LEARNING WHAT TO VERIFY: RANKING UNDER A FIXED EXACT-EVALUATION BUDGET

Anonymous authors
Paper under double-blind review

ABSTRACT

Many optimization pipelines can cheaply score far more candidates than they can exactly verify, so the scorer effectively allocates a scarce exact-evaluation budget. We formalize this setting as bounded exact-verification retrieval: rank a fixed pool, verify B candidates, and return the best verified utility. We show a sharp prediction–decision separation: as the pool grows while B remains small, normalized value error and pairwise ranking error can vanish while verification regret remains maximal. A layer-cake representation expresses regret through quality thresholds missed by the verified set; bounding each miss through predicted mass on the corresponding quality set yields decision-aligned regret envelopes and the smooth QualityHit loss. Decomposing QualityHit across thresholds shows how budget reweights competing objectives: all budgets share the same realizable zero-loss limit, while under finite capacity the restricted optimum can shift, with constructive ranking reversals across budgets. Across FJSP, CVRP, and T-SP, QualityHit improves verification quality in point estimates across all tested family–budget settings, with graded utility explaining most of the gain and budget adaptation providing a smaller additional effect. These results shift surrogate learning from global predictive accuracy toward the decisions exposed to scarce exact evaluation.

1 INTRODUCTION

Many optimization pipelines can cheaply score far more candidates than they can exactly evaluate. A scheduler may rank hundreds of schedules before expensive simulation, a routing model may score many routes before exact decoding and feasibility checks, or a configuration search may screen many settings before running only a few full evaluations. In such pipelines, the learned score does more than predict candidate value: by ranking the pool, it determines which candidates receive the scarce exact-evaluation budget.

We call this decision rule bounded exact-verification retrieval. Given a fixed candidate pool, the scorer ranks all candidates, only the top B are evaluated by the exact objective, and the best verified candidate is returned. The full candidate pool is available before verification, exact evaluations have equal cost, and no verification outcome is used to select subsequent candidates. Figure 1(a) illustrates the central asymmetry: the same candidate pool and true utilities can lead to different deployment outcomes solely because the scorer changes which candidates enter the verified prefix.

The mismatch with global surrogate accuracy can be extreme. Consider n candidates with one unit-gain candidate and $n-1$ zero-gain candidates, and rank exactly B zero-gain candidates ahead of the useful one. Then

$$E_{\text{pair}} = \frac{B}{n-1}, \qquad E_p^p = \frac{1}{n} + \frac{n-1}{n^{p+1}},$$

while the verified prefix still misses the unique unit-gain candidate. The construction requires only B strict-utility inversions, while the remaining prediction perturbations shrink with the pool size. Hence for fixed B, or more generally along any sequence $B_n = o(n)$ as $n \to \infty$, both global errors vanish although verification regret remains maximal. Proposition 1 formalizes this separation.

The natural object is therefore not full-pool prediction accuracy, but whether the verified prefix intersects the candidates above each utility threshold. For a threshold t, the verified decision reaches

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.

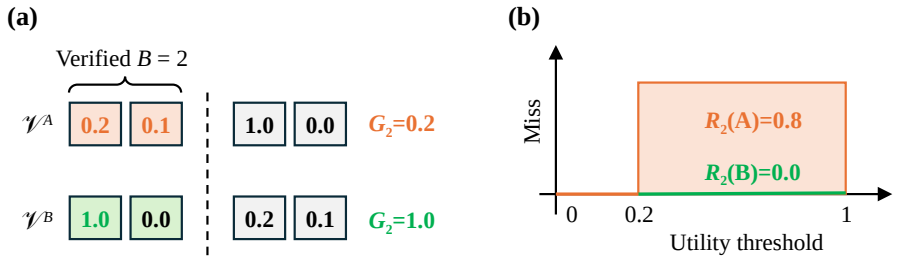

Figure 1: **Bounded exact-verification retrieval.** (a) For the same candidate pool and true utilities, different rankings expose different candidates to an exact-evaluation budget of $B = 2$, leading to different best verified utility G_2. (b) Verification regret equals the total width of utility thresholds whose corresponding quality sets are missed by the verified prefix, providing the intuition for the layer-cake analysis in Section 4.

utility at least t exactly when its top-B prefix intersects the corresponding quality set. Integrating these hit and miss events yields verification regret. The key observation is that when the verified prefix misses a quality set, the predicted probability mass that can be assigned to that set is necessarily constrained. Applying this argument threshold by threshold yields a hierarchy of decision-aligned regret envelopes. This follows the same broad metric-to-surrogate design pattern as metric-driven learning-to-rank methods such as LambdaLoss (Wang et al., 2018), but here the downstream utility is the maximum exact utility recovered from a verified prefix, leading naturally to nested set-hit events rather than a position-additive ranking metric.

The resulting smooth objective, QualityHit, aggregates predicted mass over the full quality set at each threshold. Its underlying top-subset envelope gives the tighter decision-aligned certificate, while the full-set form provides a differentiable objective suitable for training. The same threshold decomposition also reveals how the deployment budget changes the relative weighting of competing quality levels. All budgets share the same winner-concentrating zero-loss limit in a realizable class, whereas under finite model capacity this reweighting can move the restricted optimum. We characterize this effect through comparative statics and give a constructive example in which the population-optimal restricted ranking reverses across budgets.

We evaluate QualityHit on controlled candidate pools from FJSP, CVRP, and TSP with verification budgets $B \in \{4, 8\}$. Across all reported family–budget settings, QualityHit gives positive held-out point-estimate gains over value regression, a softmax listwise baseline, and matched LambdaRank@B. Structural ablations further separate the roles of graded post-verification utility and budget-dependent weighting, with graded utility accounting for most of the observed improvement.

Our main contributions are:

- **Bounded exact-verification retrieval and prediction–decision separation.** We formalize the setting in which a learned scorer allocates a limited exact-evaluation budget over a fixed candidate pool, and show that vanishing full-pool value error and pairwise ranking error need not imply vanishing verification regret.
- **Decision-aligned regret envelopes and budget-sensitive learning.** We express verification regret through missed utility thresholds and derive a hierarchy of verification-specific regret envelopes, including the smooth QualityHit objective used for training. The same decomposition shows how deployment budget reweights threshold objectives: all budgets share the same realizable zero-loss limit, while finite-capacity restrictions can induce budget-dependent optima and ranking reversals.
- **Controlled empirical evaluation.** We evaluate QualityHit across FJSP, CVRP, and TSP at $B \in \{4, 8\}$ using matched value, listwise, cutoff-aware, and LambdaRank baselines together with structural ablations that distinguish the contribution of graded utility from budget-specific weighting.

Section 3 defines bounded exact-verification retrieval and the deployment objective. Section 4 develops the decision-aligned theory and regret envelopes. Sections 5–7 present the training objectives, experimental protocol, and empirical results, with complete proofs and additional analyses provided in the appendix.

2 Related Work

Metric-driven learning to rank. Pairwise and listwise objectives replace flat or discontinuous ranking metrics with trainable surrogates (Burges et al., 2005; Cao et al., 2007). LambdaRank uses metric-sensitive pairwise updates; in the NDCG setting, the pair weights depend on the NDCG change induced by swapping two items (Burges et al., 2006; Wang et al., 2018). The closest methodological antecedent to our construction is the LambdaLoss framework of Wang et al. (2018): it places LambdaRank inside a probabilistic loss framework and defines metric-driven losses with explicit connections to ranking metrics. Thus, deriving training weights or loss structure from the deployed metric is not itself our novelty. Our difference is the deployed utility: exact verification returns the maximum actionable gain among the B selected candidates, whose layer-cake representation becomes a collection of nested set-hit events. QualityHit therefore weights probability mass on utility-threshold sets rather than pairwise swap changes or rank-position gains. We use matched LambdaRank@B as the closest metric-driven ranking control.

Precision-at-top and top-k calibration focus learning near a cutoff (Kar et al., 2015; Yang & Koyejo, 2020). Differentiable sorting and top-k methods, including NeuralSort and differentiable top-k cross-entropy objectives, provide smooth alternatives (Grover et al., 2019; Petersen et al., 2022; Struski et al., 2025). These methods are close in deployment geometry, but they do not by themselves encode the best exact actionable gain recovered after verification.

Decision-focused learning. Decision-focused methods optimize downstream decisions rather than prediction error alone. SPO+ is a convex surrogate derived from decision error under predicted objective coefficients (Elmachtoub & Grigas, 2022). Wilder et al. (2019) differentiates through continuous relaxations of combinatorial problems, while Mandi et al. (2022) formulates decision-focused learning as learning to rank over candidate solutions. Our downstream action is different: the learned score selects which utilities become observable, after which the exact objective chooses the winner.

Expensive-evaluation optimization. Classical efficient global optimization and Bayesian optimization choose new expensive evaluations sequentially through acquisition criteria, while surrogate-assisted evolutionary computation manages expensive evaluations with learned surrogates (Jones et al., 1998; Frazier, 2018; Jin, 2011). We isolate the one-shot case in which the candidate pool is fixed before verification and the learning problem is the ordering of that pool.

3 Budgeted Exact Verification

Let x be an instance, $y_0 = b(x)$ a trusted fallback, and $\mathcal{C}(x) = \{c_1, \dots, c_n\}$ a fixed candidate pool available before any exact evaluation. Exact verification returns terminal utility $V(x, c_i)$. We define the signed gain relative to the fallback and its actionable part as

$$g_i = V(x, c_i) - V(x, y_0), \qquad a_i = [g_i]_+ \geq 0.$$

A scorer $s_\theta(x, c_i)$ ranks candidates using only cheap pre-verification features. With budget B, deployment verifies

$$\mathcal{V}_B(x; \theta) = \mathrm{TopB}_{c \in \mathcal{C}(x)}\, s_\theta(x, c)$$

and obtains

$$G_B(x; \theta) = \max_{c_i \in \mathcal{V}_B} a_i.$$

The within-pool oracle is $a^\star = \max_i a_i$, so the ranking-induced regret is

$$R_B(x; \theta) = a^\star - G_B(x; \theta). \tag{1}$$

All Top-B operations use a fixed deterministic tie rule. The candidate generator is fixed in all experiments; the learning problem is only the ordering used to allocate the B exact evaluations.

Motivating instantiations. In hyperparameter optimization or architecture search, c_i can be a configuration or architecture and $V(x, c_i)$ its validation performance after the prescribed evaluation procedure; B is the number of full evaluations that can be afforded. In molecular or materials screening, the same abstraction can represent a small budget of high-fidelity docking, DFT, or molecular-dynamics evaluations. These examples motivate the decision problem only; our experiments use controlled combinatorial pools where full-pool counterfactual utilities can be observed offline.

Training access. Exact gains are available as offline supervision for the training pools. They are labels and evaluation targets, never scorer inputs. The paper therefore addresses scarce exact access at deployment, not adaptive acquisition of training labels.

4 From Verification Regret to Trainable Envelopes

The theory has two steps: global surrogate accuracy can fail at the verification cutoff, and verification regret can be written as missed utility-threshold sets. The second step produces both a tighter piecewise-smooth envelope and the smoother relaxation used for training.

4.1 Prediction–decision separation

For a value-prediction vector $\hat{a} = (\hat{a}_1, \dots, \hat{a}_n)$ whose ordering determines the verified prefix, define

$$E_p(\hat{a}, a)^p = \frac{1}{n} \sum_{i=1}^{n} |\hat{a}_i - a_i|^p,$$

and let E_{pair} be the fraction of strict-utility pairs (i, j) with $a_i > a_j$ for which $\hat{a}_i \le \hat{a}_j$.

Proposition 1 (Full-pool error need not control verification regret)**.** *For every $n \ge 2$, $1 \le B < n$, and fixed $p \ge 1$, there exist $a \in [0,1]^n$ with exactly one positive candidate and predictions $\hat{a}$ such that*

$$R_B = 1, \qquad E_{\text{pair}} = \frac{B}{n-1}, \qquad E_p(\hat{a}, a)^p = \frac{1}{n} + \frac{n-1}{n^{p+1}}.$$

Consequently, for any $B_n = o(n)$, both full-pool errors vanish while verification regret remains maximal.

The failure is concentrated at the cutoff: only B strict pairs need to be wrong, but they are exactly the pairs that keep the useful candidate outside the verified prefix.

4.2 A hierarchy of set-hit envelopes

For $t \in (0, a^\star]$, define the quality set $H_t = \{i : a_i \ge t\}$ and $m_t = |H_t|$. A verified prefix achieves less than t utility exactly when it misses H_t, hence

$$R_B = a^\star - G_B = \int_0^{a^\star} \mathbf{1}\{\mathcal{V}_B \cap H_t = \varnothing\}\, dt. \tag{2}$$

Let $p_i = \exp(s_i)/\sum_j \exp(s_j)$, $P_t = \sum_{i \in H_t} p_i$, and let $P^{\downarrow}_{t,k}$ be the mass of the k largest-probability members of H_t. For $c \in (0,1)$ define

$$\phi_c(z) = \min\left\{ \frac{-\log z}{-\log c}, \frac{1-z}{1-c} \right\}. \tag{3}$$

Set

$$\mathcal{U}^{(B)}_{\text{top}}(s; a) = \int_0^{a^\star} \min_{1 \le k \le m_t} \phi_{k/(k+B)}(P^{\downarrow}_{t,k})\, dt$$

and

$$\mathcal{L}^{(B)}_{\text{hit}}(s; a) = \int_0^{a^\star} \frac{-\log P_t(s)}{\log(1 + B/m_t)}\, dt. \tag{4}$$

Theorem 2 (Verification-regret envelope)**.** *For every finite score vector,*

$$R_B \le \mathcal{U}_{\mathrm{top}}^{(B)}(s;a) \le \mathcal{L}_{\mathrm{hit}}^{(B)}(s;a). \tag{5}$$

The first inequality uses the sharp fact that if top-B misses a k-candidate set, its predicted mass is at most $k/(k+B)$. The second chooses the full quality set and replaces the piecewise envelope by a log loss. Exact finite-sum forms and proofs are in Appendix A.

4.3 What the smooth relaxation does—and does not—justify

The top-subset envelope $\mathcal{U}_{\mathrm{top}}^{(B)}$ is continuous and piecewise smooth; kinks occur when the active subset size changes or when probability order changes inside a quality set. QualityHit uses $\mathcal{L}_{\mathrm{hit}}^{(B)}$ because it reduces every threshold to a single setwise log-mass term and avoids an inner sort/min operation during training. This is an optimization choice, not a theorem that the full-set relaxation is statistically preferable.

The distinction is most visible for broad quality sets. A miss of a set of size m only implies

$$P_t \le \frac{m}{m+B},$$

so for $m \gg B$ the certificate is weak. At the same time,

$$\frac{1}{\log(1+B/m)} = \frac{m}{B} + O(1), \qquad m/B \to \infty.$$

Thus the large QualityHit coefficient on broad threshold sets compensates for the slack of the full-set certificate. It is not a direct derivative of true verification regret with respect to that threshold. This is why the empirical normalizer interventions in Section 7 are interpreted as sensitivity checks, not as a decomposition of a verified causal mechanism.

For very small B, $w_\ell(B) \approx \Delta_\ell m_\ell / B$, so broader quality sets receive relatively more weight; as B grows, the m_ℓ-dependent normalizer differences vanish to first order. Changing B therefore changes these certificate-induced coefficients whenever threshold-set sizes differ. Appendix A.3 proves the corresponding relative-weight monotonicity and gives a restricted-model example whose optimum reverses between $B = 4$ and $B = 8$. Those results establish that budget dependence is possible; they do not imply that it must be large, or beneficial, in a particular trained network.

5 Training Objectives

All methods receive the same cheap pre-verification features and output one score per candidate.

SoftmaxListwise (SL). For $p_i = \exp(s_i)/\sum_j \exp(s_j)$, the tuned listwise baseline uses

$$q_i = \frac{\exp(g_i/\tau)}{\sum_j \exp(g_j/\tau)}, \qquad \mathcal{L}_{\mathrm{SL}} = -\sum_i q_i \log p_i,$$

with temperature and checkpoint selected on validation data.

LambdaRank@B. We use a deterministic full-pair LambdaRank-style objective targeted to actionable NDCG@B (Burges et al., 2006; Wang et al., 2018). Relevance is $a_i = [g_i]_+$, the logarithmic rank discount is truncated after position B, and each unequal-utility pair receives a logistic ranking loss weighted by the absolute change in truncated DCG induced by swapping the pair, normalized by ideal DCG@B. This is our closest metric-driven ranking control.

ActionableValueRegression (AVR). AVR predicts $a_i = [g_i]_+$ with tuned SmoothL1 loss.

QualityHit. Let $v_1 > \cdots > v_L > 0$ be the distinct positive utilities, $v_{L+1} = 0$, $H_\ell = \{i : a_i \ge v_\ell\}$, $m_\ell = |H_\ell|$, $P_\ell = \sum_{i \in H_\ell} p_i$, and $\Delta_\ell = v_\ell - v_{\ell+1}$. The smooth full-set envelope is

$$\mathcal{L}_{\mathrm{hit}}^{(B)} = \sum_{\ell=1}^{L} \frac{\Delta_\ell}{\log(1+B/m_\ell)}(-\log P_\ell).$$

The Δ_ℓ terms encode graded post-verification utility. The factor $1/\log(1 + B/m_\ell)$ is inherited from the full-set miss certificate in Section 4; we do not interpret it as a calibrated measure of threshold importance.

WinnerClassification (exploratory). As a winner-only control, WinnerClassification uses the deterministic actionable maximizer $y^\star \in \arg\max_i a_i$ as a class label and minimizes $-\log p_{y^\star}$ under the same tie rule. It discards intermediate utility levels.

Objective-component controls. The equal-interval control keeps the nested quality sets but replaces observed Δ_ℓ by equal widths. The no-normalizer control removes $1/\log(1 + B/m_\ell)$ entirely. The fixed-B_0 control keeps the same formula but freezes its budget argument at $B_0 = 4$:

$$\mathcal{L}_{\text{fixed-B}_0} = \sum_{\ell=1}^{L} \frac{\Delta_\ell}{\log(1 + B_0/m_\ell)}(-\log P_\ell).$$

At deployment $B = 4$, QualityHit and fixed-B_0 are identical term by term. At $B = 8$, the comparison changes only the budget argument inside the normalizer.

6 Experimental Design

The experiments address three questions. **Q1:** How does QualityHit compare with value, listwise, top-k, and metric-driven ranking baselines under the same verification budget? **Q2:** Which parts of the objective matter empirically—graded post-verification utility, the full-set normalizer, and budget matching? **Q3:** Do the observed candidate pools exhibit the heterogeneous threshold-set geometry required for budget-dependent reweighting?

Matched protocol. We evaluate FJSP, CVRP, and TSP with fixed candidate generators and equal-cost exact decoding. Every objective uses the same family-specific cheap action features, a 512-candidate cap, an action-only width-128 depth-3 MLP, 1,200/240 train/validation instances, seeds $\{7, 17, 27, 37, 47\}$, optimization budget, validation rule, and deployment budgets $B \in \{4, 8\}$. Model selection is completed before held-out exact values are used. Primary comparisons use 300 held-out instances per family.

Comparators. The core baselines are SoftmaxListwise, ActionableValueRegression, and matched LambdaRank@B. We also include the Yang–Koyejo ψ_5 hinge and a differentiable top-k cross-entropy control instantiated with NeuralSort (DiffTopK–NS) as cutoff-aware controls (Yang & Koyejo, 2020; Petersen et al., 2022; Grover et al., 2019). WinnerClassification is retained only as an exploratory winner-only control and is not used for confirmatory graded-utility claims; equal-interval is the better-powered graded-utility intervention. We do not report a WinnerClassification held-out effect size because the retained evaluation artifact does not contain a complete auditable per-seed/per-instance table for that control, and its singleton-positive reference stratum is too sparse for a well-powered contrast. The equal-interval, no-normalizer, and fixed-B_0 objectives are component checks rather than additional baseline families.

Inference. Point estimates pair methods within held-out instance and training seed and then average across the five seeds. Confidence intervals use a two-level paired cluster bootstrap that resamples seeds and then paired instances within seed; for background on bootstrap inference with clustered data, see Field & Welsh (2007). Three family-level tests for the same claim are adjusted by Holm's sequentially rejective procedure (Holm, 1979). Exact intervals and p-values are reported in tables rather than repeated in the main text; full details are in Appendix B.

7 Results

Q1: comparison with matched baselines. QualityHit has positive point estimates against SoftmaxListwise and ActionableValueRegression in all 12 family–budget comparisons and against matched LambdaRank@B in all six family–budget cells (Table 1). Figure 2 summarizes the same

Table 1: QualityHit minus matched comparator on held-out BestGain. Values are paired means. The final column reports $100\,\mathbb{E}[G_B^{\mathrm{QH}} - G_B^{\mathrm{SL}}]/\mathbb{E}[a^\star - G_B^{\mathrm{SL}}]$, the fraction of SoftmaxListwise's mean remaining oracle headroom closed by QualityHit.

Family	B	Δ vs. SL	Δ vs. AVR	Δ vs. LambdaRank@B	Oracle-normalized vs. SL (%)
FJSP	4	+0.370	+0.404	+0.240	4.88
FJSP	8	+0.450	+0.449	+0.310	6.23
CVRP	4	+0.00460	+0.00600	+0.00310	11.83
CVRP	8	+0.00630	+0.00800	+0.00510	16.57
TSP	4	+0.00220	+0.00300	+0.00170	2.12
TSP	8	+0.00280	+0.00350	+0.00210	2.93

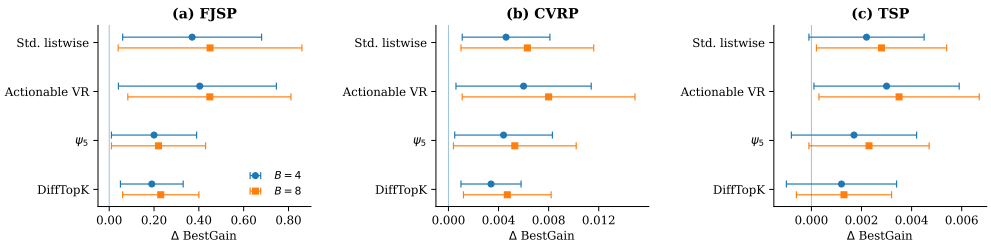

Figure 2: **Comparator summary.** Oracle-normalized point-estimate improvement of QualityHit over matched value, listwise, metric-driven, and cutoff-aware comparators. For comparator C, the normalization is $100\,\mathbb{E}[G_B^{\mathrm{QH}} - G_B^{C}]/\mathbb{E}[a^\star - G_B^{C}]$. Confidence intervals and hypothesis tests are reported in Appendix B.

comparisons together with the cutoff-aware ψ_5 and DiffTopK–NS controls on a common oracle-headroom scale. The positive direction is consistent across families and budgets, although the magnitude varies by family.

The matched LambdaRank@B comparison is the most direct metric-driven control because it is both graded and explicitly targeted to the deployment cutoff. Its two-level 95% interval excludes zero in five of six cells, with TSP at $B = 4$ as the exception. After Holm correction across the three families within each budget, all three $B = 8$ comparisons remain significant at $\alpha = 0.05$, whereas none of the $B = 4$ comparisons does. Exact intervals and adjusted p-values are reported in Appendix Table 8; full SL/AVR intervals and the ψ_5/DiffTopK–NS controls are in Appendix Tables 3 and 4.

Q2: objective-component checks. Replacing the observed utility intervals by equal widths reduces held-out BestGain in all six reported cells, with five intervals excluding zero (Table 2). Removing the full-set normalizer also reduces the point estimate in all six cells. These results make graded post-verification utility the clearest empirical contributor while showing that the normalizer also changes the learned solution.

The fixed-B_0 intervention isolates budget matching more narrowly by retaining the threshold-size normalizer but freezing its budget argument at $B_0 = 4$. At deployment $B = 8$, full QualityHit exceeds fixed-B_0 by $+0.248$ on FJSP, $+0.0136$ on CVRP, and $+0.0014$ on TSP; the first two remain significant after three-family Holm correction, while TSP does not (Appendix Table 5). Thus graded utility provides the strongest and most consistent improvement, with an additional budget-matching effect visible on FJSP and CVRP at $B = 8$.

Q3: threshold geometry in the observed pools. The budget-sensitive theory becomes nontrivial when quality thresholds induce sets of different sizes. The structural holdout exhibits this heterogeneity in all three families. Figure 3 shows the fraction of total positive utility-interval width attached to small quality sets $m_\ell \le 4$: 24.0% for FJSP, 68.1% for CVRP, and 88.3% for TSP. The median positive-set sizes are 36, 18, and 4, respectively, while the median numbers of distinct positive utility

Table 2: Objective-component ablations. Positive values favor full QualityHit. Brackets are seed×instance two-level paired-bootstrap 95% confidence intervals.

	QH − no normalizer		QH$_{B=4}$ − equal interval$_{B=4}$	
Family	deploy $B = 4$	deploy $B = 8$	deploy $B = 4$	deploy $B = 8$
FJSP	$+0.253$ $[0.018, 0.488]$	$+0.265$ $[0.021, 0.509]$	$+0.188$ $[0.012, 0.364]$	$+0.217$ $[0.014, 0.420]$
CVRP	$+0.0071$ $[0.0017, 0.0125]$	$+0.0121$ $[0.0076, 0.0166]$	$+0.0100$ $[0.0017, 0.0183]$	$+0.0083$ $[0.0003, 0.0163]$
TSP	$+0.0018$ $[0.0001, 0.0035]$	$+0.0022$ $[0.0010, 0.0034]$	$+0.0015$ $[0.0003, 0.0027]$	$+0.0018$ $[-0.0001, 0.0037]$

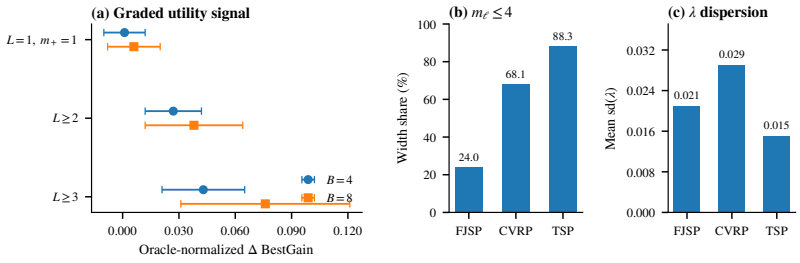

Figure 3: **Threshold-structure audit.** Fraction of total positive utility-interval width attached to quality sets with $m_\ell \leq 4$ on the deterministic reconstructed test pools. Annotations report the median positive-set size m_+ and median number L of distinct positive utility levels. Full audit statistics are in Appendix Tables 6 and 7.

levels are 6, 14, and 4. These distributions provide the heterogeneous threshold sizes required for the budget-dependent weights in Proposition 7 to act non-uniformly.

8 Limitations

The experiments cover three combinatorial families, one scorer architecture, fixed one-shot candidate pools, equal verification costs, and only $B \in \{4, 8\}$. Exact full-pool utilities are available offline for training; the study does not address label-efficient learning or adaptive acquisition.

A methodological limitation is the use of the smooth full-set relaxation. $\mathcal{U}_{\text{top}}^{(B)}$ is tighter but only piecewise smooth, whereas QualityHit optimizes the simpler smooth envelope $\mathcal{L}_{\text{hit}}^{(B)}$. The present experiments do not compare the empirical tightness of R_B, $\mathcal{U}_{\text{top}}^{(B)}$, and $\mathcal{L}_{\text{hit}}^{(B)}$, nor do they train directly with $\mathcal{U}_{\text{top}}^{(B)}$. Evaluating these alternatives would clarify the empirical trade-off between surrogate tightness and optimization smoothness.

9 Conclusion

When a learned score determines which candidates receive scarce exact evaluation, global predictive accuracy is not the deployed objective. Verification regret instead depends on which utility-threshold sets the verified prefix hits. This view yields a hierarchy from exact regret to a tighter top-subset envelope and then to the smooth QualityHit loss. On three combinatorial families, QualityHit has positive held-out point estimates against value, listwise, top-k, and matched metric-driven ranking controls; the strongest LambdaRank@B inference is at $B = 8$. The graded-utility ablation is supportive, while the normalizer experiments do not justify a clean causal story. The central takeaway is therefore the verification-specific objective construction, not a claim that one particular smooth upper bound is universally optimal.

AI USE STATEMENT

Generative AI tools were used for language editing, code debugging, and mathematical claims checking. The authors remain responsible for the mathematical claims, experimental protocol, citations, and interpretation.

REPRODUCIBILITY STATEMENT

The combinatorial experiments use a candidate cap of 512, an action-only MLP with hidden width 128 and depth 3, 1,200 training and 240 validation instances, seeds $7, 17, 27, 37, 47$, at most 80 epochs, learning rate 10^{-3}, weight decay 10^{-4}, and validation early stopping. Exact gains are offline labels and evaluation targets, never input features. Primary held-out comparisons use 300 instances per family. Point estimates average paired held-out differences within seed and then across the five training seeds. The LambdaRank@B control uses the same candidate pools, architecture, splits, seeds, optimization budget, and validation-only model selection, with actionable gains as relevance and an NDCG discount (Järvelin & Kekäläinen, 2002) truncated at B. Final confidence intervals use the two-level paired cluster bootstrap in Section 6. Additional implementation details and structural diagnostics are in Appendix B.

The source release also includes `scripts/envelope_tightness_diagnostic.py`, which computes $R_B, \mathcal{U}_{\text{top}}^{(B)}$, and $\mathcal{L}_{\text{hit}}^{(B)}$ exactly from retained per-candidate score/gain dumps. The no-data archive does not contain those score dumps.

REFERENCES

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd International Conference on Machine Learning*, pp. 89–96, 2005. doi: 10.1145/1102351.1102363. URL https://doi.org/10.1145/1102351.1102363.

Christopher J. Burges, Robert Ragno, and Quoc V. Le. Learning to rank with nonsmooth cost functions. In *Advances in Neural Information Processing Systems 19*, pp. 193–200, 2006. URL https://proceedings.neurips.cc/paper/2006/hash/af44c4c56f385c43f2529f9b1b018f6a-Abstract.html.

Zhe Cao, Tao Qin, Tie-Yan Liu, Ming-Feng Tsai, and Hang Li. Learning to rank: From pairwise approach to listwise approach. In *Proceedings of the 24th International Conference on Machine Learning*, pp. 129–136, 2007. doi: 10.1145/1273496.1273513. URL https://doi.org/10.1145/1273496.1273513.

Adam N. Elmachtoub and Paul Grigas. Smart "predict, then optimize". *Management Science*, 68 (1):9–26, 2022. doi: 10.1287/mnsc.2020.3922. URL https://doi.org/10.1287/mnsc.2020.3922.

C. A. Field and A. H. Welsh. Bootstrapping clustered data. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 69(3):369–390, 2007. doi: 10.1111/j.1467-9868.2007.00593.x. URL https://doi.org/10.1111/j.1467-9868.2007.00593.x.

Peter I. Frazier. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018. URL https://arxiv.org/abs/1807.02811.

Aditya Grover, Eric Wang, Aaron Zweig, and Stefano Ermon. Stochastic optimization of sorting networks via continuous relaxations. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=H1eSS3CcKX.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. URL https://www.jstor.org/stable/4615733.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002. doi: 10.1145/582415.582418. URL https://doi.org/10.1145/582415.582418.

Yaochu Jin. Surrogate-assisted evolutionary computation: Recent advances and future challenges. *Swarm and Evolutionary Computation*, 1(2):61–70, 2011. doi: 10.1016/j.swevo.2011.05.001. URL https://doi.org/10.1016/j.swevo.2011.05.001.

Donald R. Jones, Matthias Schonlau, and William J. Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998. doi: 10.1023/A:1008306431147. URL https://doi.org/10.1023/A:1008306431147.

Purushottam Kar, Harikrishna Narasimhan, and Prateek Jain. Surrogate functions for maximizing precision at the top. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pp. 189–198. PMLR, 2015. URL https://proceedings.mlr.press/v37/kar15.html.

Jayanta Mandi, Víctor Bucarey, Maxime Mulamba Ke Tchomba, and Tias Guns. Decision-focused learning: Through the lens of learning to rank. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 14935–14947. PMLR, 2022. URL https://proceedings.mlr.press/v162/mandi22a.html.

Felix Petersen, Hilde Kuehne, Christian Borgelt, and Oliver Deussen. Differentiable top-k classification learning. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 17656–17668. PMLR, 2022. URL https://proceedings.mlr.press/v162/petersen22a.html.

Łukasz Struski, Michal B. Bednarczyk, Igor T. Podolak, and Jacek Tabor. Lapsum – one method to differentiate them all: Ranking, sorting and top-k selection. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 56990–57007. PMLR, 2025. URL https://proceedings.mlr.press/v267/struski25a.html.

Xuanhui Wang, Cheng Li, Nadav Golbandi, Michael Bendersky, and Marc Najork. The LambdaLoss framework for ranking metric optimization. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, pp. 1313–1322. ACM, 2018. doi: 10.1145/3269206.3271784. URL https://doi.org/10.1145/3269206.3271784.

Bryan Wilder, Bistra Dilkina, and Milind Tambe. Melding the data-decisions pipeline: Decision-focused learning for combinatorial optimization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 1658–1665, 2019. doi: 10.1609/aaai.v33i01.33011658. URL https://doi.org/10.1609/aaai.v33i01.33011658.

Forest Yang and Sanmi Koyejo. On the consistency of top-k surrogate losses. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 10727–10735. PMLR, 2020. URL https://proceedings.mlr.press/v119/yang20f.html.

A Proofs and Supplementary Theory

This appendix contains complete proofs and supplementary results for Section 4. Results needed for the main argument are stated in the paper; longer comparative-statics conditions, transfer bounds, information limits, and auxiliary constructions are stated and proved here.

A.1 Proofs for prediction–decision separation

Proof overview. The construction misorders only B strict-utility pairs, yet those are exactly the pairs needed to push the unique unit-gain candidate outside the verified top-B set. The normalized full-pool errors therefore vanish with n while verification regret remains one.

Proof of Proposition 1. Let $a_1 = 1$, $a_i = 0$ for $i \geq 2$, and $\varepsilon_n = 1/n$. Define $\hat{a}_1 = 0$, $\hat{a}_i = \varepsilon_n$ for $2 \leq i \leq B+1$, and $\hat{a}_i = -\varepsilon_n$ otherwise. Exactly B zero-utility candidates outrank candidate

1, so $R_B = 1$. The only strict utility pairs compare candidate 1 with one of the $n-1$ zero-utility candidates, and exactly B are inverted. Hence

$$E_{\text{pair}} = \frac{B}{n-1}, \qquad E_p^p = \frac{1+(n-1)\varepsilon_n^p}{n} = \frac{1}{n} + \frac{n-1}{n^{p+1}}.$$

□

A.2 Technical lemmas for threshold envelopes

The main text states the verification-regret envelope. The lemmas below supply its two ingredients: a probability-mass cap for missed sets and a smooth secant–log upper envelope.

Lemma 3 (Probability-mass cap under a missed quality set)**.** *Let p be a probability vector whose ordering defines the deterministic Top-B set S_B. For any nonempty $H \subseteq [n]$, if $S_B \cap H = \varnothing$, then*

$$P_H := \sum_{i \in H} p_i \le \frac{|H|}{|H| + B}.$$

Proof overview. This lemma converts a missed set into a sharp probability-mass cap: every selected candidate must have probability at least as large as the largest missed member, so the selected mass and missed-set mass cannot jointly exceed one.

Proof. Let $m = |H|$ and $M = \max_{i \in H} p_i$. Every selected candidate has probability at least M, including under score ties, because Top-B contains only candidates no lower than any excluded candidate under the deterministic tie rule. Also $P_H \le mM$, hence $M \ge P_H/m$. Since S_B and H are disjoint,

$$1 \ge P_H + \sum_{j \in S_B} p_j \ge P_H + BM \ge P_H \left(1 + \frac{B}{m}\right),$$

which rearranges to the claim. □

Lemma 4 (Top-k subset mass extremum)**.** *Fix a nonempty set H of size m and a probability vector q. If $q^{\downarrow}_{H,1} \ge \cdots \ge q^{\downarrow}_{H,m}$ are the masses $\{q_i : i \in H\}$ in decreasing order, then for every $k \in \{1, \dots, m\}$,*

$$\max_{\substack{A \subseteq H \\ |A| = k}} \sum_{i \in A} q_i = \sum_{\ell=1}^{k} q^{\downarrow}_{H,\ell}.$$

Moreover, every missed k-subset has predicted mass at most $k/(k+B)$.

Proof overview. The first claim is the standard top-k extremum of a subset sum; the second then applies the missed-set mass cap to any k-subset. Together these facts identify the strongest partial-mass certificate inside a quality set.

Proof. For fixed k, replacing any selected index whose target mass is smaller than an unselected target mass cannot decrease the subset sum. Iterating yields the top-k partial sum. The predicted-mass statement is Lemma 3 applied to a set of cardinality k. □

Lemma 5 (Secant–log envelope)**.** *For $c \in (0,1)$, define ϕ_c as in Equation* (3)*, with $\phi_c(0) = 1/(1-c)$. Then*

$$\phi_c(z) = \begin{cases} \dfrac{1-z}{1-c}, & 0 \le z \le c, \\ \dfrac{-\log z}{-\log c}, & c \le z \le 1. \end{cases}$$

In particular, $z \le c$ implies $1 \le \phi_c(z) \le 1/(1-c)$, and for $z \in (0,1]$,

$$\phi_c(z) \le \frac{-\log z}{-\log c}.$$

Proof overview. The proof compares $-\log z$ with the secant joining its values at c and 1. This identifies which branch of ϕ_c is active and gives the logarithmic upper envelope used in the smooth QualityHit bound.

Proof. Let $f(z) = -\log z$. On $[c, 1]$, convexity of f places its graph below the secant joining $(c, f(c))$ and $(1, 0)$:

$$\frac{f(z)}{f(c)} \le \frac{1-z}{1-c}.$$

Hence the logarithmic branch is the minimum on $[c, 1]$. For $z \in (0, c]$, define

$$h(z) = \frac{-\log z}{-\log c} - \frac{1-z}{1-c}.$$

We have $h(c) = 0$ and

$$h'(z) = -\frac{1}{z(-\log c)} + \frac{1}{1-c} < 0,$$

because $z \le c$ and $-c \log c < 1 - c$ for $c \in (0, 1)$. Thus $h(z) \ge 0$ for $z \le c$, so the linear branch is the minimum there. The remaining statements follow immediately, including the continuous extension at $z = 0$. □

Proof overview. The proof works threshold by threshold. A missed quality set forces every top-k partial mass inside it below $k/(k + B)$; applying ϕ_c bounds the miss indicator, and integrating the layer-cake identity yields the top-subset envelope. Choosing the full set gives the smooth QualityHit envelope.

Proof of Theorem 2. Fix $t \in (0, a^\star]$. If the verified prefix hits H_t, then

$$\mathbf{1}\{G_B < t\} = 0$$

and every claimed pointwise upper bound is immediate. Suppose instead that the prefix misses H_t. For any $1 \le k \le m_t$, let $A_{t,k} \subseteq H_t$ contain the k largest probabilities p_i inside H_t. Since $A_{t,k}$ is also missed, Lemma 3 gives

$$P^{\downarrow}_{t,k} = \sum_{i \in A_{t,k}} p_i \le c_{B,k} = \frac{k}{k+B}.$$

Lemma 5 therefore yields $\phi_{c_{B,k}}(P^{\downarrow}_{t,k}) \ge 1$ for every k. Hence, pointwise in t,

$$\mathbf{1}\{G_B < t\} \le \min_{1 \le k \le m_t} \phi_{c_{B,k}}\left(P^{\downarrow}_{t,k}\right).$$

Using the layer-cake identity

$$R_B = a^\star - G_B = \int_0^{a^\star} \mathbf{1}\{G_B < t\}\, dt$$

and integrating proves

$$R_B \le \mathcal{U}^{(B)}_{\text{top}}(s; a).$$

For the second inequality, choose $k = m_t$ inside the minimum. Then $P^{\downarrow}_{t,m_t} = P_t$, $c_{B,m_t} = m_t/(m_t + B)$, and Lemma 5 gives

$$\begin{aligned}
\min_{1 \le k \le m_t} \phi_{c_{B,k}}(P^{\downarrow}_{t,k}) &\le \phi_{c_{B,m_t}}(P_t) \\
&\le \frac{-\log P_t}{-\log c_{B,m_t}} \\
&= \frac{-\log P_t}{\log(1 + B/m_t)}.
\end{aligned}$$

Integrating proves $\mathcal{U}^{(B)}_{\text{top}} \le \mathcal{L}^{(B)}_{\text{hit}}$. Multiplying the pointwise inequalities by arbitrary nonnegative w_B and summing proves the multi-budget statement.

For exact finite-sum evaluation, let $v_1 > \cdots > v_L > 0$ be the distinct positive values among a_i, set $v_{L+1} = 0$, and write $H_\ell = \{i : a_i \ge v_\ell\}$. Both H_t and all within-H_t probability partial sums are constant for $t \in (v_{\ell+1}, v_\ell]$. Therefore

$$\mathcal{U}^{(B)}_{\text{top}}(s;a) = \sum_{\ell=1}^{L} (v_\ell - v_{\ell+1}) \min_{1 \le k \le |H_\ell|} \phi_{c_{B,k}}\left(P^{\downarrow}_{\ell,k}\right),$$

while

$$\mathcal{L}^{(B)}_{\text{hit}}(s;a) = \sum_{\ell=1}^{L} (v_\ell - v_{\ell+1}) \frac{-\log \sum_{i \in H_\ell} p_i}{\log(1 + B/|H_\ell|)}.$$

The latter is differentiable in finite scores. The top-subset envelope is continuous and piecewise smooth; kinks can occur when the active k changes or when a partial mass crosses its cap. □

Lemma 6 (Rank-invariant scale envelope)**.** *For $p^{(\beta)} = \mathrm{softmax}(\beta s)$ with $\beta > 0$,*

$$R_B \le \inf_{\beta > 0} \mathcal{U}^{(B)}_{\text{top}}(\beta s; a) \le \mathcal{U}^{(B)}_{\text{top}}(s;a) \le \mathcal{L}^{(B)}_{\text{hit}}(s;a).$$

Proof overview. Positive score scaling preserves the deterministic ranking and therefore leaves verification regret unchanged. Applying the envelope theorem along this rank-equivalent orbit gives the stated scale refinement.

Proof of Lemma 6. For every $\beta > 0$, multiplying all scores by β preserves their deterministic ordering, so the deployed verified prefix and R_B are unchanged. Theorem 2 applied to βs gives

$$R_B \le \mathcal{U}^{(B)}_{\text{top}}(\beta s; a) \qquad \text{for every } \beta > 0.$$

Taking the infimum over β, then evaluating the infimum at $\beta = 1$, gives the first two inequalities. The final inequality is Theorem 2. □

A.3 Proofs for QualityHit geometry and finite-capacity budget sensitivity

For distinct positive utility levels $v_1 > \cdots > v_L > 0$, set $v_{L+1} = 0$, $H_\ell = \{i : a_i \ge v_\ell\}$, $m_\ell = |H_\ell|$, $P_\ell = \sum_{i \in H_\ell} p_i$, and $\Delta_\ell = v_\ell - v_{\ell+1}$. Then

$$\mathcal{L}^{(B)}_{\text{hit}} = \sum_{\ell=1}^{L} \frac{\Delta_\ell}{\log(1 + B/m_\ell)} (-\log P_\ell). \tag{6}$$

Proposition 7 (Budget changes relative certificate weights)**.** *For $m_a < m_b$, the ratio*

$$r_{ab}(B) = \frac{1/\log(1 + B/m_a)}{1/\log(1 + B/m_b)} = \frac{\log(1 + B/m_b)}{\log(1 + B/m_a)}$$

is strictly increasing for $B > 0$, from m_a/m_b as $B \downarrow 0$ toward 1 as $B \to \infty$. If all active threshold sets have the same size, changing B is only a common positive rescaling.

Proof overview. The proof first partitions the layer-cake integral over intervals on which the threshold set is constant, yielding the finite-sum weights. It then differentiates the ratio of two budget normalizers and proves monotonicity through an elementary increasing function.

Proof of Proposition 7. Let $v_1 > \cdots > v_L > 0$ be the distinct positive values and set $v_{L+1} = 0$. For every $t \in (v_{\ell+1}, v_\ell]$, the threshold set is exactly $H_t = H_\ell$, so $m_t = m_\ell$ and $P_t = P_\ell$. Splitting Equation (4) across these intervals gives

$$\mathcal{L}^{(B)}_{\text{hit}} = \sum_{\ell=1}^{L} (v_\ell - v_{\ell+1}) \frac{-\log P_\ell}{\log(1 + B/m_\ell)},$$

which is Equation (6).

For two set sizes $m_a < m_b$, write

$$r_{ab}(B) = \frac{\log(1 + B/m_b)}{\log(1 + B/m_a)}.$$

Put $x = B/m_b$ and $y = B/m_a$, so $0 < x < y$. Direct differentiation shows $r'_{ab}(B) > 0$ exactly when

$$\frac{(1+y)\log(1+y)}{y} > \frac{(1+x)\log(1+x)}{x}.$$

The function $h(z) = (1+z)\log(1+z)/z$ is strictly increasing because

$$h'(z) = \frac{z - \log(1+z)}{z^2} > 0 \qquad (z > 0).$$

Hence r_{ab} is strictly increasing. The endpoint limits are

$$r_{ab}(B) = \frac{m_a}{m_b}\left[1 + \frac{B}{2}\left(\frac{1}{m_a} - \frac{1}{m_b}\right) + O(B^2)\right] \quad (B \downarrow 0),$$

and

$$r_{ab}(B) = 1 - \frac{\log(m_b/m_a)}{\log B} + O\left(\frac{1}{(\log B)^2} + \frac{1}{B \log B}\right) \quad (B \to \infty).$$

Strict monotonicity and continuity imply a unique crossing of every $c \in (m_a/m_b, 1)$.

If every active set has the same size m, then all terms share the factor $1/\log(1 + B/m)$, so changing B multiplies the full loss and its score gradient by one positive scalar. The singleton-positive case is $m = 1$. □

Corollary 8 (Common zero-loss target under winner concentration)**.** *If there exists a sequence of score vectors whose induced softmax mass on an actionable maximizer tends to one, then* $\inf_s \mathcal{L}^{(B)}_{\text{hit}}(s; a) = 0$ *for every* $B > 0$.

Proof overview. A single winner-concentrating score sequence lies in every positive threshold set. Along that sequence all threshold masses approach one simultaneously, so every budget-specific QualityHit objective approaches the same zero-loss limit.

Proof of Corollary 8. Let $i^\star$ be any actionable maximizer. If a sequence of score distributions satisfies $p_{i^\star} \to 1$, then every positive threshold set contains $i^\star$, so $P_\ell \to 1$ for every ℓ. Each term in Equation (6) therefore tends to zero for every fixed $B > 0$. Nonnegativity gives $\inf_s \mathcal{L}^{(B)}_{\text{hit}}(s; a) = 0$. The same winner-concentrating sequence works for every budget, so the zero-loss limit is shared across B. □

Corollary 9 (Deterministic max-consistency)**.** *If the utility vector has a unique maximizer* $y^\star$ *with gap* $\delta^\star = a_{y^\star} - \max_{j \neq y^\star} a_j > 0$*, then*

$$\mathcal{L}^{(B)}_{\text{hit}}(s; a) \geq \frac{\delta^\star}{\log(1+B)}(-\log p_{y^\star}). \tag{7}$$

Consequently, pointwise QualityHit loss tending to zero forces $p_{y^\star} \to 1$ *and eventually places* $y^\star$ *in every fixed top-*k *prefix.*

Proof overview. The highest nonzero utility interval isolates the unique maximizer as a singleton quality set. Its QualityHit term lower-bounds $-\log p_{y^\star}$, so vanishing loss forces the maximizer probability to one and eventually places it in every fixed top-k prefix.

Proof of Corollary 9. Suppose the deterministic utility vector has a unique maximizer $y^\star$ with gap $\delta^\star = a_{y^\star} - \max_{j \neq y^\star} a_j > 0$. The highest threshold set is the singleton $\{y^\star\}$, hence

$$\mathcal{L}^{(B)}_{\text{hit}}(s; a) \geq \frac{\delta^\star}{\log(1+B)}(-\log p_{y^\star}).$$

If $y^\star \notin \mathrm{TopK}_k(s)$, at least k candidates have score no smaller than $s_{y^\star}$, so $p_{y^\star} \le 1/(k+1)$ and

$$\mathcal{L}_{\mathrm{hit}}^{(B)}(s;a) \ge \delta^\star \frac{\log(k+1)}{\log(1+B)}.$$

Thus pointwise QualityHit loss tending to its infimum zero eventually places $y^\star$ in every fixed top-k prefix. This is the deterministic max-oriented statement used in the main text; it is not a claim of general stochastic multiclass top-k calibration. □

A.3.1 Comparative statics of the restricted population optimum

For a parametric scorer s_θ with $\theta \in \Theta \subset \mathbb{R}^d$, define

$$J_B(\theta) = \mathbb{E}_x\left[\sum_{\ell=1}^{L(x)} w_\ell(B,x)\,\ell_\ell(\theta;x)\right], \qquad \ell_\ell(\theta;x) = -\log P_\ell(s_\theta(x)).$$

Theorem 10 (Restricted-optimum comparative statics)**.** *Let $I \subset (0,\infty)$ be open and assume Θ is open. Suppose J_B is coercive and strictly convex in θ, twice continuously differentiable in θ and once in B, differentiation may pass through the expectation, and $\mathbf{H}_B = \nabla^2_{\theta\theta} J_B(\theta^\star_B)$ is positive definite at the unique minimizer for each $B \in I$. Then $B \mapsto \theta^\star_B$ is continuously differentiable and*

$$\frac{d\theta^\star_B}{dB} = \mathbf{H}_B^{-1}\mathbb{E}\left[\sum_\ell \lambda_{m_\ell}(B) w_\ell(B,x) g_\ell(\theta^\star_B;x)\right], \tag{8}$$

where $g_\ell = \nabla_\theta \ell_\ell$ and $\lambda_m(B) = 1/[(B+m)\log(1+B/m)]$.

For compact notation define

$$g_\ell(\theta;x) = \nabla_\theta \ell_\ell(\theta;x), \qquad \ell_\ell(\theta;x) = -\log P_\ell(s_\theta(x)).$$

The QualityHit weight obeys

$$w'_\ell(B,x) = -\frac{\Delta_\ell(x)}{(B+m_\ell(x))\log^2(1+B/m_\ell(x))} = -\lambda_{m_\ell}(B) w_\ell(B,x), \tag{9}$$

where

$$\lambda_m(B) = \frac{1}{(B+m)\log(1+B/m)}.$$

Proof overview. The proof applies the implicit-function theorem to the first-order condition of the restricted population risk. Differentiating that condition with respect to budget exposes the exact Hessian-preconditioned gradient imbalance that drives optimizer motion.

Proof of Theorem 10. For fixed B, coercivity and continuity imply existence of a minimizer, and strict convexity makes it unique. Since the minimizer is interior,

$$F(B,\theta^\star_B) := \nabla_\theta J_B(\theta^\star_B) = 0.$$

The Jacobian of F with respect to θ is $\mathbf{H}_B$, which is invertible by positive definiteness. The implicit function theorem therefore gives a continuously differentiable local branch through every $B \in I$. Uniqueness makes these local branches agree on overlaps, yielding one C^1 path on I. Differentiating the first-order condition gives

$$0 = \frac{d}{dB}F(B,\theta^\star_B) = \mathbf{H}_B \frac{d\theta^\star_B}{dB} + \nabla^2_{B\theta} J_B(\theta^\star_B).$$

By differentiation under the expectation and Equation (9),

$$\nabla^2_{B\theta} J_B(\theta^\star_B) = \mathbb{E}\left[\sum_\ell w'_\ell(B,x) g_\ell(\theta^\star_B;x)\right] = -\mathbb{E}\left[\sum_\ell \lambda_{m_\ell}(B) w_\ell(B,x) g_\ell(\theta^\star_B;x)\right].$$

Multiplication by $-\mathbf{H}_B^{-1}$ proves Equation (8). Since $\mathbf{H}_B^{-1}$ is nonsingular, the derivative is nonzero exactly when the weighted gradient vector in that equation is nonzero. □

Local nonconvex form. Global strict convexity is not essential for the derivative formula. If $\bar{\theta}$ is an isolated stationary local minimum at budget $\bar{B}$ and $\nabla^2_{\theta\theta} J_{\bar{B}}(\bar{\theta})$ is positive definite, the same implicit-function argument gives a unique local C^1 branch of stationary local minima and Equation (8) in a neighborhood of $\bar{B}$. This is the form relevant to a stable basin of a nonconvex model; it does not claim that optimization will select the same basin at every budget.

Corollary 11 (Grouped-gradient criterion)**.** *At $\theta^\star_B$, let*

$$q_m(B) = \mathbb{E}\left[\sum_{\ell: m_\ell = m} w_\ell(B, x) g_\ell(\theta^\star_B; x)\right].$$

Then $\sum_m q_m(B) = 0$ and, for any scalar $\bar{\lambda}$,

$$\frac{d\theta^\star_B}{dB} = \mathbf{H}_B^{-1} \sum_m (\lambda_m(B) - \bar{\lambda}) q_m(B). \tag{10}$$

Moreover $\lambda_m(B)$ is strictly increasing in m. With exactly two active sizes $m_a \neq m_b$, opposing nonzero group gradients imply nonzero local budget drift.

Proof overview. Grouping first-order terms by threshold-set size and centering their budget decay rates removes the common component using the stationarity condition. The remaining weighted conflict gives the optimizer drift and shows that even opposing one-dimensional gradients are sufficient.

Proof of Corollary 11. The first-order condition is

$$0 = \nabla_\theta J_B(\theta^\star_B) = \sum_m q_m(B).$$

Equation (8) and $w' = -\lambda_m w$ give

$$\frac{d\theta^\star_B}{dB} = \mathbf{H}_B^{-1} \sum_m \lambda_m(B) q_m(B).$$

Subtracting $\bar{\lambda} \sum_m q_m = 0$ proves Equation (10).

To order the decay rates, let $x = B/m$. Then

$$\frac{1}{\lambda_m(B)} = (B + m)\log(1 + B/m) = B\frac{(1+x)\log(1+x)}{x} = Bh(x).$$

The function h is strictly increasing, as shown above. Increasing m decreases $x = B/m$, hence decreases $h(x)$ and therefore increases $\lambda_m(B)$. With exactly two active sizes, the first-order condition gives $q_{m_b} = -q_{m_a}$. Hence

$$\frac{d\theta^\star_B}{dB} = (\lambda_{m_a} - \lambda_{m_b}) \mathbf{H}_B^{-1} q_{m_a},$$

which is nonzero whenever $q_{m_a} \neq 0$ and $m_a \neq m_b$. □

Margin-level prediction. For any differentiable score margin $M_{ij}(\theta; x) = s_\theta(x, c_i) - s_\theta(x, c_j)$,

$$\frac{d}{dB} M_{ij}(\theta^\star_B; x) = \nabla_\theta M_{ij}(\theta^\star_B; x)^\top \mathbf{H}_B^{-1} \sum_m \lambda_m(B) q_m(B).$$

Thus the theorem makes a directly testable local prediction: estimate the grouped gradient contributions and a Hessian-vector product, then compare the predicted sign of the margin drift with finite differences across nearby training budgets. Endpoint margins of opposite sign imply at least one ranking crossover by continuity; a fixed nonzero derivative sign on the interval makes that crossover unique.

A.3.2 A CLOSED-FORM NON-REALIZABLE POPULATION CROSSOVER

Proposition 12 (A restricted optimum can reverse across budgets)**.** *There exists a one-dimensional, ten-candidate, non-realizable QualityHit population problem with a strictly convex risk whose unique optimizer satisfies*

$$\theta_B^\star < 0 \ (B < 6), \qquad \theta_6^\star = 0, \qquad \theta_B^\star > 0 \ (B > 6).$$

Hence the population-optimal restricted ranking at $B = 4$ is reversed at $B = 8$.

Proof overview. The construction reduces the population risk to a one-dimensional strictly convex function. Evaluating its derivative at the ranking boundary shows that the unique minimizer changes sign exactly when the normalizer ratio crosses a prescribed level, producing the $B = 4$ versus $B = 8$ ranking reversal.

Proof of Proposition 12. Let $\rho(B) = \log(1 + B)/\log(1 + B/2)$, set $\kappa = (8/9)\rho(6)$, and choose mixture probabilities $\pi_A = \kappa/(1 + \kappa)$ and $\pi_C = 1/(1 + \kappa)$. There are two instance types, each with ten candidates. Type A has utilities $(1, 0, \dots, 0)$ and scores $(\theta, 0, \dots, 0)$, so its only QualityHit layer has $m = 1$ and

$$-\log P_A(\theta) = \log(1 + 9e^{-\theta}).$$

Type C has utilities $(1, 1, 0, \dots, 0)$ and scores $(-\theta, -\theta, 0, \dots, 0)$, so its only layer has $m = 2$ and

$$-\log P_C(\theta) = \log(1 + 4e^{\theta}).$$

With these mixture probabilities, put

$$\alpha_B = \frac{\pi_A}{\log(1 + B)}, \qquad \beta_B = \frac{\pi_C}{\log(1 + B/2)}.$$

Then

$$J_B(\theta) = \alpha_B \log(1 + 9e^{-\theta}) + \beta_B \log(1 + 4e^{\theta}). \tag{11}$$

The risk tends to $+\infty$ as $\theta \to \pm\infty$, and

$$J_B''(\theta) = \alpha_B \frac{9e^{\theta}}{(e^{\theta} + 9)^2} + \beta_B \frac{4e^{\theta}}{(1 + 4e^{\theta})^2} > 0,$$

so there is one global minimizer.

Writing $z = e^{\theta}$, the first-order condition is

$$-\alpha_B \frac{9}{z + 9} + \beta_B \frac{4z}{1 + 4z} = 0,$$

or

$$4\beta_B z^2 + 36(\beta_B - \alpha_B)z - 9\alpha_B = 0.$$

The positive root is

$$e^{\theta_B^\star} = \frac{9(\alpha_B - \beta_B) + 3\sqrt{9(\alpha_B - \beta_B)^2 + \alpha_B\beta_B}}{2\beta_B}. \tag{12}$$

For the sign of the minimizer it is simpler to evaluate the derivative at the ranking boundary:

$$J_B'(0) = -\frac{9}{10}\alpha_B + \frac{4}{5}\beta_B = \frac{\beta_B}{10}\left(8 - 9\frac{\alpha_B}{\beta_B}\right).$$

Strict convexity implies $\theta_B^\star < 0$, $= 0$, or > 0 according as α_B/β_B is $< 8/9$, $= 8/9$, or $> 8/9$. Now

$$\frac{\alpha_B}{\beta_B} = \kappa\frac{\log(1 + B/2)}{\log(1 + B)} = \frac{\kappa}{\rho(B)}, \qquad \rho(B) = \frac{\log(1 + B)}{\log(1 + B/2)}.$$

By Proposition 7, ρ is strictly decreasing. The chosen $\kappa = (8/9)\rho(6)$ therefore gives a unique crossing at $B^\star = 6$. Numerically,

$$\theta_4^\star \approx -0.140, \qquad \theta_8^\star \approx 0.094.$$

When $\theta < 0$, the type-A positive candidate is rank 10 while the two type-C positives are ranks 1–2. When $\theta > 0$, the type-A positive is rank 1 and the two type-C positives are ranks 9–10. Thus the population-optimal restricted ranking itself reverses between training budgets 4 and 8. The two types require opposite limits $\theta \to +\infty$ and $\theta \to -\infty$ to drive their losses to zero, so the family is genuinely non-realizable. □

A.3.3 MATCHED-BUDGET RESTRICTED CALIBRATION WITNESS

For a finite restricted class $\mathcal{F}_r$, let

$$\mathcal{E}_B^{\mathrm{task}}(f) = \mathfrak{R}_B(f) - \min_{h\in\mathcal{F}_r} \mathfrak{R}_B(h)$$

be restricted task excess regret, and define the matched restricted calibration function

$$\delta_B^{\mathcal{F}_r}(\epsilon) = \inf\left\{ \mathfrak{L}_B(f) - \min_{h\in\mathcal{F}_r} \mathfrak{L}_B(h) : f\in\mathcal{F}_r,\ \mathcal{E}_B^{\mathrm{task}}(f) \geq \epsilon \right\}.$$

Using restricted *excess* task regret rather than absolute regret avoids a trivial zero caused by unavoidable approximation error. For a training budget β evaluated at deployment budget B, define $\delta_{\beta\to B}^{\mathcal{F}_r}$ analogously with surrogate excess measured by $\mathfrak{L}_\beta$ and task excess measured by $\mathcal{E}_B^{\mathrm{task}}$.

Proposition 13 (Matched-budget restricted calibration gap)**.** *There exists a single ten-candidate pool and a two-element restricted class $\mathcal{F}_r = \{f_T, f_L\}$ such that f_T is both the QualityHit and task optimum at $B = 4$, while f_L is both optima at $B = 8$. For positive task-excess ranges, the matched-budget calibration functions are strictly positive while the corresponding cross-budget calibration functions are zero.*

Proof overview. The proof constructs two fixed hypotheses whose task ordering and QualityHit ordering switch at the same budget threshold. The loss difference is an explicit function of the normalizer ratio, so matched-budget restricted calibration is positive while the cross-budget calibration gap collapses to zero.

Proof of Proposition 13. Let the utilities be $(1, u, 0, \ldots, 0)$ with

$$u = \left[1 + \frac{\log 7}{\log 4}\frac{\log(42/17)}{\log 8}\right]^{-1} \approx 0.620913.$$

Take score distributions (equivalently scores $s_i = \log p_i$)

$$p_1^T = \frac{1}{100},\quad p_2^T = \frac{1}{5},\quad p_3^T = \cdots = p_9^T = \frac{157}{1400},\quad p_{10}^T = \frac{1}{200},$$

and

$$p_1^L = \frac{2}{25},\quad p_2^L = \frac{1}{200},\quad p_3^L = \cdots = p_6^L = \frac{9}{50},\quad p_7^L = \cdots = p_{10}^L = \frac{39}{800}.$$

Both vectors are strictly positive and sum to one. Under f_T, candidate 2 has rank 1 and candidate 1 has rank 9. Under f_L, candidate 1 has rank 5 and candidate 2 has rank 10. Hence

$$G_4(f_T) = G_8(f_T) = u, \qquad G_4(f_L) = 0, \qquad G_8(f_L) = 1.$$

Thus f_T uniquely minimizes restricted task regret at $B = 4$, with the wrong hypothesis having task excess u, while f_L uniquely minimizes it at $B = 8$, with the wrong hypothesis having task excess $1 - u$.

There are two QualityHit layers: $H_1 = \{1\}$ and $H_2 = \{1, 2\}$. Since

$$\frac{p_1^L}{p_1^T} = 8, \qquad \frac{p_1^T + p_2^T}{p_1^L + p_2^L} = \frac{42}{17},$$

the surrogate-risk difference is

$$D_B := \mathfrak{L}_B(f_L) - \mathfrak{L}_B(f_T) = -\frac{(1-u)\log 8}{\log(1+B)} + \frac{u\log(42/17)}{\log(1+B/2)}. \tag{13}$$

The definition of u makes

$$\frac{(1-u)\log 8}{u\log(42/17)} = \frac{\log 7}{\log 4} = \rho(6).$$

Since $\rho(B) = \log(1+B)/\log(1+B/2)$ is strictly decreasing, $D_B > 0$ for $B < 6$, $D_6 = 0$, and $D_B < 0$ for $B > 6$. In particular,

$$D_4 \approx 0.021388, \qquad -D_8 \approx 0.009831.$$

Therefore

$$\delta_4^{\mathcal{F}_r}(\epsilon) = D_4 \quad (0 < \epsilon \leq u), \qquad \delta_8^{\mathcal{F}_r}(\epsilon) = -D_8 \quad (0 < \epsilon \leq 1-u).$$

For the mismatched loss, f_L is the $B = 8$ surrogate minimizer while it has positive task excess u at deployment budget 4, so the defining infimum already contains a zero-surrogate-excess hypothesis and $\delta_{8\to 4}^{\mathcal{F}_r} = 0$ on $(0, u]$. The $4 \to 8$ statement is symmetric. □

A.3.4 ADDITIONAL FIXED-SCORE SANITY CHECK

The following construction is weaker than Proposition 12: it changes the loss ordering of two fixed nonoptimal distributions without optimizing over a restricted family. It is included only as a minimal illustration of the same weight-ratio mechanism.

Supporting construction: fixed-score loss-order reversal. For any budgets $1 \le B_1 < B_2$, there exist a multilevel utility vector and two fixed non-optimal score distributions whose QualityHit loss ordering differs between B_1 and B_2.

Proof overview. This weaker construction keeps two score distributions fixed and uses the monotone budget-normalizer ratio to reverse only their QualityHit loss ordering. It isolates the reweighting mechanism without making a claim about the optimizer of a restricted class.

Verification of the supporting construction. Fix integer budgets $1 \le B_1 < B_2$ and a pool size $n > B_2$. Define

$$\rho(B) = \frac{\log(1+B)}{\log(1+B/2)}.$$

Proposition 7 implies that ρ is strictly decreasing. Pick any

$$\kappa \in \big(\rho(B_2), \rho(B_1)\big).$$

Consider two utility levels $u_1 = 1$ and $u_2 = 1/(1+\kappa)$, so $(u_1 - u_2)/u_2 = \kappa$. Choose any $c > 0$, set $y = 1/4$, and choose $x = ye^{-2c}/2$. Construct two strictly positive score distributions p and $\tilde{p}$ by

$$\tilde{p}_1 = x, \qquad \tilde{p}_1 + \tilde{p}_2 = y, \qquad p_1 = xe^{c}, \qquad p_1 + p_2 = ye^{-c},$$

and distribute the remaining probability mass positively over candidates $3, \dots, n$. Then

$$\log \frac{p_1}{\tilde{p}_1} = c, \qquad \log \frac{\tilde{p}_1 + \tilde{p}_2}{p_1 + p_2} = c.$$

Equation (6) yields

$$\mathcal{L}^{(B)}_{\mathrm{hit}}(p) - \mathcal{L}^{(B)}_{\mathrm{hit}}(\tilde{p}) = u_2 c \left[-\frac{\kappa}{\log(1+B)} + \frac{1}{\log(1+B/2)} \right].$$

This difference is negative exactly when $\kappa > \rho(B)$, giving the reversal. □

Lemma 14 (Exact QualityHit score gradient)**.** *With the notation of Proposition 7 and* $p = \mathrm{softmax}(s)$*,*

$$\frac{\partial \mathcal{L}^{(B)}_{\mathrm{hit}}}{\partial s_j} = p_j \sum_{\ell=1}^{L} w_\ell(B) - p_j \sum_{\ell : j \in H_\ell} \frac{w_\ell(B)}{P_\ell}.$$

Thus a candidate is promoted by every quality threshold whose nested set contains it, not only by the winner threshold.

Proof overview. Differentiating $-\log P_H$ through the softmax gives a closed-form candidate gradient. Summing this identity over nested quality sets shows exactly which thresholds promote each candidate.

Proof. For any fixed nonempty H with $P_H = \sum_{i \in H} p_i$,

$$\frac{\partial(-\log P_H)}{\partial s_j} = p_j - \mathbf{1}\{j \in H\} \frac{p_j}{P_H}.$$

Apply this identity termwise to Equation (6). □

A.3.5 Degenerate and asymptotic sanity checks

The comparative-statics theorem reduces correctly in the main edge cases.

- As $B \downarrow 0$, $w_\ell(B) = \Delta_\ell[m_\ell/B + 1/2 + O(B/m_\ell)]$. Up to the common factor $1/B$, very small budgets emphasize broader threshold sets in proportion to m_ℓ.
- As $B \to \infty$, $w_\ell(B) = \Delta_\ell(\log B)^{-1}[1 + (\log m_\ell)/(\log B) + o(1/\log B)]$. Thus the normalizer differences vanish asymptotically and only the interval widths remain at leading order.
- If all active m_ℓ are equal, all λ_m are equal and Equation (10) is exactly zero. In particular, an all-singleton-positive population has a budget-independent restricted minimizer.
- If $\Delta_\ell = 0$, that layer contributes neither risk nor drift.
- Collinearity alone is not a zero-drift condition. In one dimension, opposing scalar group gradients can balance at the optimum while distinct λ_m make their budget derivatives fail to balance; Proposition 12 is exactly such a case.
- In a fully realizable class, one common winner-concentrating sequence drives every layer loss to zero for every budget. The finite-interior-minimizer assumptions of Theorem 10 need not hold in that limit, so there is no contradiction with Corollary 8.

A.4 Population transfer and information limits

For a scorer class $\mathcal{F}$, define

$$\mathfrak{R}_B(f) = \mathbb{E}R_B(f), \qquad \mathfrak{L}_B(f) = \mathbb{E}\mathcal{L}_{\mathrm{hit}}^{(B)}(f),$$
$$\mathfrak{R}_B^\star = \inf_{f\in\mathcal{F}} \mathfrak{R}_B(f), \qquad \mathfrak{L}_B^\star = \inf_{f\in\mathcal{F}} \mathfrak{L}_B(f), \qquad \Delta_{\mathcal{F}} = \mathfrak{L}_B^\star - \mathfrak{R}_B^\star \geq 0.$$

Theorem 15 (Oracle and finite-sample excess-regret transfer)**.** *For every $f \in \mathcal{F}$,*

$$\mathfrak{R}_B(f) - \mathfrak{R}_B^\star \leq \mathfrak{L}_B(f) - \mathfrak{L}_B^\star + \Delta_{\mathcal{F}}. \tag{14}$$

If $0 \leq a_i \leq A$, every pool has n candidates, $|s_i^f| \leq S$, $\mathcal{F}$ is finite, and $\hat{f}$ is an η-approximate empirical minimizer over N i.i.d. pools, then with probability at least $1-\delta$,

$$\mathfrak{R}_B(\hat{f}) - \mathfrak{R}_B^\star \leq \Delta_{\mathcal{F}} + \eta + 2M_{B,n,S,A}\sqrt{\frac{\log(2|\mathcal{F}|/\delta)}{2N}}, \tag{15}$$

where $M_{B,n,S,A} = A(2S + \log n)/\log(1 + B/n)$.

Proof overview. The oracle inequality follows by taking expectations of the pointwise QualityHit envelope and adding the approximation gap. The finite-class statement then bounds the loss range from score and utility bounds and applies Hoeffding plus a union bound.

Proof of Theorem 15. The pointwise inequality $R_B(f;Z) \leq \mathcal{L}_{\mathrm{hit}}^{(B)}(f;Z)$ follows from Theorem 2. Taking expectations gives $\mathfrak{R}_B(f) \leq \mathfrak{L}_B(f)$ and therefore $\mathfrak{R}_B^\star \leq \mathfrak{L}_B^\star$. For any $f \in \mathcal{F}$,

$$\begin{aligned}\mathfrak{R}_B(f) - \mathfrak{R}_B^\star &\leq \mathfrak{L}_B(f) - \mathfrak{R}_B^\star \\ &= \big[\mathfrak{L}_B(f) - \mathfrak{L}_B^\star\big] + \big[\mathfrak{L}_B^\star - \mathfrak{R}_B^\star\big],\end{aligned}$$

which proves Equation (14).

For the finite-sample statement, $|s_i^f| \leq S$ implies $p_i^f \geq e^{-2S}/n$. Every nonempty threshold set therefore has $P_t \geq e^{-2S}/n$, so $-\log P_t \leq 2S + \log n$. Since $m_t \leq n$ and $a^\star \leq A$,

$$0 \leq \mathcal{L}_{\mathrm{hit}}^{(B)}(f;Z) \leq M_{B,n,S,A} := \frac{A(2S + \log n)}{\log(1 + B/n)}.$$

Let $\widehat{\mathfrak{L}}_B$ denote the empirical mean. Hoeffding's inequality and a union bound over finite $\mathcal{F}$ give, with probability at least $1-\delta$,

$$\sup_{f\in\mathcal{F}} \left|\widehat{\mathfrak{L}}_B(f) - \mathfrak{L}_B(f)\right| \leq M_{B,n,S,A}\sqrt{\frac{\log(2|\mathcal{F}|/\delta)}{2N}} =: \varepsilon_N.$$

If $\hat{f}$ is an η-approximate empirical minimizer, then $\mathfrak{L}_B(\hat{f}) \le \mathfrak{L}_B^\star + \eta + 2\varepsilon_N$. Substitution into Equation (14) proves Equation (15). □

Corollary 16 (Observable large-margin realizability is sufficient)**.** *Suppose $0 \le a_i \le A$ and every pool has n candidates. If for every $K > 0$ the class contains a scorer f_K and an observable optimal index $i^\star(x) \in \arg\max_i a_i(x)$ such that*

$$s^{f_K}(x, c_{i^\star(x)}) - \max_{j \neq i^\star(x)} s^{f_K}(x, c_j) \ge K \quad \textit{almost surely},$$

then $\mathfrak{L}_B^\star = \mathfrak{R}_B^\star = 0$ and $\Delta_{\mathcal{F}} = 0$.

Proof. The margin condition gives $p_{i^\star}^{f_K} \ge [1 + (n-1)e^{-K}]^{-1}$. Every positive threshold set contains $i^\star$, hence

$$\mathcal{L}_{\text{hit}}^{(B)}(f_K; Z) \le \frac{A \log(1 + (n-1)e^{-K})}{\log(1 + B/n)} \longrightarrow 0$$

uniformly in Z. The transfer theorem gives the remaining claims. □

Proposition 17 (Exact-access floor under indistinguishable candidates)**.** *If m candidates are observationally indistinguishable before verification, exactly one uniformly random candidate has utility $\Delta > 0$, and $B < m$, then every one-shot policy satisfies*

$$\Pr(\textit{verify the valuable candidate}) \le B/m, \qquad \mathbb{E}[R_B] \ge \Delta(1 - B/m).$$

Proof. Let J be the uniformly random valuable index and S the verified set chosen from pre-verification information. Observational indistinguishability makes the law of S independent of J. Since $|S| \le B$,

$$\Pr(J \in S) = \frac{1}{m} \sum_{j=1}^{m} \Pr(j \in S) = \frac{\mathbb{E}|S|}{m} \le \frac{B}{m}.$$

The verified utility is Δ only upon a hit, giving the regret floor. □

B Core Experimental Protocol

Matched configuration. The matched protocol uses a candidate cap of 512; an action-only MLP with hidden width 128 and depth 3; 1,200 training and 240 validation instances; seeds $7, 17, 27, 37, 47$; an 80-epoch cap; learning rate 10^{-3}; weight decay 10^{-4}; and validation early stopping. All objectives use the same family-specific cheap pre-verification action features and candidate interface. Exact gains are labels only.

Fresh holdouts. All core empirical comparisons use five seeds and complete model selection before using held-out exact test values. Tuned SoftmaxListwise and tuned ActionableValueRegression each use 300 fresh instances per family for FJSP, CVRP, and TSP. The structural holdout uses 300 fresh instances per family. Final uncertainty uses a two-level paired cluster bootstrap: training seeds are resampled as clusters and held-out instances are resampled within each sampled seed.

Structural interventions. The equal-interval control replaces true utility-interval widths by equal widths while keeping the same nested threshold sets. The no-normalizer ablation removes $1/\log(1 + B/m_t)$ entirely, while fixed-B_0 retains the same form but freezes its budget argument at $B_0 = 4$. With $B_0 = 4$, the $B = 4$ QualityHit and fixed-B_0 objectives are identical. Because each altered objective is retrained, the resulting held-out differences are reported as sensitivity checks rather than as an additive or causal decomposition of the normalizer.

The fixed-B_0 effect at $B = 8$ is close to the no-normalizer effect on FJSP and CVRP, despite being the narrower analytic perturbation. Because both comparisons retrain nonlinear models, this pattern does not identify the budget term as the causal source of the gain; optimization-path sensitivity is a plausible alternative explanation.

Table 3: Full uncertainty for the core SoftmaxListwise (SL) and ActionableValueRegression (AVR) comparisons. Δ is QualityHit minus comparator; brackets are seed$\times$instance two-level paired-bootstrap 95% confidence intervals.

Family	B	Δ vs. SL	Δ vs. AVR
FJSP	4	$+0.370$ $[0.060, 0.680]$	$+0.404$ $[0.041, 0.746]$
FJSP	8	$+0.450$ $[0.040, 0.860]$	$+0.449$ $[0.083, 0.811]$
CVRP	4	$+0.00460$ $[0.00110, 0.00810]$	$+0.00600$ $[0.00060, 0.01140]$
CVRP	8	$+0.00630$ $[0.00100, 0.01160]$	$+0.00800$ $[0.00110, 0.01490]$
TSP	4	$+0.00220$ $[-0.00010, 0.00450]$	$+0.00300$ $[0.00010, 0.00590]$
TSP	8	$+0.00280$ $[0.00020, 0.00540]$	$+0.00350$ $[0.00030, 0.00670]$

Table 4: QualityHit minus cutoff-aware controls on the matched holdout. Δ is the paired mean difference; brackets are seed$\times$instance two-level paired bootstrap 95% confidence intervals. For comparator C, the normalized gain is $100\,\mathbb{E}[G_B^{\mathrm{QH}} - G_B^C]/\mathbb{E}[a^\star - G_B^C]$, i.e., the fraction of that comparators mean remaining oracle headroom closed by QualityHit.

Family	B	Δ vs. ψ_5	Norm. gain (%)	Δ vs. DiffTopK–NS	Norm. gain (%)
FJSP	4	$+0.20$ $[0.01, 0.39]$	2.73	$+0.19$ $[0.05, 0.33]$	2.59
FJSP	8	$+0.22$ $[0.01, 0.43]$	3.11	$+0.23$ $[0.06, 0.40]$	3.27
CVRP	4	$+0.0044$ $[0.0005, 0.0083]$	11.61	$+0.0034$ $[0.0010, 0.0058]$	9.11
CVRP	8	$+0.0053$ $[0.0004, 0.0102]$	14.02	$+0.0047$ $[0.0012, 0.0082]$	12.76
TSP	4	$+0.0017$ $[-0.0008, 0.0042]$	1.65	$+0.0012$ $[-0.0010, 0.0034]$	1.17
TSP	8	$+0.0023$ $[-0.0001, 0.0047]$	2.42	$+0.0013$ $[-0.0006, 0.0032]$	1.38

Threshold-structure audit. The threshold-structure audit reconstructs 300 deterministic configuration-matched test pools per family, groups positive gains into distinct levels, and reports L, m_+, utility-width shares by m_ℓ, and the utility-width-weighted within-pool standard deviation of $\lambda_{m_\ell}(4)$. It performs no model training and uses no scorer predictions.

Certificate geometry at observed threshold sizes. Table 7 evaluates the full-set miss cap and its logarithmic multiplier at the median positive-set size from the structural holdout. These are not empirical tightness ratios—they use only observed set sizes—but they expose the direction of the relaxation: broad sets permit a miss with probability mass close to one and therefore receive a large compensating multiplier.

Envelope-tightness audit. Given retained per-candidate score vectors and exact gains, R_B, $\mathcal{U}_{\mathrm{top}}^{(B)}$, and $\mathcal{L}_{\mathrm{hit}}^{(B)}$ can all be evaluated exactly per pool. The released code includes `scripts/envelope_tightness_diagnostic.py` for this calculation. The current no-data artifact does not contain the per-instance score traces required to reconstruct an empirical tightness curve, so no numerical tightness claim is made here.

All six LambdaRank@B point estimates favor QualityHit. The two-level 95% interval excludes zero in five of six cells; TSP at $B = 4$ is the exception. After Holm correction across the three families within each budget, the $B = 8$ comparisons remain significant at $\alpha = 0.05$ and the $B = 4$ comparisons do not.

DiffTopK–NS control implementation. The matched differentiable top-k control instantiates the differentiable top-k cross-entropy construction of Petersen et al. (2022) with the NeuralSort relaxation of Grover et al. (2019). For each pool the deterministic label is the actionable maximizer $y^\star$ and $k = B$. To make 512-candidate pools practical, the implementation optionally keeps the target plus the highest-scoring $m - 1$ alternatives before the relaxation. This follows Petersen et al.'s implementation strategy of differentiably ranking only a hard top-m subset while ensuring that the ground-truth class is included. The reported configuration uses $m = 32$ and tunes the relaxation temperature on validation data; `shortlist_m=None` runs the full relaxation.

Table 5: Matched fixed-B_0 sensitivity check with $B_0 = 4$ ($n = 300$ per family). $\Delta_4 = 0$ holds algebraically. At $B = 8$, intervals and p-values use the seed×instance two-level paired bootstrap; Holm correction is across the three families. FJSP and CVRP remain significant at $\alpha = 0.05$ after adjustment; TSP does not.

Family	Δ_4	Δ_8	95% CI for Δ_8	raw p	Holm p
FJSP	0	$+0.248$	$[0.058,\ 0.438]$	0.011	0.033
CVRP	0	$+0.0136$	$[0.0027,\ 0.0245]$	0.014	0.033
TSP	0	$+0.0014$	$[-0.0003,\ 0.0031]$	0.110	0.110

Table 6: Configuration-matched threshold-structure audit; 300 deterministic reconstructed test pools per family. Width share is the fraction of total utility-interval width attached to thresholds of the stated size.

Family	median L	median m_+	$\Pr(m_+ = 1)$	width share $m_\ell \le 4$	mean sd$(\lambda_{m_\ell}(4))$
FJSP	6	36	0.0%	24.0%	0.021
CVRP	14	18	0.3%	68.1%	0.029
TSP	4	4	7.0%	88.3%	0.015

C Top-k Consistency Control

Max-oriented consistency versus multiclass top-k calibration. Yang and Koyejo define top-k calibration through the requirement that surrogate-risk minimization produce scores that are top-k preserving with respect to an arbitrary conditional class-probability vector η (Yang & Koyejo, 2020). QualityHit has a different input object: a deterministic vector of candidate utilities. For a unique utility maximizer $y^\star$, Equation (7) gives a positive surrogate gap whenever $y^\star$ is excluded from a fixed top-k prefix. Indeed, exclusion implies $p_{y^\star} \le 1/(k+1)$, yielding

$$\inf_{s:\, y^\star \notin \mathrm{TopK}_k(s)} \mathcal{L}^{(B)}_{\mathrm{hit}}(s; a) \ge \delta^\star \frac{\log(k+1)}{\log(1+B)} > 0 = \inf_s \mathcal{L}^{(B)}_{\mathrm{hit}}(s; a).$$

If the deterministic target is embedded as the degenerate class distribution $\eta = e_{y^\star}$, this gives strict separation for eventual top-k preservation at that target. It is not a proof of calibration uniformly over the simplex: QualityHit can be indifferent to reorderings among nonmaximal candidates that preserve the nested utility-set masses relevant to verification.

Yang–Koyejo control protocol. For each combinatorial family and deployment budget, the ψ_5 model is trained with

$$\psi_5(s, y^\star; B) = \left(1 + s_{[B+1]} - s_{y^\star}\right)_+,$$

using the same cheap candidate features, MLP architecture, train/validation pools, seeds, optimizer budget, and validation-only selection used by QualityHit. The actionable maximizer $y^\star$ is chosen by a fixed tie rule, and evaluation is performed on the fresh holdout reported in Table 4.

Table 7: Full-set certificate geometry at the median positive-set size m_+ from Table 6. The miss cap is $c = m/(m + B)$ and the QualityHit multiplier is $[\log(1 + B/m)]^{-1}$.

Family	median m_+	$c\,(B = 4)$	multiplier $(B = 4)$	$c\,(B = 8)$	multiplier $(B = 8)$
FJSP	36	0.900	9.49	0.818	4.98
CVRP	18	0.818	4.98	0.692	2.72
TSP	4	0.500	1.44	0.333	0.910

Table 8: Matched LambdaRank@B comparison. QualityHit and LambdaRank@B use identical candidate pools, train/validation/test splits, model architecture, optimization budget, deployment budget, and training seeds. Δ is QualityHit minus LambdaRank@B on held-out BestGain. Intervals and p-values use the seed$\times$instance two-level paired bootstrap. Holm correction is applied across the three families separately for each B.

Family	B	Δ	95% CI	raw p	Holm p
FJSP	4	$+0.24$	$[0.02, 0.46]$	0.033	0.072
CVRP	4	$+0.0031$	$[0.0004, 0.0058]$	0.024	0.072
TSP	4	$+0.0017$	$[-0.0002, 0.0032]$	0.050	0.072
FJSP	8	$+0.31$	$[0.07, 0.55]$	0.011	0.034
CVRP	8	$+0.0051$	$[0.0008, 0.0092]$	0.017	0.035
TSP	8	$+0.0021$	$[0.0001, 0.0041]$	0.040	0.040