaderboard WinoGrande matrix released by SparseEval (Zhang et al., 2026; Sakaguchi et al., 2020). Within each panel we deduplicate response profiles and retain a unique full-pool winner. All methods receive the same predictions and access gold labels only through queries. Table 6 separates source candidates, retained profiles, panel sizes and repeated runs. For breadth, 19 public raw-prediction panels cover CIFAR-10, GLUE, ImageNet-family tasks and PACS (Krizhevsky, 2009; Wang et al., 2019; Deng et al., 2009; Li et al., 2017). For cost maps, JudgeTuning supplies measured configuration costs (Salinas et al., 2025), while 135 WinoGrande panels use declared parameter count to order deployment options. The latter ordering is a proxy for inference burden.

Controlled comparisons. Full and quotient uniform (Kossen et al., 2021) measure the effect of removing constant-prediction items. Static all-pair cut and response-entropy QBC (Seung et al., 1992) rank disagreements without updating survivors. Active Comparison (Sawade et al., 2012) and Model Selector (Okanovic et al., 2025) use observed evidence to rank queries. Deployable Model Selector chooses its setting using majority-vote pseudo-labels, whereas its oracle-grid variant reports

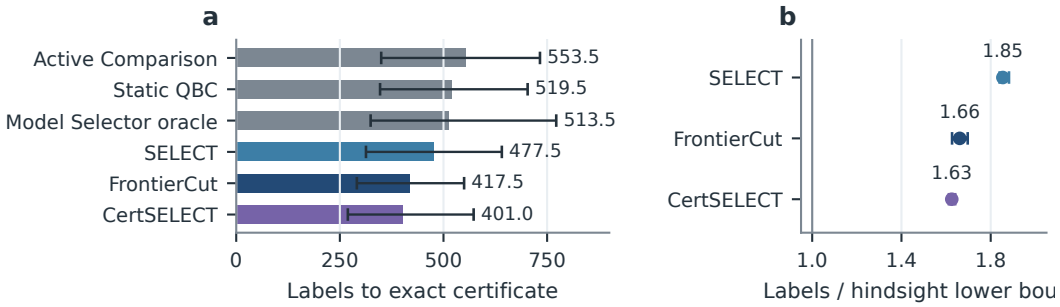

Figure 2: **Online label counts and hindsight lower bounds.** (a) Run-median labels and interquartile ranges over 706 matched runs on fixed-pool, unique-winner panels. (b) Panel-median labels divided by a full-label covering LP lower bound, with 95% panel-bootstrap intervals. The denominator lower-bounds the minimum certificate size.

the best completed run as a reference. SELECT (Durmazkeser et al., 2026) and CERTSELECT share a response-only temperature, calibrated on the full structural cohort to $\tau = 0.7$. Their comparison isolates survivor restriction. Table 7 lists each control's preprocessing, candidate updates and query objective. Every rule uses unit item weights and the same exact stopper, Theorem 3.

Aggregation. For paired counts L_m, L_b, saving is $\Delta = 1 - L_m/L_b$. We take its median within each panel and then across panels, with percentile intervals from 10,000 panel-bootstrap resamples (Davison & Hinkley, 1997). This statistic differs from a ratio of marginal median label counts. Primary SELECT and classical comparisons use $353 \times 2 = 706$ matched runs, and the uniform-control batch uses $353 \times 8 = 2{,}824$. Cross-task tables specify run or dataset aggregation separately.

6.2 FIXED BUDGETS LEAVE DECISIONS UNRESOLVED

To test the motivation for decision-specific stopping, we first sample uniformly from seven cached LLM task matrices, with 200 trials per task and budget. At 200 labels, exact-winner recovery ranges from 97.5% on IFEval to 40.0% on MMLU-Pro (Figure 1, Table 21). Low average score error can also coexist with a wrong winner: Table 22 stratifies the 50-label trials by estimation error. Winner identity depends on errors relative to the leading score gaps, which an average across candidates does not resolve.

We next examine retrospective budget truncations of the same 706 SELECT paths. Budgets come from the completed stopping-count distribution and are used only for this diagnostic. At 478 labels, 49.86% of runs remain uncertified while only 0.99% choose the wrong winner (Table 23). The certificate distinguishes a settled decision from one that remaining labels could overturn. Actual error and regret require the full labels for retrospective evaluation.

6.3 CERTIFICATE FEEDBACK REDUCES LABEL USE

Under the common exact stopper, FRONTIERCUT and CERTSELECT save a panel-median paired 9.39% and 10.50% over all-candidate SELECT (Table 1). All outputs are certified and correct on these unique-winner panels. Restricting SELECT's acquisition masses leaves its evidence update and stopping rule fixed, so the CERTSELECT comparison attributes the reduction to removing impossible winners from query selection. Organization-disjoint temperature calibration preserves this effect (Table 15).

Figure 2 separates absolute label counts from their ratio to a hindsight lower bound. That bound solves a relaxed covering problem using complete labels (Appendix E.8). Both survivor-based rules reduce the ratio relative to SELECT, while a gap to the lower bound remains. Table 1 also shows panel heterogeneity: FRONTIERCUT saves labels on 299 panels and uses more on 54. Its comparison changes both the candidate set and the query statistic.

The deployable Model Selector comparison likewise favors both rules in panel-median saving. Further controls establish the roles of preprocessing and query ranking: Table 8 reports the separate 2,824-run uniform batch, and Table 16 reports paired effects against classical selectors, with the oracle reference marked separately. These comparisons establish usefulness across query rules. CERTSELECT

Table 1: **Certificate feedback reduces label use.** Panel-median paired savings with 95% panel-bootstrap intervals on 353 fixed-pool, unique-winner panels (706 runs). All outputs are certified and correct. Signs count positive/zero/negative effects. Rows 1–3 use the full-cohort response-only $\tau = 0.7$. Rows 4–5 use deployable Model Selector, with organization-disjoint temperature selection for CERTSELECT. Dashes indicate run signs omitted from this summary.

Comparison	Saving ↑	95% interval	Panel signs	Run signs
FRONTIERCUT vs. SELECT	9.39%	[7.43, 10.61]	299/0/54	595/6/105
CERTSELECT vs. SELECT	10.50%	[9.61, 11.23]	342/5/6	680/12/14
CERTSELECT vs. FRONTIERCUT	2.10%	[0.88, 3.13]	208/1/144	414/15/277
FRONTIERCUT vs. Model Selector	26.66%	[24.52, 29.21]	342/0/11	—
CERTSELECT vs. Model Selector	30.19%	[28.15, 31.70]	346/0/7	—

Table 2: **Acquisition savings across tasks and candidate pools.** Each panel has eight matched replicates. The first block gives run-median paired savings over 152 runs and counts datasets with positive/zero/negative median effects. The second gives medians over 29 candidate panels from 11 source tasks, with 95% panel-bootstrap intervals.

Collection	Comparison	Saving	Dataset signs / interval
19 datasets	FRONTIERCUT vs. static pair cut	18.18%	19/0/0
	CERTSELECT vs. SELECT	14.00%	18/1/0
	FRONTIERCUT vs. SELECT	14.16%	19/0/0
29 candidate panels	FRONTIERCUT vs. SELECT	16.94%	[12.87, 19.23]
	CERTSELECT vs. SELECT	14.88%	[13.17, 17.17]

achieves additional median label savings over FRONTIERCUT and requires temperature selection. FRONTIERCUT avoids that calibration and retains positive median savings against SELECT at all 14 tested temperatures (Table 25).

6.4 GAINS ACROSS LABEL ALPHABETS AND PANEL SIZES

To test the acquisition benefit beyond binary responses, we retain the natural categorical predictions and compare against static all-pair cut and categorical SELECT. Table 2 shows positive median savings across the 19 datasets, with every returned winner certified and correct. The table also separates a candidate-pool sensitivity study: 29 disjoint candidate groups drawn from the same 11 multiclass source tasks. Its intervals resample candidate panels and describe variation within this task collection.

The multiclass setting also tests a distinct part of the certificate: a disagreement can reveal a gold label matching neither candidate, contributing zero to their observed score difference. Table 19 measures this event among winner–rival disagreements. Lemma 2 establishes validity for these answers, while Table 2 establishes the empirical acquisition savings.

Finally, Figure 7 shows positive median savings and lower bootstrap endpoints in all 12 organization strata defined by candidate count, winner gap and informative-item fraction. Savings increase across candidate-count quartiles, consistent with pruning more irrelevant comparisons in larger pools. These are associations across panels whose other properties also vary.

6.5 SHARED EVIDENCE CLOSELY BOUNDS FIXED-MAP CERTIFICATE SIZE

To quantify the covering problem in Theorem 7, we use full labels to bound certificate sizes offline. The canonical map selects the highest-accuracy feasible candidate at each cap. Its LP relaxation lower-bounds the minimum certificate size, and its greedy cover supplies a feasible upper bound. The any-output floor separately lower-bounds the size attainable when regional choices may change.

On JudgeTuning's full 331-region ordering, the greedy size is within 4% of the fixed-map LP bound at both tolerances (Table 3). Tightening the tolerance increases the evidence demands and the required label subset. The close bracket locates the canonical map's certificate cost despite the hardness of finding an exact minimum.

Table 3: **Greedy certificates approach fixed-map LP bounds.** Offline bounds from full labels. JudgeTuning reports counts and pool fractions. LP values are fractional relaxation objectives, and greedy counts are integer feasible sizes. WinoGrande reports medians over 135 panels, with Greedy/LP computed within each panel before aggregation. Floor allows any map. LP and Greedy fix the canonical map. These are deterministic optimization bounds.

Workload	Tol.	Floor ↑	Fixed LP ↑	Greedy ↓	Greedy/LP ↓
JudgeTuning	2 pp	1078 (30.38%)	2032.8 (57.29%)	2111 (59.50%)	1.039
	1 pp	1100 (31.00%)	2151.9 (60.65%)	2216 (62.46%)	1.030
WinoGrande	2 pp	15.00%	27.15%	27.70%	1.012
	1 pp	15.47%	29.76%	29.99%	1.008

Allowing local changes to regional choices finds smaller feasible certificates (Table 17). This measured improvement is under 4%, but the global any-output brackets remain wider. Thus the fixed-map certificate size is closely determined, while the best output choices remain less tightly resolved.

Organization cost orderings. The same comparison on 135 WinoGrande panels also gives close median fixed-map bounds, and all constructed subsets meet every regional constraint. Parameter count defines the deployment ordering, while the table's fractions measure gold labels as a share of the item pool. Table 18 shows that most panels have several canonical winners, so one shared label subset supports distinct choices at different caps.

6.6 Uniform Labels Often Miss the Joint Deployment Decision

We separately test fixed-budget recovery on 20 JudgeTuning caps with six distinct full-pool winners, using 5,000 uniform trials per budget. This cap subset measures joint selection reliability, whereas the full 331-region study above measures offline certificate size. Regret is the full-pool accuracy gap to the best feasible candidate at a cap. Joint recovery within two percentage points requires the maximum cap regret in a trial to meet that tolerance.

At 400 labels, 20.66% of trials meet the joint tolerance, and the 95th percentile of maximum cap regret is 5.21 points (Table 13, Figure 8). Individual caps can perform acceptably while the complete deployment map fails. Uniform sampling already shares each observed label across caps. What a fixed budget lacks is the joint certificate that bounds every regional comparison simultaneously.

7 Conclusion

In this paper we asked which observed labels settle a model choice under every completion of a shared pool and which candidates those labels leave in play. The certificate makes stopping a decision about the remaining evidence: an observed lead is sufficient exactly when no assignment of the unobserved labels can overturn it. Shared weights make the pairwise attainable bounds exact for both questions, and feeding the survivor set back into acquisition reduces labels under one exact stopping rule on organization panels and natural categorical tasks. For deployment across cost caps, the shared evidence matrix expresses joint certification as an integer cover. Feasible greedy certificates closely approach their fixed-map linear relaxations, locating the annotation burden near the minimum for those deployment choices.

Limitations. The certificate treats acquired labels as final observations on a finite pool, and the main single-region evaluation assumes a unique full-pool winner. The main SELECT comparison uses a binary unit-weight adaptation, with categorical comparisons reported separately. A direct implementation stores $O(M^2)$ pairwise state.

AI USE STATEMENT

In this work, we used generative AI tools for tasks with required disclosure. Generative AI tools helped develop the theoretical models and conceptual frameworks, helped formulate the mathematical claims, provided critical ingredients for proving those claims, and assisted in the writing of the proofs. The mathematical claims, the proofs, and every experimental result were checked and verified by the authors. We additionally used generative AI tools for tasks with recommended disclosure, namely to aid and polish the writing and to edit the paper for readability. We have reviewed all AI-assisted work, including the mathematical derivations and the text. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Anastasios N. Angelopoulos, Stephen Bates, Clara Fannjiang, Michael I. Jordan, and Tijana Zrnic. Prediction-powered inference. *Science*, 382(6671):669–674, 2023. doi: 10.1126/science.adi6000.

Ofir Arviv, Kristjan Greenewald, Yotam Perlitz, Hadar Mulian, Michal Shmueli-Scheuer, and Leshem Choshen. Stop guessing when to stop testing: Efficient model evaluation with just enough data. *arXiv preprint arXiv:2607.08522*, 2026. doi: 10.48550/arXiv.2607.08522. URL https://arxiv.org/abs/2607.08522.

Shir Ashury Tahan, Ariel Gera, Benjamin Sznajder, Leshem Choshen, Liat Ein-Dor, and Eyal Shnarch. Label-efficient model selection for text generation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics*, pp. 8384–8402. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.acl-long.456. URL https://aclanthology.org/2024.acl-long.456/.

Jean-Yves Audibert, Sébastien Bubeck, and Rémi Munos. Best arm identification in multi-armed bandits. In *Conference on Learning Theory*, 2010.

Gabrielle Berrada, Jannik Kossen, Freddie Bickford Smith, Muhammed Razzak, Yarin Gal, and Thomas Rainforth. Scaling up active testing to large language models. In *Advances in Neural Information Processing Systems*, volume 38, pp. 173670–173697, 2025. URL https://papers.nips.cc/paper_files/paper/2025/file/fdb11be1acf5e3724737dd585e590146-Paper-Conference.pdf.

Lingjiao Chen, Matei Zaharia, and James Zou. FrugalGPT: How to use large language models while reducing cost and improving performance. *arXiv preprint arXiv:2305.05176*, 2023.

Vašek Chvátal. A greedy heuristic for the set-covering problem. *Mathematics of Operations Research*, 4(3):233–235, 1979. doi: 10.1287/moor.4.3.233.

Ferdinando Cicalese, Sergio Filho, Eduardo Laber, and Marco Molinaro. Teaching with limited information on the learner's behaviour. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 2016–2026. PMLR, 2020. URL https://proceedings.mlr.press/v119/cicalese20a.html.

Anthony C. Davison and David V. Hinkley. *Bootstrap Methods and Their Application*. Cambridge University Press, 1997. doi: 10.1017/CBO9780511802843.

Rémy Degenne and Wouter M. Koolen. Pure exploration with multiple correct answers. In *Advances in Neural Information Processing Systems*, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009. doi: 10.1109/CVPR.2009.5206848.

Yavuz Durmazkeser, Patrik Okanovic, Andreas Kirsch, Torsten Hoefler, and Nezihe Merve Gürel. Large language model selection with limited annotations. *arXiv preprint arXiv:2605.24981*, 2026. doi: 10.48550/arXiv.2605.24981. URL https://arxiv.org/abs/2605.24981.

Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of Machine Learning Research*, 7(39):1079–1105, 2006. URL http://jmlr.org/papers/v7/evendar06a.html.

Uriel Feige. A threshold of $\ln n$ for approximating set cover. *Journal of the ACM*, 45(4):634–652, 1998. doi: 10.1145/285055.285059.

Aurélien Garivier and Emilie Kaufmann. Optimal best arm identification with fixed confidence. In *Conference on Learning Theory*, 2016.

Daniel Golovin and Andreas Krause. Adaptive submodularity: Theory and applications in active learning and stochastic optimization. *Journal of Artificial Intelligence Research*, 2011.

Daniel Golovin, Andreas Krause, and Debajyoti Ray. Near-optimal Bayesian active learning with noisy observations. In *Advances in Neural Information Processing Systems*, volume 23, 2010.

Shervin Javdani, Yuxin Chen, Amin Karbasi, Andreas Krause, Drew Bagnell, and Siddhartha Srinivasa. Near optimal Bayesian active learning for decision making. In *Proceedings of the Seventeenth International Conference on Artificial Intelligence and Statistics*, volume 33 of *Proceedings of Machine Learning Research*, pp. 430–438. PMLR, 2014.

Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Tom Rainforth. Active testing: Sample-efficient model evaluation. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 5753–5763. PMLR, 2021. URL https://proceedings.mlr.press/v139/kossen21a.html.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Tetsuji Kuboyama. How many labels does model choice need? Certificates and budgets for selective prediction. *arXiv preprint arXiv:2609.18622*, 2026. URL https://arxiv.org/abs/2609.18622.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M. Hospedales. Deeper, broader and artier domain generalization. In *IEEE International Conference on Computer Vision*, pp. 5542–5550, 2017. doi: 10.1109/ICCV.2017.591.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinybenchmarks: Evaluating LLMs with fewer examples. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 34303–34326. PMLR, 2024. URL https://proceedings.mlr.press/v235/maia-polo24a.html.

Charles F. Manski. Nonparametric bounds on treatment effects. *The American Economic Review*, 1990.

Oded Maron and Andrew W. Moore. Hoeffding races: Accelerating model selection search for classification and function approximation. In *Advances in Neural Information Processing Systems*, volume 6, pp. 59–66. Morgan Kaufmann, 1994.

Márcio Monteiro, Weichen Li, Puyu Wang, Marius Kloft, and Sophie Fellenz. Landmark-guided policy optimization for multi-objective language model selection. In *Proceedings of the 43rd International Conference on Machine Learning*, Proceedings of Machine Learning Research. PMLR, 2026. URL https://openreview.net/forum?id=fexmo79Kxo.

Aidar Myrzakhan, Sondos Mahmoud Bsharat, and Zhiqiang Shen. Open-LLM-leaderboard: From multi-choice to open-style questions for LLMs evaluation, benchmark, and arena. *arXiv preprint arXiv:2406.07545*, 2024. doi: 10.48550/arXiv.2406.07545. URL https://arxiv.org/abs/2406.07545.

Patrik Okanovic, Andreas Kirsch, Jannes Kasper, Torsten Hoefler, Andreas Krause, and Nezihe Merve Gürel. All models are wrong, some are useful: Model selection with limited labels. In *Proceedings of the 28th International Conference on Artificial Intelligence and Statistics*, volume 258 of *Proceedings of Machine Learning Research*, pp. 2035–2043. PMLR, 2025. URL https://proceedings.mlr.press/v258/okanovic25a.html.

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. *arXiv preprint arXiv:2406.18665*, 2024.

Lorenzo Pacchiardi, Lucy G. Cheke, and José Hernández-Orallo. 100 instances is all you need: Predicting the success of a new LLM on unseen data by testing on a few instances. *arXiv preprint arXiv:2409.03563*, 2024.

Yotam Perlitz, Elron Bandel, Ariel Gera, Ofir Arviv, Liat Ein-Dor, Eyal Shnarch, Noam Slonim, Michal Shmueli-Scheuer, and Leshem Choshen. Efficient benchmarking of language models. *arXiv preprint arXiv:2308.11696*, 2023.

Antonio Purificato, Maria Sofia Bucarelli, Andrea Bacciu, Fabrizio Silvestri, and Amin Mantrach. Select, label, evaluate: Active testing in NLP. *arXiv preprint arXiv:2603.21840*, 2026. doi: 10.48550/arXiv.2603.21840. URL https://arxiv.org/abs/2603.21840.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do ImageNet classifiers generalize to ImageNet? In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 5389–5400. PMLR, 2019. URL https://proceedings.mlr.press/v97/recht19a.html.

Mohammad Reza Karimi, Nezihe Merve Gürel, Bojan Karlaš, Johannes Rausch, Ce Zhang, and Andreas Krause. Online active model selection for pre-trained classifiers. In *Proceedings of the 24th International Conference on Artificial Intelligence and Statistics*, volume 130 of *Proceedings of Machine Learning Research*, pp. 307–315. PMLR, 2021. URL https://proceedings.mlr.press/v130/reza-karimi21a.html.

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*, volume 34, pp. 8732–8740, 2020. doi: 10.1609/aaai.v34i05.6399.

David Salinas, Omar Swelam, and Frank Hutter. Tuning LLM judge design decisions for 1/1000 of the cost. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 52728–52744. PMLR, 2025. URL https://proceedings.mlr.press/v267/salinas25a.html.

Christoph Sawade, Niels Landwehr, and Tobias Scheffer. Active comparison of prediction models. In *Advances in Neural Information Processing Systems*, volume 25, 2012. URL https://papers.nips.cc/paper/4495-active-comparison-of-prediction-models.

H. Sebastian Seung, Manfred Opper, and Haim Sompolinsky. Query by committee. In *Proceedings of the Fifth Annual Workshop on Computational Learning Theory (COLT)*, pp. 287–294, 1992. doi: 10.1145/130385.130417.

Hadas Shachnai, Rotem Shavitt, and Andreas Wiese. Finding possible winners in spatial voting with incomplete information. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 4048–4056, 2025. doi: 10.24963/ijcai.2025/451.

Hans Ulrich Simon and Jan Arne Telle. MAP- and MLE-based teaching. *Journal of Machine Learning Research*, 25(96):1–34, 2024. URL https://jmlr.org/papers/v25/23-1086.html.

Wei Song, Zhenya Huang, Cheng Cheng, Weibo Gao, Bihan Xu, Guanhao Zhao, Fei Wang, and Runze Wu. IRT-Router: Effective and interpretable multi-LLM routing via item response theory. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2025.

Elad Tolochinsky, Yaniv Tenzer, and Yaniv Romano. Valid best-model identification for LLM evaluation via low-rank factorization. *arXiv preprint arXiv:2605.10405*, 2026. doi: 10.48550/arXiv.2605.10405. URL https://arxiv.org/abs/2605.10405.

Rajan Vivek, Kawin Ethayarajh, Diyi Yang, and Douwe Kiela. Anchor points: Benchmarking models with much fewer examples. *arXiv preprint arXiv:2309.08638*, 2023.

Toby Walsh. Uncertainty in preference elicitation and aggregation. In *Proceedings of the Twenty-Second AAAI Conference on Artificial Intelligence*, pp. 3–8. AAAI Press, 2007.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=rJ4km2R5t7.

Ganghua Wang, Zhaorun Chen, Bo Li, and Haifeng Xu. Cer-eval: Certifiable and cost-efficient evaluation framework for LLMs. *arXiv preprint arXiv:2505.03814*, 2025. URL https://arxiv.org/abs/2505.03814.

Laurence A. Wolsey. An analysis of the greedy algorithm for the submodular set covering problem. *Combinatorica*, 2(4):385–393, 1982. doi: 10.1007/BF02579435.

L. Xia and V. Conitzer. Determining possible and necessary winners given partial orders. *Journal of Artificial Intelligence Research*, 41:25–67, 2011. doi: 10.1613/jair.3186.

Taolin Zhang, Hang Guo, Wang Lu, Tao Dai, Shu-Tao Xia, and Jindong Wang. Sparseeval: Efficient evaluation of large language models by sparse optimization. In *International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=CZAzAedGSV.

A Extended Related Work

A.1 Active Testing and Limited-Label Model Selection

Active testing chooses evaluation points to estimate model performance more efficiently than uniform sampling (Kossen et al., 2021). Active comparison focuses directly on comparing predictive models and can exploit disagreement structure (Sawade et al., 2012). Model Picker studies online active selection among pretrained classifiers (Reza Karimi et al., 2021). Model Selector develops a limited-label selection method that combines model predictions with an acquisition mechanism and reports strong empirical performance (Okanovic et al., 2025). DiffUse targets label-efficient selection for text generation (Ashury Tahan et al., 2024). These works establish that query choice matters and that prediction disagreement is valuable side information.

Our target differs in the required guarantee. We condition on a fixed finite pool and ask whether one candidate is the strict winner under every completion of the unobserved labels. This is not a confidence statement under a sampling model. It is an exact finite-pool statement about the cached matrix and the labels already revealed. A probabilistic method can be connected to our stopper, and we do exactly that in the experiments. The resulting policy inherits exact completion validity from the stopper while preserving the acquisition preferences of the original method.

This distinction also changes what constitutes a fair baseline. Comparing label counts under different stopping rules would mix acquisition quality with guarantee strength. We therefore attach a common exact stopper to quotient uniform sampling and query-by-committee as well as Active Comparison and Model Selector, with SELECT-style acquisition evaluated under the same stopping condition. Post hoc oracle hyperparameter grids are reported as deliberately favorable diagnostic baselines, not as deployable procedures. The primary SELECT comparison uses a response-only structurally calibrated temperature.

A.2 Efficient LLM Evaluation

The rapid growth of LLM candidate pools has produced methods that estimate aggregate benchmark performance from fewer items. tinyBenchmarks uses item and model structure to reduce evaluation size (Maia Polo et al., 2024). Scaling Active Testing extends active evaluation to large language models (Berrada et al., 2025). SparseEval frames efficient evaluation through sparse optimization (Zhang et al., 2026). Cer-Eval studies certifiable and cost-efficient LLM evaluation (Wang et al., 2025). Other recent work studies valid best-model identification with low-rank structure (Tolochinsky et al., 2026), active testing in NLP (Purificato et al., 2026), and adaptive stopping with sufficient data (Arviv et al., 2026).

SELECT-LLM is the closest acquisition baseline for our main experiment (Durmazkeser et al., 2026). It constructs a acquisition weighting over candidates and chooses queries using candidate response patterns. We reproduce its query objective with high-precision tie refinement, adapt it to our finite-pool target, and compare it under the same exact completion stopper. This comparison is intentionally narrow. It establishes that certificate feedback can improve the acquisition module on our target. It does not establish global dominance over the original method on every target or stopping rule.

LLM evaluation is an application rather than a training contribution in this paper. The candidate response matrices are fixed. Acquiring a label reveals correctness for all candidates that answered the item. The framework can therefore be used after expensive inference has already been cached, or when inference is inexpensive relative to expert labeling. It uses cached predictions without gradients or model internals and requires neither reward-model training nor policy optimization.

A.3 Cost-Aware Configuration Selection

JudgeTuning demonstrates that LLM judge design involves a large configuration space and that careful tuning can greatly reduce evaluation cost (Salinas et al., 2025). LAMPS studies multi-objective language model selection with landmark-guided policy optimization (Monteiro et al., 2026). These settings motivate deployment maps in which the preferred candidate varies with a resource cap. Our contribution is not a new cost predictor. We take measured or declared costs as fixed inputs and ask how many shared labels are required to certify the selected configuration at every relevant cap.

The cost-map target is stronger than selecting one global winner. A single acquired label can support several regions, but the procedure must simultaneously rule out different challengers in different feasible sets. This coupling produces a structured covering problem. It also reveals a practical limit: even an optimized certificate can require a large share of the pool when many region-specific alternatives remain viable.

A.4 Teaching, Set Cover, and Certificates

Machine teaching studies how a teacher selects examples that identify a target concept under assumptions about the learner (Cicalese et al., 2020; Simon & Telle, 2024). Our completion certificate is related in spirit because a subset of labeled items must force a decision. The learner model and target are different. Here the candidate predictions are fixed, the desired object is an empirical winner or a deployment map, and validity is quantified over all completions of one finite label vector.

Once a fixed cost-map certificate is derived, classical covering theory applies. Greedy Set Cover has a logarithmic guarantee (Chvátal, 1979). Submodular cover admits related analyses (Wolsey, 1982), and logarithmic hardness is classical (Feige, 1998). We use these results as inherited optimization tools. We distinguish this derivation from the established approximation algorithms used afterward.

Table 4 summarizes the conceptual boundary. No single binary feature fully captures the differences among these works. In particular, a method without an adversarial-completion guarantee may still provide a strong statistical guarantee for its intended data-generating process. Our claim is narrower: when the desired guarantee is exact on a fixed pool, the completion certificate gives a common boundary on which acquisition methods can be compared without changing the validity target.

B Extended Setting and Certificate Material

This appendix supports Sections 3 and 4 with the detailed material below.

B.1 Practical Instances of the Finite-Pool View

This section describes finite-pool applications in support of Section 3.

The finite-pool view matches several practical workflows. A laboratory may cache responses from a set of fixed LLMs and send a subset of prompts for expert grading. A benchmark owner may have predictions from many submitted systems but can afford to verify only part of the test set. A company

Table 4: Relationship to representative limited-label evaluation approaches. "Common stopper" means that our experiment attaches the same exact completion condition to the acquisition module. Entries describe the target used in this paper and do not redefine the original scope of cited methods.

Approach	Primary object	Shared item label	Exact finite-pool completion guarantee	Deployment cost map	Role in our study
Active Testing (Kossen et al., 2021)	Performance estimation	Yes	No	No	Conceptual precursor
Active Comparison (Sawade et al., 2012)	Pairwise model comparison	Yes	No	No	Adapted acquisition baseline
Model Picker (Reza Karimi et al., 2021)	Online model selection	Yes	No	No	Related active selector
Model Selector (Okanovic et al., 2025)	Limited-label selection	Yes	No	No	Adapted strong baseline with common stopper
SELECT-LLM (Durmazkeser et al., 2026)	LLM selection with annotations	Yes	No	No	Closest acquisition baseline with common stopper
Cer-Eval (Wang et al., 2025)	Certifiable LLM evaluation	Yes	Different certificate target	No	Contemporary certifiable evaluation
This work	Empirical winner and fixed cost map	Yes	Yes	Yes	Exact stopper, acquisition feedback, and burden theory

may compare several judge prompts or inference settings on the same set of human-rated instructions. In all three cases, one gold label can be reused and the output is discrete.

This definition deliberately separates finite-pool correctness from sampling uncertainty. A confidence interval can be useful when the target is expected performance under a population distribution. Here the primary target is the empirical winner on the fixed evaluation pool. The only uncertainty before full labeling is which labels occupy the unrevealed coordinates.

Strictness removes ambiguity from the main theorem. A deterministic tie rule can be incorporated by changing the pairwise inequalities according to the rule. Our empirical panels require a unique full-pool winner, and every reported exact run returns that winner. Panels without a unique population winner are excluded before evaluation rather than resolved using labels seen by an acquisition method.

B.2 Why Estimation Error Is Not the Decision

This section describes the distinction between estimation error and model choice in support of Section 3.

Small aggregate estimation error does not imply correct selection. Suppose two candidates differ on only a small subset of items. A score estimator can be accurate on average while missing precisely those items, leading to a reversed winner. The opposite can also occur. If all remaining disagreement favors the current leader by too little to change the order, the winner is already determined even though the absolute scores of both candidates remain uncertain.

Figure 3 measures this distinction on seven LLM task matrices. Uniform sampling improves exact-winner recovery with budget, but several tasks remain below 50% even after 200 labels. At 50 labels the mean regret of the selected winner ranges from 2.84 to 5.70 percentage points. The result does

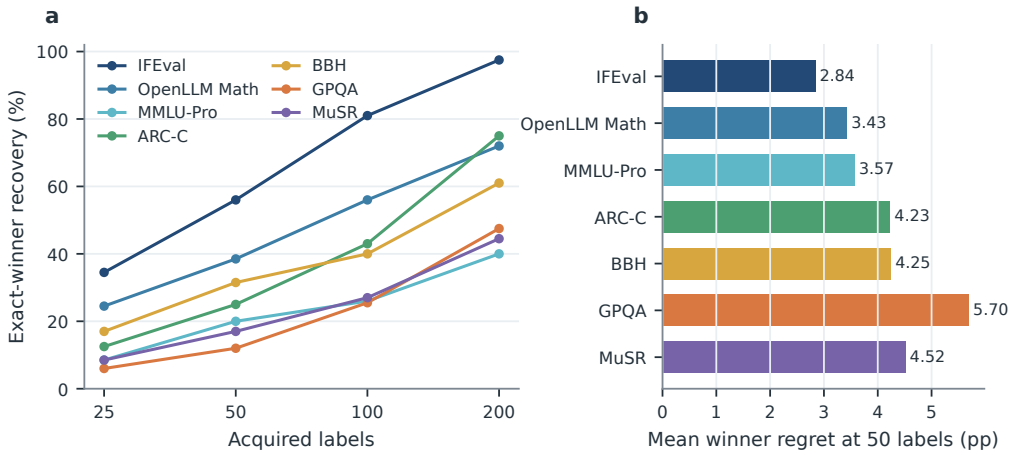

Figure 3: Zero-method pain measurement on seven cached LLM task matrices with 200 random trials per budget and task. (a) Probability that uniform sampling recovers the exact full-pool winner. (b) Mean regret of the selected winner at 50 labels. A low-cost estimate can be numerically plausible while the discrete decision remains unstable.

not say that uniform estimates are useless. It says that an estimation budget is not itself a decision certificate.

B.3 Information Interface

This section describes the shared-label information interface in support of Section 3.

Our framework assumes that candidate predictions are already available on the pool. It counts acquired gold labels, not candidate inference calls. This interface is suitable when responses have been cached, when multiple methods share a benchmark repository, or when expert adjudication dominates inference cost. It is not a substitute for methods whose primary cost is model inference.

This boundary matters for comparisons with inference-row sampling. If a method requests a separate sample for each model, mapping that request to reusable shared item labels can change the cost substantially. In our measured interface, the default ten-samples-per-model setting corresponds to an average of 496.6 to 1,267.6 distinct item labels across panels, a duplication factor of 49.66 to 126.76 relative to the nominal per-model count. We report this mismatch to explain why the interfaces differ. We do not use it to reject such methods on the tasks for which they were designed.

B.4 Scope Boundaries of the Claims

This section describes the scope of the stated guarantees in support of Section 3.

Several boundaries are important. We do not train or fine-tune a large language model. The large models appear as fixed candidates whose cached outputs define the prediction matrix. No preference optimization or reinforcement learning from human feedback is used. We do not claim that the original SELECT-LLM problem and our completion-safe target are identical. We compare its acquisition module only after adapting it to the same candidate pool and attaching the identical exact stopper. We also claim no acquisition saving for a complete cost map. The cost-map contribution is the certificate formulation, the measured burden and the optimization tightness.

B.5 Evaluation Quantities

This section describes the reported evaluation quantities in support of Section 6.

For a method m on run t, let L_{mt} be labels acquired at the exact stopping time. Against baseline b, the paired relative saving is

$$\Delta_{m,b,t} = 1 - \frac{L_{mt}}{L_{bt}}, \tag{7}$$

with positive values favoring m. When several replicates share one organization panel, the panel is the primary inference unit. We take the median of paired savings within each panel and then take the median across panels. Bootstrap intervals resample panels. Run-level summaries are retained as secondary descriptions.

For a deployment map, we report certificate size as a fraction of the item pool. The any-output lower bound, fixed-map linear relaxation and greedy fixed-map certificate answer different questions. The first is a universal floor for any valid output. The second and third concern one fixed target map. A fixed-map linear-program value is not claimed as a lower bound for the more flexible any-output problem.

The distinction among three cost-map objects is important throughout the paper. The pain experiment uses 20 JudgeTuning caps and asks whether all 20 decisions are recovered together. The certificate-complexity experiment expands the same workload to 331 distinct feasible regions. At a two-point tolerance, the regional any-output lower bounds are minimized by 16 distinct candidates. Nine minimize the bound in three or more regions. The WinoGrande study contains 135 organization panels, each with its own three to 15 cost regions. Statistics from these objects are not interchangeable.

B.6 Strict Dominance Between Histories

This section describes dominance comparisons between observed histories in support of Section 3.

The certificate also induces a partial order over histories. Consider two histories with the same cached matrix. History H_1 dominates H_2 for candidate a if every pairwise slack

$$\gamma_{ab} = d_{ab} - r_{ab} \tag{8}$$

is at least as large under H_1 and one is strictly larger. If H_2 certifies a, then H_1 certifies a. This order provides a deterministic way to compare evidence states even when they have acquired different items.

The order does not totally rank acquisition policies. One query can improve slack against one challenger while another query improves a different constraint. This is precisely why acquisition remains nontrivial after the exact stopper is known. FrontierCut uses a coarse but effective surrogate for the number of unresolved constraints an item can affect. CertSELECT instead feeds the survivor set back into a probabilistic query objective.

B.7 Finite Alphabets and Preset Weights

This section describes finite label alphabets and preset item weights in support of Section 3.

Theorem 3 holds for any finite label alphabet and any preset nonnegative weights. Multiclass labels do not enlarge the pairwise extremum because a disagreement item still has only three possible pairwise contributions: $+w_i$, 0, and $-w_i$. A label matching neither prediction contributes zero. Negative weights are excluded because they would reverse the interpretation of correctness evidence and invalidate the monotone disagreement radius.

Weights must be fixed independently of unseen labels. Examples include importance weights fixed in advance for a benchmark mixture, or known repeated-observation weights. If weights are estimated adaptively from acquired labels, an additional argument is needed because querying an item can change both its observed contribution and the scoring functional itself.

B.8 Mechanical Verification

This section describes the finite enumeration checks in support of Section 3.

The enumeration checks robust-winner equivalence and possible-winner equivalence. It also checks simultaneous maxima and pairwise minima. Further checks concern elimination order and the equivalence between a singleton survivor set and a robust winner. Table 5 gives the totals.

Mechanical verification is not a substitute for proof. Its role is to detect sign errors and quantifier mistakes that are easy to make in completion arguments. In particular, it distinguishes the simultaneous favorable completion used for possible-winner reasoning from the challenger-specific adverse completion used for robust-winner reasoning.

Table 5: Finite-state verification of the exact certificate. Every listed check had zero mismatches.

Verification quantity	Count	Verification quantity	Count
Prediction matrices	26,278	Candidate checks	4,994,268
Partial histories	1,688,292	Random weighted states	10,000
Completion evaluations	5,590,672	Observed mismatches	0

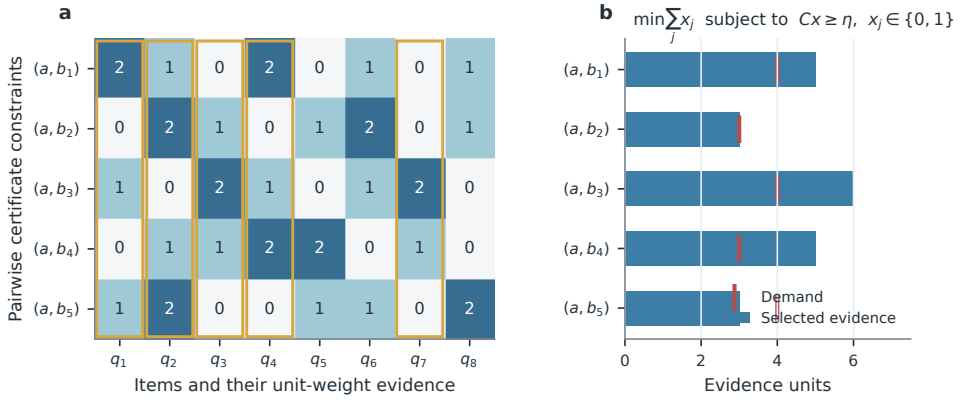

Figure 4: Fixed-map certificate as a unit-weight integer cover. (a) Rows are region–challenger constraints and columns are items. Gold outlines mark selected labels. Coefficients are exact evidence contributions in $\{0, 1, 2\}$. (b) A certificate covers every row demand. The displayed matrix is schematic, while all reported experiments solve constraints derived from cached response matrices.

B.9 What the Certificate Does Not Assume

This section describes the assumptions of completion certificates in support of Section 3.

The certificate does not assume independent items or calibrated model probabilities. It also does not require low rank, exchangeability or a realizable data-generating model. It also does not require that the acquisition rule ignore observed labels. Any adaptive policy may use the current history as long as it applies the exact condition before output.

These strengths come with a cost. Completion safety is deliberately conservative relative to a high-probability statistical guarantee. If many unobserved items separate close candidates, the remaining disagreement radius can stay large even when a statistical model strongly favors one candidate. Our experiments therefore evaluate label burden directly rather than assuming exactness will always be inexpensive.

B.10 Cover Structure of a Fixed Map

This section describes the covering structure of fixed maps in support of Section 4.

Figure 4 illustrates how one selected item can contribute to several pairwise constraints.

In binary tasks, a disagreement item must match either a or b. Therefore its coefficient is either two or zero. Dividing both sides by two turns each pairwise condition into a multicover demand over witness items that favor the target. This specialization makes the combinatorial meaning transparent: every challenger must be refuted by enough selected labels on which the target is correct and the challenger is wrong.

The witness view also explains why pure disagreement sampling can be wasteful. Two items may separate exactly the same candidates before their labels are known. After labeling, one may support the desired map while the other supports a challenger. A certificate constructed from realized labels uses realized evidence. An online policy must choose before observing which direction the evidence will take.

The fixed-map formulation assumes that the target winner for each region is specified. An any-output policy may instead return whichever map becomes cheapest to prove. Introduce a binary choice for

each feasible candidate–region assignment and require one certified output per region. The resulting program is a disjunction over fixed-map covers with cross-region sharing of item variables. This flexibility cannot make the problem harder, so any-output certificate size is at most the best fixed-map certificate size. At the same time, a lower bound for one canonical map is not a universal lower bound for any-output certification. We maintain this distinction in every empirical figure. Our simple any-output floor takes the maximum of necessary pair-demand and demand-volume bounds across regions. It is valid for all output maps but can be looser than the fixed-map linear relaxation.

Theorem 8 does not imply that every practical instance is difficult. It explains why a general closed-form minimum should not be expected and motivates lower bounds plus constructive approximations.

B.11 Certificate Burden Versus Acquisition Savings

This section describes the distinction between certificate burden and acquisition savings in support of Section 4.

The cost-map formulation serves two roles. First, it measures an evidence floor: some deployment maps intrinsically require many labels under completion safety. Second, it provides a target based on realized labels against which online acquisition can be compared. These roles do not by themselves establish a successful new online method.

Allowing the output map to switch reduces the two-point certificate count on JudgeTuning from 2,111 to 2,029, or 3.88%. The cost-map results therefore measure certificate burden, and the acquisition claims in the paper are restricted to single-region exact selection.

B.12 Shared-Label Scores and Histories

This section supports the score and observation definitions in Section 3. It explains their special cases and the information available to an acquisition policy.

Here M and N are the numbers of candidates and items. Candidate indices are a and b, while i and j index items.

A zero weight excludes an item from the score, while unit weights give the ordinary count of correct predictions.

An acquisition policy may also use internal randomness to resolve query choices. A stopping rule either requests another label or outputs a candidate.

This section describes applications of this shared-label interface.

Equation (1) defines the scores that a completion certificate must compare. However, evaluating every completion would require enumerating all assignments to the unobserved items. We instead seek exact bounds on each pairwise score difference. To obtain them, we separate the observed lead from the unobserved disagreements that can change it.

The extremes in Lemma 2 give attainable upper and lower bounds for every comparison. We now need to turn these bounds into decisions about stopping and candidate elimination. We use the lower bounds to certify a strict winner and the simultaneous upper bounds to identify possible winners.

Existing finite-state checks of these equivalences are reported in Table 5.

The smallest lead in Lemma 2 must be positive against every rival to permit a completion-safe output.

B.13 Proofs of the Single-Region Results

This section supports the stopping and elimination results of Section 3. We restate each result before its proof argument. The full itemwise and pathwise proofs are in Appendix H.

A positive d_{ab} means that a currently leads b. If $r_{ab} = 0$, their score difference is already fixed. Agreement items cancel because their correctness indicators are identical for every gold label. On a disagreement item, the contribution is $+w_i$ when the label matches a and $-w_i$ when it matches b. A label matching neither prediction contributes zero. Consequently, the full score difference lies between $d_{ab} - r_{ab}$ and $d_{ab} + r_{ab}$.

The certificate also determines whether another stopping rule could validly stop earlier on the same sequence of queries and observed labels.

B.13.1 Favorable Extreme in Lemma 2

Lemma 2 (restated). For any candidate a, the completion $y_i = p_{ia}$ for all $i \in \mathcal{U}$ attains $\max_z(S_a(y^{(z)}) - S_b(y^{(z)})) = d_{ab} + r_{ab}$ simultaneously for every challenger b.

Proof sketch. This assignment contributes $+w_i$ on each disagreement with a and zero on every agreement. Appendix H gives the complete itemwise proof. The same assignment attains every pairwise upper bound because it uses only a's predictions.

B.13.2 Adverse Extreme in Lemma 2

Lemma 2 (restated). For any ordered pair (a, b), $\min_z(S_a(y^{(z)}) - S_b(y^{(z)})) = d_{ab} - r_{ab}$.

One minimizing completion assigns p_{ib} on each unobserved item where $p_{ia} \neq p_{ib}$. *Proof sketch.* The stated assignment contributes $-w_i$ on every unobserved disagreement and zero on every agreement. Summing these contributions attains the lower bound, as detailed in Appendix H.

Different rivals may require different minimizing completions. Checking every pairwise lower bound nevertheless suffices because strict victory requires superiority over every rival.

B.13.3 Proof of Theorem 3

Theorem 3 (restated). Candidate a is a robust strict winner if and only if $d_{ab} > r_{ab}$ for all $b \neq a$.

Proof sketch. When all inequalities hold, Lemma 2 gives a positive score difference against every rival under every completion. The full proof and deterministic tie-rule variants appear in Appendix H. If an inequality fails, its minimizing completion makes that rival tie or beat a.

B.13.4 Proof of Corollary 4

Corollary 4 (restated). Candidate a is excluded from the possible-winner set if $d_{ab} + r_{ab} < 0$ for some rival b.

Equivalently, a is a possible winner if and only if $d_{ab} + r_{ab} \geq 0$ for every $b \neq a$. *Proof sketch.* Lemma 2 attains all upper bounds in the same completion. Their nonnegativity is therefore equivalent to a matching or exceeding every rival in that completion. Appendix H states the strict and weak winner variants.

A singleton possible-winner set is equivalent to a robust strict winner. If a rival could tie or beat its sole member under some completion, a different candidate would also attain a best score.

B.13.5 Proof of Corollary 6

Corollary 6 (restated). Along any fixed acquisition path, the first history at which one candidate satisfies Theorem 3 is the earliest history at which any completion-safe policy can output a strict winner.

Proof sketch. Appendix H gives the pathwise argument. Theorem 3 makes its condition necessary for every valid output and sufficient for immediate stopping.

B.14 Regional Evidence Contributions

This section supports the cost-map definitions and evidence coefficients in Section 4. It relates regional tolerance to pairwise evidence and explains the binary multicover specialization.

Theorem 3 certifies one choice among all candidates. Deployment under different inference cost caps requires several choices supported by the same labels. We define this joint decision target and express its certificate as an integer covering problem. We then use the covering structure to bound the minimum number of required labels.

Larger tolerances permit more regional choices.

By Lemma 2, candidate $a = g(r)$ meets the tolerance against rival b under every completion exactly when

$$r_{ab} - d_{ab} \leq \epsilon_N,$$

where $r_{ab} - d_{ab}$ is the largest advantage that b can attain over a. Thus a regional certificate bounds the score shortfall of its chosen candidate.

The regional requirement derived from Lemma 2 bounds each rival's attainable advantage. However, minimizing the joint label count requires tracking how one item supports several regional comparisons. We therefore express each acquired label as evidence toward these comparisons. This gives a covering problem in which an item is counted once even when it supports several choices.

The following terms refer to Equation (3). The outer indicator removes agreement items. For a disagreement item, the constant term removes its unresolved unit of advantage for b. The two correctness indicators then add the observed support for a and subtract the support for b. A label matching a gives two units of evidence, whereas a label matching b gives zero. A multiclass label matching neither prediction gives one unit.

A zero demand means that the initial disagreement already lies within tolerance. The comparison is certified exactly when its selected evidence reaches η_{ab}. We now combine these requirements across regions.

For binary labels, each disagreement must match one of the two predictions. Equation (3) then takes only the values zero and two. Dividing the evidence constraints by two gives a multicover problem. Each selected item supports the target against some rivals, and each rival requires a specified number of such supporting items. This section explains the binary interpretation and the any-output formulation that also optimizes the regional choices. The evidence matrix supplies the link from completion safety to this established covering structure.

Greedy certificate construction. For a certifiable fixed map, index evidence rows by $k = (r, b)$ and initialize residual demands $s_k = \eta_k$ and the selected set $T = \varnothing$. At each step select an unselected item maximizing $\sum_k \min\{s_k, C_{kj}\}$, add it to T, and update $s_k \leftarrow \max\{0, s_k - C_{kj}\}$. Stop when all residual demands are zero. The selected set then satisfies every row of Eq. (4), so $|T|$ is a feasible upper bound on the minimum certificate size. For the binary implementation, use witness coefficients $C_{kj}/2 \in \{0, 1\}$ and demands $\lceil \eta_k/2 \rceil$; its integer gain counts currently unmet rows supported by the item. Item-index order resolves equal gains. If unmet rows have no remaining supporting item, the map is infeasible.

B.15 Covering Proofs and Greedy Bound

This section supports Theorems 7 and 8 in Section 4. It also records the inherited greedy bound used to interpret certificate size. Full covering derivations are in Appendix I.

B.15.1 Proof of Theorem 7

Theorem 7 (restated). For a unit-weight finite pool with realized labels and a fixed target map g, the minimum fixed-map certificate size at tolerance ϵ_N is the value of

$$\min_{x \in \{0,1\}^N} \sum_{j=1}^{N} x_j \quad \text{subject to} \quad \sum_{j=1}^{N} C_{g(r)b,j} x_j \geq \eta_{g(r)b} \ \text{ for every } r \in \mathcal{R},\ b \in \mathcal{A}_r \setminus \{g(r)\},$$

where the binary vector x selects the shared labeled subset.

Proof sketch. Each coefficient in Equation (3) is the exact reduction in a rival's attainable advantage. These reductions add across selected items. Satisfying all demands is therefore equivalent to meeting the tolerance in every region. Appendix I provides the complete derivation.

Equation (4) characterizes the minimum label count for a fixed map. Computing that minimum requires solving a discrete optimization problem. We establish its computational difficulty and then construct lower and upper bounds using the covering formulation.

The difficulty already arises when there is only one region and the labels are binary.

B.15.2 Proof of Theorem 8

Theorem 8 (restated). Minimum certification with one region and binary labels is NP-hard for a fixed target map and, on the same instances, in the any-output form.

Proof sketch. We reduce Set Cover to certification by introducing a target candidate and a challenger for each universe element. Each set becomes an item whose label supports the target against the challengers corresponding to elements in that set. The construction makes the target the only acceptable output. A certificate with a given number of labels exists exactly when a set cover of that size exists. Appendix I gives the full construction. Thus the reduction covers both fixed-map and any-output certification.

B.15.3 Integer-Residual Greedy Bound

This bound supports the greedy certificate construction in Section 4. For a feasible fixed map with positive total demand, let k denote its minimum integral certificate size. Let R be the initial total integer demand and let Q_{greedy} be the selected item set. The standard integer-residual argument gives

$$|Q_{\mathrm{greedy}}| \le k + \lceil k \log(R/k) \rceil,$$

where $\log$ is the natural logarithm. The logarithmic term bounds the steps needed to reduce the residual to at most k. The remaining k term allows one additional label per residual unit. Appendix I.4 gives the general integer-demand proof, and Section 6.5 compares feasible certificates with their fixed-map LP bounds. If $R = 0$, the empty set is already a certificate.

B.16 Consequences of the Score Assumptions

This section collects the score special cases supporting Section 3. We use $w_i = 1$ in the experiments unless stated otherwise. Shared nonnegative weights make the itemwise bounds attainable while allowing arbitrary dependence among items. The guarantee therefore applies to adaptively chosen queries without assigning probabilities to completions. When $\mathcal{U}$ is empty, the strict certificate reduces to ordinary strict comparison of the fully observed scores. The coefficient matrix characterizes the smallest sufficient subset of realized labels. The greedy optimization principle is inherited from covering methods.

B.17 Why the Pairwise Test Is Exact Only for Shared Weights

Corollary 4 decides the possible-winner set through $M - 1$ inequalities per candidate. The reason is the simultaneous attainment in Lemma 2. Setting $y_i = p_{ia}$ on every unobserved item maximizes $S_a(y^{(z)}) - S_b(y^{(z)})$ for all rivals b at the same time, because the weight w_i that this choice contributes to the pair (a, b) does not depend on b. One completion therefore witnesses all $M - 1$ pairwise maxima, and passing every pairwise test is sufficient as well as necessary.

This simultaneity fails once each candidate carries its own item weights, as in the risk model of Kuboyama (2026). Write $w_{ia} \ge 0$ for the weight that candidate a places on item i, and keep the shared gold label. Consider three candidates a, b and c, two unobserved items 1 and 2 on which the three candidates predict alike, and one observed item 3 on which only a is wrong. Let $w_{\cdot a} = (1, 1)$, $w_{\cdot b} = (2, 0)$ and $w_{\cdot c} = (0, 2)$ on the two unobserved items, and let item 3 carry weight one for every candidate. Write $e_i \in \{0, 1\}$ for the event that the completion makes the shared prediction on item i wrong. The penalties are $e_1 + e_2 + 1$ for a, $2e_1$ for b and $2e_2$ for c, and the smaller penalty wins. Reading the same weights through Equation (1) gives the scores $2 - e_1 - e_2$, $3 - 2e_1$ and $3 - 2e_2$, which produce the same survivors and the same absence of a best completion. Candidate a survives its test against b only at $(e_1, e_2) = (1, 0)$ and survives its test against c only at $(e_1, e_2) = (0, 1)$. Both pairwise tests pass, yet no single completion makes a best, so the pairwise survivor set is a strict superset of the possible winners. Neither b nor c beats a under every completion either, so pairwise domination does not remove a. Under shared weights the same configuration cannot arise, because $y_i = p_{ia}$ resolves every rival at once.

Exhaustive enumeration of the four completions confirms this construction. A synthetic check then draws 10,000 random instances with three candidates and four unobserved binary items, where the item weights and the observed-item offsets are uniform on $\{0, 1, 2, 3\}$ and the shared predictions are fair coin flips. At least one such false survivor appears in 2.89% of them with a Wilson 95% interval

Algorithm 1 CERTSELECT with exact stopping

Require: Binary cached predictions P, unit item weights, calibrated $\tau > 0$, label oracle and seeded tie stream. The full pool has a unique winner

Set $E_a = 0$, $\mathcal{O} = \varnothing$, $\mathcal{U} = \mathcal{Q}$, $A = \mathcal{A}$ and informative query pool $\mathcal{I} = \{i : \exists a, b,\ p_{ia} \neq p_{ib}\}$
Initialize $d_{ab} = 0$ and r_{ab} by Eq. (2)
while $|A| > 1$ **do**
  Recompute masses $\widetilde{\pi}_a \propto \exp(E_a/\tau)$ on A, normalized to sum to one; set other masses to zero
  For $i \in \mathcal{I} \cap \mathcal{U}$, compute $u_i = \sum_{a:p_{ia}=1} \widetilde{\pi}_a$ and minimize $u_i^2 + (1-u_i)^2$
  Refine numerical near-ties, then draw one minimizing item using the seeded tie stream
  Acquire y_i, move i from $\mathcal{U}$ to $\mathcal{O}$, and set $E_a \leftarrow E_a + \mathbb{I}[p_{ia} = y_i]$ for all a
  Update d_{ab}, r_{ab} and retain a in A exactly when $d_{ab} + r_{ab} \geq 0$ for every rival
end while
return The unique candidate in A and its completion certificate

from 2.58% to 3.24%. Of the 27,999 candidates that pass every pairwise test in these instances, 289 are not possible winners. Repeating the same enumeration with shared weights produces no disagreement between the pairwise tests and the enumerated possible winners. These frequencies describe the stated synthetic distribution and do not estimate how often candidate-specific weights arise in practice.

C ACQUISITION DETAILS

This appendix supports Section 5 with the detailed material below.

C.1 CERTIFICATE STATE AND COMPLEXITY

This section describes certificate state and complexity in support of Section 5.

At initialization, every candidate is a survivor. We store observed margins $d_{ab} = 0$ and full-pool disagreement weights r_{ab} for every pair. Revealing y_i adds $w_i(\mathbb{I}[p_{ia} = y_i] - \mathbb{I}[p_{ib} = y_i])$ to d_{ab} and subtracts $w_i \mathbb{I}[p_{ia} \neq p_{ib}]$ from r_{ab}. Corollary 4 removes a candidate only when it cannot be best, even jointly with a rival, under any completion. This exclusion is permanent because additional labels only reduce the set of completions.

Initializing the pairwise state takes $O(M^2 N)$ time and $O(M^2)$ additional storage beyond P. Each label update and survivor test takes $O(M^2)$ time. Separately, grouping surviving candidates by their predicted labels computes all FRONTIERCUT query scores in $O(N|A_t|)$ time per round, compared with $O(N|A_t|^2)$ for explicit pair enumeration. Items in $\mathcal{I}$ are the available informative queries; globally constant prediction columns stay in the evaluation pool.

Identical response patterns can share a query-score computation. The implementation draws one concrete remaining item from tied patterns and observes only its label. Other items in that pattern remain unlabeled. After each observation, elimination changes A_t and the pair-separation scores are recomputed.

C.2 NUMERICAL TIE REFINEMENT IN SELECT ACQUISITION

This section describes numerical tie refinement in select acquisition in support of Section 5.

Float64 query scores shortlist patterns within 10^{-10} of the apparent minimum. We then group candidates by integer evidence level and rank the shortlisted patterns by their absolute imbalance using extended precision. If m is the smallest extended-precision imbalance, retain values within $256\epsilon_{\mathrm{ld}} \max\{1, |A_t|\} \max\{1, m\}$ of it, where ϵ_{ld} is the arithmetic's machine epsilon. Distinct unresolved coefficient vectors are compared using 80-digit decimal arithmetic; exact equality is passed to the seeded tie stream. Canonicalizing a binary split and its complement makes the ordering invariant to label orientation. These calculations refine acquisition scores, while the exact certificate alone determines stopping.

The masses determine which query is useful, while the survivor test determines which output is certified. Temperature therefore affects acquisition efficiency and leaves the stopping condition unchanged.

C.3 Static and Adaptive Controls

This section describes static and adaptive controls in support of Section 5.

Quotient uniform samples a random permutation after removing constant columns. Static QBC ranks items once by response entropy across all candidates. Static all-pair cut ranks items once by the number of separated candidate pairs. Active Comparison and Model Selector are adapted to the shared-label matrices and connected to the same exact stopper. For Model Selector, we also report the best result across a fixed post hoc grid to avoid weakening that baseline through one unfavorable choice.

These controls isolate three sources of gain. Comparing to full uniform measures the value of removing information-free columns. Comparing to quotient uniform measures the value of nonuniform acquisition. Comparing FrontierCut to static pair cut measures the value of updating the candidate frontier. Comparing CertSELECT to all-candidate SELECT measures the value of exact survivor feedback within the same acquisition-mass split acquisition family.

C.4 Pair-Separation Baselines and State Computation

This section supports the structural query rule in Section 5. It explains static all-pair cut and the computation of the pair-separation score and certificate state.

Section 4 characterizes certificate size using realized labels. We now choose items before observing their labels to reach the single-region stopping condition in Theorem 3. We study two ways to use the possible winners from Corollary 4. Counting the pairs separated by an item gives FrontierCut. Restricting evidence-based candidate weights to possible winners gives CertSELECT. We describe the shared state before presenting each query rule.

Acquiring item i adds its observed weighted score difference to d_{ab}. It also subtracts w_i from r_{ab} whenever the pair disagrees on that item.

Before acquisition we remove items on which all candidates predict the same label. These items remain unobserved and contribute zero to Equation (2). The remaining items form the initial query pool. Randomly ordering this pool gives the quotient-uniform baseline used in Section 6. The two rules below rank these informative items using the current candidate state.

The survivor set from Corollary 4 records which candidates remain relevant to selection. A direct structural baseline nevertheless ranks items once by their disagreement across all candidates. This *static all-pair cut* baseline continues to count pairs involving candidates eliminated by later observations. We seek to focus each query on comparisons among current survivors. FrontierCut does so by recomputing the same pair-separation count over A_t after each acquired label.

The count $n_{i\ell}^{(t)}$ and score $q_t(i)$ are defined in Equation (5). The superscript records the current survivor set.

The term $|A_t|^2$ counts ordered pairs including self-pairs. The sum subtracts pairs giving the same answer, which also removes every self-pair. Dividing by two counts each remaining unordered pair once. Hence $q_t(i) = 0$ when all survivors agree on item i. For binary labels, the score is the product of the two response-group sizes and is largest for a balanced split.

The score counts the current pairwise comparisons that an item can affect. The acquired label then determines the direction of its evidence and which candidates can be eliminated.

Grouping candidates by their response computes Equation (5) in $O(N|A_t|)$ time per step. Explicit pair enumeration would take $O(N|A_t|^2)$ time. The stored pairwise state occupies $O(M^2)$ space. This section describes response-pattern grouping and state updates. The next subsection incorporates cumulative correctness through the evidence-weighted acquisition baseline SELECT and its restriction to survivors.

C.5 SELECT Weights and Temperature Calibration

This section supports the SELECT baseline and its survivor restriction in Section 5. It explains the binary objective and the response-only temperature choice.

Equation (5) treats every survivor equally when counting separated pairs. It leaves accumulated correctness evidence out of the query ranking. To incorporate that evidence, we adapt the acquisition objective of SELECT-LLM (Durmazkeser et al., 2026). The objective favors items that divide candidate weight between different answers. This fits the shared-label interface because the acquired answer updates the evidence for every candidate.

The masses $\pi_a^{(t)}$ are defined in Section 5. The masses sum to one over all candidates. A smaller temperature concentrates mass on candidates with more observed correct answers. A larger temperature spreads the mass more evenly.

The mass predicting 0 is then $1 - u_i$. SELECT chooses an item minimizing

$$h_t(i) = u_i^2 + (1 - u_i)^2,$$

where $h_t(i)$ is the binary query objective reproduced from the SELECT acquisition module. Its two terms measure the squared masses of the response groups. The objective is minimized at $u_i = 1/2$ and maximized at $u_i = 0$ or $u_i = 1$. Minimization therefore favors a balanced division of candidate mass.

If $A_t = \mathcal{A}$, Equation (6) recovers SELECT. With uniform masses over binary survivors, minimizing $h_t(i)$ gives the same query ranking as maximizing Equation (5). In Equation (6), the numerator keeps the original evidence weight. The denominator redistributes all mass among survivors, while eliminated candidates receive zero.

Response-only calibration. For item i, candidate a receives the observable proxy $M^{-1} \sum_b \mathbb{I}[p_{ia} = p_{ib}]$. We accumulate this agreement proxy in place of gold correctness and measure, after each query, whether a candidate with maximal observed proxy score also maximizes the full-pool proxy total. The calibration criterion averages this indicator over query steps, two paths per panel and panels.

Canonicalizing binary label orientation and deduplicating candidate response profiles yields 376 calibration panels with at least five distinct profiles. Panel inclusion uses response structure, without requiring gold correctness or a unique full-pool winner. Each calibration path queries at most 256 informative items. The grid is $\{0.1, 0.2, \ldots, 1.0, 2, 3, 4, 5\}$, and the smallest temperature with maximal average proxy accuracy is selected. This procedure gives $\tau = 0.7$ for the primary comparison (Table 25).

For organization-disjoint calibration, a seeded permutation partitions the same 376 organizations into five folds. Each evaluation organization receives the temperature selected using only the other four folds. The grid, proxy criterion and path budgets are unchanged. Section E.13 reports the selected temperatures and acquisition outcomes.

C.6 Executable Survivor-Pair Acquisition

This section gives the pseudocode corresponding to the query steps in Section 5. The procedure maintains the same pairwise margins and possible-winner set.

Algorithm 2 combines this query rule with the certificate state. For a pool with a unique full-pool winner, repeated querying terminates with that winner and its completion certificate. Appendix C.4 derives the pair count and gives state-update costs. The query therefore targets comparisons still relevant to the next elimination.

Algorithm 2 FRONTIERCUT with exact stopping

Require: Cached predictions P, item weights w and a label oracle. The full pool has a unique winner

Define the informative query pool $\mathcal{I} = \{i : \exists a, b,\ p_{ia} \neq p_{ib}\}$
Set $t = 0$ and $\mathcal{O} = \varnothing$. Set $\mathcal{U} = \mathcal{Q}$
Initialize $d_{ab} = 0$ and r_{ab} by Equation (2). Set $A_0 = \mathcal{A}$
while $|A_t| > 1$ **do**
    Compute $q_t(i)$ for each remaining query item $i \in \mathcal{I} \cap \mathcal{U}$ using Equation (5)
    Select an item i maximizing $q_t(i)$ using a seeded random tie stream
    Acquire y_i and move i from $\mathcal{U}$ to $\mathcal{O}$
    Update d_{ab} and r_{ab} using Equation (2)
    Apply Corollary 4 to obtain A_{t+1} and increment t
end while
return The unique candidate in A_t and its completion certificate

C.7 FROM POSSIBLE WINNERS TO QUERY CHOICE

Building on this characterization, we direct acquisition toward disagreements among candidates still able to win (Section 5). To identify these candidates, we assign each candidate's own predictions to the unlabeled items and check whether it can match or exceed every rival's score. This favorable assignment simultaneously maximizes its margins against all rivals. Using the resulting candidate set, FRONTIERCUT queries the item that separates the most remaining pairs. Similarly, CERTSELECT restricts and renormalizes SELECT-style candidate weights over this set before applying the same query objective. Both rules therefore use certificate information to decide which label to acquire next.

D FULL EXPERIMENTAL DESIGN

This appendix supports Section 6 with the detailed material below.

D.1 QUESTIONS

This section describes the experimental questions in support of Section 6.

The experiments address five questions.

1. Does limited uniform labeling leave the discrete winner or deployment map unstable on real workloads?
2. Does the exact certificate agree with exhaustive completion enumeration?
3. Under the same exact stopper, does certificate-aware acquisition use fewer labels than structural and classical baselines as well as recent strong baselines?
4. Do the effects persist across task families and label alphabets? Do they also persist across candidate counts, winner gaps and informative-item ratios?
5. How large is the intrinsic evidence burden of certifying a full deployment cost map, and how close are constructive certificates to fixed-map lower bounds?

D.2 LLM PAIN MATRICES

This section describes the cached LLM correctness matrices in support of Section 6.

The zero-method study uses seven cached LLM correctness matrices. These include IFEval, Open-LLM Math and MMLU-Pro. The other matrices are ARC-Challenge and BBH alongside GPQA and MuSR. The Open LLM Leaderboard provides part of the surrounding evaluation context (Myrzakhan et al., 2024). For each matrix and label budget in $\{25, 50, 100, 200\}$, we draw 200 uniform samples. Metrics include exact-winner recovery and selected-winner regret. They also include top-five recovery and overlap where applicable. We additionally measure mean absolute score error and Kendall rank correlation.

Table 6: Experimental workloads and primary units. A panel-level unit prevents panels with more replicates from receiving extra inferential weight. Organization-selection panels retain 4,083 response profiles from 6,161 source deployment candidates after within-panel deduplication and unique-winner filtering. Candidate counts in the table are per panel. Runs count replicates within a method, or deterministic instances for cost-map bounds.

Workload	Panels or tasks	Items	Candidates	Runs	Primary unit	Primary target
LLM pain matrices	7	task-dependent	task-dependent	5,600	task and seed	Uniform-budget instability
LLM structural selection	353	up to 1,267	5–74	2,824	organization panel	Exact winner, structural controls
LLM strong baselines	353	up to 1,267	5–74	706	organization panel	Exact winner, recent selectors
Cross-task predictions	19	71–50,000	8–114	152	dataset panel	Binary and multiclass exact winner
JudgeTuning pain	20 caps	3,548	397	5,000 per budget	trial	Joint cap decisions
JudgeTuning cost map	331 regions	3,548	397	2 tolerances	full map	Certificate burden
WinoGrande proxy cost	135	up to 1,267	4–71	270	organization panel	External cost-map burden

All seven tasks show both decision instability and a measurable separation between aggregate estimation quality and winner correctness. The complete study contains 5,600 budget–seed records. Figure 3 reports the primary winner metrics.

D.3 Organization-Level LLM Panels

This section describes organization-level llm panels in support of Section 6.

We build the main workload from the processed Open LLM Leaderboard WinoGrande correctness matrix released by SparseEval (Zhang et al., 2026), based on WinoGrande (Sakaguchi et al., 2020). Candidate models are grouped by organization. Duplicate response profiles are removed within each panel. We retain 353 panels with a unique full-pool winner. The 6,161 deployment candidates with organization and model identifiers supply these panels. The retained panels hold 4,083 deduplicated response profiles in aggregate and use up to 1,267 items.

The structural comparison uses eight matched replicates per panel, for 2,824 runs. The stronger classical comparison and SELECT comparison use two replicates per panel, for 706 runs each. All methods within a replicate receive matched random seeds and the identical full-pool prediction matrix. Gold labels are revealed only in the order requested by the acquisition rule. The full-pool winner is computed from the complete gold label vector only for final correctness evaluation.

For SELECT temperature calibration, 376 response-only structural panels are formed before the unique-winner filter. No gold scores enter cohort selection or the calibration objective. The 353 evaluation panels are the primary inference units. The two replicates within a panel are first aggregated by a median paired saving.

D.4 Cross-Task Raw-Prediction Panels

This section describes cross-task raw-prediction panels in support of Section 6.

To test finite-alphabet breadth, we use 19 public raw-prediction panels with eight matched replicates each. Eleven panels are multiclass. Class counts include 2, 3 and 4. Other panels have 6 or 7 classes as well as 10 or 1,000 classes. The collection includes CIFAR-10 (Krizhevsky, 2009) and GLUE tasks (Wang et al., 2019). It also includes ImageNet (Deng et al., 2009), ImageNetV2 (Recht et al., 2019) and PACS (Li et al., 2017). Pool sizes range from 71 to 50,000 items, and candidate counts range from eight to 114.

The methods include quotient uniform and static QBC. They also include static all-pair cut and FrontierCut. Every method uses the same exact multiclass certificate. The primary effect is saving against static all-pair cut because that baseline already absorbs response disagreement without adaptive frontier updates.

D.5 JUDGETUNING COST WORKLOAD

This section describes judgetuning cost workload in support of Section 6.

JudgeTuning provides 397 judge configurations, 3,548 instructions, and measured configuration costs (Salinas et al., 2025). The pain study uses 20 caps spanning 21 to 397 feasible configurations. These caps have six distinct full-pool winners. The budgets are 50 and 100 alongside 200 and 400. At each budget we repeat uniform acquisition 5,000 times and evaluate the entire 20-cap decision vector.

The full certificate study expands the cost ordering to 331 distinct feasible regions. The any-output lower bound selects a candidate minimizing the regional bound at each cap. These choices use 16 distinct candidates at a two-point tolerance and 17 at one point. At tolerances of one and two percentage points, we compute a simple any-output lower bound, the fixed canonical-map linear relaxation and a greedy integral fixed-map certificate. Fixed-map certificates are verified directly against every region–challenger demand.

The 20-cap pain vector and the 331-region certificate map answer different questions. The former shows that a seemingly substantial uniform label budget can leave at least one deployment choice wrong. The latter measures how much evidence is required to make an entire fine-grained map completion-safe.

D.6 WINOGRANDE PROXY-COST MAPS

This section describes winogrande proxy-cost maps in support of Section 6.

For external cost-map structure, we use 135 WinoGrande organization panels. Declared parameter count in billions defines the candidate cost ordering. This proxy orders model configurations by declared size, without estimating hardware-specific deployment costs. Of the deployment candidates, 4,258 identifiers have a uniquely parseable parameter count. Organization eligibility filtering and within-panel response deduplication yield 1,773 response profiles across the 135 proxy-cost panels.

Panels contain four to 71 models, with median nine. They contain three to 15 cost regions, with median four. Across panels, 83.70% have multiple canonical region winners. We compute the same any-output floor, fixed-map LP, and greedy fixed-map certificate at one- and two-point tolerances. All 270 constructed certificates are checked against their derived constraints.

D.7 BASELINES AND FAIR STOPPING

This section describes baselines and fair stopping in support of Section 6.

Table 7 describes the acquisition controls. Full uniform is intentionally weak and appears only to quantify the cost of uninformative columns. Quotient uniform is the meaningful random-order baseline. Static pair cut is the closest nonadaptive structural control for FRONTIERCUT. The Model Selector oracle grid is favorable to the baseline but nondeployable, so we also run the deployable form whose grid value comes from majority-vote pseudo-labels on the cached predictions. SELECT all-candidate with structurally calibrated $\tau = 0.7$ is the main recent external acquisition baseline.

Fair stopping is central to the design. A baseline is never forced to use its original approximate stopping threshold in the primary label-count comparison. It receives the exact completion condition. Thus every reported saving is relative to the number of labels the acquisition order needs to reach the same completion-safe boundary.

D.8 STATISTICS

This section describes statistics in support of Section 6.

For the main comparisons, the organization panel is the primary inference unit. We calculate the two paired relative savings within each panel and take their median. The reported point estimate is the median across the 353 panel values. A percentile bootstrap resamples panels 10,000 times with a fixed seed (Davison & Hinkley, 1997). We report the 2.5th and 97.5th percentiles. Positive, zero, and negative panel counts supplement the median.

Table 7: Acquisition baselines. Every row in the exact-selection experiments uses the same completion stopper.

Method	Constant items	Survivor restriction	Query objective and setting
Full uniform	Retained	No	Random order
Quotient uniform	Removed	No	Random order
Static QBC	Uninformative	No	Fixed response entropy
Static all-pair cut	Zero score	No	Fixed pair-separation count
Active Comparison	Uninformative	No	Updated comparison score, fixed implementation
Model Selector	Uninformative	No	Updated hypothetical-label entropy, pseudo-label grid selection (deployable) or best completed grid run (oracle)
SELECT	Uninformative	No	Updated acquisition-mass split, response-only temperature
FRONTIERCUT	Removed	Yes	Updated survivor pair-separation count
CERTSELECT	Uninformative	Yes	Updated SELECT mass split on survivors, shared temperature protocol

The cross-task collection has only eight repeats per dataset and strongly varying pool sizes. We therefore report run-level medians and one median per dataset, plus the fraction of datasets with positive median saving. Stratified robustness uses four quartiles for candidate count, full-pool winner gap, and informative-item ratio. For each of the 12 strata we report the median saving, its bootstrap interval and the replicate-stratum medians.

Cost-map results are descriptive optimization quantities on fixed finite instances. LP objectives are deterministic lower bounds for their fixed maps. Greedy-to-LP ratios are approximation-gap diagnostics, not sampling confidence intervals.

D.9 REPRODUCIBILITY AND INTEGRITY CHECKS

This section describes reproducibility and integrity checks in support of Section 6.

Every experiment includes a settings file with parameters and random seeds, together with its result JSON file. Appendix Q lists the result files and seeds. Independent 3090 and 4090 executions of the final P8B run produce the same deterministic result core. Their wall-clock times were 1,292.29 and 836.04 seconds. Seventy-five local tests satisfied their expected conditions with one process and one numerical thread.

Every cost-map certificate is verified after construction against its derived constraints. Figure files in this manuscript are generated directly from the result JSON files by a single-process script. No plotted value is transcribed from an image.

D.10 ACQUISITION BASELINE DESIGN AND AGGREGATION

This section supports the baseline comparisons and statistical summaries in Sections 6 and 6.3. It describes the query objectives and how matched runs contribute to panel-level effects.

We include CIFAR-10 (Krizhevsky, 2009) and GLUE (Wang et al., 2019) to cover image and language classification. We also use ImageNet (Deng et al., 2009) and ImageNetV2 (Recht et al., 2019) for larger label alphabets, alongside PACS (Li et al., 2017) for domain variation. This section lists the datasets and candidate counts and Appendix K reports uniform-sampling comparisons. The collection tests acquisition with natural categorical predictions.

All single-region acquisition rules stop at the strict condition $d_{ab} > r_{ab}$ against every rival. During acquisition we retain possible winners satisfying $d_{ab} + r_{ab} \geq 0$ against every rival, including candidates that can tie. This common stopping condition makes the comparison measure how efficiently each query rule collects decisive evidence.

Weighting the survivors. To assess the value of cumulative correctness within the survivor set, we compare CERTSELECT directly with FRONTIERCUT. The CERTSELECT versus FRONTIERCUT row of Table 1 gives a positive median saving for CERTSELECT, while FRONTIERCUT uses fewer labels on some panels. The rules optimize different splits: FRONTIERCUT balances survivor counts, whereas CERTSELECT balances masses derived from cumulative correct-item counts. Their query rankings coincide when survivor masses are uniform in the binary setting, as shown in Section 5. The full method summaries and oracle-temperature comparisons appear in Appendix L. This comparison supports combining exact elimination with graded evidence when ranking the remaining queries.

These experiments connect the acquisition claim in Section 1 to the common stopping condition of Theorem 3. Section 6.1 describes the shared evaluation setup. Section 6.3 compares acquisition rules before examining the contribution of adaptive queries and their behavior across task families. Section 6.5 measures fixed-map certificate size. Appendix E.10 examines joint decisions under uniform labeling.

This section lists all datasets and candidate counts. Additional uniform-sampling results on cached task matrices (Myrzakhan et al., 2024) appear in Appendix K to characterize the selection problem under a limited label budget.

Structural baselines. Full uniform (Kossen et al., 2021) samples all items and measures the cost of querying items on which all candidates agree. Quotient uniform removes those items before random ordering, isolating the benefit of informative-item selection. Static all-pair cut ranks items once by pairwise disagreement and isolates the effect of updating the survivor set in FRONTIERCUT. Static query-by-committee (QBC) ranks by response entropy (Seung et al., 1992), giving a complementary disagreement-based comparison.

Evidence-based baselines. We include Active Comparison (Sawade et al., 2012) for its direct focus on performance differences. We adapt Model Selector (Okanovic et al., 2025) to the shared-label matrices to compare against information-based model selection. Its oracle grid (Okanovic et al., 2025) reports the smallest label count among the evaluated hyperparameter settings for each run. This oracle is a favorable performance reference because choosing its setting uses the completed acquisition outcomes. We therefore also evaluate a deployable form of the same selector. Its query ranking is the hypothetical-label entropy of the original method and its grid value maximizes the identification rate along pseudo-label paths. The pseudo-labels are majority votes of the candidate responses, so the choice reads no gold label. The two forms search the same grid and stop at the same exact certificate.

SELECT (Durmazkeser et al., 2026) is the main recent acquisition baseline because it incorporates cumulative correctness when balancing candidate responses. Both SELECT and CERTSELECT use $\tau = 0.7$ and the cumulative correct-item count $E_a^{(t)}$ defined in Section 5. For temperature selection we replace gold correctness with average agreement with all candidate responses. The chosen temperature maximizes path accuracy averaged across calibration panels, as defined in Section 5. Appendix L gives the calibration table and oracle-grid SELECT (Durmazkeser et al., 2026) variants, which serve as favorable references using the same outcome-based choice of temperature.

Matched randomness allows paired comparisons while the repeated queries capture variation in item ordering.

We test whether using the possible-winner set from Corollary 4 improves acquisition under the shared strict certificate. Section 6.3 gives the main comparison with SELECT (Durmazkeser et al., 2026). Appendix E.7.1 separates structural acquisition gains from constant-item removal and compares classical selectors. Section 6.4 and Appendix E.9.1 examine task breadth and variation across organization panels.

E EXTENDED RESULTS

This appendix supports Sections 6.3 and 6.5 with the detailed material below.

E.1 STRUCTURAL ACQUISITION GAIN

This section describes structural acquisition gain in support of Section 6.3.

Table 8: Structural comparison on 353 LLM organization panels with eight replicates each. Label summaries are over 2,824 matched runs.

Method	Q25	Median	Q75	Correct
Full uniform	1054.75	1159.0	1223.25	2824
Quotient uniform	362.00	571.5	783.00	2824
FRONTIERCUT	290.00	**418.0**	547.00	2824

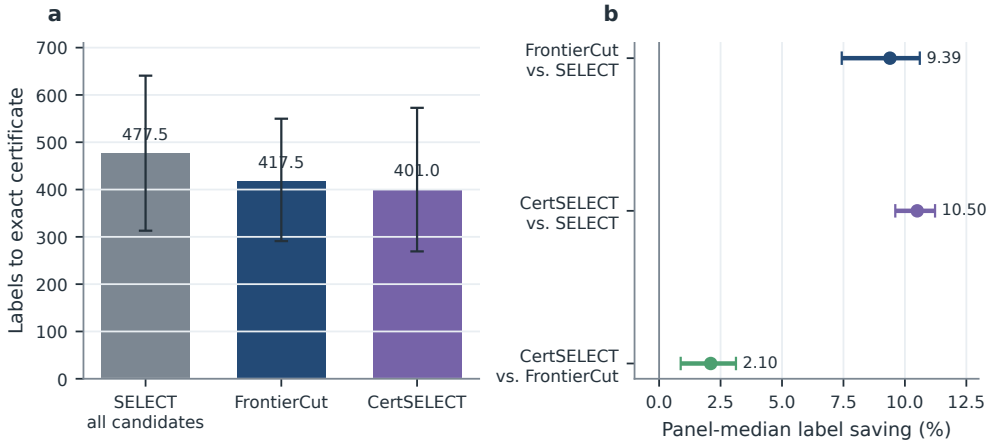

Figure 5: Strong SELECT acquisition comparison on 353 organization panels and 706 matched runs. (a) Median labels with interquartile ranges. All methods use the identical exact stopper and all 706 outputs are correct. (b) Panel-median paired relative savings with 95% panel-bootstrap intervals. Each panel first aggregates its two paired savings.

Table 8 gives the quartiles behind the structural comparison of Section 6.3. The median number of current candidate edges that FRONTIERCUT removes per acquired label is 11.39.

Table 8 separates removing constant items from using an adaptive query ranking. The latter comparison changes the order as well as restricting attention to survivors. The direct CERTSELECT–SELECT comparison in Table 1 isolates survivor restriction within a shared query objective.

E.2 SELECT ACQUISITION COMPARISON

This section describes select acquisition comparison in support of Section 6.3.

Table 1 reports the matched run signs and panel effects. The two survivor-based rules use different information: FRONTIERCUT balances survivor counts, while CERTSELECT balances acquisition masses computed from cumulative correctness. The latter preserves graded evidence among candidates that remain logically possible, complementing exact elimination.

E.3 BINARY AND MULTICLASS BREADTH

This section describes binary and multiclass breadth in support of Section 6.3.

Table 9 records absolute label counts, while Figure 6 gives one paired median per dataset. Each dataset contributes eight replicates, so dataset aggregation also avoids letting absolute pool size dominate the comparison. Table 2 reports the principal paired effects, and Table 19 checks the multiclass zero-difference case handled by Lemma 2.

E.4 STRATIFIED ROBUSTNESS

This section describes stratified robustness in support of Section 6.3.

Figure 7 shows panel-median saving against quotient uniform for all 12 strata.

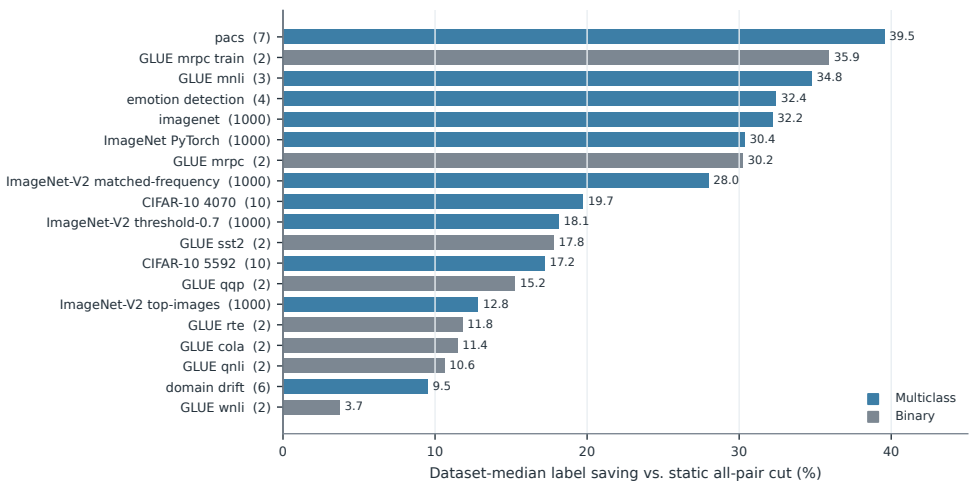

Figure 6: Dataset-median saving of FRONTIERCUT relative to static all-pair cut on 19 natural raw-prediction panels. Parentheses give label-alphabet size. All panels are positive, including 11 multiclass datasets and four ImageNet-family panels with 1,000 classes.

Table 9: Cross-task label counts over 152 matched runs.

Method	Q25	Median	Q75	Correct
Quotient uniform	884	5045.0	9615.5	152
Static QBC	714	3505.0	8784.3	152
Static pair cut	732	3501.5	8744.0	152
FRONTIERCUT	597	**2835.5**	5713.0	152

Figure 7 shows increasing savings across candidate-count quartiles and positive effects across winner-gap and informative-item strata. These associations are consistent with survivor pruning redirecting queries away from eliminated alternatives. Since panel characteristics vary together, the strata establish breadth across observed conditions rather than the causal effect of changing one characteristic.

E.5 JUDGETUNING PAIN AND COST-MAP DETAIL

This section describes judgetuning pain and cost-map detail in support of Section 6.5.

Figure 8 reports the 20-cap JudgeTuning pain experiment, and Figure 9 the certificate burden of the full map.

E.6 EXTERNAL PROXY-COST PANEL STRUCTURE

This section describes external proxy-cost panel structure in support of Section 6.5.

Table 18 separates source coverage, panel composition and deployment-map structure. The selected configurations are ordered by declared parameter count. Table 10 then measures the fraction of gold labels needed to certify each resulting map.

These results establish the prevalence of multiple regional choices under a reproducible model-size ordering. The label fractions quantify evidence requirements within each fixed pool, while measured latency, memory and monetary cost would require separate deployment measurements.

E.7 STRUCTURAL AND CLASSICAL ACQUISITION COMPARISONS

This section supplements the acquisition comparisons in Section 6.3 with label-count quartiles under the shared exact stopping condition.

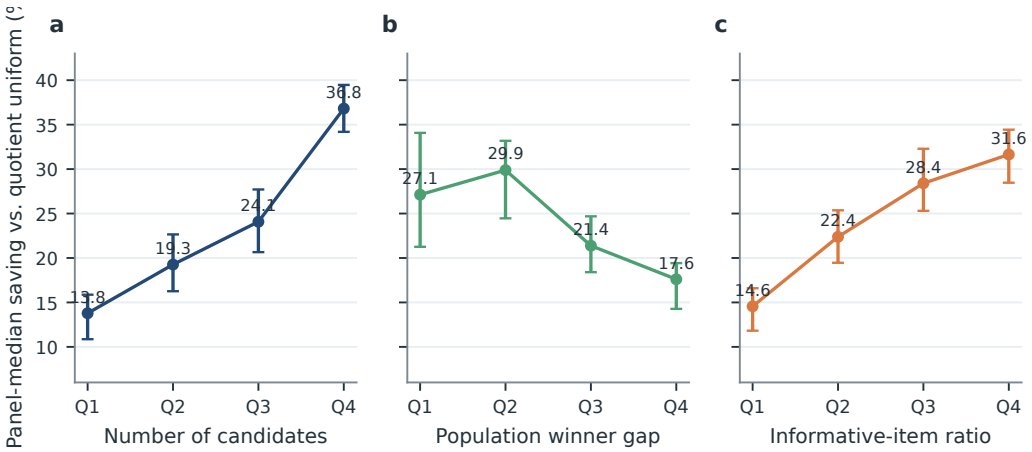

Figure 7: Stratified panel-median saving of FRONTIERCUT against the quotient uniform baseline. Points show stratum medians and bars show 95% panel-bootstrap intervals. All 12 strata have positive lower endpoints. Quartiles are formed separately for candidate count, population winner gap, and informative-item ratio.

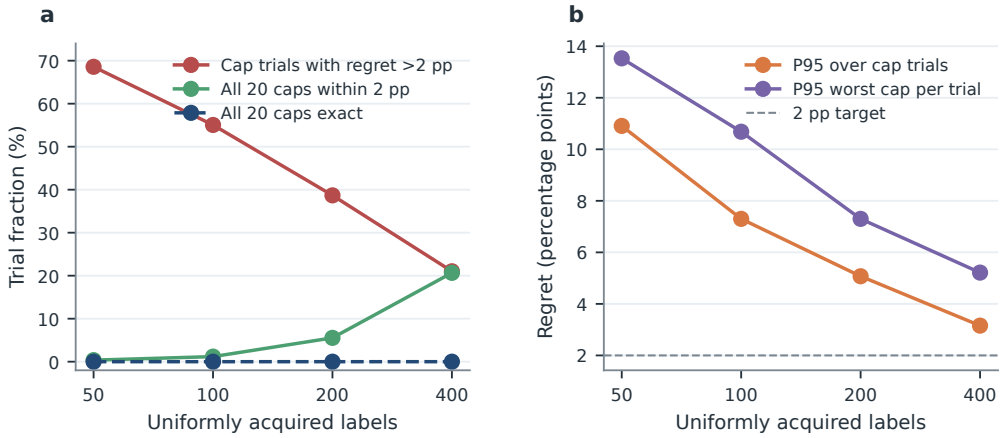

Figure 8: JudgeTuning 20-cap pain measurement over 5,000 uniform-label trials per budget. (a) A large share of individual cap trials exceeds two percentage points of regret, while exact recovery of all caps remains zero. (b) The 95th percentiles of pooled individual-cap regret and of the maximum cap regret per trial both decline with budget. At 400 labels they are 3.16 and 5.21 percentage points, respectively.

E.7.1 STRUCTURAL AND CLASSICAL COMPARISONS

This comparison supports the structural and classical baseline conclusions in Section 6.3.

Informative-item selection. To separate adaptive acquisition from removing constant items, we compare FRONTIERCUT with both uniform baselines. Table 8 shows lower median label use for quotient uniform than full uniform and a further reduction for FRONTIERCUT. All structural-comparison outputs satisfy the certificate and match the unique full-pool winner. Agreement items cancel from every pairwise margin, which explains the reduction obtained by removing them. The additional reduction follows a query rule that recomputes disagreement among survivors after each label. Hence the structural gain extends beyond avoiding items that cannot change the decision.

Comparison-based acquisition. To test against selectors that already prioritize informative comparisons, we evaluate the classical baselines with the same strict stopper. Figure 2 gives the median counts and Table 1 the paired savings, while Table 11 adds the label-count quartiles. The oracle Model Selector (Okanovic et al., 2025) row benefits from selecting its best evaluated setting for every run. Even against that reference, the paired comparison favors FRONTIERCUT. Unlike a static disagreement ranking, FRONTIERCUT directs later queries toward comparisons that remain unresolved

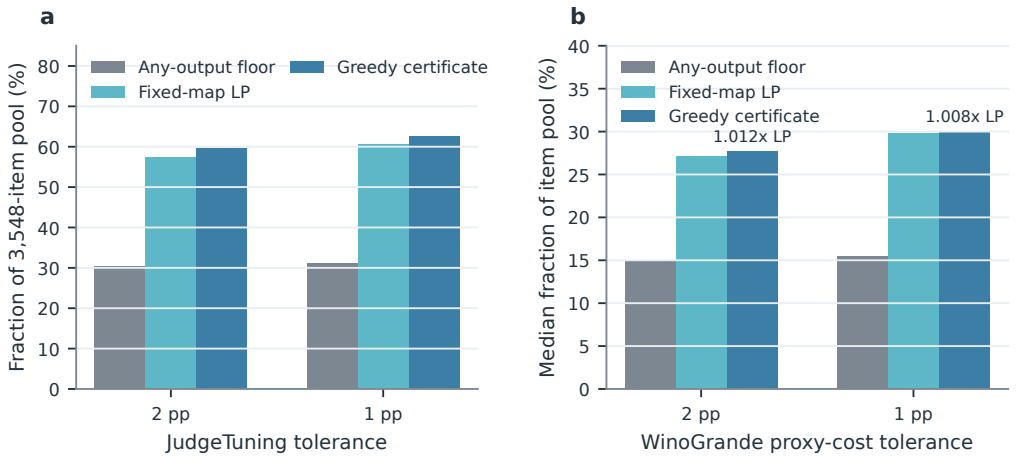

Figure 9: Cost-map certificate burden. (a) JudgeTuning full-map quantities as fractions of the 3,548-item pool. The fixed-map LP and greedy certificate concern the canonical map. The any-output floor is universal. (b) Median fractions across 135 WinoGrande organization panels using parameter count only as a proxy cost.

Table 10: WinoGrande proxy-cost panel summaries. Fractions are medians across 135 organization panels. Parameter count is a declared cost proxy only.

Tolerance	Any-output floor	Fixed-map LP	Greedy certificate	Greedy/LP	Floor $\geq 5\%$	Verified
2 pp	15.00%	27.15%	27.70%	1.012	99.26%	135/135
1 pp	15.47%	29.76%	29.99%	1.008	99.26%	135/135

after elimination. These results support survivor-based acquisition alongside the improvement over evidence-weighted SELECT.

Deployable Model Selector. The oracle grid chooses its setting after the label counts are known, so we also compare against Model Selector with a grid value taken from majority-vote pseudo-labels. Against that baseline FRONTIERCUT saves a panel-median paired 26.66% with interval $[24.52, 29.21]\%$ and CERTSELECT saves 30.19% with interval $[28.15, 31.70]\%$. The savings are positive on 342 and 346 of the 353 panels and negative on 11 and 7. The CERTSELECT comparison uses the organization-disjoint temperature of Appendix C.5. Both rules therefore keep a margin over an external selector that a practitioner could run without gold labels for tuning.

E.8 HINDSIGHT CERTIFICATE BOUNDS

This section supports the hindsight comparison in Section 6.3. For each of the 353 organization panels we solve the single-region covering program of Theorem 7 under full labels. The demands are the strict $\eta_{ab} = D_{ab} + 1$ against every rival b of the full-pool winner a, and the coefficients are $C_{ab,j} = 2$ on items where a is correct and b is wrong. Relaxing the item variables to $x_j \in [0, 1]$ gives a lower bound on the smallest certificate for that panel, and the greedy cover over the same demands gives a feasible upper bound. Neither value is an asserted integer optimum, so the ratios below are distances to a bracket rather than approximation guarantees. All 353 programs reach optimality with maximum primal violation 5.7×10^{-14} and maximum primal-dual gap 1.7×10^{-13}. The panel-median ratio of greedy labels to the LP bound is 1.0184 with a 95% interval of [1.0162, 1.0214].

Table 12 reports for each acquisition rule the median over panels of the within-panel median ratio of labels used to each hindsight reference. The runs are the same 706 runs as in Table 1 and the intervals are 10,000-resample panel bootstraps.

The ordering matches the label counts of Figure 2, and both survivor-restricted rules sit closer to the hindsight bounds than all-candidate SELECT. A ratio above 1.6 remains for every deployable rule at the calibrated temperature. An adaptive path that reveals labels one at a time therefore does not reach what full knowledge of the labels would allow, which bounds how much any acquisition order alone

Table 11: Label-count quartiles for the classical baselines of Figure 2. All methods use the common exact stopper on the same 706 matched runs.

Method	Q25	Q75
Static QBC (Seung et al., 1992)	347.00	703.00
Active Comparison (Sawade et al., 2012)	349.75	732.75
Model Selector oracle (Okanovic et al., 2025)	324.25	772.00
FRONTIERCUT	291.00	549.75

Table 12: Labels used relative to hindsight certificate bounds on the 353 organization panels. Smaller values lie closer to the hindsight reference. The LP column divides by a lower bound on the smallest certificate and the greedy column divides by a feasible cover. The oracle-grid rows take the smallest label count over completed hyperparameter-grid runs and are not deployable settings.

Method	Labels / LP bound	Labels / greedy cover
FRONTIERCUT	1.661 [1.627, 1.697]	1.612 [1.591, 1.643]
CERTSELECT	1.625 [1.613, 1.641]	1.578 [1.566, 1.591]
SELECT (Durmazkeser et al., 2026)	1.855 [1.840, 1.882]	1.810 [1.792, 1.839]
CERTSELECT oracle grid	1.500 [1.481, 1.515]	1.450 [1.424, 1.469]
SELECT oracle grid	1.765 [1.742, 1.784]	1.719 [1.695, 1.735]

can still save. The two oracle-grid rows use tuning information unavailable at acquisition time and are reported only as a reference point.

E.9 PANEL COMPOSITION AND ACQUISITION GAINS

This section supports the panel-stratum summaries in Table 14. It relates acquisition gains to candidate count and winner gap as well as the fraction of informative items.

E.9.1 GAINS ACROSS PANEL STRATA

These strata support the panel-composition conclusion in Section 6.4.

To determine how the structural gain varies with panel composition, we form quartiles separately by candidate count and full-pool winner gap. We also stratify by the fraction of items on which candidates disagree. Table 14 summarizes the smallest effects, and Table 29 in Appendix N gives every stratum. All stratum medians and their lower bootstrap endpoints favor FRONTIERCUT over quotient uniform. The gain grows across candidate-count quartiles, consistent with eliminating candidates removing more irrelevant pairwise comparisons from later queries. Positive effects across both winner-gap and informative-item strata indicate that the benefit spans the observed panel conditions. Across these comparisons, feeding the exact possible-winner set into acquisition reduces the labels needed to certify a unique choice.

E.10 REGIONAL OUTPUT CHOICES AND UNIFORM-LABEL RECOVERY

This section supports the cost-map interpretation in Section 6.5. It reports changes in regional outputs and complete uniform-label joint-recovery results.

We evaluate the introduction's claim that shared evidence can certify choices across deployment cost caps. The covering formulation in Theorem 7 lets us compare a feasible certificate with a lower bound for the same fixed map. We report aggregate bounds for JudgeTuning (Salinas et al., 2025) and WinoGrande. This section reports output-map switching and uniform-label joint recovery. The main comparison measures how tightly shared evidence bounds fixed-map certificate size.

As in Section 4, we convert accuracy tolerance to correct-item counts through $\epsilon_N = N\epsilon/100$. The cost-map target is non-strict ϵ-optimality: every regional choice satisfies $r_{ab} - d_{ab} \leq \epsilon_N$ against each feasible rival. A map may therefore retain tied choices within tolerance.

The increase follows the larger pairwise evidence demands induced by reducing ϵ_N in Equation (4). Each selected item can support several regional comparisons, and the cover accounts for that sharing while satisfying every demand.

Table 13: **Joint map recovery remains uncertain under uniform labeling.** Results use 20 JudgeTuning caps and 5,000 trials per budget. All exact is the fraction recovering every full-pool decision. Within 2 pp is the fraction whose maximum cap regret is within tolerance. Cap > 2 pp is the fraction of individual-cap trials exceeding tolerance. P95 denotes the 95th percentile of regret in percentage points. P95 cap pools individual-cap regrets, whereas P95 max uses the largest cap regret per trial. Higher recovery and lower regret are better.

Budget	All exact ↑	Within 2 pp ↑	Cap $>$ 2 pp ↓	P95 cap ↓	P95 max ↓
50	0.00%	0.36%	68.58%	10.91	13.53
100	0.00%	1.18%	55.05%	7.30	10.68
200	0.00%	5.54%	38.68%	5.07	7.30
400	0.00%	20.66%	21.02%	3.16	5.21

The shared item variables reuse each label across these comparisons, as specified by Theorem 7.

Section 6.5 measures this comparison on JudgeTuning (Salinas et al., 2025) and examines the effect of changing regional outputs. Section 6.5 tests the same quantities across organization panels. Section 6.6 then examines why joint certification matters when labels are acquired uniformly.

Output choices. To assess how much the specified outputs contribute to the evidence burden, we allow regional choices to vary within tolerance. The canonical map fixes one output per region. Table 17 reports the fixed-map greedy size, the feasible size found by local output switching, and both any-output lower bounds. The switched maps satisfy the regional constraints at each tolerance. Their smaller certificate sizes quantify the benefit found by this local search. The remaining intervals measure uncertainty about the global any-output optimum.

Any-output relaxation. We also relax the any-output program (25) over all admissible maps, with 331 regions and 3,548 item variables and the demands $\eta_{ab} = \max\{0, \lceil D_{ab} - \epsilon_N \rceil\}$. Solving the projected formulation by constraint generation gives a verified dual bound at both tolerances. Table 17 uses the larger of the regional floor and the any-output relaxation as the valid lower bound. The relaxation improves the bound at one point and is weaker than the floor at two points. A fractional output variable can spread a region's choice over several candidates, weakening the relaxation when regional demands are small.

E.10.1 Joint Decisions Under Uniform Labeling

This experiment supports the joint-recovery comparison in Section 6.5.

To relate the certificate burden to budgeted labeling, we evaluate whether uniform samples (Kossen et al., 2021) recover every choice in a deployment map. This study uses 20 JudgeTuning caps with six distinct full-pool winners. We use 5,000 uniform-label trials per budget to measure variation in both individual and joint decisions. For each cap, regret is the full-pool accuracy gap between the selected candidate and the best feasible candidate. Smaller regret is better, while larger exact-recovery and within-tolerance fractions indicate more reliable selection. The full cost ordering used above serves the certificate-size analysis, whereas this cap set measures joint decision recovery under sampled labels.

Table 13 shows that increasing the uniform budget improves both individual-cap and joint within-tolerance selection. Even at the largest reported budget, no trial recovers every exact decision and only a minority keeps all caps within tolerance. The individual-cap regret rate also remains positive at that budget. This difference follows the joint target: acceptable decisions at some caps cannot compensate for an incorrect choice at another cap. A shared-label certificate directly addresses this requirement by bounding every regional comparison simultaneously. The experiment therefore supports certifying the complete deployment decision when the available cost cap can vary.

E.11 Aggregate Task and Stratum Comparisons

This section supports the cross-task and panel-composition conclusions in Section 6.3. It collects the aggregate effects with their comparison units.

Table 1 reports the SELECT comparison and Table 14 reports aggregate task and stratum effects.

Table 14: **Acquisition gains extend across tasks and panel strata.** Cross-task statistics use 19 datasets with eight matched replicates each. The named static method is the reference for each saving. Organization-panel strata compare FRONTIERCUT with quotient uniform. Higher values favor FRONTIERCUT. The lower endpoint is from a 95% panel-bootstrap interval. Each row states its aggregation unit to distinguish run medians from dataset and stratum summaries.

Collection	Statistic and reference	Value ↑
Cross-task	Run-median saving vs. static QBC	19.50%
	Run-median saving vs. static all-pair cut	18.18%
	Median of dataset savings vs. static all-pair cut	18.12%
	Minimum dataset saving vs. static all-pair cut	3.71%
Organization strata	Minimum stratum-median saving vs. quotient uniform	13.78%
	Minimum lower interval endpoint	10.86%
	Minimum fraction of positive panels	96.59%

Table 15: Organization-disjoint temperature sensitivity on the same 353 panels and 706 runs. Intervals use 10,000 panel-bootstrap resamples. Calibration uses response-only structural objectives in the other four folds.

Comparison	Saving	95% interval	Panel $+/0/-$
FRONTIERCUT vs. SELECT	9.41%	[7.64, 10.94]	300/0/53
CERTSELECT vs. SELECT	10.53%	[9.62, 11.52]	343/4/6

Table 14 reports positive run-level savings against both rankings and positive dataset-median savings against static all-pair cut on every dataset.

E.12 READING THE MAIN RESULT TABLES

This section supports the aggregation and metric conventions of Tables 1 and 3. The descriptions specify every sign count and comparison unit.

E.12.1 SELECT COMPARISON CONVENTIONS

Run-level label medians in Figure 2 summarize absolute counts across matched runs whereas the panel-level effects of Table 1 aggregate paired relative savings within each panel before comparing organizations, allowing the same experiment to describe annotation burden and the typical paired improvement separately. A difference between marginal count medians therefore need not equal the reported median paired saving.

E.12.2 COST-MAP COMPARISON CONVENTIONS

The greedy-to-LP ratio compares quantities from the same JudgeTuning instance, whereas for WinoGrande we compute the ratio within each panel before taking its median rather than dividing marginal medians that can reflect different panels in the collection.

E.13 ORGANIZATION-DISJOINT TEMPERATURE CALIBRATION

This comparison tests whether sharing organizations between structural calibration and evaluation explains the label savings. The five-fold construction is specified in Appendix C.5. Four folds select $\tau = 0.7$ and the fifth selects $\tau = 1.0$. Table 15 compares the held-out panel effects with the full-cohort calibration in Table 1. The similar estimates support stability to the tested organization-disjoint calibration protocol.

E.14 ADDITIONAL COMPARISON AND DIAGNOSTIC TABLES

F EXTENDED DISCUSSION, LIMITATIONS AND IMPACT

This appendix expands the interpretation and limitations summarized in Section 7.

Table 16: FRONTIERCUT paired savings against classical acquisition controls. Each effect is the median of within-panel median savings on 353 panels with two replicates. The oracle row uses the best completed hyperparameter-grid run and is a hindsight reference. Deployable Model Selector effects appear in Table 1.

Information protocol	Baseline	Saving
Fixed rule	Active Comparison	19.55%
Fixed rule	Static QBC	17.01%
Hindsight grid reference	Model Selector oracle	15.86%

Table 17: JudgeTuning output switching and any-output bounds from complete labels. All sizes count gold labels, with fractional objectives for LP relaxations. The valid lower bound is the maximum of the floor and relaxed objective. The upper bound is a verified switched-map certificate. Regret and its certified bound are maximum regional accuracy shortfalls, in percentage points. Local switching does not establish a globally optimal output map.

Quantity	Tolerance 2 pp	Tolerance 1 pp
Canonical-map greedy	2111	2216
Locally switched feasible size	2029	2135
Saving from local switching	3.88%	3.66%
Any-output regional floor	1078	1100
Any-output relaxed lower bound	318.55	1539
Best available any-output bracket	[1078, 2029]	[1539, 2135]
Switched-map maximum realized regret	1.72	0.90
Switched-map certificate regret bound	1.97	0.99

F.1 PRACTICAL INTERPRETATION

This section expands the practical interpretation of the certificate in Section 3.

In plain terms, the method answers: "Have we bought enough shared labels that no possible labeling of the remaining items can change the model we would deploy?" The answer is useful when predictions are already cached and labels are expensive. A benchmark owner can stop expert annotation as soon as the winner is forced. A model marketplace can certify the best eligible model at a resource cap. An LLM evaluation team can compare prompt and judge configurations alongside decoding or routing configurations without fully labeling every instruction.

The certificate is also useful when it does not trigger early. It exposes why stopping is impossible. The unresolved pair a, b with small slack identifies a concrete challenger and the remaining items capable of reversing the decision. This information can guide annotation, explain continued cost to stakeholders, and distinguish a genuinely close race from a poorly chosen random sample.

The workflow is modular. Cached responses define reusable structure. The stopper converts observed margins and remaining disagreement into an exact decision. An acquisition rule then tries to reach that condition quickly. Changing the acquisition rule does not require re-proving validity, and changing the deployment map does not require retraining any candidate model.

F.2 RELATIONSHIP TO LARGE MODELS

This section explains the cached-prediction setting used in Section 6.

The framework is about evaluating and selecting large models, not about training one. Each LLM is a candidate row in a response or correctness matrix. The method consumes cached outputs and newly acquired gold labels. It has no trainable transformer, reward model, or policy network. It can therefore run on commodity compute once candidate inference traces exist.

No reinforcement learning from human feedback is used. Human feedback can supply gold labels in an application, but the algorithm treats those labels as fixed observations. It neither learns a reward function nor updates model parameters. This makes the method compatible with limited hardware and also narrows its claim. It improves evaluation efficiency, not model capability.

Table 18: WinoGrande proxy-cost panel structure. Parameter count orders deployment options. Counts of profiles aggregate within-panel deduplicated candidates. Ranges describe the 135 retained panels.

Quantity	Value
Source identifiers with parseable parameter count	4258
Retained panels / response profiles	135 / 1773
Models per panel: median [range]	9 [4, 71]
Cost regions per panel: median [range]	4 [3, 15]
Canonical winners per panel: median [range]	3 [1, 8]
Panels with multiple canonical winners	83.70%
Largest greedy/LP ratio at 2 pp	1.155

Table 19: Third-label diagnostic in the 11 multiclass panels. For each full-pool winner–rival pair, the fraction counts disagreement items whose gold label matches neither prediction. Pair fractions are averaged within a panel and then across panels. These answers contribute zero to the observed pairwise score difference. The interval resamples panels 10,000 times.

Quantity	Value	95% interval
Macro-average third-label fraction	17.99%	[12.82, 22.90]
Minimum panel fraction	4.97%	—
Panels with positive fraction	11/11	—

The strong SELECT comparison is valuable in this context because it tests whether ideas from modern LLM selection can be combined with an exact finite-pool certificate. The result is positive. Restricting the acquisition masses to exact survivors improves the acquisition path while the stopper preserves validity. This is a concrete interaction between a recent LLM-oriented acquisition method and certificate logic, without pretending to train a large model end to end.

F.3 Why the Exact Boundary Matters

This section interprets the exact stopping condition in Section 3.

An approximate stopping threshold can always be loosened to save labels, which makes acquisition comparisons hard to interpret. The pathwise lower bound removes this degree of freedom. Once an item order is fixed, no completion-safe method can stop earlier than Corollary 6. Thus label counts measure how quickly an order creates decisive evidence, not how much risk a method silently accepts.

Exactness is particularly meaningful for benchmark governance. A benchmark result may be published once and reused for years. A one-time erroneous winner can propagate into model cards, procurement choices, and later research. A completion certificate is independently checkable from the cached matrix, acquired labels, and reported item set.

The same exactness may be unnecessary in exploratory development. A practitioner willing to accept a small failure probability could combine certificate slack with a statistical model of unseen labels and stop earlier. Our theorem would remain useful as the deterministic endpoint and as a way to quantify the additional assumption being made.

F.4 Why Full Maps Are Expensive

This section interprets the cost-map evidence requirements measured in Section 6.5.

A cost map asks many decisions at once. Different budget regions admit different challengers and may have different winners. Although one shared label can satisfy several constraints, no single favorable comparison necessarily resolves them all. JudgeTuning makes this visible: the simple any-output floor already exceeds 30% of the pool, while a fixed-map certificate uses about 60%.

This burden should guide deployment design. If users occupy only a few cost ranges, certifying every microscopic cap may be unnecessary. Regions with identical feasible sets can be merged. A system can also predeclare a smaller menu of operating points. These choices change the decision target before labels are acquired, which can be more effective than optimizing an already over-detailed map.

F.5 LIMITATIONS

This section expands the limitations summarized in Section 7.

Finite item pool. The guarantee concerns the given item pool. It does not by itself establish generalization to future prompts or distribution shift. A new pool requires a new certificate.

Cached-prediction interface. We count gold labels and assume candidate predictions are available. When inference dominates annotation cost, a joint inference-and-label policy is needed. The present label savings should not be interpreted as total monetary savings without a workload-specific cost model.

Noisy or disputed labels. The theory treats acquired gold labels as final. Human disagreement, adjudication error, and stochastic judge outputs require either repeated labeling or a robust observation model. One direct extension is to certify relative to a set of admissible labels on disputed items, but the current experiments do not evaluate it.

Unique strict winner. Main single-region panels have a unique full-pool winner. Deterministic tie rules can be encoded, and cost maps allow tolerance, but empirical behavior under large equivalence classes deserves separate study.

Conservatism. Adversarial completion can require many labels. The cost-map results demonstrate this rather than hiding it. Statistical assumptions may substantially reduce burden when exact finite-pool safety is not required.

Calibration overlap. The response-only structural cohort does not use gold labels and does not apply a unique-winner filter. Its organizations also appear among the evaluation panels, so we repeat the assignment in organization-disjoint folds and report the resulting savings in Appendix C.5. Calibration on a separate benchmark family remains open.

Baseline scope. SELECT-LLM is compared through its acquisition module after adaptation to our target and common stopper. The original research question, data interface, and end-to-end guarantee are not identical. Our evidence supports a controlled acquisition claim only.

Cost proxies. JudgeTuning uses measured configuration costs from its workload. The WinoGrande extension uses declared parameter count, which is only a proxy. Hardware latency and energy conclusions would require direct measurement.

Cost-map construction from realized labels. The fixed-map integer cover sees realized labels and is therefore a certificate-size reference, not an online acquisition policy.

Scale of pairwise state. A direct implementation stores $O(M^2)$ pair quantities. The tested panels are tractable. However, candidate pools with millions of distinct profiles would require compressed dominance structures, challenger screening that preserves exactness or distributed updates.

F.6 BROADER IMPACT AND USE

This section discusses practical uses of the completion certificate in Section 3.

Reducing annotation can lower the barrier to comparing many models. Exact stopping can also improve transparency because a reported winner comes with a finite witness set. These benefits depend on the quality and representativeness of the pool. An exact certificate on a biased benchmark can confidently select the wrong model for a real population.

Cost-aware selection can reinforce a narrow view of deployment value if cost and accuracy are the only axes. Safety and fairness may need to enter the feasible set before certification. Privacy and domain-specific constraints may also need to enter that set. The framework can handle a filtered candidate set, but it does not decide which constraints society or an organization should impose.

Labels may encode sensitive human judgments. Caching model responses and releasing item-level certificates can expose benchmark content or annotator decisions. Practical deployments should apply access controls and publish aggregate evidence when raw labels cannot be shared.

F.7 Future Directions

This section considers extensions of the certificate and cost-map formulations in Sections 3 and 4.

One direction is a risk ladder between exact completion safety and probabilistic stopping. The exact inequality supplies the top rung. Lower rungs could replace the full remaining disagreement radius with a validated uncertainty set, while reporting precisely which assumption permits earlier stopping.

A second direction is joint model inference and shared-label acquisition. If not all candidate responses are cached, the policy must decide whether to run another model on an item or purchase the item's gold label. The shared structure then becomes a two-resource problem.

A third direction is disjunctive cost-map optimization. The gap between the simple any-output floor and fixed-map LP motivates stronger lower bounds. Our small output-switching gain warns that flexibility alone may not be enough. A useful method should first show a real workload in which the canonical map is materially more expensive than another valid map.

A fourth direction is certificate compression. The selected labeled items, pairwise witnesses, and region mapping form a verifiable object. Finding a compact explanation that preserves exactness could make model-selection decisions easier to inspect and contest.

Finally, response-only structural calibration deserves broader study. The 376-panel cohort is sufficient for the comparison reported here, but cross-benchmark calibration could test whether acquisition temperature reflects stable response geometry or one collection's organization structure.

F.8 Interpretation and Scope of Completion Certificates

This section expands the interpretation of the stopping condition in Section 3 and the experimental scope in Section 6.

In plain terms, the method answers: "Have we bought enough shared labels that no possible labeling of the remaining items can change the model we would deploy?" It is also informative when it does not trigger early: the unresolved pair with small slack names a concrete challenger and the items that could reverse the decision. An approximate threshold, by contrast, can always be loosened to save labels. The pathwise lower bound removes that freedom, so label counts measure how quickly an order creates decisive evidence rather than how much risk a method silently accepts. A practitioner accepting a small failure probability could combine certificate slack with a model of unseen labels and stop earlier, with our theorem as the deterministic endpoint.

However, several limitations bound the claims. The guarantee concerns the given item pool. It counts gold labels while assuming cached predictions. It treats acquired labels as final and assumes a unique full-pool winner on the main panels. Adversarial completion is conservative by construction, and the cost-map results of Section 6.5 show how large the resulting certificates can be. The response-only calibration cohort uses no gold labels, and an organization-disjoint fold assignment gives the savings reported in Appendix C.5. SELECT-LLM is compared only through its acquisition module. This section expands each point, adds the $O(M^2)$ pairwise-state and cost-model caveats, and discusses broader impact. The WinoGrande extension uses declared parameter count as a proxy rather than a latency or energy measurement.

G Notation and Scope

This supplement expands the deterministic finite-pool results and provides complete experimental tables. It follows the same scope as the main paper. Candidate predictions are fixed, one queried gold label scores every candidate, and the guarantee concerns the fixed item pool. The main single-region theorem permits finite label alphabets and preset nonnegative weights. The cost-map integer-residual results use unit weights.

The label alphabet may contain more values than any candidate predicts on a given item. Such a value can make both candidates wrong. This case is essential for the one-unit cost-map coefficient. A binary correctness matrix is sufficient for binary tasks. General multiclass certification requires raw cached answers so that equal wrong answers can be distinguished from different wrong answers.

Table 20: Core notation.

Symbol	Meaning
$\mathcal{A}$	Finite candidate model or configuration set
$\mathcal{Q}$	Finite evaluation item pool
p_{ia}	Cached prediction of candidate a on item i
y_i	Gold label of item i
w_i	Preset nonnegative item weight
$\mathcal{O}$,$\mathcal{U}$	Observed and unobserved item sets
S_a	Full-pool weighted correctness score
d_{ab}	Observed score margin of a over b
r_{ab}	Remaining weighted disagreement of a and b
D_{ab}	Full unit-weight disagreement count
$C_{ab,j}$	Realized evidence supplied by item j
η_{ab}	Fixed-map evidence demand
A_t	Candidate survivor set after t labels

H Full Single-Region Proofs

H.1 Itemwise Pair Contribution

Fix candidates a and b. For one item i, define

$$\delta_i(y_i) = w_i\big(\mathbb{I}[p_{ia} = y_i] - \mathbb{I}[p_{ib} = y_i]\big). \tag{9}$$

If $p_{ia} = p_{ib}$, both correctness indicators are equal for every label and $\delta_i = 0$. If $p_{ia} \neq p_{ib}$, the label p_{ia} gives $\delta_i = w_i$ and the label p_{ib} gives $\delta_i = -w_i$. Every other label gives $\delta_i = 0$. Therefore

$$\max_{y_i \in \mathcal{Y}} \delta_i(y_i) = w_i, \qquad \min_{y_i \in \mathcal{Y}} \delta_i(y_i) = -w_i \tag{10}$$

on disagreement items.

Lemma 9 (Simultaneous favorable completion)**.** *For a fixed candidate a, assigning $y_i = p_{ia}$ on every $i \in \mathcal{U}$ simultaneously maximizes $S_a - S_b$ for every challenger b and gives*

$$\max_z \{S_a(y^{(z)}) - S_b(y^{(z)})\} = d_{ab} + r_{ab}. \tag{11}$$

Proof. The observed items contribute d_{ab}. On every unobserved agreement item, the contribution is zero. On every unobserved disagreement item, choosing p_{ia} attains the maximum w_i in (10). The same choice $y_i = p_{ia}$ works for every challenger because a is fixed. Summing gives $d_{ab} + r_{ab}$. No completion can exceed the itemwise maxima. □

Lemma 10 (Pairwise adverse completion)**.** *For an ordered pair (a, b),*

$$\min_z \{S_a(y^{(z)}) - S_b(y^{(z)})\} = d_{ab} - r_{ab}. \tag{12}$$

Proof. Assign $y_i = p_{ib}$ on every unobserved disagreement item. Equation (10) shows that this contributes $-w_i$ on each such item. Agreement items contribute zero. The completion reaches $d_{ab} - r_{ab}$, and no completion can be smaller than the sum of the itemwise minima. □

The favorable completion is simultaneous across all challengers. The adverse completion is pair specific. Robust victory requires a pairwise condition for every challenger, so a different counterexample completion for each failed pair is sufficient for necessity.

H.2 Robust Winner Theorem

Theorem 11 (Exact robust strict winner)**.** *Candidate a is the strict winner under every completion consistent with the observed history if and only if*

$$d_{ab} > r_{ab} \qquad \text{for all } b \neq a. \tag{13}$$

Proof. Suppose (13) holds. The adverse-completion lemma gives

$$S_a(y^{(z)}) - S_b(y^{(z)}) \geq d_{ab} - r_{ab} > 0 \tag{14}$$

for every completion z and challenger b. Hence a is always the unique strict winner.

Conversely, suppose the inequality fails for challenger b. The pairwise adverse completion gives $S_a - S_b = d_{ab} - r_{ab} \leq 0$. Candidate a ties or loses under a completion consistent with the history, so it is not a robust strict winner. □

Corollary 12 (Possible strict winner)**.** *Candidate a is a strict winner under at least one completion if and only if*

$$d_{ab} + r_{ab} > 0 \qquad \textit{for all } b \neq a. \tag{15}$$

Proof. The favorable completion for a attains every maximum simultaneously. If all maxima are positive, that one completion makes a strictly best. If one maximum is nonpositive, no completion makes a strictly beat the corresponding challenger. □

Weak-winner and deterministic tie-rule variants replace the strict inequalities according to whether a tie is accepted for candidate a. The empirical single-region design avoids this complication by retaining panels with a unique full-pool winner.

H.3 Pathwise Lower Bound

Theorem 13 (No exact policy can stop earlier)**.** *Let a policy be completion-safe and suppose it outputs candidate a at history H. Then $d_{ab} > r_{ab}$ holds at H for every challenger b.*

Proof. If an inequality fails, the adverse-completion lemma constructs a completion on which a ties or loses to b. The policy observes the same history before stopping on that completion, so its output is not the unique strict winner. This contradicts completion safety. □

Corollary 14 (Earliest exact stopping time)**.** *Along any fixed realized acquisition path, the first time at which one candidate satisfies* (13) *is the earliest valid stopping time among all completion-safe policies following that path.*

Sufficiency follows from the robust-winner theorem. Necessity follows from the pathwise lower bound. The result is stronger than a lower bound in expectation because it applies separately to every query and label trajectory.

H.4 Monotonicity and Survivor Safety

Once a candidate becomes impossible under every completion, revealing more labels cannot make it possible again. To see this, let $\mathcal{C}(H)$ be the set of completions consistent with history H. If H' extends H, then $\mathcal{C}(H') \subseteq \mathcal{C}(H)$. A candidate that wins on no element of the larger set cannot win on the subset. Thus eliminating an impossible candidate and restricting later acquisition calculations to survivors preserves exactness.

Acquiring a disagreement item i raises the pairwise robust slack $\gamma_{ab} = d_{ab} - r_{ab}$ by $2w_i$ when the label favors a and leaves it unchanged when the label favors b. In the multiclass neutral case it rises by w_i, so γ_{ab} is nondecreasing along any acquisition path. Pairwise favorable possibility $d_{ab} + r_{ab}$ is nonincreasing when labels arrive. These monotonicities provide independent implementation invariants.

H.5 Weights and Tolerances

For preset weights, the same itemwise proof gives contributions in $\{-w_i, 0, w_i\}$. The exact strict certificate remains $d_{ab} > r_{ab}$. If candidate a need only be within weighted tolerance ϵ of every challenger, the condition becomes

$$r_{ab} - d_{ab} \leq \epsilon. \tag{16}$$

This is a non-strict epsilon-optimality statement. The cost-map experiments use unit weights and tolerances converted into count units.

I Full Cost-Map Derivations

I.1 Exact Pairwise Reduction

Under unit weights, let Q be the queried set. Challenger b's largest possible advantage over candidate a is

$$U_{b,a}(Q) = r_{ab}(Q) - d_{ab}(Q). \tag{17}$$

This follows directly from the adverse-completion lemma. Candidate a is robustly within ϵ_N of challenger b exactly when $U_{b,a}(Q) \le \epsilon_N$.

Before querying a disagreement item j, it contributes one to $U_{b,a}$. After querying it, its realized challenger advantage is

$$\mathbb{I}[p_{jb} = y_j] - \mathbb{I}[p_{ja} = y_j]. \tag{18}$$

The reduction in the worst-case upper bound is therefore

$$C_{ab,j} = \mathbb{I}[p_{ja} \neq p_{jb}]\big(1 + \mathbb{I}[p_{ja} = y_j] - \mathbb{I}[p_{jb} = y_j]\big). \tag{19}$$

The coefficient is two if the label favors a and zero if it favors b. It is one if both are wrong on a multiclass item. Starting from D_{ab}, the tolerance condition requires total reduction at least $D_{ab} - \epsilon_N$.

Theorem 15 (Fixed-map cover equivalence)**.** *For a fixed region map g, a queried set Q is an epsilon-optimality certificate if and only if its incidence vector satisfies*

$$\sum_j C_{g(r)b,j} x_j \ge \max\{0, \lceil D_{g(r)b} - \epsilon_N \rceil\} \tag{20}$$

for every region r and feasible challenger b.

Proof. Equation (19) is the exact reduction in (17) caused by querying item j. Reductions add across distinct items. The initial upper bound is D_{ab} and the required final bound is at most ϵ_N. Hence (20) is necessary and sufficient for one pair. Conjunction over all required region–challenger pairs proves the map result. Minimizing $\sum_j x_j$ gives the smallest certificate because one binary variable counts each shared label once. □

For binary labels, the neutral coefficient cannot occur. Define $M_{ab,j} = 1$ when item j is a disagreement whose observed label favors a. Then $C_{ab,j} = 2M_{ab,j}$ and the witness demand is

$$\kappa_{ab} = \max\left\{0, \left\lceil \frac{D_{ab} - \epsilon_N}{2} \right\rceil \right\}. \tag{21}$$

I.2 Any-Output Integer Program

Let z_{ra} select the output candidate for region r, and let $B_{ab} \ge \eta_{ab}$ be a valid big-M constant. The exact any-output formulation is

$$\min_{x,z} \quad \sum_j x_j \tag{22}$$

$$\text{subject to} \quad \sum_{a \in \mathcal{A}_r} z_{ra} = 1 \qquad \forall r \tag{23}$$

$$\sum_j C_{ab,j} x_j \ge \eta_{ab} - B_{ab}(1 - z_{ra}) \qquad \forall r,\ a, b \in \mathcal{A}_r \tag{24}$$

$$x_j, z_{ra} \in \{0, 1\}. \tag{25}$$

Candidates that are not epsilon-optimal under the realized complete labels may be removed. The realized labeling is one allowed completion, so such a candidate cannot be completion-safe.

Fixing z recovers the fixed-map cover. Allowing z creates a disjunction over maps while retaining shared item variables. The LP of one fixed canonical map is therefore not a universal lower bound for (25).

I.3 NP-Hardness Construction

Theorem 16. *Minimum cost-map certification is NP-hard for binary labels in a single region with equal candidate costs and a unique population epsilon-optimal candidate. The statement holds for a fixed target map and for the any-output form.*

Proof. Begin with a Set Cover instance in which every universe element occurs in at least one set. Create one query item for every set. Create a target candidate a and one challenger b_u for each universe element u. Candidate a is correct on every item. On the item representing set j, challenger b_u disagrees with a exactly when u belongs to set j.

Let d_u be the number of sets containing u, and set $L = \max\{2, \max_u d_u\}$. Add $L - d_u$ private items for element u on which only b_u disagrees with a. Every pair (a, b_u) now has $D_{ab_u} = L$, and every disagreement label favors a. Choose $\epsilon_N = L - 2$. Candidate a beats each challenger by L, so it is the unique population epsilon-optimal candidate. Equation (21) gives witness demand one for every element.

Selecting an original set item covers exactly the elements in that set. A private item covers only its own element. Replacing a selected private singleton by any original set containing that element never increases cardinality. Therefore the minimum certificate size equals the minimum Set Cover size. A polynomial certificate algorithm would solve Set Cover. Since a is the only candidate any completion-safe policy may output on this instance, the output map is determined, so the same reduction establishes hardness of the fixed-map problem. The any-output form inherits it, and its extra difficulty is the disjunction over maps in instances where several outputs are admissible. □

I.4 Greedy Bound

For a feasible fixed map with nonnegative integer coefficients and demands, define

$$F(Q) = \sum_{(a,b)} \min \left\{ \eta_{ab}, \sum_{j \in Q} C_{ab,j} \right\}. \tag{26}$$

The sum ranges over the required regional comparisons. This coverage function is integer-valued and monotone submodular. Let $R = \sum_{(a,b)} \eta_{ab}$ and let k be the minimum certificate size. For binary witness multicover, substituting $M_{ab,j}$ and κ_{ab} gives the corresponding witness form.

Proposition 17. *If $R > 0$, active-demand greedy returns a certificate Q_g satisfying*

$$|Q_g| \le k + \lceil k \log(R/k) \rceil \le k(1 + \log R) + 1. \tag{27}$$

Proof. Write Q_t for the greedy set and O for an optimal set of size k. For a comparison with residual $\rho_{ab} = \max\{0, \eta_{ab} - \sum_{j \in Q_t} C_{ab,j}\}$, feasibility of O gives $\sum_{j \in O \setminus Q_t} C_{ab,j} \ge \rho_{ab}$. Consequently $\sum_{j \in O \setminus Q_t} \min\{C_{ab,j}, \rho_{ab}\} \ge \rho_{ab}$, since either a coefficient reaches the residual by itself or capping leaves the entire coefficient sum unchanged. Summing over comparisons shows that the individual marginal gains of the optimal items total at least $\Delta_t = R - F(Q_t)$. Some optimal item therefore gains at least Δ_t / k. Greedy gains at least as much, yielding

$$\Delta_{t+1} \le \Delta_t (1 - 1/k) \le R \exp(-(t+1)/k). \tag{28}$$

A feasible certificate can be constructed with at most R positive-gain selections because each reduces the integer residual by at least one, so $k \le R$. After $\lceil k \log(R/k) \rceil$ selections the residual is at most k, requiring at most k further selections. The second inequality follows from $k \ge 1$ and rounding up by at most one. If $R = 0$, the empty set suffices. □

This is an inherited cover guarantee (Chvátal, 1979; Wolsey, 1982) applied to the completion-induced evidence matrix and its joint regional constraints.

J Mechanical Falsification Tests

The single-region enumerator covers 26,278 prediction matrices with 1,688,292 partial histories and 5,590,672 label completions for 4,994,268 candidate checks. It checks the robust and possible-winner formulas through simultaneous maxima and pairwise minima, while also testing elimination order

Table 21: Uniform-sampling pain results. Exact-winner recovery is a percentage. Regret is in percentage points. Each budget–task entry has 200 random trials.

Task	Exact at 25	Exact at 50	Exact at 100	Exact at 200	Regret at 50	Regret at 200
IFEval	34.5	56.0	81.0	97.5	2.845	0.102
OpenLLM Math	24.5	38.5	56.0	72.0	3.433	0.814
MMLU-Pro	8.5	20.0	26.0	40.0	3.573	1.692
ARC-Challenge	12.5	25.0	43.0	75.0	4.234	0.771
BBH	17.0	31.5	40.0	61.0	4.249	1.317
GPQA	6.0	12.0	25.5	47.5	5.696	2.159
MuSR	8.5	17.0	27.0	44.5	4.519	1.884

Table 22: Winner errors by score-estimation error at a budget of 50 labels. MAE cutoffs are accuracy percentage points. Wrong/total counts use the lower ($\leq Q_{25}$) and upper ($\geq Q_{75}$) MAE quartiles within each task.

Task	Low cutoff	Low: wrong/total	High cutoff	High: wrong/total
IFEval	4.05	20/50	5.46	24/50
OpenLLM Math	3.15	30/50	4.37	28/50
MMLU-Pro	4.41	39/50	6.57	37/50
ARC-Challenge	3.70	34/50	6.34	38/50
BBH	4.44	34/50	5.94	36/50
GPQA	4.50	46/50	5.49	41/50
MuSR	4.26	40/50	6.01	42/50

and singleton-survivor equivalence. An additional 10,000 randomly weighted states check preset nonnegative weights. Every mismatch count is zero.

The cost-map tests cover 493,163 exact pairwise comparisons on reduced binary states up to seven items. They include 240 random tied-cost maps checked against all unqueried-label completions, 80 fixed maps whose direct minimum certificates are compared with multicover optima, and three Set Cover instances. Multiclass tests include a deterministic zero–one–two coefficient example, 80 random finite-alphabet maps checked by completion enumeration, and 80 fixed maps compared with integer-cover optima.

The single-region totals above are reproduced by the shipped enumerator artifact, and the cost-map totals are the case counts of the deterministic cost-map test suite, which passes in full. The reduced binary state enumeration is exhaustive for its item count because every agreement column is common mode and each disagreement has three relevant states: unqueried, queried in favor of the candidate, or queried in favor of the challenger. Multiclass adds the neutral state in which both candidates are wrong.

K Additional Pain Measurements

Table 21 gives the complete exact-winner trajectory for the seven LLM task matrices. IFEval approaches full recovery at 200 labels while MMLU-Pro and GPQA remain below 50%, as does MuSR. Increasing the uniform budget reduces mean regret on every task, yet exact recovery can remain low because close candidates are easily reordered.

At 50 labels, Table 22 compares winner errors in the low- and high-MAE subsets of the same trials. MAE averages absolute accuracy-estimation errors across candidates. The task-specific 25th and 75th percentiles define the two subsets, including values equal to the cutoffs. Each subset contains exactly 50 of the 200 saved trials. Low mean score error can leave the leading candidates in the wrong order, motivating direct checks of their unresolved margins.

The SELECT diagnostic in Table 23 uses the same 706 acquisition paths as the primary comparison. Budgets are rounded-up quartiles and the 90th percentile of completed stopping counts. A path that certifies by the budget retains its certified winner. An unresolved path uses its current selected winner. The three certificate columns partition all runs, while full labels determine actual error and regret

Table 23: Retrospective SELECT budgets: certification and actual error. Certificate columns count runs with stopping time $T < B$, $T = B$ or $T > B$, respectively, out of 706. Error is the percentage of all 706 runs selecting a wrong winner. Regret is in accuracy percentage points over all runs, including certified ones.

Budget	$T < B$	$T = B$	$T > B$	Wrong (%)	Mean regret	P95	Max regret
313	176	3	527	3.68	0.1287	0	23.05
478	353	1	352	0.99	0.0092	0	1.89
641	529	2	175	0.28	0.0038	0	2.29
778	635	0	71	0.14	0.0001	0	0.08

Table 24: Full structural label-count summary over 2,824 runs.

Method	Min	Q25	Median	Q75	Mean
Full uniform	736	1054.8	1159.0	1223.3	1128.2
Quotient uniform	16	362.0	571.5	783.0	580.5
FRONTIERCUT	13	290.0	418.0	547.0	427.2

independently. Because fewer than 5% of runs have a wrong winner at every budget, the median and 95th percentile of regret are both zero. The mean and maximum expose the remaining tail.

L COMPLETE SINGLE-REGION TABLES

L.1 STRUCTURAL AND CLASSICAL COMPARISONS

The structural comparison contains 353 panels, eight replicates, and 2,824 matched runs per method. Full uniform, quotient uniform, and FRONTIERCUT each certify the correct winner in all runs. Table 24 includes the mean and full range omitted from the main text.

Against quotient uniform, run-level savings range from -14.40% to 52.01%. Their median is 22.95%, with lower and upper quartiles of 13.59% and 33.61%. The few negative runs show why a deterministic claim of per-run dominance would be false. The panel-level median is 23.02%, with panel range -4.32% to 51.04%.

The classical-baseline comparison contains two replicates per panel. FRONTIERCUT has mean 427.37 labels. Static QBC has 528.72, Active Comparison has 552.21, and the Model Selector oracle grid has 553.90. The mean and median order differ for Model Selector because a subset of panels has large label counts. Paired savings are therefore preferred to comparing unpaired marginal means.

L.2 RESPONSE-ONLY TEMPERATURE CALIBRATION

The SELECT grid contains 14 temperatures. Calibration uses 376 response-only structural panels, two replicates per panel, and at most 256 acquisition steps. Each temperature has 183,702 path observations. Table 25 reports macro path accuracy. The selected value $\tau = 0.7$ has the highest score, with $\tau = 0.8$ and 1.0 close behind.

Calibration selection does not use gold correctness, full-pool scores, or a unique-winner filter. The response-only cohort contains the evaluation organizations, and Appendix C.5 reports the organization-disjoint fold assignment together with the savings it gives. Temperature selection uses response agreement rather than evaluation correctness.

Method ordering by median label count changes across the temperature grid, with FRONTIERCUT preceding CERTSELECT at $\tau = 0.1$–0.5 and the order reversed at tested temperatures from 0.6 onward.

L.3 ALL P8B VARIANTS

Table 26 includes favorable oracle-grid envelopes in addition to the deployable structurally calibrated methods. Oracle rows choose the best temperature after observing the number of labels required on a run. They are diagnostic upper envelopes and not deployable selectors.

Table 25: Response-only structural temperature calibration.

τ	Successes	Macro accuracy
0.1	110,975	0.61596
0.2	106,378	0.59229
0.3	116,452	0.64422
0.4	118,594	0.65592
0.5	119,187	0.65916
0.6	119,074	0.65863
0.7	**119,466**	**0.66087**
0.8	119,423	0.66054
0.9	119,343	0.65994
1.0	119,399	0.66051
2.0	118,286	0.65403
3.0	117,464	0.64918
4.0	117,013	0.64686
5.0	115,993	0.64147

Table 26: Full P8B method summary over 706 runs. Every method certifies and returns the correct winner in 706 of 706 runs.

Method	Min	Q25	Median	Q75	Max	Mean
FRONTIERCUT	13	291.00	417.5	549.75	1159	427.37
SELECT all, calibrated	10	313.00	477.5	640.75	1161	478.15
SELECT all, oracle grid	10	297.50	448.0	600.75	1153	452.76
CERTSELECT, calibrated	11	269.25	401.0	572.75	1170	425.41
CERTSELECT, oracle grid	9	260.25	378.0	508.00	1134	386.94

FRONTIERCUT saves a panel median 4.23% against all-candidate oracle SELECT, with interval $[3.30, 5.73]\%$. Oracle CERTSELECT saves 8.33% over FRONTIERCUT, with interval $[7.49, 9.46]\%$. These oracle comparisons diagnose headroom from temperature selection. The main claims use the calibrated rows only.

High-precision query ranking is included in every final P8B variant. Float64 first screens response patterns within a narrow objective radius. The implementation then ranks exact integer imbalance coefficients in extended precision. A decimal fallback at 80 digits resolves any remaining ambiguity. The procedure orders close but genuinely different objectives rather than declaring them tied.

M CROSS-TASK DATASET DETAILS

Table 27 reports one median paired saving per dataset. The class count in parentheses ranges from two to 1,000. All 19 values are positive. The weakest panel is GLUE WNLI at 3.71%, and the strongest is PACS at 39.54%.

The aggregate run-level saving against quotient uniform is 41.01%. This larger value combines informative-column removal with acquisition. Against static QBC it is 19.50%, and against static all-pair cut it is 18.18%. The multiclass-only medians are 52.70%, 27.91%, and 27.97% in the same order.

All 152 runs for all four methods have valid certificates and correct winners. This corresponds to 608 method runs. The result directly verifies the finite-alphabet extension on natural multiclass matrices rather than only on synthetic enumeration.

Comparison with SELECT acquisition on the same panels. The static rankings above are non-adaptive. We therefore repeat the 19 panels with the evidence-weighted SELECT objective at the calibrated $\tau = 0.7$ carried to categorical predictions, which keeps its candidate acquisition masses and all-candidate scope while allowing more than two labels per item. Table 28 lists the dataset medians. Over the 152 matched runs the run-median paired saving against this baseline is 14.00% for CERTSELECT and 14.16% for FRONTIERCUT, with run wins, ties and losses of 146/3/3 and 148/1/3. The dataset medians are 13.96% and 14.15%, positive on 18 and 19 of the 19 datasets, and

Table 27: Cross-task dataset census and dataset-median saving of FRONTIERCUT relative to static all-pair cut. Each dataset has eight matched runs.

Dataset	Classes	Items	Models	Saving	Dataset	Classes	Items	Models	Saving
CIFAR-10 4070	10	10,000	80	19.73%	GLUE QQP	2	40,430	98	15.24%
CIFAR-10 5592	10	10,000	80	17.19%	GLUE RTE	2	277	76	11.78%
Domain drift	6	3,600	9	9.50%	GLUE SST-2	2	872	88	17.79%
Emotion detection	4	5,509	8	32.42%	GLUE WNLI	2	71	68	3.71%
GLUE CoLA	2	1,043	97	11.45%	ImageNet	1000	50,000	92	32.17%
GLUE MNLI	3	9,815	80	34.77%	ImageNet PyTorch	1000	50,000	114	30.36%
GLUE MRPC	2	408	74	30.22%	ImageNetV2 MF	1000	10,000	114	27.97%
GLUE MRPC train	2	3,668	51	35.90%	ImageNetV2 T0.7	1000	10,000	114	18.12%
GLUE QNLI	2	5,463	78	10.59%	ImageNetV2 top	1000	10,000	114	12.81%
PACS	7	9,991	30	39.54%					

Table 28: Dataset-median paired label saving against SELECT acquisition carried to categorical predictions. Each dataset has eight matched runs and all methods share the same exact stopping rule.

Dataset	CERTSELECT	FRONTIERCUT	Dataset	CERTSELECT	FRONTIERCUT
CIFAR-10 4070	5.07%	3.76%	GLUE RTE	17.76%	14.15%
CIFAR-10 5592	12.84%	16.57%	GLUE SST-2	14.78%	13.59%
Domain drift	3.77%	2.43%	GLUE WNLI	0.00%	1.52%
Emotion detection	27.66%	23.72%	ImageNet	10.62%	13.36%
GLUE CoLA	13.96%	13.77%	ImageNet PyTorch	17.17%	18.63%
GLUE MNLI	6.35%	5.28%	ImageNetV2 MF	13.81%	14.27%
GLUE MRPC	16.78%	16.96%	ImageNetV2 T0.7	17.21%	19.42%
GLUE MRPC train	23.55%	21.50%	ImageNetV2 top	20.39%	23.38%
GLUE QNLI	5.85%	3.01%	PACS	22.83%	25.68%
GLUE QQP	3.91%	4.16%			

the single exception is the 71-item GLUE WNLI panel where CERTSELECT ties. Within the 88 multiclass runs the savings are 13.81% and 16.57%. All 456 method runs return a valid certificate and the correct full-pool winner.

Expanded multiclass candidate panels. The multiclass comparison above rests on eleven panels, one per multiclass source. To widen it we split the deduplicated candidate pool of each source into disjoint groups and keep the groups with a unique full-pool winner. This gives 29 candidate panels from the same eleven sources, of which 27 are new groupings with 648 new runs. The panel-median paired saving against SELECT is 16.94% with interval $[12.87, 19.23]\%$ for FRONTIERCUT and 14.88% with interval $[13.17, 17.17]\%$ for CERTSELECT. Both are positive on all 29 panels, and the two point estimates stay near the 16.57% and 13.81% that the original candidate pools give for the same two rules. Merging the new groupings with those pools and removing duplicates leaves 38 configuration panels with 304 paired runs per method. A grouping changes the candidate set of a source rather than adding a source, so these panels measure stability across candidate sets and not broader data coverage.

N COMPLETE STRATIFIED ROBUSTNESS RESULTS

Table 29 gives all 12 strata. Feature ranges can overlap at quartile boundaries because tied feature values are retained in deterministic panel order. The effect is paired saving of FRONTIERCUT against quotient uniform.

Eight replicate-specific medians are computed within every stratum. The smallest across all 96 replicate–stratum cells is 12.82%. The worst panel fraction positive is 96.59%. These checks guard against a median that is positive only after mixing replicates.

Table 29: All LLM robustness strata. CI is the 95% panel-bootstrap interval for the stratum median. Positive is the fraction of panels with positive paired effect.

Family	Quartile	Panels	Median	Positive	95% CI
Candidate count	Q1	89	13.78%	97.75%	[10.86, 15.87]%
Candidate count	Q2	88	19.26%	98.86%	[16.26, 22.65]%
Candidate count	Q3	88	24.08%	100.00%	[20.66, 27.71]%
Candidate count	Q4	88	36.81%	100.00%	[34.19, 39.47]%
Winner gap	Q1	89	27.12%	100.00%	[21.26, 34.07]%
Winner gap	Q2	88	29.89%	100.00%	[24.46, 33.18]%
Winner gap	Q3	88	21.38%	100.00%	[18.40, 24.68]%
Winner gap	Q4	88	17.61%	96.59%	[14.28, 19.42]%
Informative ratio	Q1	89	14.56%	97.75%	[11.82, 16.60]%
Informative ratio	Q2	88	22.38%	100.00%	[19.46, 25.36]%
Informative ratio	Q3	88	28.40%	98.86%	[25.30, 32.29]%
Informative ratio	Q4	88	31.64%	100.00%	[28.47, 34.43]%

Table 30: JudgeTuning certificate sizes and ratios.

Quantity	2 pp	1 pp
Any-output floor	1078	1100
Fixed-map LP	2032.7549	2151.8765
Greedy certificate	2111	2216
Greedy/LP	1.03849	1.02980

O Additional Cost-Map Statistics

O.1 JudgeTuning Caps and Regions

The eligible configuration counts at the 20 sampled caps are

$$\boldsymbol{m} = (21, 40, 60, 80, 100, 120, 139, 159, 179, 199, \\ 219, 239, 259, 278, 298, 321, 338, 361, 378, 397).$$

The corresponding canonical winner indices are

$$\boldsymbol{v} = (122, 122, 355, 355, 355, 355, 355, 355, 355, 339, \\ 158, 160, 160, 160, 154, 154, 154, 154, 154, 154).$$

There are six distinct winners.

The full cost ordering contains 331 distinct regions. The any-output lower bound selects a candidate minimizing the regional bound at each cap. These choices use 16 distinct candidates at a two-point tolerance and 17 at one point. At two points, nine of the 16 minimize the bound in three or more regions. The four most frequent choices cover 281 regions. At two points, 1,780 distinct fixed-map pair constraints remain after merging repeated region–pair requirements. Pair demands range from one to 1,082, with median 339 and 95th percentile 962. Available witnesses range from 43 to 1,414, with median 455. The LP solver terminates optimally after 369 iterations.

At one point, the any-output lower bound is 1,100 and the fixed-map LP is 2,151.8765. At two points, they are 1,078 and 2,032.7549. Table 30 expands the optimization ratios.

O.2 WinoGrande Proxy-Cost Distributions

The 135 external panels contain four to 71 models with median nine and lower and upper quartiles of six and 16, while the 90th percentile of 27 describes the upper end of the observed candidate-count distribution. Cost-region counts range from three to 15 with median four and 90th percentile seven. Canonical winner counts range from one to eight with median three and 90th percentile 4.6.

At two points, fixed-map LP fractions range from 7.89% to 59.19%, with median 27.15%. Greedy fractions range from 7.97% to 61.33%, with median 27.70%. Greedy-to-LP ratios have median 1.0123 and first quartile 1.0000. The 90th percentile is 1.0596 and the maximum is 1.1554.

At one point, fixed-map LP fractions range from 8.92% to 62.19%, with median 29.76%. Greedy fractions range from 8.92% to 64.48%, with median 29.99%. Greedy-to-LP ratios have median 1.0084 and first quartile 1.0000. The 90th percentile is 1.0535 and the maximum is 1.1478.

All 270 greedy certificates pass direct feasibility checks. Independent 3090 and 4090 recalculations agree on the deterministic result core.

P Output-Map Switching

Each run starts from the greedy certificate of one of three initial maps and tries to delete its labels in greedy order, in reverse greedy order or in one of eight random orders. A label is deleted whenever every region still has a candidate that the remaining labels certify within tolerance. At two points, the best switched map reduces the canonical greedy certificate from 2,111 to 2,029 labels, or 3.88%. Maximum realized regret is 1.7193 points and the certificate bound is 1.9729.

At one point, the count decreases from 2,216 to 2,135, or 3.66%. Maximum realized regret is 0.9019 and the bound is 0.9865. Both switched maps are valid certificates.

These results narrow the next research question. A future any-output method should first locate a natural workload where alternative valid maps have materially different certificate sizes. It should not assume that combinatorial output freedom automatically produces large annotation savings.

Q Sampling Details

Q.1 Panel Construction

Organization panels are formed from a fixed candidate–item correctness matrix and the accompanying candidate metadata. Candidate identifiers must be unique. Within one organization, candidates with identical response profiles are deduplicated before evaluation. Columns that are constant across the retained profiles are marked uninformative. The main evaluation set retains panels with one unique full-pool winner.

The response-only calibration view is constructed earlier. Binary response orientation is canonicalized by XOR with one reference profile, which makes a response split invariant to exchanging label names. Unique response profiles are retained when the panel meets the minimum profile count. This selection does not inspect correctness scores or require a unique population winner.

The cross-task panels are derived from public raw prediction collections. Each record identifies the dataset and its full-pool winner alongside the item and class counts while also giving the candidate count and informative-item count together with eight matched acquisition records that share the same predictions and labels. Multiclass certification uses raw predicted labels rather than only correctness indicators.

JudgeTuning costs and configuration outputs are taken from the workload revision released with the original study (Salinas et al., 2025). The 20 pain caps are fixed quantile-like points in the cost ordering. The 331-region map contains every distinct feasible-set change after preprocessing. No cap is selected according to the performance of a proposed acquisition rule.

WinoGrande proxy costs are parsed from declared parameter-size tokens in deployment identifiers. Panels without enough parseable distinct-cost profiles are excluded under the parsing rules. The cost variable is labeled as a proxy in every result table.

Q.2 Matched Randomness

For each panel and replicate, a fixed seed sequence generates child streams for all methods. Random-order baselines use their stream to construct a permutation. Adaptive methods use their stream only to break exact acquisition ties or choose an item from an identical response-pattern group. The gold label vector is fixed across every method.

Table 31: Random seeds by result file.

Result file	Seeds
`P0_RESULTS.json`	0–199, one per uniform sample, for every task and budget
`P1_RESULTS.json`	0–99 for the invariance trials, 10000–10099 for their random candidate sequences and 0–199 for the initial-batch simulation
`P2D_RESULTS.json`	1252026080702
`P2E_RESULTS.json`	1252026080703
`P2F_RESULTS.json`	125260807
`P2I_RESULTS.json`	125260807 combined with job indices 0–7 for the 10,000 weighted states
`P2J_4090_RESULTS.json`	20260807
`P5B_4090_RESULTS.json`	20260807
`P6D_RESULTS.json`	125640 as the base seed of the ten deletion orders
P8B result files	1252026080703 for acquisition, 1252026080800 for temperature calibration and 1252026080802 for the 10,000 paired bootstrap resamples

Matched design reduces variation from arbitrary item-order ties. It does not force methods to query the same item at a step. Once acquisition paths diverge, each method follows its own state and stops at its own first exact certificate.

Q.3 Panel-Level Bootstrap

Let Δ_{pr} be relative saving on panel p and replicate r. The primary panel value is

$$\bar{\Delta}_p = \mathrm{median}_r\, \Delta_{pr}. \tag{29}$$

The point estimate is $\mathrm{median}_p\, \bar{\Delta}_p$. One bootstrap draw samples 353 panel indices with replacement and recomputes that median. The reported interval uses the 2.5th and 97.5th percentiles of 10,000 draws. Replicates stay grouped inside the resampled panel.

This procedure matches the scientific unit. An organization with two candidates and an organization with dozens of candidates each contribute one panel value. A run-level bootstrap is retained only as a secondary summary.

R Machine and Numerical Details

The 4090 experiment uses Linux 5.4 with Python 3.8.10 and NumPy 1.22.4. It runs 16 worker processes with one numerical thread per process. No accelerator kernel is used because the workload is dominated by finite matrix and acquisition-state operations. Wall-clock time is 836.04 seconds. The 3090 execution uses Linux 5.15, Python 3.10.16 and NumPy 1.22.4 with 16 worker processes and takes 1,292.29 seconds.

Both machines write the same deterministic result core, and `P8B_V21_CROSS_MACHINE_RECALCULATION.json` recomputes the core from each machine's result file and records that the two are equal. The core excludes machine labels, wall-clock time, and machine-specific environment fields. It includes the calibrated temperature and panel results together with label counts and certificates. The comparison also covers correctness and query ordering. The P0 result file records Python 3.12.4, NumPy 2.3.2 and SciPy 1.16.2, and the P7A result file records Python 3.12.3 and NumPy 2.5.2. All fixed-map certificates are verified after construction by recomputing every active demand from the selected item set.

R.1 Random Seeds

Table 31 lists the random seeds of each result file. All seeds initialize NumPy's default generator. Result files not listed there use no random numbers.

R.2 NUMERICAL TIE REFINEMENT

SELECT query scores can be extremely close when many candidates have similar evidence. Using one floating-point comparison threshold as a tie rule can change an acquisition path across hardware. The final implementation therefore decomposes the score into exact integer response-level coefficients and exponential temperature levels.

Float64 identifies patterns near the apparent minimum. Long-double arithmetic ranks their imbalance values. If two remain numerically indistinguishable, an 80-digit decimal calculation resolves the order. Exact equality is retained as a tie and passed to the fixed random tie stream. Tests cover orientation invariance and response-pattern grouping alongside tie refinement and survivor restriction. They also check stopper equality and agreement between the machines.