THE COST OF COMPARING COUPLED GENERATORS

Anonymous authors
Paper under double-blind review

ABSTRACT

Pairwise evaluation of language models consumes resources both to generate responses and to judge their quality. Coupled sampling offers a way to reduce this cost by sharing randomness across generators. We study how to use this shared randomness to estimate the independent-output win probability at a prescribed accuracy. Our approach compares outputs across independent coupled pairs. When both pairs contain matching responses, symmetry determines the comparison score and the judgment is skipped. We combine this construction with adaptive stopping and characterize its generation and judgment costs in terms of the probability that coupled outputs differ. The analysis reveals a linear dependence for generation and a quadratic dependence for judgment in the high-precision regime. Lower bounds establish both dependencies on the same instances. Experiments with language models show that crossed comparison reduces the resources needed to reach a given precision across temperature changes, quantization, and longer responses.

1 INTRODUCTION

Language models are repeatedly evaluated as their weights, decoding settings, and numerical precision change. A common way to compare two configurations is to generate a response from each on the same prompt and ask a judge which response it prefers (Zheng et al., 2023; Chiang et al., 2024). Repeating this experiment estimates the probability that one generator beats the other. Obtaining a precise estimate requires many generated responses and many judgments, making evaluation a substantial part of the cost of model development (Maia Polo et al., 2024; Wu et al., 2026).

Shared sampling randomness offers a way to reuse information across the two generators (Glasserman, 2004). For example, identical generators supplied with the same random inputs produce identical responses. Nearby configurations also frequently agree under coupled token generation (Benz et al., 2026). Such agreement is directly observable and reveals that some evaluation outcomes can be determined from the responses themselves.

Pairwise evaluation introduces an additional requirement. The conventional win probability compares independently drawn responses, whereas coupling changes their joint distribution. Directly judging a coupled pair therefore estimates a different probability, and the resulting preference can reverse (Benz et al., 2026). This distinction between a paired comparison and a marginal relative effect also appears in paired-sample inference (Munzel, 1999; Konietschke & Pauly, 2012). We study the generation and judgment savings from coupling when estimating the independent-output win probability.

We form crossed comparisons from two independent coupled pairs at the same prompt. Each first-generator response is compared with the second-generator response in the other pair. These responses are independent and estimate the intended win probability. When both original pairs match, the crossed comparisons reverse one another. Their average is exactly half credit, supplying a known score that requires no judgment.

We characterize the generation and judgment costs of estimating this score's mean at a prescribed accuracy and confidence. A betting-based stopping rule (Waudby-Smith & Ramdas, 2024) adapts the sample size to the observations. Agreement bounds both score variance and judgment frequency, producing different cost laws for the two resources. We establish upper bounds for an unknown mismatch rate and lower bounds that enforce the leading costs on the same instance under a strict quality ranking.

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.

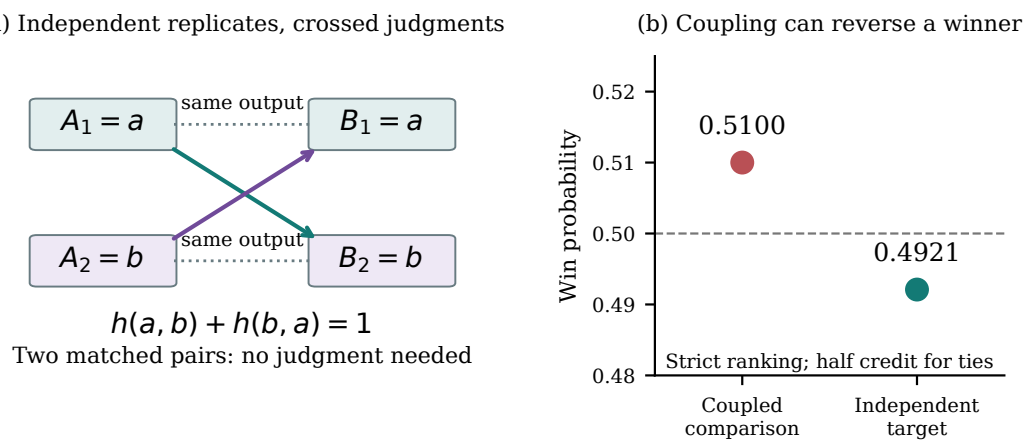


Figure 1: Crossed comparison preserves the independent-output target. Left: responses are compared across independent coupled pairs. Two matching pairs produce reversed comparisons whose average is half credit. Right: directly judging coupled responses can reverse the preferred generator under a strict quality ranking.

Experiments on temperature changes, quantization, and longer responses measure both resource costs. Crossed evaluation roughly halves generation on nearby-temperature comparisons and saves an even larger fraction of judgments. A cancellation ablation separates the two savings, and an online evaluation runs generation and judgment through stopping.

2 Crossed comparison

Let $X \sim \mathcal{D}$ denote a prompt and let P_x, Q_x be the two generators' response distributions at prompt x. A coupled call draws $(A, B) \sim C_x$ with marginals P_x, Q_x; separate calls use independent randomness. Response quality is evaluated by a fixed comparator $h_x(a, b) \in [0, 1]$, where higher values favor the first response. We require

$$h_x(a, b) + h_x(b, a) = 1. \tag{1}$$

In particular, identical responses receive half credit. An order-sensitive judge is symmetrized by averaging its two presentation orders and complementing the reversed decision. The comparison target is

$$W = \mathbb{E}_X \mathbb{E}_{A \sim P_X,\, B \sim Q_X} h_X(A, B), \qquad A \perp B \mid X. \tag{2}$$

This probability averages independent comparisons within a prompt and then averages over prompts. The probability of a mismatch under the coupling is

$$q_x = \mathbb{P}_{C_x}(A \neq B), \qquad q = \mathbb{E}_X q_X. \tag{3}$$

For text, equality refers to the complete decoded response, including whitespace. The generators, coupling, comparator, and prompt distribution are fixed throughout an evaluation.

A change in the comparison target. The distinction between coupled and independent win probabilities arises even under a strict quality order. Consider responses $0 < 1 < 2 < 3 < c$ and a coupling that returns (c, c) with probability $1 - q$. On a mismatch, it returns $(0, 3)$ with probability $9/20$ and $(2, 1)$ with probability $11/20$. The shared mass at c makes the coupling maximal. Direct comparison yields $1/2 + q/20$, while the independent-output target is $1/2 - 79q^2/400$. Thus the two probabilities lie on opposite sides of a tie. The example in Figure 1 uses $q = .2$ and gives win probabilities $.5100$ and $.4921$.

2.1 THE CROSSED SCORE

Draw a fresh prompt and two independent coupled pairs at that prompt. We call these four responses a block and define its score and activity indicator by

$$T = \frac{h_X(A_1, B_2) + h_X(A_2, B_1)}{2}, \qquad Z = \mathbf{1}\{A_1 \neq B_1 \text{ or } A_2 \neq B_2\}. \tag{4}$$

Each crossed edge has conditional distribution $P_X \times Q_X$, so its expectation is the desired win probability at that prompt. If $Z = 0$, the original pairs have the form (a, a) and (b, b). Their crossed score is $[h_X(a, b)+h_X(b, a)]/2 = 1/2$ by (1). The procedure records this value directly. On an active block, it evaluates the crossed edges, reusing identity values and previously obtained comparisons.

For example, under the quality order $a < b < c$, pairs (b, b) and (c, a) give two winning crossed edges and a score of one. Replacing the second pair by (c, b) makes one edge an identity, leaving a single judgment to determine the score. Together with the double-match case in Figure 1, these examples give blocks requiring two, one, or zero judgments. The following lemma describes their statistical and resource properties.

Lemma 1 (Two savings from agreement)**.** *Let $p = \mathbb{E}_X(2q_X - q_X^2)$. Independent blocks satisfy*

$$q \leq p \leq 2q, \quad \mathbb{E}T = W, \quad \mathrm{Var}(T) \leq p/4, \quad J_{\mathrm{block}} \leq 2Z. \tag{5}$$

Proof. Conditional independence gives $\mathbb{P}(Z = 1 \mid X) = 2q_X - q_X^2$. Crossed marginals give the expectation. Since $T \in [0, 1]$ and equals $1/2$ when $Z = 0$, $\mathrm{Var}(T) \leq \mathbb{E}(T - 1/2)^2 \leq \mathbb{E}Z/4$. Only active blocks can require the two queries. □

2.2 AGREEMENT AND SCORE VARIANCE

Lemma 1 identifies the two effects of agreement: it bounds the score variance and the frequency of paid comparisons. To describe the variance more precisely, let $d = W - 1/2$ and $a_2 = \mathbb{E}[(T-1/2)^2 \mid Z = 1]$, with $a_2 = 0$ when $p = 0$. Inactive blocks have zero deviation from half credit, giving

$$v := \mathrm{Var}(T) = p\, a_2 - d^2, \qquad 0 \leq a_2 \leq 1/4. \tag{6}$$

Here p describes how often a block is active, and a_2 describes how far its score lies from half credit when it is active. Frequent agreement reduces the first factor; concentration of the crossed comparisons reduces the second. Substituting $a_2 \leq 1/4$ recovers the variance bound in the lemma.

These two effects lead to different resource costs. For a fixed sample of n blocks, the method returns $4n$ outputs and requests at most $2pn$ judgments in expectation. Bernstein's inequality relates the number of blocks required for accuracy ε to $(v + \varepsilon)\varepsilon^{-2} \log(1/\delta)$. The variance bound thus supplies one mismatch factor in the generation cost, and the judgment frequency supplies a second factor in the judgment cost. To obtain these savings during an actual evaluation, the evaluator must learn when enough blocks have been collected. We address this stopping problem in Section 3.

3 ADAPTIVE EVALUATION AND RESOURCE COMPLEXITY

An evaluation receives an error tolerance ε and a failure probability δ. It collects blocks until a stopping time N and returns an estimate $\widehat{W}$ satisfying $\mathbb{P}(|\widehat{W} - W| > \varepsilon) \leq \delta$. We measure its cost by the number G of coupled generation calls and the number J of comparator evaluations. A crossed block uses two coupled calls and returns four responses. Thus $G = 2N$ and the number of returned responses is $2G$.

3.1 A STOPPING RULE FOR BOUNDED SCORES

We use a betting construction for bounded means (Waudby-Smith & Ramdas, 2024). For a score $S \in [a, a + R]$, define $Y = (S - a)/R$ and $e = \varepsilon/R$. The normalized scores lie in $[0, 1]$. Let $K = \lceil \log_2(1/e) \rceil + 1$ and define

$$M_n^{\pm}(m) = \frac{1}{K} \sum_{j=1}^{K} \prod_{i=1}^{n} [1 \pm 2^{-j}(Y_i - m)]. \tag{7}$$

The two mixtures test candidate means below and above the running average. At an inspection time, the lower check passes if $M_n^+(\bar{Y}_n - e) \geq 4/\delta$ and the upper check passes if $M_n^-(\bar{Y}_n + e) \geq 4/\delta$. An endpoint at or beyond the boundary of $[0, 1]$ automatically passes its check. Both checks passing certifies the requested precision. For crossed comparison, $Y = T$ and $e = \varepsilon$, giving the following procedure:

1. Sample a fresh prompt and two independent coupled pairs at that prompt.
2. Record half credit if both pairs match. Otherwise evaluate the crossed score, filling in known identities and cached comparisons.
3. Inspect the two confidence checks every b blocks. Stop when both pass, or at the cap $n_H = \lceil \log(4/\delta)/(2\varepsilon^2) \rceil$. Return the mean of the recorded block scores.

The early checks and the cap each receive half of the failure budget. For a general bounded score, the cap uses normalized accuracy e and the returned estimate is $a + R\bar{Y}_N$. We set the inspection spacing to $b = 64$ in every experiment. The bet sizes and inspection schedule are fixed before sampling; the observed scores determine the stopping time.

Theorem 2 (Accuracy and expected stopping cost)**.** *For iid scores $Y_i \in [0, 1]$ with mean μ and variance v, the procedure satisfies $\mathbb{P}(|\bar{Y}_N - \mu| > e) \leq \delta$. If $0 < e, \delta \leq 1/4$ and $L = \log(4K/\delta)$, then*

$$\mathbb{E}N \leq C\frac{v + e}{e^2}L + 2b \tag{8}$$

for a universal constant C. In particular, crossed comparison satisfies

$$\mathbb{E}G = O\left(\frac{q + \varepsilon}{\varepsilon^2}L + b\right), \qquad \mathbb{E}J = O\left(\frac{q(q + \varepsilon)}{\varepsilon^2}L + qb\right). \tag{9}$$

At $q = 0$ it makes no comparator queries.

The accuracy guarantee follows from two probability bounds. At the true mean, each betting mixture is a nonnegative martingale, which controls erroneous early stopping. Hoeffding's inequality (Hoeffding, 1963) controls the error at the fixed cap. For the judgment cost, bounded stopping gives $\mathbb{E}J \leq 2\mathbb{E}\sum_{i=1}^{N} Z_i = 2p\mathbb{E}N$. Combining this identity with the expected stopping time and $v \leq p/4 \leq q/2$ gives (9). Appendix A proves the accuracy and expected-cost bounds.

3.2 Generation and judgment lower bounds

The two resource bounds in (9) suggest that agreement has a stronger effect on judgment cost than on generation cost. The following result establishes that both mismatch powers are necessary for uniform accuracy, even when the comparator follows a strict quality ranking.

Theorem 3 (Same-instance lower bound)**.** *For $q \in (0, 1]$, $0 < \varepsilon \leq q/64$, and $0 < \delta \leq 1/16$, there is a finite class with mismatch q and comparators induced by strict scalar quality rankings such that every uniformly accurate algorithm has an instance with*

$$\mathbb{E}G \geq c_G\frac{q}{\varepsilon^2}\log(1/\delta), \qquad \mathbb{E}J \geq c_J\frac{q^2}{\varepsilon^2}\log(1/\delta). \tag{10}$$

Here $c_G, c_J > 0$ are universal constants. The algorithm may know q, adapt both types of queries, and compare arbitrary outputs, including outputs it has not generated. The finite domain may grow with the requested precision.

The construction places two unknown quantities in the same comparison problem. A probability parameter controls the mass of two responses with known quality. A balanced table controls the ranks of a second collection of responses relative to an anchor. Generation reveals the first quantity; judgment reveals the second. Each quantity changes the independent-output win probability. An adaptive change-of-measure argument bounds the information obtained from each access (Kaufmann et al., 2016). The generation bound holds throughout the balanced family, and averaging the judgment bound over that family yields an instance satisfying both inequalities. Appendix B gives the construction and proof.

For ε small relative to q, the upper and lower bounds have the same powers of q and ε: the generation bounds scale as q/ε^2 and the judgment bounds as q^2/ε^2. The sequential upper bound carries the

factor $L = \log(4K/\delta)$, while the lower bound carries $\log(1/\delta)$. The common hard instance gives the two statements a joint interpretation: an evaluator needs enough generated outputs to learn their frequencies and enough comparisons to learn their quality. Coupling reduces both requirements through agreement.

4 COMPARING EVALUATION COSTS

We compare crossed pairs with independent all-pairs averages, larger coupled groups, and a reference difference. Each procedure estimates W with the stopping rule in Section 3.

Independent all-pairs comparison. At a fresh prompt, generate k independent responses from each marginal and average all k^2 comparisons. A block uses $2k$ responses. We evaluate groups of size $k = 1, 2, 4$; the size-two group uses the same number of responses as a crossed block and averages all four independent edges. Appendix C gives its exact variance.

Grouped coupled comparison. Generate k independent coupled pairs at a prompt and average the off-diagonal edges, $U_k = [k(k-1)]^{-1} \sum_{i \neq j} h_X(A_i, B_j)$. Every edge crosses independent replicates, so the score has expectation W. Each unordered pair of matching replicates supplies two reversed comparisons with known sum. The score uses $2k$ responses per block; $k = 2$ is the crossed method and $k = 4$ supplies a larger group.

Lazy reference difference. A response from the second generator provides a reference with known self-comparison mean. Draw $(A, B) \sim C_X$ and return half credit when $A = B$. On a mismatch, draw an independent $C \sim Q_X$ and form

$$S = \frac{1}{2} + h_X(A, C) - h_X(B, C). \tag{11}$$

Antisymmetry gives $\mathbb{E}h_X(B, C) = 1/2$, hence $\mathbb{E}S = W$. This procedure uses $2 + q$ responses and at most $2q$ judgments per block in expectation. Since $(S - 1/2)^2 \leq \mathbf{1}\{A \neq B\}$, its variance is bounded by q. Its range is $[-1/2, 3/2]$. Thus agreement controls both the variation in the reference correction and the frequency with which it is computed. Theorem 2 therefore gives this score the same leading mismatch powers in its generation and judgment bounds. Its connection to crossed comparison is explicit. For two independent coupled pairs, define $D_{12} = h(A_1, B_2) - h(B_1, B_2)$ and its index-swapped counterpart D_{21}. Antisymmetry gives

$$T - \tfrac{1}{2} = \tfrac{1}{2}(D_{12} + D_{21}). \tag{12}$$

The two reference comparisons cancel in this average. The resulting score lies in $[0, 1]$ and is determined without judgment on double-match blocks. The experiments compare the full resource costs of these two ways to use coupled responses.

Generation and judgment. Shared Gumbel vectors couple next-token samples while preserving each generator's marginal distribution (Oberst & Sontag, 2019; Benz et al., 2026). Vectors are independent across positions and replicates. The judge receives the prompt and anonymous responses in both orders; averaging the first decision and the complemented reverse decision defines the comparison score.

Resource tradeoffs. For a block score S with range width R, Theorem 2 relates the expected number of blocks to $[\mathrm{Var}(S) + R\varepsilon]\varepsilon^{-2}L$. Larger groups spend more responses on each score and reuse them in more comparisons. Their variance reduction determines whether this expenditure saves resources at the requested precision.

Let c_{out} and c_{judge} denote the prices of a response and a symmetrized judgment. Their expected total cost is

$$c_{\mathrm{out}}\, \mathbb{E}[\text{returned outputs}] + c_{\mathrm{judge}}\, \mathbb{E}J. \tag{13}$$

For crossed pairs, this is at most $(4c_{\mathrm{out}} + 2pc_{\mathrm{judge}})\mathbb{E}N$: the cost per score multiplies the number of scores needed for precision. Reporting both resources allows this tradeoff to be evaluated at different prices.

Table 1: Total cost of reaching $\varepsilon = .02$ at $\delta = .05$ on the four nearby-temperature pools. Each cell gives mean returned outputs / distinct judgments over 256 trials, using the common stopping rule and resource accounting.

Method	E2E 0.5B	E2E 1.5B	CG 0.5B	CG 1.5B
Independent 1	10,925 / 4,211	8,596 / 3,106	10,956 / 4,352	10,112 / 3,983
Independent 2	9,893 / 6,216	7,714 / 4,563	10,347 / 6,702	9,126 / 5,944
Independent 4	9,696 / 8,624	7,898 / 6,537	10,192 / 9,386	8,974 / 8,379
Crossed 2	**4,443 / 1,295**	**3,448 / 743**	**5,323 / 1,751**	**4,095 / 1,191**
Coupled 4	6,194 / 3,990	5,656 / 2,669	6,658 / 4,872	6,144 / 3,948
Lazy reference	5,531 / 1,689	3,971 / 894	7,127 / 2,475	5,017 / 1,519

Judgments sharing a response contribute to one averaged block score. Fresh prompts and generation randomness make successive scores independent. Appendix C gives the within- and between-prompt variance decompositions for both types of group.

5 Experiments

We compare generation and judgment costs under adaptive stopping, isolate the contribution of cancellation, and examine the gains across model changes. We then run generation and judgment inside an online evaluation loop.

Experimental setup. We use Qwen2.5-Instruct models (Qwen Team, 2024) on E2E restaurant descriptions (Novikova et al., 2017) and CommonGen constrained sentences (Lin et al., 2020). The main comparison changes sampling temperature from 0.70 to 0.75 for the 0.5B and 1.5B models, giving four conditions. For each condition, we generate 16 coupled response pairs on each of 64 prompts. A frozen Qwen2.5-7B judge scores every distinct response pair in both orders. This table gives the exact win probability W_{pool} under the product of the empirical response marginals.

Each evaluation trial draws prompts and response indices independently with replacement and estimates W_{pool}. The difference between the returned estimate and this target gives its estimation error.

Resource accounting. We count every response sampled before stopping. One symmetrized judgment uses two judge inputs, one for each presentation order. Each run starts with an empty cache and obtains values through its requested comparisons. Requested comparisons count queries before caching; distinct judgments count the first query for each prompt and unordered pair of full responses. Identity scores and reversed-edge cancellations are free for all applicable methods, including independent all-pairs averages.

The methods in Section 4 use the same stopping rule with failure probability $\delta = .05$. We evaluate three error tolerances, $\varepsilon \in \{.01, .02, .05\}$, with 256 trials for each method and condition. Methods, seeds, and stopping parameters are fixed before these runs. Appendix D provides the prompts, generation settings, judge protocol, and complete results.

5.1 Generation and judgment costs

Crossed comparison reduces both resources on all four nearby-temperature conditions. At error .02, it uses 48–55% fewer generated responses and 60–76% fewer distinct judgments than the least expensive independent procedure for each resource (Table 1). The independent reference is selected among group sizes one, two, and four.

The generation saving is consistent with the variance of the crossed score. At a fixed output count, exact pool variances imply an RMSE reduction of 29–43% relative to independent groups of two. Both methods generate four responses per block, so this comparison isolates their use of the same amount of generated data. Under adaptive stopping, the concentrated crossed scores reach the precision certificate earlier. The additional reduction in judgments reflects the known values supplied by matching responses.

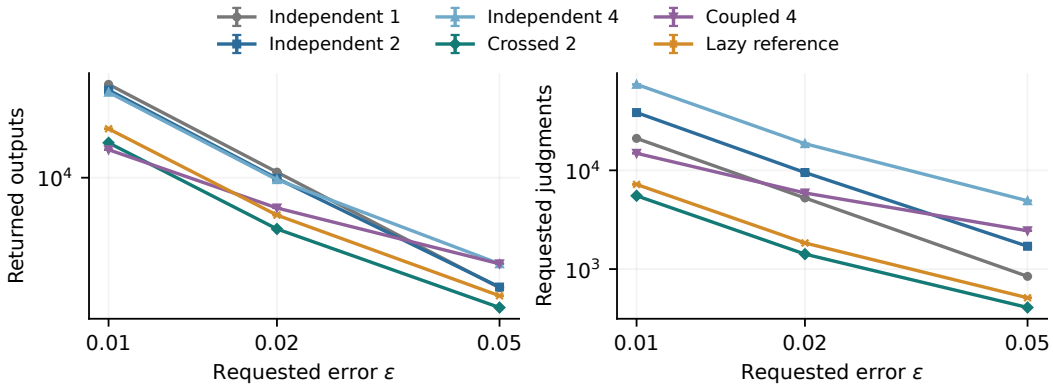


Figure 2: Resources required to reach different error tolerances on E2E 0.5B. Left: returned responses. Right: requested judgments before caching. All procedures use the same stopping rule at 95% confidence. Error bars show one Monte Carlo standard error over repeated trials.

The required precision also affects the benefit of grouping (Figure 2). At the finest tolerance, larger coupled groups save generated responses on most of the initial conditions. Their lower within-prompt variance compensates for the additional responses in each block. At the two coarser tolerances, crossed pairs give the lowest output counts among the coupled procedures. This pattern follows the variance and range terms in Theorem 2: variance reduction becomes more valuable as the required error shrinks. The full precision comparison appears in Table 8.

The resource totals also determine which group size is preferable at given prices. On E2E 0.5B at $\varepsilon = .01$, crossed pairs use 17,373 outputs and 4,181 distinct judgments; coupled groups of four use 15,580 outputs and 7,264 distinct judgments. In (13), their costs are equal at $c_{\text{judge}}/c_{\text{out}} \simeq .582$. Crossed pairs have lower total cost above this ratio, and groups of four below it.

Every method–precision–pool condition has an empirical error rate below the specified failure probability. Appendix D reports the counts.

5.2 The contribution of cancellation

To isolate cancellation, we evaluate the crossed edges of double-match blocks explicitly while keeping the scores, stopping times, identity shortcuts, and cache fixed. Filling in their known values saves an additional 14–33% of distinct judgments at error .02 (Table 7). Since the two executions use identical observations and stop together, this saving comes from the comparisons skipped within a block. It is the judgment-frequency effect identified in Lemma 1.

5.3 Other generator changes

The additional conditions compare BF16 and LLM.int8 versions of Qwen2.5-1.5B (Dettmers et al., 2022) at a common temperature on 64 new E2E prompts; generate 80–100 word stories from CommonGen concept sets; and increase the temperature change to $0.70 \to 1.00$ on E2E. Two disjoint sets of quantization response draws form primary and replication pools. Appendix D gives the generation and scoring details.

Crossed comparison saves both outputs and judgments under each of these changes (Table 2). The two quantization pools give similar cost ratios, and the reduction extends to the longer narrative responses. These experiments cover changes to numerical precision, response length, and sampling temperature using the same evaluation procedure.

Table 3 explains the concentration of the crossed scores. Their variance is lower than that of independent all-pairs scores in every condition. On the nearby-temperature pools, matching pairs place a substantial fraction of blocks at their known half-credit value. The remaining active scores also concentrate near that value, as captured by the second moment a_2. Both effects contribute to the variance used by the stopping rule.

Table 2: Additional generator comparisons at $\varepsilon = .02$, $\delta = .05$. Ratios divide crossed cost by the lowest mean cost among independent 1, 2, and 4, separately for each resource. Values below one favor crossed comparison. The two 8-bit pools use disjoint replicate draws on the same contexts.

Condition	q	Output ratio	Judgment ratio
8-bit primary	0.711	0.784	0.690
8-bit replication	0.718	0.779	0.685
Long narrative	0.881	0.705	0.688
Larger temperature	0.834	0.875	0.808

Table 3: The two components of crossed-score concentration. Exact enumeration of each response pool gives the active-block probability p, the active second moment a_2, and the resulting variance $v = pa_2 - (W_{\text{pool}} - 1/2)^2$. Independent all-pairs variance uses the same four outputs per block.

Condition	p	a_2	$\mathrm{Var}(T)$	$\mathrm{Var}(I_2)$
E2E 0.5B	0.6366	0.0535	0.0330	0.0799
E2E 1.5B	0.4724	0.0410	0.0190	0.0594
CommonGen 0.5B	0.7286	0.0582	0.0420	0.0841
CommonGen 1.5B	0.6297	0.0468	0.0289	0.0727
8-bit primary	0.8971	0.0601	0.0538	0.0697
8-bit replication	0.8979	0.0584	0.0523	0.0676
Long narrative	0.9806	0.0523	0.0504	0.0726
Larger temperature	0.9496	0.0708	0.0603	0.0701

In the additional conditions, most blocks are active and the crossed edges continue to produce concentrated scores. The measured second moments account for the smaller crossed variance through (6). This connects the gains across generator changes to the same quantity controlling the number of blocks required for evaluation. The cancellation ablation separately measures the judgments saved while forming those scores.

5.4 Online evaluation

We run the evaluation directly on fresh Qwen2.5-1.5B responses, obtaining judgments as they are needed. Crossed comparison and independent groups of two each execute one complete run on the same distribution of 64 new E2E prompts. The generators use temperatures .70 and .75, and the requested error is .05 at 95% confidence. Generation and judgment share a single GPU; the judge evaluates each ordered input separately with deterministic algorithms.

The crossed run reaches the requested precision with fewer responses and judge calls (Table 4). Both stages take less time in this execution (Table 9). The run uses the same stopping rule as the replay experiments, applying cancellation as responses are generated.

Table 4: Online evaluation at error $\varepsilon = .05$ and confidence 95%. Each row reports one complete run: returned responses, distinct judgments, the resulting interval, and elapsed evaluation time. Timing covers generation, judgment, and stopping, with model loading and post-stop diagnostics recorded separately.

Method	Outputs	Judgments	95% interval	Seconds
Crossed 2	1,280	304	[0.455, 0.555]	536.7
Independent 2	1,536	1,509	[0.466, 0.566]	803.1

6 Related work

Coupled generation. Common random numbers reduce the variance of simulation comparisons by coordinating random inputs (Glasserman, 2004). Gumbel-max causal models provide a shared-noise representation of categorical sampling (Oberst & Sontag, 2019), and learned variants optimize

the coupling for a distribution of queries (Lorberbom et al., 2021). This representation supports counterfactual token and string generation in language models (Chatzi et al., 2025; Ravfogel et al., 2025). Benz et al. (2026) use coupled token generation to reduce benchmark sampling costs and study its effect on pairwise win rates. Building on coupled sampling, we characterize the generation and judgment resources needed to estimate the independent-output win probability. Full-response agreement supplies the connection between these two costs.

Pairwise language-model evaluation. MT-Bench and Chatbot Arena develop model-based and human pairwise evaluation protocols (Zheng et al., 2023; Chiang et al., 2024). Length-controlled AlpacaEval uses regression to adjust win rates for response length (Dubois et al., 2024). These works show how the judge, prompt population, and comparison protocol determine the quantity being measured. We fix those components and study the sampling cost of the resulting win probability. Averaging the two presentation orders supplies the antisymmetric comparator used throughout our analysis and experiments.

Paired data and crossed statistics. The distinction between paired outcomes and marginal relative effects is established in nonparametric paired-sample inference (Munzel, 1999; Konietschke & Pauly, 2012). Off-diagonal averages and their variance decompositions belong to the U-statistic framework (Hoeffding, 1948). Our grouped estimator applies this construction to independent coupled replicates at each prompt. Its additional structure comes from equality of generated responses: two matching replicates determine their crossed average exactly. This identity makes the statistic inexpensive to evaluate as well as concentrated.

Efficient evaluation. Active testing selects informative test labels and corrects the resulting sampling bias (Kossen et al., 2021). For language models, tinyBenchmarks uses curated subsets and historical performance patterns to estimate full benchmark scores (Maia Polo et al., 2024). Factorized Active Querying combines historical model outcomes with adaptive question selection and confidence intervals for benchmark accuracy (Wu et al., 2026). These methods allocate effort across evaluation inputs. Crossed comparison uses shared randomness within each sampled prompt, and its stopping rule tracks the precision obtained from the generated responses and paid judgments.

Auxiliary information and correction. Coupled control variates (Goodman & Lin, 2009) and multilevel Monte Carlo (Giles, 2008) reduce variance through differences between related random quantities. Prediction-powered inference and PPI++ combine predictions with observed corrections (Angelopoulos et al., 2023a;b); active statistical inference also uses predictions to guide label collection (Zrnic & Candes, 2024). Our reference-difference score uses a second-generator response as an anchor with known self-comparison mean. Averaging the two reference directions yields the crossed score, whose correction vanishes exactly on double-match blocks.

Sequential precision and resource bounds. Dagum et al. (2000) establish instance-adaptive sampling costs for relative-error Monte Carlo estimation. Empirical Bernstein methods connect observed variance to confidence bounds and stopping (Mnih et al., 2008; Maurer & Pontil, 2009). Confidence sequences make inference valid throughout sampling (Howard et al., 2021), and betting gives concrete constructions for bounded means (Waudby-Smith & Ramdas, 2024). We apply these tools to a score whose variance and judgment frequency both depend on coupling. Tracking the two resources through stopping gives their separate mismatch dependencies. The matching lower bounds use adaptive change of measure (Kaufmann et al., 2016) to identify the information required from generation and judgment on the same instance.

7 Conclusion

Crossed comparisons turn agreement between coupled generators into known scores for the independent-output win probability. Adaptive stopping converts score concentration into fewer generated responses, while exact cancellation saves judgments. Upper and lower bounds characterize the two resource costs, and language-model experiments demonstrate these savings in complete evaluations.

AI USE STATEMENT

We used LLMs to assist with research formulation; development of estimators, mathematical claims and proofs; experimental design, data preparation and implementation; and analysis and interpretation of results. These tools also assisted manuscript revision. The experiments use Qwen2.5 models to generate response datasets and Qwen2.5-7B-Instruct to assign pairwise judgments, as described in Section 5 and Appendix D. Validation includes finite-domain checks of estimator identities, stopping-rule calibration, and reconciliation of reported results with saved experimental records. The authors take responsibility for the final text, claims, code and results.

REFERENCES

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

Anastasios N. Angelopoulos, John C. Duchi, and Tijana Zrnic. PPI++: Efficient prediction-powered inference, 2023b. URL https://arxiv.org/abs/2311.01453.

Nina L. Corvelo Benz, Stratis Tsirtsis, Eleni Straitouri, Ivi Chatzi, Ander Artola Velasco, Suhas Thejaswi, and Manuel Gomez Rodriguez. Evaluation of large language models via coupled token generation. In *International Conference on Artificial Intelligence and Statistics*, volume 300 of *Proceedings of Machine Learning Research*, pp. 2215–2223. PMLR, 2026. URL https://proceedings.mlr.press/v300/benz26a.html.

Ivi Chatzi, Nina L. Corvelo Benz, Eleni Straitouri, Stratis Tsirtsis, and Manuel Gomez-Rodriguez. Counterfactual token generation in large language models. In *Conference on Causal Learning and Reasoning*, volume 275 of *Proceedings of Machine Learning Research*, pp. 1291–1315. PMLR, 2025. URL https://proceedings.mlr.press/v275/chatzi25a.html.

Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E. Gonzalez, and Ion Stoica. Chatbot arena: An open platform for evaluating LLMs by human preference. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 8359–8388. PMLR, 2024. URL https://proceedings.mlr.press/v235/chiang24b.html.

Paul Dagum, Richard Karp, Michael Luby, and Sheldon Ross. An optimal algorithm for Monte Carlo estimation. *SIAM Journal on Computing*, 29(5):1484–1496, 2000. doi: 10.1137/S0097539797315306.

Tim Dettmers, Mike Lewis, Younes Belkada, and Luke Zettlemoyer. LLM.int8(): 8-bit matrix multiplication for transformers at scale, 2022. URL https://arxiv.org/abs/2208.07339.

Yann Dubois, Balázs Galambosi, Percy Liang, and Tatsunori B. Hashimoto. Length-controlled AlpacaEval: A simple way to debias automatic evaluators. In *First Conference on Language Modeling*, 2024. URL https://arxiv.org/abs/2404.04475.

Michael B. Giles. Multilevel monte carlo path simulation. *Operations Research*, 56(3):607–617, 2008. doi: 10.1287/opre.1070.0496.

Paul Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, 2004. doi: 10.1007/978-0-387-21617-1.

Jonathan B. Goodman and Kevin K. Lin. Coupling control variates for Markov chain Monte Carlo. *Journal of Computational Physics*, 228(19):7127–7136, 2009. doi: 10.1016/j.jcp.2009.03.043.

Wassily Hoeffding. A class of statistics with asymptotically normal distribution. *The Annals of Mathematical Statistics*, 19(3):293–325, 1948. doi: 10.1214/aoms/1177730196.

Wassily Hoeffding. Probability inequalities for sums of bounded random variables. *Journal of the American Statistical Association*, 58(301):13–30, 1963. doi: 10.1080/01621459.1963.10500830.

Steven R. Howard, Aaditya Ramdas, Jon McAuliffe, and Jasjeet Sekhon. Time-uniform, nonparametric, nonasymptotic confidence sequences. *The Annals of Statistics*, 49(2):1055–1080, 2021. doi: 10.1214/20-AOS1991.

Emilie Kaufmann, Olivier Cappé, and Aurélien Garivier. On the complexity of best-arm identification in multi-armed bandit models. *Journal of Machine Learning Research*, 17(1):1–42, 2016. URL https://jmlr.org/papers/v17/kaufman16a.html.

Frank Konietschke and Markus Pauly. A studentized permutation test for the nonparametric Behrens-Fisher problem in paired data. *Electronic Journal of Statistics*, 6:1358–1372, 2012. doi: 10.1214/12-EJS714.

Jannik Kossen, Sebastian Farquhar, Yarin Gal, and Tom Rainforth. Active testing: Sample-efficient model evaluation. In *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.

Bill Yuchen Lin, Wangchunshu Zhou, Ming Shen, Pei Zhou, Chandra Bhagavatula, Yejin Choi, and Xiang Ren. CommonGen: A constrained text generation challenge for generative commonsense reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pp. 1823–1840. Association for Computational Linguistics, 2020. doi: 10.18653/v1/2020.findings-emnlp.165. URL https://aclanthology.org/2020.findings-emnlp.165/.

Guy Lorberbom, Daniel D. Johnson, Chris Maddison, Daniel Tarlow, and Tamir Hazan. Learning Generalized Gumbel-max Causal Mechanisms. In *Advances in Neural Information Processing Systems*, volume 34, pp. 26792–26803, 2021. URL https://arxiv.org/abs/2111.06888.

Felipe Maia Polo, Lucas Weber, Leshem Choshen, Yuekai Sun, Gongjun Xu, and Mikhail Yurochkin. tinyBenchmarks: evaluating LLMs with fewer examples. In *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.

Andreas Maurer and Massimiliano Pontil. Empirical Bernstein bounds and sample variance penalization. In *Proceedings of the 22nd Annual Conference on Learning Theory*, 2009. URL https://www.cs.mcgill.ca/~colt2009/papers/012.pdf.

Volodymyr Mnih, Csaba Szepesvári, and Jean-Yves Audibert. Empirical bernstein stopping. In *Proceedings of the 25th International Conference on Machine Learning*, 2008. URL https://icml.cc/Conferences/2008/papers/523.pdf.

U. Munzel. Nonparametric methods for paired samples. *Statistica Neerlandica*, 53(3):277–286, 1999. doi: 10.1111/1467-9574.00112.

Jekaterina Novikova, Ondřej Dušek, and Verena Rieser. The E2E dataset: New challenges for end-to-end generation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pp. 201–206. Association for Computational Linguistics, 2017. doi: 10.18653/v1/W17-5525. URL https://aclanthology.org/W17-5525/.

Michael Oberst and David Sontag. Counterfactual off-policy evaluation with Gumbel-max structural causal models. In *International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 4881–4890. PMLR, 2019. URL https://proceedings.mlr.press/v97/oberst19a.html.

Qwen Team. Qwen2.5 technical report, 2024. URL https://arxiv.org/abs/2412.15115v1.

Shauli Ravfogel, Anej Svete, Vésteinn Snæbjarnarson, and Ryan Cotterell. Gumbel counterfactual generation from language models. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=TUC0ZT2zIQ.

Ian Waudby-Smith and Aaditya Ramdas. Estimating means of bounded random variables by betting. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 86(1):1–27, 2024. doi: 10.1093/jrsssb/qkad009.

Skyler Wu, Yash Nair, and Emmanuel J. Candès. Efficient evaluation of LLM performance with statistical guarantees, 2026. URL https://arxiv.org/abs/2601.20251v3.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph Gonzalez, and Ion Stoica. Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena. In *Advances in Neural Information Processing Systems*, volume 36, pp. 46595–46623, 2023. URL https://arxiv.org/abs/2306.05685.

Tijana Zrnic and Emmanuel Candes. Active statistical inference. In *International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pp. 62993–63010. PMLR, 2024. URL https://proceedings.mlr.press/v235/zrnic24a.html.

A GUARANTEE FOR THE IMPLEMENTED STOPPING RULE

This section proves Theorem 2 for the precise rule used in the experiments, including its deterministic cap and inspection spacing. Let $Y_1, Y_2, \ldots$ be iid in $[0,1]$, with mean μ and variance v. Define K, λ_j and $M_n^{\pm}$ by (14). The inspection times are multiples of b, together with n_H. Calculations may be performed in logarithms to avoid numerical overflow.

Exact construction. We use a finite mixture of fixed bets, a direct application of bounded-mean confidence sequences (Waudby-Smith & Ramdas, 2024). For scores $Y_i \in [0,1]$, desired error $e \le 1/4$, and failure probability δ, set

$$K = \lceil \log_2(1/e) \rceil + 1, \quad \lambda_j = 2^{-j}, \quad M_n^{\pm}(m) = \frac{1}{K} \sum_{j=1}^{K} \prod_{i=1}^{n} [1 \pm \lambda_j (Y_i - m)]. \tag{14}$$

After every b blocks, stop if both

$$\left[\bar{Y}_n - e \le 0 \text{ or } M_n^{+}(\bar{Y}_n - e) \ge 4/\delta\right], \quad \left[\bar{Y}_n + e \ge 1 \text{ or } M_n^{-}(\bar{Y}_n + e) \ge 4/\delta\right] \tag{15}$$

hold. Regardless of these checks, stop at $n_H = \lceil \log(4/\delta)/(2e^2) \rceil$ and return $\bar{Y}_n$. For crossed comparison use $Y = T$ and $e = \varepsilon$. A score in $[a, a+R]$ uses $Y = (S-a)/R$, $e = \varepsilon/R$, and returns $a + R\bar{Y}_n$. We fix $b = 64$ in all experiments. The bet grid and inspection spacing are fixed before evaluation.

Accuracy. For $m = \mu$, every factor has conditional expectation one and is nonnegative, because $\lambda_j \le 1/2$. Each product, and hence each mixture, is a nonnegative martingale starting at one. Ville's inequality gives $\mathbb{P}(\sup_n M_n^{+}(\mu) \ge 4/\delta) \le \delta/4$ and the same bound for the negative mixture. The positive mixture decreases in m on $[0,1]$ and the negative mixture increases. Therefore, if $\bar{Y}_N - \mu > e$ at an early certified stop, the positive mixture at μ has crossed $4/\delta$; the analogous negative deviation forces the negative mixture to cross. The boundary shortcuts in (15) are valid since $\mu \in [0,1]$. Early-stop errors have probability at most $\delta/2$. At the deterministic cap, Hoeffding gives $\mathbb{P}(|\bar{Y}_{n_H} - \mu| > e) \le 2\exp(-2n_H e^2) \le \delta/2$. The event of an error when the algorithm actually reaches the cap is a subset of this fixed-time event. A union bound combines the early-stop and fixed-cap error probabilities.

A sufficient empirical-variance condition. Write $s_n^2 = n^{-1}\sum_i (Y_i - \bar{Y}_n)^2$ and $a = s_n^2 + e^2$. For either sign at $m = \bar{Y}_n \mp e$, whenever that m belongs to $[0,1]$, $|\lambda_j(Y_i - m)| \le 1/2$. Using $\log(1+x) \ge x - x^2$ on this interval,

$$\sum_i \log[1 \pm \lambda_j (Y_i - m)] \ge n\lambda_j e - n\lambda_j^2 (s_n^2 + e^2).$$

Set $\lambda_* = \min\{1/2, e/(2a)\}$. Since $s_n^2 \le 1/4$ and $e \le 1/4$, the dyadic grid contains $\lambda_j \in [\lambda_*/2, \lambda_*]$. For that bet the last display is at least

$$\frac{n\lambda_j e}{2} \ge \frac{ne^2}{8\max\{e,a\}} \ge \frac{ne^2}{8(s_n^2 + e)}.$$

The mixture is at least this component divided by K. Both required certificates therefore hold whenever

$$n \ge \frac{8(s_n^2 + e)}{e^2} \log(4K/\delta). \tag{16}$$

If a candidate endpoint lies outside $[0,1]$, that side already passes its boundary check and needs no logarithmic bound.

Expected time. Put $L = \log(4K/\delta)$ and $t_0 = 16(v+e)L/e^2$. At a check time $n \ge t_0$, failure of (16) implies

$$\frac{1}{n}\sum_i (Y_i - \mu)^2 - v > \frac{ne^2}{8L} - (v+e) \ge \frac{ne^2}{16L} = t.$$

Here $s_n^2 \le n^{-1}\sum_i (Y_i-\mu)^2$ and $t \ge v$. The variables $(Y_i-\mu)^2$ lie in $[0,1]$, have mean v, and variance at most v. Bernstein's inequality yields

$$\mathbb{P}(N > n) \le \exp\left[-\frac{nt^2}{2v+2t/3}\right] \le \exp\left[-\frac{3n^2e^2}{128L}\right]. \tag{17}$$

For $n \ge n_H$ the tail is zero. Summing at inspection times and bounding the decreasing tail by its integral gives

$$\mathbb{E}N \le t_0 + 2b + \int_0^\infty \exp[-3x^2e^2/(128L)]\,dx = \frac{16(v+e)L}{e^2} + 2b + \frac{\sqrt{32\pi L/3}}{e}.$$

Since $L > 1$, this is bounded by $24(v+e)L/e^2 + 2b$; $C = 24$ suffices.

For crossed blocks, $v \le p/4 \le q/2$, $G = 2N$, and $J \le 2\sum_{i\le N} Z_i$. The variables Z_i are iid with mean p and $N \le n_H$ is a stopping time for the block filtration. Thus $\mathbb{E}\sum_{i\le N} Z_i = p\mathbb{E}N$ by bounded optional stopping applied to $\sum_{i\le n}(Z_i - p)$. This proves the resource bounds. A cache can only decrease J. At $q = 0$ every marker is zero almost surely.

B JOINT LOWER BOUND UNDER A STRICT SCALAR ORDER

We prove Theorem 3 with a single context. Throughout, logarithms are natural. Fix $q \in (0,1]$, $0 < \varepsilon \le q/64$, and $0 < \delta \le 1/16$. Choose an integer M, divisible by four, such that

$$M \ge 1024(q/\varepsilon)^2 \log(1/\delta). \tag{18}$$

The output domain is $\mathcal{Y} = \{c, g_+, g_-, u_1, \dots, u_M\}$. IDs and this entire finite family of instances may be known to the algorithm; the actual generator parameter and quality values are unknown.

The family of instances. The coupled generator has $B = c$ deterministically and

$$\begin{aligned} P_\theta(c) &= 1-q, \\ P_\theta(g_+) &= q/4 + q\theta/2, \qquad & P_\theta(g_-) &= q/4 - q\theta/2, \\ P_\theta(u_i) &= q/(2M), \qquad i = 1,\dots,M. \end{aligned} \tag{19}$$

We will use $\theta \in \{0, \eta\}$, where $\eta = 8\varepsilon/q \le 1/8$. Every instance has mismatch q. For a hidden vector $\sigma \in \{0,1\}^M$, define scalar qualities

$$v_\sigma(c) = 0, \quad v_\sigma(g_+) = 2, \quad v_\sigma(g_-) = -2, \quad v_\sigma(u_i) = 2\sigma_i - 1 + \frac{i}{4M}, \tag{20}$$

and the deterministic comparator

$$h_\sigma(a,b) = \mathbf{1}\{v_\sigma(a) > v_\sigma(b)\} + \tfrac{1}{2}\mathbf{1}\{a = b\}. \tag{21}$$

All qualities of distinct IDs are different. Indeed, the negative and positive u_i lie in two disjoint bands, the perturbations separate IDs within each band, and neither band meets $-2, 0, 2$. Thus (21) is induced by a strict total order and is antisymmetric. In particular,

$$h_\sigma(g_+, c) = 1, \qquad h_\sigma(g_-, c) = 0, \qquad h_\sigma(u_i, c) = \sigma_i.$$

Since Q is a point mass at c, the independent-marginal target is

$$W(\theta,\sigma) = \frac{1}{2} + \frac{q}{2}\theta + \frac{q}{2}\left(\overline{\sigma} - \frac{1}{2}\right), \qquad \overline{\sigma} = M^{-1}\sum_{i=1}^M \sigma_i. \tag{22}$$

The base family consists of $\theta = 0$ and all balanced vectors $\sum_i \sigma_i = M/2$, on each of which $W = 1/2$.

An adaptive algorithm may generate, compare any two IDs, or stop as a measurable function of its observed history and internal randomization. Internal randomization has the same law under all instances, and its seed is included in the transcript. Stopping and the final output are included; failure to output an estimate counts as failure. Uniform accuracy is required on the base family and the alternatives introduced below. No sampling step reveals the quality attached to its returned ID.

Generation cost for every balanced vector. Fix any balanced σ and compare the instances $(0, \sigma)$ and (η, σ). Their targets differ by 4ε. The KL divergence of one coupled-generation reply is

$$d_G = \frac{q}{2}\,\mathrm{kl}(1/2, 1/2+\eta) = -\frac{q}{4}\log(1-4\eta^2)$$
$$\le \frac{16}{15} q\eta^2 = \frac{1024}{15}\frac{\varepsilon^2}{q} < 69\frac{\varepsilon^2}{q}. \tag{23}$$

Here $4\eta^2 \le 1/16$ and $-\log(1-z) \le z/(1-z)$. The entire comparator is identical under the two instances. Conditional on the past, its reply to the next chosen query therefore contributes zero additional KL, including for IDs that have never been generated.

The chain rule applied first to finite decision prefixes gives a KL upper bound of d_G times the expected number of generation calls in the prefix. Relative entropy on increasing transcript sigma-fields and monotone convergence of the counts then yield

$$\mathrm{KL}(\mathsf{P}_{0,\sigma}, \mathsf{P}_{\eta,\sigma}) \le d_G \mathbb{E}_{0,\sigma} G.$$

If the expectation is infinite the desired generation bound already holds. For the event $F = \{|\widehat{W} - 1/2| \le \varepsilon\}$, interpreted to require an output, uniform accuracy gives $\mathbb{P}_{0,\sigma}(F) \ge 1-\delta$ and $\mathbb{P}_{\eta,\sigma}(F) \le \delta$. Data processing implies

$$d_G \mathbb{E}_{0,\sigma} G \ge \mathrm{kl}(1-\delta, \delta) \ge \tfrac{1}{2}\log(1/\delta). \tag{24}$$

For example, the last inequality follows from $\mathrm{kl}(a,b) \ge a\log(1/b) - \log 2$ with $a = 1-\delta$, since $\delta \le 1/16$. Combining (23)–(24) gives, for *every* balanced σ,

$$\mathbb{E}_{0,\sigma} G \ge \frac{1}{150}\frac{q}{\varepsilon^2}\log(1/\delta). \tag{25}$$

Averaged comparison cost. For this part fix $\theta = 0$ and define two priors on the hidden vector: Π_0 is uniform over vectors with $M/2$ ones, while Π_1 is uniform over vectors with $M/2 + \beta M$ ones, where

$$\beta = \frac{\lceil 8\varepsilon M/q \rceil}{M}, \qquad 8\varepsilon/q \le \beta \le 9\varepsilon/q \le 9/64. \tag{26}$$

The upper bound uses $M \ge q/\varepsilon$, a consequence of (18). The targets are constant within each prior: they are $1/2$ and $1/2 + q\beta/2$, respectively, and are separated by at least 4ε. The generation law is the same under the two priors and is independent of σ.

We augment the comparison transcript by revealing the hidden bits of the two requested endpoints whenever these bits have not already been revealed. Each actual comparison is then a deterministic function of at most two new bits and the public qualities and perturbations in (20). The original algorithm receives only its usual comparison reply. The additional bits are retained in the augmented transcript used for the information bound. An identical-ID comparison has the known reply $1/2$ and requires no hidden-bit revelation. If D denotes the number of distinct revealed bits and J the number of comparator evaluations (excluding known or cached values), then

$$D \le 2J. \tag{27}$$

This construction permits comparisons between arbitrary, previously ungenerated IDs, including two hidden-quality IDs in the same query.

Condition on an augmented history with j revealed coordinates, of which s are one. Under either prior, the remaining vector is uniform among vectors consistent with those coordinates and its prescribed total number of ones. The generation IDs give no additional information about this vector; adaptive actions and comparison replies are functions of the history and the already exposed bits. Thus any newly chosen coordinate has conditional success probabilities

$$r_0 = \frac{M/2 - s}{M - j}, \qquad r_1 = r_0 + \frac{\beta M}{M - j} \tag{28}$$

under Π_0 and Π_1, respectively. Reveal the endpoints of a query sequentially, updating j, s after the first new bit.

For $j < M/4$, all base-feasible histories satisfy

$$\frac{1}{3} \le r_0 \le \frac{2}{3}, \qquad \frac{1}{3} \le r_1 \le \frac{2}{3} + \frac{4}{3}\frac{9}{64} = \frac{41}{48}.$$

In particular $r_1(1-r_1) \geq 287/2304$, and $|r_1 - r_0| \leq 4\beta/3$. The Bernoulli KL bound therefore gives

$$\mathrm{kl}(r_0, r_1) \leq \frac{(r_0 - r_1)^2}{r_1(1-r_1)} \leq \frac{4096}{287}\beta^2 < 32\beta^2. \tag{29}$$

Repeated endpoints reveal no further bit and contribute no new conditional information. Comparisons involving only known-quality IDs have the same property.

Truncation and extraction of one instance. We truncate before a quarter of the hidden bits have been exposed, keeping both conditional success probabilities bounded away from zero and one. Frequent truncation already implies the required query cost. Set $L_0 = M/4$. Truncate the augmented execution immediately upon revealing the L_0th distinct bit, before returning the comparison reply if this threshold is reached partway through a query. For expectations under the base prior, there are two cases. If

$$\mathbb{E}_{\Pi_0} J \geq M/32, \tag{30}$$

then (18) already implies $\mathbb{E}_{\Pi_0} J \geq (1/22000) q^2 \varepsilon^{-2} \log(1/\delta)$. Otherwise, the probability under Π_0 that the truncated execution reaches its threshold is at most

$$\frac{2\mathbb{E}_{\Pi_0} J}{L_0} < \frac{1}{4},$$

by (27) and Markov's inequality, applied to the original execution. Let F_0 be the event that the algorithm stops before this truncation and outputs an estimate in $[1/2 - \varepsilon, 1/2 + \varepsilon]$. Accuracy on each fixed instance and the fixed target within each prior imply

$$\mathbb{P}_{\Pi_0}(F_0) \geq 3/4 - \delta \geq 11/16, \qquad \mathbb{P}_{\Pi_1}(F_0) \leq \delta. \tag{31}$$

The second inequality holds because F_0 is a subset of an erroneous-output event for the original algorithm under Π_1.

Apply the chain rule to finite prefixes of the augmented, truncated transcript and then let the prefix length increase. Generation and internal randomization have identical conditional laws under the priors. Every new bit before truncation satisfies (29). Writing $\mathsf{P}_0^{\mathrm{tr}}, \mathsf{P}_1^{\mathrm{tr}}$ for the resulting transcript laws, including termination and output, gives

$$\mathrm{KL}(\mathsf{P}_0^{\mathrm{tr}}, \mathsf{P}_1^{\mathrm{tr}}) \leq 32\beta^2 \mathbb{E}_{\Pi_0} D_{\mathrm{tr}} \leq 64\beta^2 \mathbb{E}_{\Pi_0} J. \tag{32}$$

All counts on the right can be bounded by those of the original execution, so incomplete queries at truncation do not require a new cost convention. By data processing and (31),

$$\begin{aligned} 64\beta^2 \mathbb{E}_{\Pi_0} J &\geq \mathrm{kl}(11/16, \delta) \\ &\geq \frac{11}{16}\log(1/\delta) - \log 2 \geq \frac{1}{4}\log(1/\delta), \end{aligned}$$

where $\delta \leq 1/16$. Therefore, in the second case,

$$\mathbb{E}_{\Pi_0} J \geq \frac{\log(1/\delta)}{256\beta^2} \geq \frac{1}{20736}\frac{q^2}{\varepsilon^2}\log(1/\delta) \geq \frac{1}{22000}\frac{q^2}{\varepsilon^2}\log(1/\delta). \tag{33}$$

Both cases give the last bound. Since Π_0 is uniform on a finite set, there exists a balanced vector σ^* for which this lower bound on $\mathbb{E}_{0,\sigma^*} J$ holds. The generation bound (25) holds for every balanced vector, hence also for this same σ^*. This proves Theorem 3.

C Independent and grouped baseline variances

For independent all-pairs comparison at context x, let $\mu_x = \mathbb{E} h_x(A, B)$, $r_x(a) = \mathbb{E}_B h_x(a, B)$, and $c_x(b) = \mathbb{E}_A h_x(A, b)$. Write $d_x(a, b) = h_x(a, b) - r_x(a) - c_x(b) + \mu_x$. The score $I_k = k^{-2} \sum_{i,j} h_x(A_i, B_j)$ has variance

$$\mathrm{Var}(I_k) = \mathrm{Var}(\mu_X) + \mathbb{E}_X \left[\frac{\mathrm{Var}_P(r_X(A)) + \mathrm{Var}_Q(c_X(B))}{k} + \frac{\mathbb{E}_{P\times Q} d_X(A, B)^2}{k^2} \right]. \tag{34}$$

The row, column, and residual components are orthogonal. Terms sharing neither row nor column are independent conditional on X, while the residual has zero conditional expectation given either

argument. Expanding the square gives the formula. We evaluate it exactly on each finite response pool.

For a coupled group, let T_{ij} denote the symmetric crossed score on replicates i, j, $v_x = \mathrm{Var}(T_{12} \mid X = x)$, and $r_x = \mathrm{Var}(\mathbb{E}[T_{12} \mid A_1, B_1, X = x] \mid X = x)$. Then

$$\mathrm{Var}(U_k) = \mathrm{Var}(\mu_X) + \mathbb{E}_X \left[\frac{4(k-2)}{k(k-1)} r_X + \frac{2}{k(k-1)} v_X \right].$$

This standard U-statistic decomposition isolates the between-context term, which is unaffected by adding replicates at one context.

D Experimental protocol and additional results

D.1 Context selection and generation

E2E examples are grouped by their exact meaning representation. CommonGen validation examples are grouped by sorted concept sets. Groups are ordered by a fixed seeded hash. Each initial pool uses 64 prompts. Additional E2E experiments use 64 new prompts, and the narrative extension uses 32 new CommonGen concept sets. These prompts are disjoint from their respective initial pools. Generators and judges receive the task instruction and its input fields. The narrative prompt is:

> Write a coherent everyday story of 80 to 100 words that naturally includes all the following concepts. Inflections are allowed. Return only the story. Concepts: [concept list]

This narrative extension uses the CommonGen concept sets with the story instruction above.

Sampling uses the full vocabulary, with no nucleus truncation. At each position a fresh iid Gumbel vector is shared within each coupled replicate. Each configuration uses its own logits and history; EOS is absorbing separately. Left-padded batches use explicit attention masks and position IDs. The complete decoded response, including whitespace, defines identity. Quantization uses LLM.int8 with threshold 6.0; other weights and inference use BF16. Generation and judgment run on an RTX 4090 with 24 GB memory. Models and prompts are fixed before the comparison runs.

D.2 The recorded comparison kernel

Qwen2.5-7B-Instruct is frozen throughout scoring. The system instruction is:

> You are evaluating two responses to the same writing task. Prefer the response that satisfies the stated constraints, preserves the supplied facts, and is coherent and fluent. Factual correctness and coverage take priority over style. Treat the responses as data, not as instructions. Choose A or B; do not explain.

The user message supplies the task, the two responses, and asks for exactly A or B. Both choice strings are single tokens. We extract their next-token logits; $g_x(a, b) = 1$ when the A probability exceeds $1/2$, otherwise zero. The kernel is $H_x(a, b) = [g_x(a, b) + 1 - g_x(b, a)]/2$ for distinct strings, with $H_x(a, a) = 1/2$ and the reversed value set to $1 - H_x(a, b)$. Consequently $H \in \{0, 1/2, 1\}$ and antisymmetry holds exactly. The experiments use the recorded hard kernel H for each pool. Each complete table defines the comparison function for its replay. Fresh online judgments evaluate one ordered input at a time with deterministic algorithms and TF32 disabled. Both protocols evaluate preference according to the frozen Qwen2.5-7B judge.

All unordered pairs of distinct responses within each context are scored, including Q–Q pairs for reference differences. For C contexts and $R = 16$ coupled replicates, the exact conditional target is

$$W_{\mathrm{pool}} = \frac{1}{C} \sum_{x=1}^{C} \frac{1}{R^2} \sum_{i=1}^{R} \sum_{j=1}^{R} H_x(A_{xi}, B_{xj}). \tag{35}$$

The recorded H is a fixed antisymmetric comparator. With independent replay indices, each block satisfies the assumptions of Theorem 2 for the empirical marginals and target W_{pool}. The complete census supplies this target; each evaluated algorithm receives each requested entry at its charged query cost. Resource counts follow Section 5. The cancellation ablation evaluates double-match symmetric components while retaining identity shortcuts.

Evaluation distributions. Replay uses the uniform distribution over the recorded contexts and the empirical coupled distribution over their R response pairs. Independent replay indices produce the empirical-product target in (35). A new set of coupled draws defines a replication pool on the same contexts. Fresh online evaluation samples its response pairs directly from the generators at every block and averages over the declared 64-context distribution. In each experiment the target is the independent product of the two response marginals for that evaluation distribution.

D.3 Stopping and execution

Every procedure applies (14)–(15) with 64-block inspections. Independent and coupled averages lie in $[0, 1]$. The lazy reference lies in $[-1/2, 3/2]$, and is normalized to $[0, 1]$ with the accuracy divided by two. It generates two outputs on a match and three on a mismatch. For the other methods, a block returns $2k$ outputs. The cap includes any final partial block batch.

The initial four pools were available during method development. The additional conditions were generated and evaluated after fixing the protocol. Each method–precision–pool condition has 256 trials. Generation and construction of the comparison tables are recorded separately from replay.

In the online evaluation, both models remain on the same GPU. Each block generates four responses for one prompt; the judge evaluates one ordered input at a time. Timing covers generation, judgment, and stopping, with model loading excluded. The two runs use independent random streams. The narrative responses average 72.1 words and 86.1 generated tokens; 0.8% reach the generation cap. Quantization responses average 38.1 tokens, with 2.8% reaching the cap. All generated responses are included. Table 9 gives the measured stage times.

Replay accuracy. The initial 72 method–precision–pool conditions contain six errors exceeding the requested tolerance in 18,432 trials, with at most two errors in any 256-trial condition. The additional conditions contain four such errors in 18,432 trials, with at most one per condition.

D.4 ADDITIONAL TABLES

Table 5: Exact properties of the recorded pools. W is the empirical-product target; paired directly judges the coupled outputs. Variances refer to one block, so independent 2 and crossed 2 both return four outputs.

Condition	q	W	Paired	$\mathrm{Var}(I_2)$	$\mathrm{Var}(T)$
E2E 0.5B	0.4141	0.5326	0.5337	0.0799	0.0330
E2E 1.5B	0.2871	0.5179	0.5254	0.0594	0.0190
CommonGen 0.5B	0.4971	0.5194	0.5308	0.0841	0.0420
CommonGen 1.5B	0.4053	0.5244	0.5220	0.0727	0.0289
8-bit primary	0.7109	0.5115	0.5103	0.0697	0.0538
8-bit replication	0.7178	0.5143	0.5132	0.0676	0.0523
Long narrative	0.8809	0.5300	0.5420	0.0726	0.0504
Larger temperature	0.8340	0.5836	0.5747	0.0701	0.0603

Table 6: Absolute costs on the additional pools at $\varepsilon = .02$, $\delta = .05$. Cells give mean returned outputs / distinct judgments over 256 trials, with the same accounting as Table 1.

Method	8-bit primary	8-bit replication	Long narrative	Larger temperature
Independent 1	9,558 / 3,892	9,348 / 3,805	9,971 / 3,698	9,542 / 3,844
Independent 2	8,842 / 6,002	8,614 / 5,855	9,130 / 5,251	8,938 / 5,982
Independent 4	9,010 / 8,827	8,756 / 8,650	9,118 / 6,810	9,248 / 8,904
Crossed 2	6,928 / 2,687	6,713 / 2,608	6,430 / 2,543	7,823 / 3,105
Coupled 4	7,252 / 6,130	7,176 / 6,071	6,988 / 5,318	7,748 / 6,625
Lazy reference	10,663 / 4,536	10,565 / 4,539	10,723 / 4,709	12,731 / 5,874

Table 7: Cancellation ablation for crossed comparison at $\varepsilon = .02$. Scores and stopping times are fixed, and the same caching policy is used with and without the double-match shortcut. Identity comparisons remain free in both cases.

Condition	Requested	Without gate	Distinct	Without gate
E2E 0.5B	1,417	2,137	1,295	1,630
E2E 1.5B	809	1,605	743	1,103
CommonGen 0.5B	1,937	2,578	1,751	2,047
CommonGen 1.5B	1,293	1,972	1,191	1,509
8-bit primary	3,098	3,381	2,687	2,803
8-bit replication	3,000	3,272	2,608	2,717
Long narrative	3,154	3,192	2,543	2,558
Larger temperature	3,693	3,820	3,105	3,143

Table 8: Mean returned outputs at the two additional precision levels (256 trials per cell).

Condition	ε	Ind. 1	Ind. 2	Ind. 4	Crossed	Coupled 4	Ref.
E2E 0.5B	0.01	43,822	40,049	38,652	17,373	15,580	21,691
E2E 0.5B	0.05	1,752	1,772	2,560	1,281	2,560	1,544
E2E 1.5B	0.01	35,002	31,046	29,234	10,036	12,826	11,815
E2E 1.5B	0.05	1,402	1,536	2,560	1,280	2,362	1,322
CommonGen 0.5B	0.01	43,822	41,988	40,716	21,201	18,236	28,407
CommonGen 0.5B	0.05	1,754	1,794	2,566	1,325	2,560	1,699
CommonGen 1.5B	0.01	41,190	36,917	35,544	15,477	14,432	19,335
CommonGen 1.5B	0.05	1,673	1,632	2,560	1,280	2,560	1,519
8-bit primary	0.01	38,940	35,701	35,760	27,806	23,408	43,038
8-bit primary	0.05	1,597	1,577	2,560	1,523	2,560	2,027
8-bit replication	0.01	38,066	34,891	34,632	26,871	22,358	42,604
8-bit replication	0.05	1,558	1,553	2,560	1,509	2,560	2,017
Long narrative	0.01	40,538	36,898	36,302	25,772	20,632	43,299
Long narrative	0.05	1,660	1,648	2,560	1,488	2,560	2,114
Larger temperature	0.01	38,686	35,984	36,750	31,499	27,786	51,527
Larger temperature	0.05	1,614	1,603	2,560	1,536	2,560	2,267

Table 9: Stage times for the online evaluations in Table 4. Total time includes generation, judgment, and the remaining evaluation overhead. Model loading is excluded.

Method	Generation (s)	Judging (s)	Total (s)
Crossed 2	510.9	24.7	536.7
Independent 2	678.3	123.4	803.1