BOUNDARYMORPH: BUDGETED CROSS-ENCODER DOCUMENT SELECTION FOR DIFFUSE RETRIEVAL

Anonymous authors
Paper under double-blind review

ABSTRACT

Open-ended queries in modern Retrieval-Augmented Generation (RAG) are increasingly "diffuse," requiring a large set of documents to be assembled into a finite LLM context window. To ensure retrieval quality, systems use fast dual-encoders and more expensive cross-encoders (CEs) to score candidates. However, the CE budget B is strictly bounded by latency and is often smaller than the context window capacity k. This mismatch makes standard reranking structurally flawed: it wastes compute verifying obvious top candidates while ignoring relevant documents further down the initial ranking. To address this, we introduce BoundaryMORPH, a novel algorithm that allocates CE budget specifically for the LLM's context capacity k. Using a Gaussian Process, BoundaryMORPH treats the initial dual-encoder ranking as a structural prior and intelligently spends CE calls on resolving top-k set membership at the boundary, rather than seeking a single most-relevant document. Information from each CE call propagates to unscored documents, maximizing the utility of the budget. We demonstrate that BoundaryMORPH achieves state-of-the-art set retrieval quality across multiple models and datasets with open-ended queries ($+5.4$ pp nCG@100 over the strongest baseline).

1 INTRODUCTION

The challenge of serving queries whose relevant material is scattered across many documents has long been recognized in IR (Bates, 1989; Marchionini, 2006; Clarke et al., 2008), and they have grown more prominent as users bring broader, more open-ended requests to LLM-based systems (Suri et al., 2024; Spatharioti et al., 2025) and on to deep-research agents that survey and synthesize many sources per query (Huang et al., 2025; Coelho et al., 2025). For RAG, this takes the form of queries with *many* relevant documents; for example, in a deep research setting, a query like

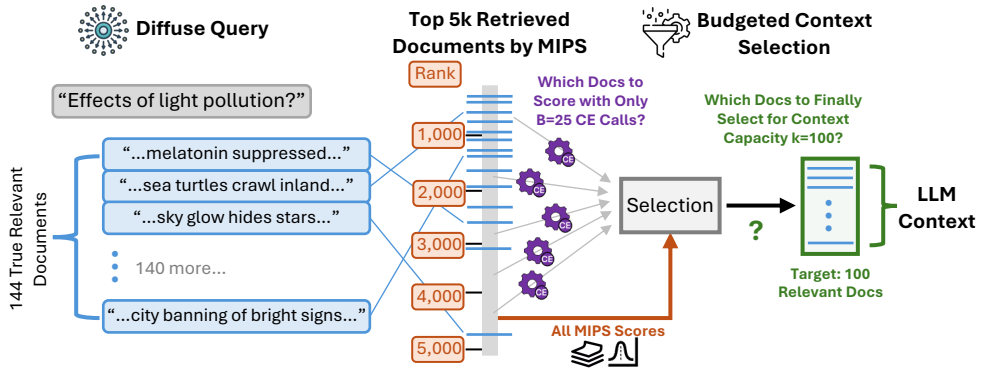

Figure 1: The diffuse query has *many* gold relevant documents (144), and they are spread across the MIPS top-5k ranking. The LLM generator has been allotted a context capacity $k = 100$, but we have only a CE budget of $B = 25$. The selection problem then becomes: how should we optimally spend these CE calls and finally select the 100 most relevant documents for the LLM?

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.

"Effects of light pollution?" is expected to surface tens, if not hundreds of relevant documents (see Figure 1)! Such **diffuse** queries cannot be well-responded to by referencing a single, or even handful of relevant documents; rather, they require a larger set of relevant documents to be inserted into the LLM generator's context for the best response.

Modern RAG systems typically rest on a two-stage architecture: a cheap **dual-encoder** embeds the query and every document into a shared vector space and returns a candidate pool ranked via **Maximum Inner Product Search** (MIPS) (Hambarde & Proença, 2023; Zhao et al., 2024; Huang & Huang, 2026), and an expensive reranker, such as a **cross-encoder** (CE) or LLM, re-scores a small number of top candidates far more accurately (Gao et al., 2024; Fan et al., 2024).

Because the CE is an expensive test-time process, it is typically applied to only a limited number, B (budget calls), of candidates. Likewise, a RAG system is always limited by its *context capacity*, k: the size of the context that has been allotted for documents being inserted into the RAG generator's context before generating a response (Cuconasu et al., 2024; Shi et al., 2024; Liu et al., 2024; Louis et al., 2025). These two limits become problematic for diffuse queries, for two reasons: (1) re-scoring only the top B dual-encoder documents risks missing out on relevant documents further down, and (2) when the CE budget is smaller than the context capacity ($B < k$), the remaining $k - B$ documents cannot anyway be evaluated by the CE due to budget constraint, and thus we gain no information about which documents to fit into the context, and which to leave out.

Figure 1 shows this clearly, where we assume a context capacity of $k = 100$, but only a $B = 25$ CE call budget: the query is diffuse, with 144 truly relevant documents, spread across the MIPS ranking. The context capacity $k = 100$ means we have 100 slots to fill our context with (hopefully) $100/144$ relevant documents. However, our budget allows for only 25 CE calls, which demonstrates problem (2). This begets a resource allocation problem: how should we optimally *spend* these CE calls and finally *select* the 100 most relevant documents for the RAG generator?

To address this, we introduce **BoundaryMORPH**, a Gaussian Process (GP) selector (Williams & Rasmussen, 1995; Srinivas et al., 2012) that actively estimates the CE's relevance landscape to optimally allocate the budget of B calls. The use of a GP allows us to *propagate* information from each CE call, so that even with only, say, 25 CE calls, we may still gain information about far more than 25 documents. BoundaryMORPH is built on two core insights regarding the nature of diffuse queries and the RAG pipeline:

1. **Morphing the MIPS's ranking rather than discarding it:** While the cheap dual-encoder is imperfect and locally miscalibrated, it provides a strong, scale-free global surface. This signal should be leveraged as a structural baseline rather than discarded.
2. **Targeting the capacity boundary rather than the maximum:** Because the RAG context capacity k is known in advance, spending budget to confidently confirm the absolute highest-scoring documents is wasteful; they will be included in the context regardless. Genuine uncertainty only exists at the **boundary** of inclusion.

BoundaryMORPH operationalizes these insights through a single running GP estimator. To satisfy the first insight, we set the GP prior mean to the document's MIPS percentile rank. This allows the model to smoothly fall back to the dual encoder's judgment in unscored regions, spending calls only to *bend* this surface rather than rebuild it. To satisfy the second, BoundaryMORPH abandons traditional global-maximum seeking, and instead, actively targets the inclusion boundary, spending its limited budget on the most contested boundary gaps, where a CE call can actually change which documents are delivered to the generator.

In summary, our main contributions are:

- **A New Formulation for RAG Selection:** We frame document selection for RAG as a budgeted top-k set-selection problem, and identify the structural mismatch of traditional approaches with this context selection problem.
- **State-of-the-Art Diffuse Retrieval:** We propose an efficient, parameter-free Gaussian Process selector that leverages a MIPS prior to warm-start the relevance landscape and utilizes a standardized boundary gap to explicitly target the top-k decision threshold. We demonstrate through extensive evaluation that BoundaryMORPH consistently outperforms existing active selection baselines (BAGEL (Kim et al., 2026), RGS (Xu & Chen, 2025), and single-stage reranking) at equal bud-

gets across multiple embedding-CE pairs, achieving the highest selection quality on both human relevance judgments (nCG@k) and cross-encoder fidelity (CE-nCG@k).
- **Analysis of Query Multimodality:** We introduce a geometric measure of query *multimodality*, revealing that our performance advantage stems from navigating complex relevance landscapes. Because diffuse queries often exhibit multiple semantic "peaks" or sub-topics, the primary mode is often over-measured. By avoiding redundant calls on securely top-k documents, Boundary-MORPH naturally explores secondary peaks, yielding gains on highly multimodal queries.

2 Problem Setting

Setup. We define a *diffuse* query to be a query which requires more than a single or a handful of relevant documents for the best response. Unlike simple factoid queries that require a single document, diffuse queries (e.g., "Effects of Light Pollution?") require tens or hundreds. In this work, we focus on the latter. A query q has an ground truth, unobserved relevance function $\mathrm{rel}(q, d) \in [0, 1]$ over documents $d \in D$ from corpus D. Two scoring functions can estimate it. The **retriever** $s(q, d)$ is cheap and amortized across queries, and narrows the corpus to a candidate pool C with $|C| = N$. In practice, N is a standard first-stage retrieval depth chosen to be large enough to bound maximum recall. The cross-encoder **(CE)**, $f_{\mathrm{CE}}(q, d) \in [0, 1]$, is the more accurate estimate and the more expensive one: it must be evaluated separately for each query-document pair, and we may call it at most $B \ll N$ times per query. The consumer of the results accepts a fixed number k of documents, the **context capacity**. The system returns a set $S \subseteq C$ with $|S| = k$.

Objective. We want the returned set to carry as much relevance as possible,

$$S^* = \underset{S \subseteq C,\ |S|=k}{\arg\max} \sum_{d \in S} \mathrm{rel}(q, d). \tag{1}$$

Given rel, this is trivial: return the k most relevant documents. The difficulty is that rel is unknown, that f_{CE} is the closest estimate of it we can obtain, and that we can afford only $B \ll N$ calls to f_{CE}. The problem is therefore one of resource allocation: which B documents to score, so that the k we return afterwards are as relevant as possible.

Context capacity. The capacity k is the number of retrieved documents given to the RAG generator before it answers. In deployed RAG pipelines this is a managed allotment: a finite context window may be divided among the system prompt, tool schemas, conversation history, retrieved documents, and response headroom. Hence, the slice reserved for retrieved documents, k, is known, and the task is one of **selection**: to choose which k documents to return (a set). We view arranging them within the context as a separate problem Cuconasu et al. (2025), and in this work, focus on the task of *selecting* the best k documents to fit into the model context. The problem therefore contains two decisions: **where** to spend the B CE calls, and **which** k documents to return once they are spent.

3 Method

At a high level, our approach uses a Gaussian Process (GP) fitted with specialized components for this task. We first introduce GPs and the standard way such an estimator is used to choose what to score. The two subsections that follow are the specializations fitting it to budgeted selection under a known capacity: (1) what the estimator starts from, and (2) how it picks the next document. Full pseudocode is given in Appendix B.1.

3.1 Preliminaries: Gaussian-process surrogates

The estimation problem. The cross-encoder f_{CE} may be evaluated at B points but must inform a decision over all N. That calls for a *surrogate*: a model that predicts $f_{\mathrm{CE}}(q, d)$ at every unscored document and reports how far each prediction can be trusted. Uncertainty is necessary, since the selector needs it to tell a resolved region from an unexamined one.

Gaussian processes. A Gaussian process is a distribution over functions, $g \sim \mathcal{GP}(\mu_0, K)$, specified by two objects (Williams & Rasmussen, 1995): a **prior mean** $\mu_0(d)$, the estimate of $g(d)$ before any evidence, and a **kernel** $K(d, d')$, which sets how strongly an observation at d' constrains the

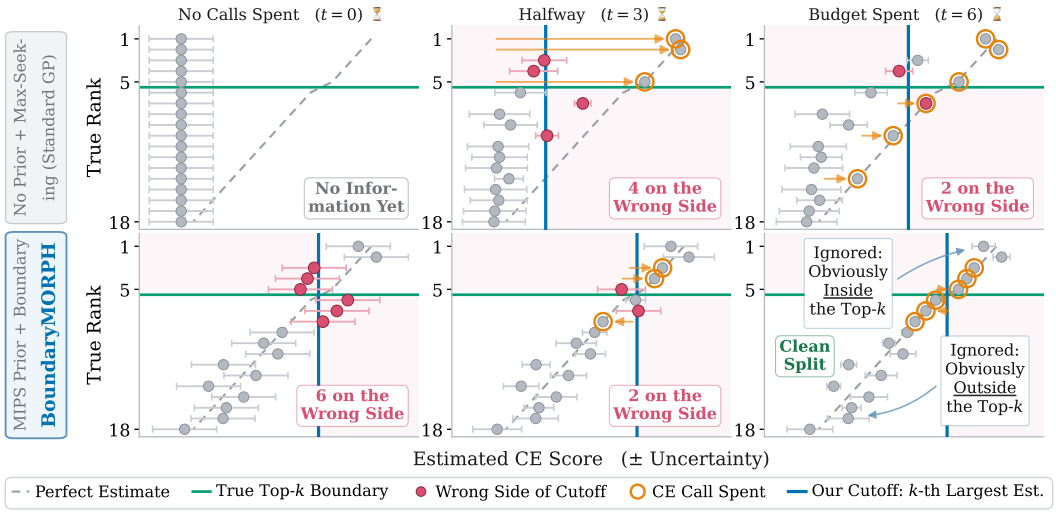

Figure 2: BoundaryMORPH vs. a standard GP. **Top row:** A standard GP (zero prior, max-seeking acquisition) wastes limited budget repeatedly scoring obvious top documents. **Bottom row:** BoundaryMORPH leverages a MIPS prior and targets the top-k boundary. By ignoring documents safely inside or outside the cutoff, it spends its budget exclusively on resolving contested edge cases.

estimate at d and is typically a decreasing function of the distance between the two documents. It is nonparametric (no weights), and all posteriors are exactly computed. Here g is a generic function placed under the GP; §3.2 instantiates it as the residual of f_{CE} against the MIPS percentile prior.

Closed-form conditioning. After t calls of g on $A_t = \{d_1, \ldots, d_t\}$ returning scores y_t, the posterior at any document d is Gaussian with mean and variance

$$\mu_t(d) = \mu_0(d) + \mathbf{k}_t(d)^\top \widetilde{K}_t^{-1}\big(y_t - \mu_0(A_t)\big), \qquad \sigma_t^2(d) = K(d,d) - \mathbf{k}_t(d)^\top \widetilde{K}_t^{-1}\,\mathbf{k}_t(d), \tag{2}$$

where $\mathbf{k}_t(d) = \mathbf{k}(d, A_t)$ is the vector of kernel values between d and the scored documents and $\widetilde{K}_t = K(A_t, A_t) + \sigma_n^2 I$ with σ_n^2 a very small observation-noise (jitter) term. Both are available for all N documents at once, and both update as each new score arrives.

Two properties of these formulas are what selectors can run on. First, **uncertainty collapses around evidence**: at a scored document $\sigma_t^2 \approx \sigma_n^2 \approx 0$, and at its kernel-neighbors σ_t is reduced too, so a region can be recognized as resolved. Second, **the mean reverts to the prior far from evidence**: as $\mathbf{k}_t(d) \to 0$ the correction term vanishes and $\mu_t(d) \to \mu_0(d)$. The posterior is a *local deformation* of the prior around the points that were paid for.

Acquisition: GP-UCB. Given a surrogate, the standard rule for choosing the next evaluation is the upper confidence bound (Srinivas et al., 2012; Djolonga et al., 2013),

$$d^\star \;=\; \operatorname*{arg\,max}_{d \notin A_t} \; \mu_t(d) \;+\; \beta\,\sigma_t(d), \tag{3}$$

which trades **exploiting** a high estimated score against **exploring** an uncertain one, with the weight β setting the balance (commonly tuned or annealed). GP-UCB is formulated and analyzed as a method for locating the **global maximum** of g with few function evaluations. A GP surrogate thus leaves exactly two things to specify: the pair (μ_0, K), and the acquisition rule.

3.2 The MIPS percentile prior

MIPS prior. While the MIPS ranking may not be as accurate as CE relevance, it is certainly not noise, and it is free (pre-computed). It encodes mostly-reliable geometry that may be miscalibrated and locally wrong. To take advantage of this, we have the GP model the **residual**, $r(d) = f_{\text{CE}}(q, d) - \mu_0(d)$, rather than the raw CE field. That is, we aim to approximate where,

and how much to **morph** the embedding similarity in order to reach CE relevance. Because the CE outputs are calibrated probabilities in $[0, 1]$, we can meaningfully model the residual against a $[0, 1]$ MIPS percentile rank. We set $\mu_0(d)$ to the document's **MIPS percentile rank** within C (≈ 1 at the head, ≈ 0 in the tail), which is scale-free, requiring no per-query calibration. The posterior mean is then μ_0 plus kernel-weighted observed residuals. Far from any observation $\mu_t \to \mu_0$: the surrogate falls back to trusting the MIPS's ranking. Calls *bend* the MIPS's surface; they never rebuild it. This is also the easier estimation problem: the residual is near zero across most of the pool, so "no correction" is a good default almost everywhere and the GP only has to represent the surprises.

Kernel and query observation. We use a directional kernel (vMF) on unit embeddings, $K(d, d') = \exp\left(\kappa\left(\cos(d, d') - 1\right)\right)$ (Mukhopadhyay et al., 2020; Mardia & Jupp, 2009). Alternative kernels are compared in Appendix B.4. Residuals are Z-standardized before conditioning and de-standardized after. Similar to Kim et al. (2026), we enter the query vector into the GP before any budget is spent, as a zero-cost observation at maximal relevance ($y = 1$), as if we had observed a maximally relevant document. This costs no CE call and gives the surrogate one anchor of known value from the start, and pinches the variance near the query.

3.3 The boundary acquisition rule

The mismatch of maximum-seeking acquisition. The objective from §2 targets *membership* in a set of size k. We argue that standard maximum-seeking acquisition (e.g., GP-UCB) is fundamentally misaligned with this goal. First, it wastes calls on documents already securely inside the top-k. Their exact *value* is uncertain, but their *membership* is not; they will enter the context window whether or not we pay to measure them. Symmetrically, it explores regions securely below the cutoff, confirming exclusions that were never in doubt. A maximum-seeker has no concept of the known capacity k, so it cannot ignore documents whose fate is already settled. Instead, a useful CE call is one that could plausibly *move a document across* the boundary, as only these calls actually change the final context fed to the LLM.

Relation to active set ordering. Our selection rule is adapted from Mean Prediction in Nguyen et al. (2024) and, in the independent-arm case, from LUCB (Kalyanakrishnan et al.), which estimates the **top-k set** of a blackbox function rather than its maximizer, and spends its evaluations on the boundary comparisons the posterior leaves ambiguous. Notably, standard GP-UCB is the $k = 1$ special case: a max-seeking selector is inherently minimizing regret of the global maximum.

Rule. At each step, we partition C by the current posterior mean μ_t into its **top-k**, S_t (the *incumbents*, our running guess at the returned set S), and the rest, $C \setminus S_t$ (the *challengers*). See Figure 2's purple line. Membership is in genuine doubt only at the boundary: an incumbent whose lower confidence bound dips below a challenger risks being wrongly retained, and a challenger whose upper confidence bound rises above an incumbent risks being wrongly dropped. The contested pairs are those close in estimated score *relative to how uncertain we are about them*, so we select the pair with the smallest **standardized boundary gap**,

$$(j^*, i^*) \;=\; \underset{j \in S_t,\; i \in C \setminus S_t}{\arg\min} \; \frac{\mu_t(j) - \mu_t(i)}{\sigma_t(j) + \sigma_t(i)}, \tag{4}$$

the pair most likely to sit on the wrong sides of the cutoff under the current estimates. We then spend the CE call on whichever of the two has **higher** σ (the outcome we can least predict). Equivalently, j^* is the most-threatened incumbent and i^* the strongest challenger.

Complexity. The kernel is built once, $O(N^2)$: one entry per document pair at test time. Updating the posterior requires $O(B^3 N)$, due to the matrix inversion, but we find in Appendix D.4 that $O(B)$ CE inference dominates the method's latency. Full derivation in Appendix B.2.

4 Experimental Setup

4.1 Datasets

Datasets. We evaluate on three datasets featuring broad, open-ended queries over news, government, and encyclopedic corpora: NeuCLIRBench-monolingual-eng (Lawrie et al., 2025), Robust04 (Voorhees, 2005), and TravelDest (Wen et al., 2024), with relevance labels from Kim et al. (2026).

To ensure diffuseness, we filter out the minority of queries with < 20 relevant documents, leaving 85, 187, and 99 queries, respectively, with median relevant document counts of **51 / 62 / 462**. To make the setting tractable and realistic, we restrict evaluation to the top $N = 10,000$ by Qwen3-Embedding-4B MIPS (Zhang et al., 2025) for each query. Full statistics are in Appendix C.1.

4.2 METRICS

Using nCG@k and CE-nCG@k. We use normalized cumulative gain (**nCG@k**) against human relevance judgments as our primary metric (Järvelin & Kekäläinen, 2002). Intuitively, this metric measures, *"compared to the **best possible set** of k documents that could have been retrieved, how relevant was the set that **was** retrieved?"*. We also report **CE-nCG@k**, the same quantity computed against the CE's scores rather than human judgments. Concretely:

$$\text{nCG@}k = \frac{\sum_{d \in S} \text{rel}(q,d)}{\sum_{d \in S^*} \text{rel}(q,d)}, \qquad \text{CE-nCG@}k = \frac{\sum_{d \in S} f_{\text{CE}}(q,d)}{\sum_{d \in S^\dagger} f_{\text{CE}}(q,d)}, \tag{5}$$

where S is the selected set of size k, S^* is the optimal set from Eq. 1, and $S^\dagger$ is the analogous optimal set of size k under the CE's own scores. Hence, **nCG@k** is the count of relevant documents returned, divided by the number a perfect selector would fit into k slots. It reaches 1 when the returned set is as relevant as the most relevant possible set. Reporting both metrics separates two claims: CE-nCG measures how well the budget is allocated to approximate f_{CE}, while nCG@k measures the retrieval quality against human judgments, rather than only tracking the CE more faithfully.

Why not recall or nDCG@k. Because relevant documents often exceed k, nCG@k directly asks: *of the k documents the generator can read, how many are relevant?* Metrics such as nDCG and MAP score the *ordering* within the returned set, which are not well-suited for measuring which k documents reach the generator's context (Trappolini et al., 2025; Samuel et al., 2026).

4.3 BASELINES

The dual-encoder (embedding model) and CE are **multilingual-e5-large** (Wang et al., 2024) and **zerank-2** (Pipitone et al., 2025) respectively, with other models in Appendix D.2, and implementation details in B.3). **BM25** (Robertson et al., 1994) is a lexical floor: applied as a re-ranking of the shared top-10k pool and truncated at k, giving one fixed ordering per query. **Single-Stage Rerank (SS)** is the standard policy: score the MIPS top-B with the CE, place them above the tail, and leave the tail in MIPS order. **BAGEL** (Kim et al., 2026) is a previous GP approach using zero-prior, and standard UCB over the whole pool with an explore/exploit term which we sweep. **RGS** (Xu & Chen, 2025) is a greedy kNN graph-based approach. Because it lacks a way to rank documents it never scored, scored documents are placed as a band above the unscored tail, which is left in MIPS order. RGS and BAGEL's hyperparameters are each tuned and selected on the evaluation metric per budget. Full implementation and sweep details for every baseline are in Appendix C.2.

5 RESULTS AND ANALYSIS

The evaluation addresses four questions in turn: (§5.1) Does BoundaryMORPH select better sets than existing selectors at equal budget? (§5.2) Are both of its components necessary (MIPS prior/boundary acquisition), and robust to the capacity k? (§5.3) How does BoundaryMORPH achieve its improvement, and which queries, geometrically speaking, yield the greatest advantage? In Appendix D.4, we also compare wall-clock latency, including against an agentic retrieval loop.

5.1 MAIN RESULTS

BoundaryMORPH outperforms baselines in all configurations. Table 1 shows the main experimental results. BoundaryMORPH outperforms baselines in all (dataset $\times$ B $\times$ k) combinations, with the biggest gains being at higher k values (average of +3.8 pp nCG@100 and +7.3 pp CE-nCG@200 over second-best baseline). BAGEL outperforms the other baselines, because it is the only baseline which also can use CE calls to update the relevance of *unscored* documents. In Appendix D.2, we show BoundaryMORPH's performance gap holds across choice of embedding and

Table 1: Top-k relevance (%) under CE budget B for various methods. Best per column in **bold**, second-best underlined. BoundaryMORPH outperforms baselines, with BAGEL a clear second, due to the ability to estimate relevance of unscored documents.

		nCG@k (%, human judgments)						CE-nCG@k (%)					
		B=25			B=50			B=25			B=50		
	Method / k	50	100	200	50	100	200	50	100	200	50	100	200
NeuCLIR	BM25	38.7	46.8	57.6	38.7	46.8	57.6	69.8	67.3	66.5	69.8	67.3	66.5
	SS	39.6	46.2	55.4	39.6	46.2	55.4	73.1	70.0	68.5	73.1	70.0	68.5
	RGS	39.1	46.9	55.7	37.0	47.5	56.6	72.3	70.3	68.7	68.8	70.3	69.1
	BAGEL	47.3	51.4	58.5	52.9	55.7	62.3	79.7	71.5	66.9	86.9	77.6	71.6
	MORPH	**49.8**	**55.1**	**62.7**	**54.7**	**59.2**	**65.1**	**85.7**	**80.9**	**77.3**	**90.5**	**85.7**	**81.9**
Robust04	BM25	34.7	40.0	49.7	34.7	40.0	49.7	49.7	47.6	47.8	49.7	47.6	47.8
	SS	34.4	37.1	44.8	34.4	37.1	44.8	54.5	51.5	51.0	54.5	51.5	51.0
	RGS	35.0	38.1	45.7	34.2	38.8	46.6	54.8	52.0	51.4	53.5	52.3	51.8
	BAGEL	40.3	41.0	46.1	48.7	48.0	53.0	59.9	53.2	50.2	68.4	59.6	55.2
	MORPH	**45.2**	**46.4**	**52.1**	**52.3**	**51.5**	**57.1**	**66.0**	**60.5**	**57.3**	**72.8**	**65.8**	**61.6**
TravelDest	BM25	23.0	21.9	23.0	23.0	21.9	23.0	57.3	57.4	57.7	57.3	57.4	57.7
	SS	30.9	28.4	27.9	30.9	28.4	27.9	64.2	64.5	64.8	64.2	64.5	64.8
	RGS	32.3	29.4	28.6	32.9	30.4	29.3	63.9	64.5	65.0	64.4	65.3	65.5
	BAGEL	40.9	35.5	33.7	46.6	40.7	37.5	75.9	72.4	69.9	82.0	77.5	74.5
	MORPH	**45.1**	**39.8**	**37.8**	**48.9**	**43.1**	**40.8**	**81.1**	**78.4**	**76.5**	**86.5**	**82.9**	**80.3**

CE model. We also see that **CE gains carry over to human judgments.** The trend persists across the two metrics, indicating that better approximation of the CE's own top-k translates into more human-judged-relevant documents delivered.

5.2 Ablations and Sensitivity

Table 2: Ablations: nCG@k (%) averaged over datasets and embedder×CE pairs; indented rows give the pp change.

	B=25			B=50		
k	50	100	200	50	100	200
MORPH	47.1	47.9	52.6	51.3	51.3	54.9
w/o acq.	-2.6	-2.5	-2.1	-1.5	-2.3	-1.5
w/o prior	-3.0	-3.1	-3.8	-2.1	-2.3	-2.3

Both the MIPS prior and the boundary rule are necessary. To test the contribution of BoundaryMORPH's two components, we perform two isolated removals, averaged over the four model pairs and three datasets: **without acquisition**, the boundary acquisition rule reverts to standard UCB at its own best swept β; **without prior**, the MIPS percentile prior mean becomes a zero mean.

As seen from Table 2, removing boundary acquisition lowers nCG@k by 1.5–2.6 percentage points, and removing the prior by 2.1–3.8. The drop holds in every per-dataset and CE-nCG cell as well (Appendix D.1), which verifies the utility of each component separately.

BoundaryMORPH exploits its knowledge of the context capacity. To test how the input context capacity k impacts our algorithm, we run BoundaryMORPH with varying values of k (the optimized value, k_{opt}). However, instead of evaluating on the corresponding CE-nCG@k_{opt}, we evaluate at *many* capacity values k_{eval}, to see what happens when we evaluate at a *different* capacity from what we gave our method. As seen in Figure 3(a), for nearly all evaluated k_{eval}, the run that was given $k_{\text{eval}} = k_{\text{opt}}$ is the best of all runs. That is, BoundaryMORPH truly is optimizing for k_{opt}, and fundamentally exploits the final evaluation's disregard for order within and outside the top-k.

5.3 Sources of Improvement

Budget is spent at the boundary rather than on secure incumbents. To investigate how BoundaryMORPH achieves performance gains, we analyze how each method allocates its budget. Figure 3(b) reports for 3 methods: among the sum relevance in the final top-k, how much relevance came from *CE-scored* documents vs. *unscored*. We see that BoundaryMORPH scores far *fewer* of the documents it finally returns. It declines to spend calls on high-confidence, high-posterior documents whose membership is not in doubt, freeing budget for the boundary where a call can flip the answer, whereas BAGEL spends far more calls at the top: seeking a global maximum. Purely

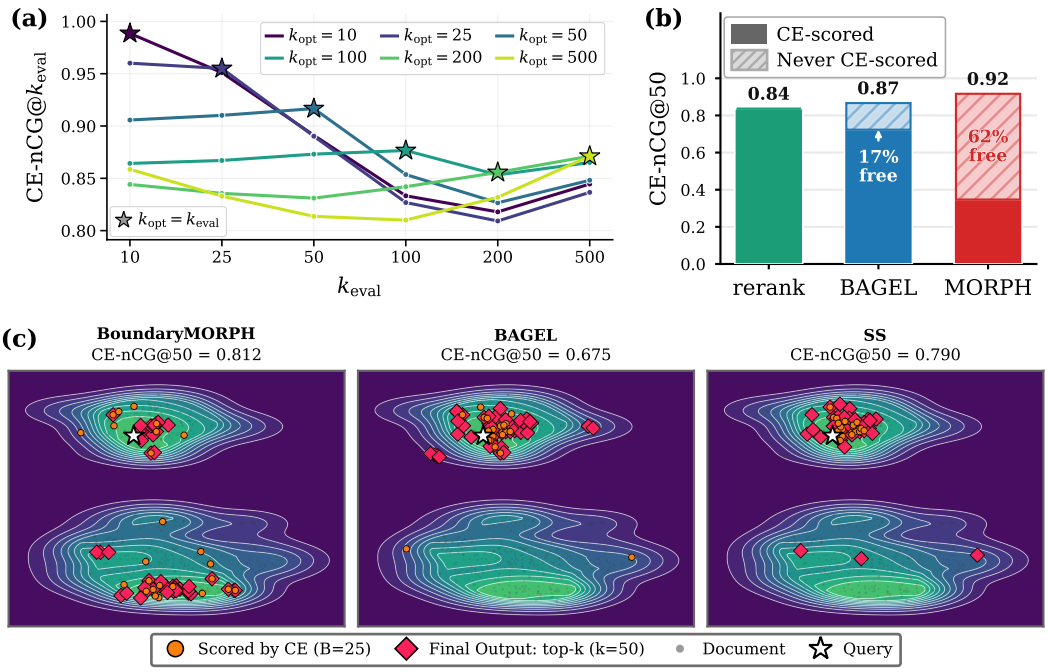

Figure 3: ***(a)***: **the acquisition rule optimizes for the exact capacity it is given.** We plot CE-nCG at different *evaluated* capacities k_{eval} (x-axis), with one curve per *optimized* capacity k_{opt} (the one given to the algorithm); when evaluated at the optimized k, k_{eval}=k_{opt} (stars), performance peaks there. ***(b)***: **BoundaryMORPH wins *because* it does not spend its budget measuring the obvious top.** Share of each method's top-k CE-nCG total coming from documents it never scored ("free"), at B=50, k=50. ***(c)***: **Skipping the obvious top buys the budget to reach a second mode.** One query's pool, plotted via t-SNE over document embeddings and contoured by CE relevance, with each method's *scored* and finally *selected* documents marked. Because BoundaryMORPH does not score documents obviously inside the top-k, it has budget left to explore, find a second mode, and exploit it, while max-seeking baselines concentrate calls on the primary mode.

reranking with the CE by definition can only include documents that are scored. Figure 3(c) shows one NeuCLIR query's *scored* and *finally selected* documents on its relevance contour: by avoiding over-scoring the main mode, BoundaryMORPH is able to cross over and explore a secondary mode.

The advantage increases with query multimodality. Figure 3(c) suggests the advantage comes from reaching secondary relevance modes, so we measure how *multimodal* each query is: whether its relevant documents form one tight group or several distinct ones. Treating CE score as height over an embedding kNN graph, we extract relevance *peaks* with persistence-based clustering (Chazal et al., 2013) and define a query's *multimodal mass* as the prominence-weighted relevance outside its tallest peak (Appendix E.1). Figure 4 contrasts a multimodal query (*light pollution*), whose peaks are distinct sub-topics, with a unimodal one (*Amazon deforestation*). On NeuCLIR, multimodal mass predicts per-query advantage over BAGEL (Spearman $\boldsymbol{\rho = +0.59}$, $p = 3.5 \times 10^{-9}$): **BoundaryMORPH gains most where it must reach "across" valleys to separate modes.** An LLM audit confirms that peaks are semantically coherent groups of documents (Appendix E.1).

6 Related Work

Budgeted reranking. *Adaptive* retrieval dynamically chooses what to score at query time. GAR (MacAvaney et al., 2022), Quam (Rathee et al., 2025b), SlideGar (Rathee et al., 2025a), and RGS (Xu & Chen, 2025) iteratively expand graph neighborhoods of high-scoring candidates, while budget-aware cascades (Rashid et al., 2024; Li et al., 2026) and list truncation (Meng et al., 2024) dynamically halt reranking over a fixed order. Crucially, all are value-seeking and cannot estimate

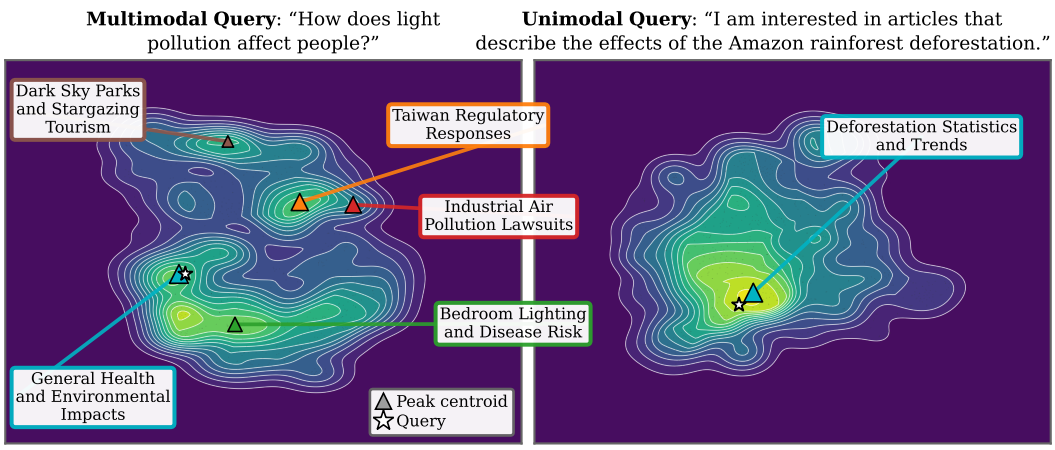

Figure 4: **Geometric peaks are genuinely distinct sub-topics.** CE relevance contours over t-SNE embedding space for a *multimodal* and a *unimodal* query, with the extracted peaks and LLM-generated peak labels. The multimodal query retains 5 peaks, each a distinct facet of the information need rather than one blurred high-relevance region; the unimodal query (Amazon deforestation) retains one. Peak extraction details are in Appendix E.1.

an unscored document. Closest in intent is AcuRank (Yoon et al., 2025), which maintains Bayesian beliefs to resolve top-k membership doubt. However, its beliefs update independently, meaning an unscored document's posterior never updates based on a scored neighbor's. Methodologically closest is BAGEL (Kim et al., 2026), a pool GP differing via deterministic warm-starting, a zero prior mean, and a maximum-seeking acquisition (GP-UCB).

Budgeted Selection Goals. Level-set estimation samples where confidence intervals straddle a known threshold (Bryan et al., 2005; Gotovos et al., 2013). Relatedly, active set ordering (Nguyen et al., 2024) and LUCB-style bandits over independent arms (Kalyanakrishnan et al., 2012; Jiang et al., 2017; Pony et al., 2026) resolve top-k membership for a known k. We observe that budgeted retrieval is fundamentally an active set ordering problem.

Indexing-time methods and agentic retrieval. Orthogonal offline methods optimize the corpus before querying via document expansion (Nogueira et al., 2019), reranker distillation (Schlatt et al., 2025), or LLM-built graphs (Sarthi et al., 2024; Edge et al., 2025; Gutiérrez et al., 2024). These change *what the pool contains*; we optimize which members reach the generator. Another orthogonal family uses LLMs to interleave search and generation (Khattab et al., 2023; 2024; Jiang et al., 2023; Jeong et al., 2024), sometimes via reinforcement learning (Jin et al., 2025), which forms the basis of deep-research agents (Huang et al., 2025; Coelho et al., 2025). Because these systems treat the retriever as a standalone tool, our budgeted selector can trivially plug into their loops to raise accuracy and cut search calls (Sharifymoghaddam & Lin, 2026; Meng et al., 2026).

7 Conclusion

We introduced BoundaryMORPH, a budget-aware document selection method for diffuse Retrieval-Augmented Generation. Rather than wasting limited cross-encoder calls on documents securely within the context capacity, BoundaryMORPH uses a Gaussian Process with a MIPS prior to explicitly target the top-k decision boundary. This approach achieves state-of-the-art selection quality over existing maximum-seeking baselines. By avoiding redundant measurements of primary topics, it successfully navigates multimodal queries to capture diverse semantic peaks, making it exceptionally well-suited for complex, open-ended retrieval applications. While BoundaryMORPH offers significant advantages for diffuse retrieval, we acknowledge certain system and algorithmic limitations. Specifically, the method currently relies on sequential execution, assumes a discrete document capacity k rather than strict token limits, and scales cubically with extreme budgets. We discuss these limitations and potential future directions in detail in Appendix A.

AI USE STATEMENT

In this work, we used generative AI tools for polishing writing, including shortening and feedback. All text was manually written or reviewed. Additionally, AI tools were used for literature discovery and summarization. All cited works were manually curated with a citation software, and checked for the individual claims for which they were cited. AI tools were also used for coding, but artifacts were manually verified and tested. We have reviewed all AI-assisted work. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

Appendix B contains all details necessary for reproducing our method (including pseudocode), while Appendix C contains all details necessary to replicate our experiments, including dataset processing details, baseline details, and prompts. We also include with this submission an anonymized codebase implementing our method and the baselines.

REFERENCES

Marcia J. Bates. The design of browsing and berrypicking techniques for the online search interface. *Online Review*, 13(5):407–424, May 1989. ISSN 0309-314X. doi: 10.1108/eb024320. URL https://doi.org/10.1108/eb024320.

Brent Bryan, Robert C. Nichol, Christopher R Genovese, Jeff Schneider, Christopher J. Miller, and Larry Wasserman. Active Learning For Identifying Function Threshold Boundaries. In *Advances in Neural Information Processing Systems*, volume 18. MIT Press, 2005. URL https://proceedings.neurips.cc/paper/2005/hash/8e930496927757aac0dbd2438cb3f4f6-Abstract.html.

Frédéric Chazal, Leonidas J. Guibas, Steve Y. Oudot, and Primoz Skraba. Persistence-Based Clustering in Riemannian Manifolds. *Journal of the ACM (JACM)*, 60(6):41:1–41:38, November 2013. ISSN 0004-5411. doi: 10.1145/2535927. URL https://dl.acm.org/doi/10.1145/2535927.

Hung-Ting Chen, Xiang Liu, Shauli Ravfogel, and Eunsol Choi. Beyond Single Embeddings: Capturing Diverse Targets with Multi-Query Retrieval, 2025. URL https://arxiv.org/abs/2511.02770. Version Number: 1.

Charles L.A. Clarke, Maheedhar Kolla, Gordon V. Cormack, Olga Vechtomova, Azin Ashkan, Stefan Büttcher, and Ian MacKinnon. Novelty and diversity in information retrieval evaluation. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '08, pp. 659–666, New York, NY, USA, July 2008. Association for Computing Machinery. ISBN 978-1-60558-164-4. doi: 10.1145/1390334.1390446. URL https://dl.acm.org/doi/10.1145/1390334.1390446.

João Coelho, Jingjie Ning, Jingyuan He, Kangrui Mao, Abhijay Paladugu, Pranav Setlur, Jiahe Jin, Jamie Callan, João Magalhães, Bruno Martins, and Chenyan Xiong. DeepResearchGym: A Free, Transparent, and Reproducible Evaluation Sandbox for Deep Research, December 2025. URL http://arxiv.org/abs/2505.19253. arXiv:2505.19253 [cs.IR].

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. The Power of Noise: Redefining Retrieval for RAG Systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 719–729, July 2024. doi: 10.1145/3626772.3657834. URL http://arxiv.org/abs/2401.14887. arXiv:2401.14887 [cs.IR].

Florin Cuconasu, Simone Filice, Guy Horowitz, Yoelle Maarek, and Fabrizio Silvestri. Do RAG Systems Really Suffer From Positional Bias?, October 2025. URL http://arxiv.org/abs/2505.15561. arXiv:2505.15561 [cs.CL].

Josip Djolonga, Andreas Krause, and Volkan Cevher. High-Dimensional Gaussian Process Bandits. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates,

Inc., 2013. URL https://proceedings.neurips.cc/paper_files/paper/2013/hash/8d34201a5b85900908db6cae92723617-Abstract.html.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From Local to Global: A Graph RAG Approach to Query-Focused Summarization, February 2025. URL http://arxiv.org/abs/2404.16130. arXiv:2404.16130 [cs.CL].

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, KDD '24, pp. 6491–6501, New York, NY, USA, August 2024. Association for Computing Machinery. ISBN 979-8-4007-0490-1. doi: 10.1145/3637528.3671470. URL https://dl.acm.org/doi/10.1145/3637528.3671470.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-Augmented Generation for Large Language Models: A Survey, March 2024. URL http://arxiv.org/abs/2312.10997. arXiv:2312.10997 [cs.CL].

Alkis Gotovos, Nathalie Casati, Gregory Hitz, and Andreas Krause. Active learning for level set estimation. In *Proceedings of the Twenty-Third international joint conference on Artificial Intelligence*, IJCAI '13, pp. 1344–1350, Beijing, China, August 2013. AAAI Press. ISBN 978-1-57735-633-2.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically Inspired Long-Term Memory for Large Language Models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59532–59569. Curran Associates, Inc., 2024. doi: 10.52202/079017-1902. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/6ddc001d07ca4f319af96a3024f6dbd1-Abstract-Conference.html.

Kailash A. Hambarde and Hugo Proença. Information Retrieval: Recent Advances and Beyond. *IEEE Access*, 11:76581–76604, 2023. ISSN 2169-3536. doi: 10.1109/ACCESS.2023.3295776. URL https://ieeexplore.ieee.org/abstract/document/10184013.

Yizheng Huang and Jimmy Xiangji Huang. A Survey on Retrieval-Augmented Text Generation for Large Language Models. *ACM Computing Surveys*, 58(12):300:1–300:38, May 2026. ISSN 0360-0300. doi: 10.1145/3805774. URL https://dl.acm.org/doi/10.1145/3805774.

Yuxuan Huang, Yihang Chen, Haozheng Zhang, Kang Li, Huichi Zhou, Meng Fang, Linyi Yang, Xiaoguang Li, Lifeng Shang, Songcen Xu, Jianye Hao, Kun Shao, and Jun Wang. Deep Research Agents: A Systematic Examination And Roadmap, September 2025. URL http://arxiv.org/abs/2506.18096. arXiv:2506.18096 [cs.AI].

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong Park. Adaptive-RAG: Learning to Adapt Retrieval-Augmented Large Language Models through Question Complexity. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 7036–7050, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.389. URL https://aclanthology.org/2024.naacl-long.389/.

Haotian Jiang, Jian Li, and Mingda Qiao. Practical Algorithms for Best-K Identification in Multi-Armed Bandits, May 2017. URL http://arxiv.org/abs/1705.06894. arXiv:1705.06894 [cs.LG].

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active Retrieval Augmented Generation. In Houda Bouamor, Juan Pino, and Kalika Bali (eds.), *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 7969–7992, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.495. URL https://aclanthology.org/2023.emnlp-main.495/.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-R1: Training LLMs to Reason and Leverage Search Engines with Reinforcement Learning, August 2025. URL http://arxiv.org/abs/2503.09516. arXiv:2503.09516 [cs.CL].

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, October 2002. ISSN 1046-8188. doi: 10.1145/582415.582418. URL https://dl.acm.org/doi/10.1145/582415.582418.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. PAC Subset Selection in Stochastic Multi-armed Bandits.

Shivaram Kalyanakrishnan, Ambuj Tewari, Peter Auer, and Peter Stone. PAC subset selection in stochastic multi-armed bandits. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, ICML'12, pp. 227–234, Madison, WI, USA, June 2012. Omnipress. ISBN 978-1-4503-1285-1.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. Demonstrate-Search-Predict: Composing retrieval and language models for knowledge-intensive NLP, January 2023. URL http://arxiv.org/abs/2212.14024. arXiv:2212.14024 [cs.CL].

Omar Khattab, Arnav Singhvi, Paridhi Maheshwari, Zhiyuan Zhang, Keshav Santhanam, Sri Vardhamanan, Saiful Haq, Ashutosh Sharma, Thomas T. Joshi, Hanna Moazam, Heather Miller, Matei Zaharia, and Christopher Potts. DSPy: Compiling Declarative Language Model Calls into Self-Improving Pipelines. In *The Twelfth International Conference on Learning Representations*, 2024.

Junyoung Kim, Anton Korikov, Jiazhou Liang, Justin Cui, Yifan Simon Liu, Qianfeng Wen, Mark Zhao, and Scott Sanner. Bayesian Active Learning with Gaussian Processes Guided by LLM Relevance Scoring for Dense Passage Retrieval, April 2026. URL http://arxiv.org/abs/2604.17906. arXiv:2604.17906 [cs.IR].

Dawn Lawrie, James Mayfield, Eugene Yang, Andrew Yates, Sean MacAvaney, Ronak Pradeep, Scott Miller, Paul McNamee, and Luca Soldani. NeuCLIRBench: A Modern Evaluation Collection for Monolingual, Cross-Language, and Multilingual Information Retrieval, November 2025. URL http://arxiv.org/abs/2511.14758. arXiv:2511.14758 [cs.IR].

Minghan Li, Eric Gaussier, Juntao Li, and Guodong Zhou. EviRerank: Adaptive Evidence Construction for Long-Document LLM Reranking, June 2026. URL http://arxiv.org/abs/2411.06254. arXiv:2411.06254 [cs.IR] version: 6.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638. URL https://aclanthology.org/2024.tacl-1.9/.

Yifan Simon Liu, Qianfeng Wen, Jiazhou Liang, Mark Zhao, Justin Cui, Anton Korikov, Armin Toroghi, Junyoung Kim, and Scott Sanner. Multimodal Item Scoring for Natural Language Recommendation via Gaussian Process Regression with LLM Relevance Judgments. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 36859–36876, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1836. URL https://aclanthology.org/2026.findings-acl.1836/.

Maxime Louis, Hervé Déjean, and Stéphane Clinchant. PISCO: Pretty Simple Compression for Retrieval-Augmented Generation. In Wanxiang Che, Joyce Nabende, Ekaterina Shutova, and Mohammad Taher Pilehvar (eds.), *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 15506–15521, Vienna, Austria, July 2025. Association for Computational Linguistics. ISBN 979-8-89176-256-5. doi: 10.18653/v1/2025.findings-acl.800. URL https://aclanthology.org/2025.findings-acl.800/.

Sean MacAvaney, Nicola Tonellotto, and Craig Macdonald. Adaptive Re-Ranking with a Corpus Graph. In *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, pp. 1491–1500, October 2022. doi: 10.1145/3511808.3557231. URL http://arxiv.org/abs/2208.08942. arXiv:2208.08942 [cs.IR].

Gary Marchionini. Exploratory search: from finding to understanding. *Communications of the ACM*, 49(4):41–46, April 2006. ISSN 0001-0782. doi: 10.1145/1121949.1121979. URL https://dl.acm.org/doi/10.1145/1121949.1121979.

K.V. Mardia and P.E. Jupp. *Directional Statistics*. Wiley Series in Probability and Statistics. Wiley, 2009. ISBN 978-0-470-31781-5. URL https://books.google.com/books?id=PTNiCm4Q-M0C.

Chuan Meng, Negar Arabzadeh, Arian Askari, Mohammad Aliannejadi, and Maarten de Rijke. Ranked List Truncation for Large Language Model-based Re-Ranking. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 141–151, July 2024. doi: 10.1145/3626772.3657864. URL http://arxiv.org/abs/2404.18185. arXiv:2404.18185 [cs.IR].

Chuan Meng, Litu Ou, Sean MacAvaney, and Jeff Dalton. Revisiting Text Ranking in Deep Research, June 2026. URL http://arxiv.org/abs/2602.21456. arXiv:2602.21456 [cs.IR].

Minerva Mukhopadhyay, Didong Li, and David B. Dunson. Estimating densities with nonlinear support using Fisher-Gaussian kernels. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 82(5):1249–1271, December 2020. ISSN 1369-7412, 1467-9868. doi: 10.1111/rssb.12390. URL http://arxiv.org/abs/1907.05918. arXiv:1907.05918 [stat.ME].

Quoc Phong Nguyen, Sunil Gupta, Svetha Venkatesh, Bryan Kian Hsiang Low, and Patrick Jaillet. Active Set Ordering. In *Advances in Neural Information Processing Systems*, volume 37, pp. 108518–108541. Curran Associates, Inc., 2024. doi: 10.52202/079017-3445. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/c4546b4f9e1a44ed15c253dd43307dd5-Abstract-Conference.html.

Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. Document Expansion by Query Prediction, September 2019. URL http://arxiv.org/abs/1904.08375. arXiv:1904.08375 [cs.IR].

Nicholas Pipitone, Ghita Houir Alami, Advaith Avadhanam, Anton Kaminskyi, and Ashley Khoo. zELO: ELO-inspired Training Method for Rerankers and Embedding Models, September 2025. URL http://arxiv.org/abs/2509.12541. arXiv:2509.12541 [cs.AI].

Roi Pony, Adi Raz Goldfarb, Oshri Naparstek, Idan Friedman, Udi Barzelay, and Eli Schwartz. Col-Bandit: Query-Time Top-K Estimation for Late-Interaction Retrieval, June 2026. URL http://arxiv.org/abs/2602.02827. arXiv:2602.02827 [cs.IR] version: 2.

Muhammad Shihab Rashid, Jannat Ara Meem, Yue Dong, and Vagelis Hristidis. EcoRank: Budget-Constrained Text Re-ranking Using Large Language Models, May 2024. URL http://arxiv.org/abs/2402.10866. arXiv:2402.10866 [cs.CL].

Mandeep Rathee, Sean MacAvaney, and Avishek Anand. Guiding Retrieval using LLM-based Listwise Rankers, January 2025a. URL http://arxiv.org/abs/2501.09186. arXiv:2501.09186 [cs.IR].

Mandeep Rathee, Sean MacAvaney, and Avishek Anand. Quam: Adaptive Retrieval through <u>Qu</u>ery <u>A</u>ffinity <u>M</u>odelling. In *Proceedings of the Eighteenth ACM International Conference on Web Search and Data Mining*, WSDM '25, pp. 954–962, New York, NY, USA, March 2025b. Association for Computing Machinery. ISBN 979-8-4007-1329-3. doi: 10.1145/3701551.3703584. URL https://dl.acm.org/doi/10.1145/3701551.3703584.

S. Robertson, S. Walker, S. Jones, M. Hancock-Beaulieu, and M. Gatford. Okapi at TREC-3. 1994. URL https://www.semanticscholar.org/paper/Okapi-at-TREC-3-Robertson-Walker/fa8ffa9f8d8367b987845ae88f2adcac2689f110.

Saron Samuel, Alexander Martin, Eugene Yang, Andrew Yates, Dawn Lawrie, Ian Soboroff, Laura Dietz, and Benjamin Van Durme. Beyond Relevance: On the Relationship Between Retrieval and RAG Information Coverage, June 2026. URL http://arxiv.org/abs/2603.08819. arXiv:2603.08819 [cs.IR].

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D. Manning. RAPTOR: Recursive Abstractive Processing for Tree-Organized Retrieval, January 2024. URL http://arxiv.org/abs/2401.18059. arXiv:2401.18059 [cs.CL].

Ferdinand Schlatt, Maik Fröbe, Harrisen Scells, Shengyao Zhuang, Bevan Koopman, Guido Zuccon, Benno Stein, Martin Potthast, and Matthias Hagen. Rank-DistiLLM: Closing the Effectiveness Gap Between Cross-Encoders and LLMs for Passage Re-Ranking. volume 15574, pp. 323–334. 2025. doi: 10.1007/978-3-031-88714-7_31. URL http://arxiv.org/abs/2405.07920. arXiv:2405.07920 [cs.IR].

Sahel Sharifymoghaddam and Jimmy Lin. Rerank Before You Reason: Analyzing Reranking Tradeoffs through Effective Token Cost in Deep Search Agents. In Maria Liakata, Viviane P. Moreira, Jiajun Zhang, and David Jurgens (eds.), *Findings of the Association for Computational Linguistics: ACL 2026*, pp. 25874–25886, San Diego, California, United States, July 2026. Association for Computational Linguistics. ISBN 979-8-89176-395-1. doi: 10.18653/v1/2026.findings-acl.1289. URL https://aclanthology.org/2026.findings-acl.1289/.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. REPLUG: Retrieval-Augmented Black-Box Language Models. In Kevin Duh, Helena Gomez, and Steven Bethard (eds.), *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8371–8384, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.463. URL https://aclanthology.org/2024.naacl-long.463/.

Seokwon Song, Sohyeon Kim, and Gunhee Kim. MULTI3IR: A Benchmark for Multi-perspective Multi-domain Multi-modal Information Retrieval, September 2026. URL https://arxiv.org/abs/2608.30949. Version Number: 1.

Sofia Eleni Spatharioti, David Rothschild, Daniel G Goldstein, and Jake M Hofman. Effects of LLM-based Search on Decision Making: Speed, Accuracy, and Overreliance. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, CHI '25, pp. 1–15, New York, NY, USA, April 2025. Association for Computing Machinery. ISBN 979-8-4007-1394-1. doi: 10.1145/3706598.3714082. URL https://dl.acm.org/doi/10.1145/3706598.3714082.

Niranjan Srinivas, Andreas Krause, Sham M. Kakade, and Matthias W. Seeger. Information-Theoretic Regret Bounds for Gaussian Process Optimization in the Bandit Setting. *IEEE Transactions on Information Theory*, 58(5):3250–3265, May 2012. ISSN 1557-9654. doi: 10.1109/TIT.2011.2182033. URL https://ieeexplore.ieee.org/abstract/document/6138914.

Siddharth Suri, Scott Counts, Leijie Wang, Chacha Chen, Mengting Wan, Tara Safavi, Jennifer Neville, Chirag Shah, Ryen W. White, Reid Andersen, Georg Buscher, Sathish Manivannan, Nagu Rangan, and Longqi Yang. The Use of Generative Search Engines for Knowledge Work and Complex Tasks, March 2024. URL http://arxiv.org/abs/2404.04268. arXiv:2404.04268 [cs.IR].

Giovanni Trappolini, Florin Cuconasu, Simone Filice, Yoelle Maarek, and Fabrizio Silvestri. Redefining Retrieval Evaluation in the Era of LLMs, 2025. URL https://arxiv.org/abs/2510.21440. Version Number: 1.

Ellen M. Voorhees. The TREC robust retrieval track. *SIGIR Forum*, 39(1):11–20, June 2005. ISSN 0163-5840. doi: 10.1145/1067268.1067272. URL https://doi.org/10.1145/1067268.1067272.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. Multilingual E5 Text Embeddings: A Technical Report, February 2024. URL http://arxiv.org/abs/2402.05672. arXiv:2402.05672 [cs.CL].

Qianfeng Wen, Yifan Liu, Joshua Zhang, George Saad, Anton Korikov, Yury Sambale, and Scott Sanner. Elaborative Subtopic Query Reformulation for Broad and Indirect Queries in Travel Destination Recommendation, October 2024. URL http://arxiv.org/abs/2410.01598. arXiv:2410.01598 [cs.IR].

Christopher Williams and Carl Rasmussen. Gaussian Processes for Regression. In *Advances in Neural Information Processing Systems*, volume 8. MIT Press, 1995. URL https://proceedings.neurips.cc/paper_files/paper/1995/hash/7cce53cf90577442771720a370c3c723-Abstract.html.

Haike Xu and Tong Chen. Beyond Sequential Reranking: Reranker-Guided Search Improves Reasoning Intensive Retrieval, September 2025. URL http://arxiv.org/abs/2509.07163. arXiv:2509.07163 [cs.IR].

Soyoung Yoon, Gyuwan Kim, Gyu-Hwung Cho, and Seung-won Hwang. AcuRank: Uncertainty-Aware Adaptive Computation for Listwise Reranking, October 2025. URL http://arxiv.org/abs/2505.18512. arXiv:2505.18512 [cs.IR].

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 Embedding: Advancing Text Embedding and Reranking Through Foundation Models. *arXiv preprint arXiv:2506.05176*, 2025.

Wayne Xin Zhao, Jing Liu, Ruiyang Ren, and Ji-Rong Wen. Dense Text Retrieval Based on Pretrained Language Models: A Survey. *ACM Transactions on Information Systems*, 42(4):89:1–89:60, February 2024. ISSN 1046-8188. doi: 10.1145/3637870. URL https://dl.acm.org/doi/10.1145/3637870.

A LIMITATIONS

While BoundaryMORPH successfully optimizes retrieval for diffuse queries under budget constraints, we identify several limitations and system assumptions that offer directions for future work:

Sequential execution and hardware utilization. Because BoundaryMORPH updates the GP posterior after each cross-encoder call, it fundamentally requires sequential execution. As discussed in Appendix D.4, while this is competitive on a single GPU, heavily parallelized production environments can amortize single-stage reranking via massive batching. Extending our boundary acquisition rule to select batches of documents simultaneously (e.g., auto-kriging) could improve hardware utilization. In addition, many applications in practice are willing to trade off some latency for higher quality retrieval (e.g., long deep research loops).

GP scaling at massive budgets. The exact GP posterior update scales cubically with the budget and linearly with the pool size ($O(B^3N)$). While our latency analysis (Appendix D.5) shows the algorithm is lightweight for realistic RAG budgets ($B \leq 200$), the matrix inversion bottleneck becomes significant at extreme budgets (e.g., > 30s at $B = 500$). If CE inference becomes drastically cheaper in the future, the relative overhead of the GP will increase. Future work could explore algorithmic improvements, e.g. using rank-1 matrix updates.

Document counts vs. token limits. We formulate the RAG capacity as a strict document count k, whereas real-world LLM context windows are strictly bounded by *tokens*. However, because modern RAG pipelines heavily rely on text chunking prior to indexing, candidate documents typically fall into a relatively narrow and predictable token band. This system practice is what allows

the discrete document count k to serve as a highly realistic and practical proxy for context capacity. Nonetheless, adapting the boundary gap for RAG systems that utilize variable-length chunks remains an interesting future challenge.

Scope of queries. As we explicitly evaluate on *diffuse* queries requiring many sources, our method is tailored to exploratory search and deep-research agents. For simple factoid queries requiring only a single, specific document ($k = 1$), the exploratory benefits of BoundaryMORPH do not apply, and standard maximum-seeking approaches will suffice.

Selection versus within-context ordering. As defined in §2, BoundaryMORPH focuses purely on set *selection* (which k documents are retrieved). Because we actively avoid spending budget on the secure top-k (Figure 3(b)), the exact relative ordering of those unscored documents relies entirely on the GP estimation. However, we argue that *whether* a document is included in the RAG context is fundamentally more important than *where* it is placed. Recent research has demonstrated that even with perfect relevance scores, sophisticated context-rearrangement strategies often fail to outperform random shuffling (Cuconasu et al., 2025). Therefore, determining the optimal arrangement of context is a distinct, largely open research question, and we view maximizing the sheer volume of relevant information delivered to the context as the primary bottleneck for diffuse queries.

B Method Details

B.1 Algorithm pseudocode

Algorithms 1–2 give full pseudocode for the procedure described in §3.3. As in §3.2, the GP is fit not to the raw CE score but to the **residual** against the MIPS prior, and that residual is Z-standardized before conditioning and de-standardized after (Algorithm 1); Algorithm 2 is the outer acquisition loop that calls it.

Algorithm 1 Posterior: residual GP conditioning

function Posterior(observed set A, observed scores $\mathbf{y}$, prior mean μ_0, kernel K, pool C)
  $\mathbf{e} \leftarrow \mathbf{y} - \mu_0(A)$ ▷ residuals of the observations, $r(d) = f_{\mathrm{CE}}(q, d) - \mu_0(d)$, §3.2
  $\mathbf{z} \leftarrow \big(\mathbf{e} - \mathrm{mean}(\mathbf{e})\big)/\mathrm{std}(\mathbf{e})$ ▷ Z-standardize the residuals
  $(\tilde{\mu}, \tilde{\sigma}^2) \leftarrow$ zero-mean GP posterior of K given $(A, \mathbf{z})$, for all $d \in C$ ▷ Eq. 2 with $\mu_0 \equiv 0$
  $\mu(d) \leftarrow \mu_0(d) + \mathrm{mean}(\mathbf{e}) + \mathrm{std}(\mathbf{e}) \cdot \tilde{\mu}(d)$ for all $d \in C$ ▷ de-standardize
  $\sigma^2(d) \leftarrow \mathrm{std}(\mathbf{e})^2 \cdot \tilde{\sigma}^2(d)$ for all $d \in C$
  return (μ, σ^2)
end function

Algorithm 2 BoundaryMORPH

Require: query q; retriever s; cross-encoder f_{CE}; budget B; capacity k
Ensure: selected set $S \subseteq C$, $|S| = k$
$C \leftarrow$ top-N candidates from $s(q)$
Build kernel K over $C \cup \{q\}$ (§3.2); set prior mean $\mu_0(d) \leftarrow$ MIPS percentile rank of d in C
$A \leftarrow \{q\}$, $\mathbf{y} \leftarrow \{1\}$ ▷ query enters as a zero-cost observation at $y = 1$
$(\mu, \sigma^2) \leftarrow$ Posterior$(A, \mathbf{y}, \mu_0, K, C)$
for $t = 1, \ldots, B$ **do**
  $S_t \leftarrow$ top-k of C by μ ▷ incumbents; challengers are $C \setminus S_t$
  $(j^*, i^*) \leftarrow \arg\min_{j \in S_t,\, i \in C \setminus S_t} \frac{\mu(j) - \mu(i)}{\sigma(j) + \sigma(i)}$ ▷ standardized boundary gap, Eq. 4
  $d \leftarrow \arg\max \big(\sigma(j^*), \sigma(i^*)\big)$ ▷ score whichever of the pair is less certain
  $y \leftarrow f_{\mathrm{CE}}(q, d)$ ▷ one CE call
  $A \leftarrow A \cup \{d\}$, $\mathbf{y} \leftarrow \mathbf{y} \cup \{y\}$
  $(\mu, \sigma^2) \leftarrow$ Posterior$(A, \mathbf{y}, \mu_0, K, C)$
end for
return the top-k documents of C ranked by μ (scored documents at their true CE score, unscored documents at μ)

B.2 TIME COMPLEXITY

This section expands on the costs stated in §3.3's **Complexity** paragraph. The kernel K is a Gram matrix over the pool (augmented with the query row of §3.2), one entry per pair of documents, so building it once before any CE call is spent costs $O(N^2)$.

Each of the B steps of Algorithm 2's loop re-conditions the GP on a growing observed set A_t, $|A_t| = t$. Rather than refactorizing from scratch, the Cholesky factor of the $t \times t$ observed-block kernel is updated incrementally at $O(t^2)$, but reading off the posterior mean and variance at *every* one of the N pool documents requires a triangular solve against that factor per document, an $O(t^2)$ solve repeated N times, i.e. $O(t^2 N)$ at step t. Summing over the run, $\sum_{t=1}^{B} O(t^2 N) = O(B^3 N)$, which is the matrix-inversion cost the main text refers to. The acquisition rule (Eq. 4) is comparatively cheap: it evaluates the standardized boundary gap over the full incumbent×challenger cross product, $O(k(N-k))$ per step and $O(Bk(N-k))$ over the run. Because $B, k \ll N$ in practice, both terms are far below their worst case, which is why Appendix D.4 finds that the $O(B)$ sequential CE calls, not the $O(B^3 N)$ GP bookkeeping, dominate wall-clock latency at the budgets used in this paper.

B.3 GP IMPLEMENTATION

The kernel is the vMF kernel $K(d, d') = \exp(\kappa(\cos(d, d') - 1))$ on unit-normalized embeddings, with the diagonal forced to 1. Its bandwidth is set per query by the median heuristic, $\kappa = \log 2/(1 - \mathrm{median}_{i<j} \cos(d_i, d_j))$ over the pool, falling back to $\kappa = 1$ if that denominator degenerates; no other hyperparameter is tuned. The prior mean is the MIPS percentile rank within C: because the pool is returned MIPS-sorted, this is simply $\mu_0 = (N - 1 - \mathrm{pos})/(N - 1)$ for a document at position pos. The query enters as an extra row of the kernel, which is therefore the augmented $(N+1) \times (N+1)$ matrix with query entries $\exp(\kappa(\cos(q, d) - 1))$, and is conditioned on at $y = 1$ before the first CE call is spent.

Each posterior update takes the Cholesky factor of $\widetilde{K}_t = K(A_t, A_t) + \sigma_n^2 I$, with jitter $\sigma_n^2 = 10^{-4}$ added to the observed-block diagonal, and performs two triangular solves: one for $\alpha = \widetilde{K}_t^{-1}(y_t - \mu_0(A_t))$ and one for $v = \widetilde{K}_t^{-1}\mathbf{k}(A_t, U)$ over the unobserved set U. The posterior is then $\mu_{\mathrm{post}} = \mu_0 + \mathbf{k}^\top \alpha$ and $\sigma^2 = \mathrm{diag}(K)_U - \sum_i \mathbf{k}_{iU} v_{iU}$, with the variance clipped at 0. If the factorization fails, the jitter is escalated to 10^{-2} and, failing that, the solve falls back to a pseudo-inverse. As described in §3.2, the quantity conditioned on is the residual against μ_0, Z-standardized before conditioning and de-standardized afterwards (Algorithm 1). When ranking the pool by μ, documents with equal posterior means are ordered by MIPS percentile descending.

The resulting costs are $O(N^2)$ for the one-time kernel build, $O(t^2 N)$ per posterior update and $O(B^3 N)$ over the run, and $O(k(N-k))$ per acquisition step and $O(Bk(N-k))$ over the run; Appendix B.2 derives these.

B.4 KERNEL ABLATION

To ensure robustness of results with respect to the kernel used in the GP, we run both GP methods (BoundaryMORPH and BAGEL) with three different kernels. We test the vMF kernel, described in the Methods section, along with the squared-exponential (RBF) and Matérn-5/2 kernels (Williams & Rasmussen, 1995). Both alternatives are functions of the squared chordal distance $\|d - d'\|^2 = 2\,(1 - \cos(d, d'))$ on the unit sphere, a monotone reparametrization of the same cosine geometry the vMF kernel uses. Writing $u = 1 - \cos(d, d')$ and $\rho = \sqrt{5u/\ell^2}$,

$$K_{\mathrm{RBF}}(d, d') = \exp\big(-u/\ell^2\big), \qquad K_{\mathrm{Mat}}(d, d') = \Big(1 + \rho + \tfrac{\rho^2}{3}\Big)\, e^{-\rho}. \tag{6}$$

The three differ only in how fast covariance decays with angle, and thus in the smoothness they assume of the CE surface: vMF and RBF are infinitely differentiable, whereas Matérn-5/2 is only twice differentiable and so admits a rougher residual field. The length-scale is set by the same median heuristic used for κ in the vMF case, $\ell^2 = \mathrm{median}_{i<j}(1 - \cos(d_i, d_j))$, so no kernel receives a tuning advantage. Jitter, the query-as-observation anchor (which enters through the same kernel), residual standardization, and the pool are unchanged.

Table 3: Kernel ablation: Δ CE-nCG from replacing the vMF kernel (kernel − vMF, percentage points), by dataset and evaluation cutoff k. Positive = the alternative kernel is better. † marks the 5/72 cells significant after multiple-comparison correction (see text).

		Δ CE-nCG@k (pp)			Δ nCG@k (pp, human qrels)		
Method	Kernel	50	100	200	50	100	200
NeuCLIR							
MORPH	RBF	$+0.12$	$+0.29$	-0.15	$+0.01$	$+0.22$	$+0.03$
MORPH	Matérn-5/2	$+0.43$	$+0.08$	$+0.15$	$+1.23$	-0.36	$+0.20$
BAGEL	RBF	-0.61	-0.71	$-1.33^{\dagger}$	-0.60	-0.99	$-1.48^{\dagger}$
BAGEL	Matérn-5/2	-0.12	$+0.28$	$+0.31$	-0.55	-0.79	-0.79
Robust04							
MORPH	RBF	$+0.44$	$+0.19$	$+0.30$	$+0.76$	$+0.55$	$+0.08$
MORPH	Matérn-5/2	$+0.17$	$+0.38$	$+0.33$	$+0.44$	$+0.81$	$+0.11$
BAGEL	RBF	$+0.01$	$-0.58^{\dagger}$	$-0.74^{\dagger}$	$+0.13$	-0.65	-0.67
BAGEL	Matérn-5/2	$+0.44$	$+0.30$	$+0.23$	$+0.57$	$+0.44$	$+0.32$
TravelDest							
MORPH	RBF	$+0.06$	-0.04	-0.42	$+0.81$	-0.33	$+0.14$
MORPH	Matérn-5/2	$+0.10$	-0.13	-0.09	-0.22	$+0.17$	-0.25
BAGEL	RBF	$+0.12$	-0.06	-0.13	$+0.22$	-0.32	-0.26
BAGEL	Matérn-5/2	$+0.16$	$+0.39$	$+0.67^{\dagger}$	$+0.46$	$+0.37$	$+0.47$

We fix acquisition and prior for each of the two methods — so the only thing that varies is the kernel — and show the change in performance from switching to RBF or Matérn-5/2 over the vMF kernel in Table 3. Both metrics are reported, since the two answer different questions (§4.2) and a kernel could in principle help one and hurt the other. Each query's score is averaged over the two budgets $B \in \{25, 50\}$ before the vMF score is subtracted, and the paired tests are run on those per-query B-averaged values. BAGEL's β_0 is re-selected within each kernel, so neither alternative is handicapped by a choice tuned for vMF.

The table shows that choice of kernel has no significant effect. Every one of the 72 deltas is under 1.5 pp and the median is 0.32 pp, and both the signs and the ordering of the two alternatives are inconsistent across datasets: Matérn-5/2 is nominally best for BoundaryMORPH on NeuCLIR at CE-nCG@50 ($+0.43$ pp) but RBF leads there on Robust04 ($+0.44$ vs. $+0.17$ pp), and on TravelDest both drift slightly negative at the larger cutoffs. Across the 72 paired Wilcoxon signed-rank tests behind the table, 13 reach $p < 0.05$ — against ≈ 4 expected by chance — and after Benjamini-Hochberg correction **5** survive at $q < 0.05$. All five are BAGEL, none our method: four are BAGEL under RBF performing *worse* than under vMF (NeuCLIR, -1.33 pp CE-nCG@200 and -1.48 pp nCG@200; Robust04, -0.58 pp CE-nCG@100 and -0.74 pp CE-nCG@200), and the fifth is Matérn-5/2 *helping* BAGEL on TravelDest ($+0.67$ pp CE-nCG@200). So the only detectable kernel effects move the baseline, in both directions, and none is large enough to matter: for scale, the same-grid gap between the two *methods* at a fixed vMF kernel is $+4.8$ to $+10.4$ pp CE-nCG. The *smallest* method effect is more than three times the largest kernel effect anywhere in the table, and fifteen times the median one. Selection skill in this problem is governed by the acquisition rule and the prior, not by the covariance shape. Hence, we report the paper's main results using the vMF kernel.

C Experimental Setup

C.1 Dataset statistics

NeuCLIRBench (Lawrie et al., 2025) pairs open-ended English information needs, each stated as a keyword line followed by a question, against a newswire corpus machine-translated into English from Persian, Russian and Chinese.

Robust04 (Voorhees, 2005) uses TREC topics, each a description-style sentence, over the TREC Disks 4 & 5 newswire and government collection.

TravelDest (Wen et al., 2024) uses broad travel-intent topics naming a desired quality of a destination rather than a place, over Wikivoyage-derived passages describing individual cities.

Table 4: Active-testbed statistics: query counts, relevant documents per query (showing these queries are genuinely diffuse), MIPS pool coverage, and document length, per dataset. Definitions and the query filter are given below.

	NeuCLIR	Robust04	TravelDest
Full-corpus docs	10,038,768	528,155	131,268
Substrate pool / query	10,000	10,000	10,000
Substrate union docs	731,335	422,904	90,727
# Queries (original)	105	249	100
# Queries (post-filter)	85	187	99
# docs/query: median	51	62	462
# docs/query: mean	70.3	89.2	839.5
# docs/query: IQR (p25–p75)	37–85	35–116	140–1,062
# docs/query: max	246	448	5,667
Pool coverage: mean	0.92	0.88	0.70
Pool coverage: median	0.96	0.92	0.74
Pool coverage: p10	0.81	0.73	0.47
Pool coverage: min	0.58	0.42	0.29
Doc tokens: median	335	426	68
Doc tokens: IQR (p25–p75)	218–517	226–805	38–109

Table 5: Relevant documents per retained query, full percentiles.

Dataset	min	p10	p25	median	mean	p75	p90	max
NeuCLIR	22	27	37	51	70.3	85	122	246
Robust04	20	27	35	62	89.2	116	194	448
TravelDest	20	67	140	462	839.5	1,062	1,975	5,667

Table 6: Pool coverage, full percentiles. All columns are macro-averaged over queries except the last, which is the micro-averaged coverage (pooled over all judged-relevant documents).

Dataset	min	p10	p25	median	mean	p75	p90	max	micro
NeuCLIR	0.582	0.810	0.897	0.957	0.922	0.981	1.000	1.000	0.904
Robust04	0.422	0.729	0.831	0.917	0.885	0.975	1.000	1.000	0.860
TravelDest	0.287	0.466	0.607	0.737	0.699	0.821	0.882	0.943	0.599

Relevance basis: NeuCLIR qrels grade > 0 (grades in $\{0, 1, 3\}$); Robust04 qrels grade in $\{1, 2\}$; TravelDest an explicit binary relevant set. Pool coverage = fraction of a query's judged-relevant documents present in its top-10,000 MIPS pool, macro-averaged, denominator = all judged-relevant documents in the full qrels, so $1-$ coverage is relevant mass discarded before any selection and bounds what any selector could recover. Token counts are Qwen3-Embedding-4B tokens over the pool union. The ≥ 20 filter drops 20 / 62 / 1 queries.

Relevant documents in the qrels against relevant documents inside the pool: 5,976 $\rightarrow$ 5,400 (NeuCLIR), 16,684 $\rightarrow$ 14,351 (Robust04), 83,112 $\rightarrow$ 49,822 (TravelDest).

Table 7: Document tokens over the pool union, full percentiles.

Dataset	min	p25	median	mean	p90	p99	max
NeuCLIR	3	218	335	444.0	824	2,261	18,053
Robust04	15	226	426	639.1	1,264	3,299	927,351
TravelDest	6	38	68	81.4	157	289	918

C.2 DETAILED BASELINES

All baselines are scored on the **same fixed per-query top-10,000 pool** as BoundaryMORPH, so every comparison isolates allocation and ordering within a shared candidate set rather than retrieval from scratch. Every baseline that has hyperparameters has them **selected on the evaluation metric** (pooled-mean CE-nCG@50) per cell per budget, and reused across k; BoundaryMORPH runs at one fixed configuration and tunes nothing.

BM25. Okapi BM25 with $k_1 = 1.5$, $b = 0.75$ and full-corpus statistics, applied as a re-ranking of the shared pool and truncated at k. It makes no CE calls, so its ordering is fixed per query and its value is identical at every B; and its ordering uses neither MIPS nor the CE, so its nCG is a single value per (dataset, k) and its CE-nCG depends only on which CE supplies the gold. Both properties are verified rather than assumed.

SS (single-stage rerank). Score the MIPS top-B with the CE, place those B above everything else, and leave the tail in MIPS order. For $k \geq B$ this returns exactly the MIPS top-k, so the budget buys nothing in that regime.

BAGEL. Zero-prior GP-UCB. It shares our pool, our vMF kernel, and our median-heuristic bandwidth, and differs in exactly three places: the prior mean (zero rather than MIPS percentile), the acquisition rule (UCB over the whole pool rather than the boundary), and the warm-start.

- β_0 swept over $\{0, 0.5, 1, 2\}$ with square-root annealing $\beta_t = \beta_0/\sqrt{t+1}$, selected per (dataset $\times$ embedder $\times$ CE) cell per B. $\beta_0 = 0$ — pure exploitation of the posterior mean — is selected in 4 of the headline cells, all at large B.
- BAGEL is **warm-started**: it spends $\lfloor B/2 \rfloor$ calls deterministically on the MIPS top-$\lfloor B/2 \rfloor$, leaving $B - \lfloor B/2 \rfloor$ acquisition steps.

RGS. Greedy graph expansion under a max-heap keyed by CE score: pop the highest-scored not-yet-expanded document, fetch its graph neighbors, score any that are unscored, push them, and truncate the heap to L_s. The graph is a DiskANN cosine graph over the pool embeddings (one greedy search + robust prune pass, $L = 100$, $\alpha = 1.2$).

- Swept: graph degree, which is also the expansion branching factor, $R \in \{10, 32, 64\}$; the seed/expansion split $n_{\text{seeds}} = \text{round}(B/\text{seed_div})$; and the heap cap $L_s = \max(n_{\text{seeds}}, \text{round}(\text{frac} \cdot B))$ with $\text{frac} \in \{0.2, 0.5, 1.0\}$. All headline cells select $R = 10$, $\text{seed_div} = 2$, $\text{frac} = 1.0$. The seed_div grid is $\{5, 4, 3, 2\}$ on the main configuration and $\{2, 4, 8\}$ on the model-swap configurations; the selected value, 2, is in both.
- Ranking convention: scored documents receive $\text{offset} + \text{CE score}$ with $\text{offset} = 10$ and unscored receive 0, so scored documents form a band strictly above unscored and a stable descending sort leaves the unscored tail in MIPS order. This is the most charitable construction of the tail for a method that cannot rank what it never scored.
- At $n_{\text{seeds}} = B$ the search reduces to single-stage rerank to within 10^{-12}, which checks the implementation against its own degenerate case.

Agentic loop. A multi-hop search agent (DSP) run with plain instruction prompts, generated by Qwen3.5-9B with thinking disabled and greedy decoding. Retrieval is *within-pool*, so the loop is scored on the same candidate set as the selectors. Each hop the model writes one query from the information need and its accumulated notes, that query is embedded with the same retriever and its top documents by within-pool cosine are shown, and the model scores each document 0–10 and records new notes. Documents shown in earlier hops are excluded from later ones (meaning the agent always scores *exactly* docs/hop new documents in each hop, and *exactly* docs/hop $\times$ hops total documents). The final ranking is surfaced documents by assigned score descending, ties broken by cosine to the original query, followed by the never-surfaced tail in MIPS order. Grid: per-hop depth $\in \{5, 10, 20, 50, 100\} \times$ hops $\in \{1, 2, 3, 4, 6, 8\}$.

The query-generation prompt is shown in Figure 5, and the note-extraction prompt in Figure 6.

Across all 29 configurations, NeuCLIR, $n = 85$, nCG@100 ranges from 0.554 (5 documents per hop, 1 hop, 57.6 s) to 0.618 (100 per hop, 3 hops, 884.6 s), and latency from 57.6 s to 1,922.8 s. BoundaryMORPH reaches nCG@100 0.621 in 25.7 s at $B = 100$, $N = 10{,}000$, so no configuration of the loop matches it on either quality or cost.

You are a search agent gathering documents to satisfy an information need. Given the information need and the notes gathered so far, write ONE new search query that will retrieve MORE relevant documents for the information need. This can mean exploring a different facet OR retrieving more documents from facets already covered — do whatever will surface the most additional relevant documents. The query MUST be at least 100 words long — write a long, detailed, keyword-rich query (at least 100 words), not a short phrase. Return ONLY the query text, no preamble, no quotes.
Claim: {claim}
Notes so far: {notes}
New search query:

Figure 5: Prompt for generating the next search query in the agentic retrieval loop (DSP baseline).

You are a search agent. Below is the information need, the notes gathered so far, and a set of newly retrieved documents (each labelled [DOC <index>]). For EACH document, assign a relevance SCORE from 0 to 10 (0 = irrelevant to the information need, 10 = perfectly relevant). Then extract any NEW facts from the relevant documents that help satisfy the information need.
Respond with STRICT JSON ONLY (no markdown fence, no prose) of the form: {"doc_scores": {"0": 8, "3": 2}, "new_notes": ["...", "..."]} Put doc_scores FIRST. Use the EXACT integer indices shown in the [DOC <index>] labels as the keys, and an INTEGER 0-10 as each value. Score EVERY document shown. Keep each note to ONE short sentence.
Claim: {claim}
Notes so far: {notes}
Retrieved documents: {context}
JSON:

Figure 6: Prompt for scoring retrieved documents and extracting notes in the agentic retrieval loop (DSP baseline).

Table 8: Full BoundaryMORPH ablations, per dataset and for both metrics (Table 2 in the main text averages the nCG@k columns over datasets). Each row independently replaces one component: boundary acquisition→ standard UCB; MIPS prior→ 0 prior. We report the pp change from full BoundaryMORPH. Both components are crucial for the gains, and *always* improve results.

	nCG@k (%, human qrels)						CE-nCG@k (%)					
	B=25			B=50			B=25			B=50		
Method \ k	50	100	200	50	100	200	50	100	200	50	100	200
NeuCLIR												
MORPH	51.2	56.4	64.4	54.8	59.7	66.6	84.6	80.8	78.8	89.9	85.3	82.4
w/o acq.	-2.1	-2.1	-1.3	-1.2	-2.7	-1.5	-3.9	-5.2	-4.6	-2.6	-5.2	-5.1
w/o prior	-2.3	-4.3	-5.5	-1.5	-3.0	-2.9	-2.9	-3.9	-4.4	-2.3	-2.3	-2.7
Robust04												
MORPH	46.4	48.2	55.1	51.4	52.0	57.7	69.8	67.9	68.1	76.1	72.3	71.1
w/o acq.	-3.4	-3.5	-2.7	-2.0	-2.5	-1.5	-3.4	-3.6	-2.9	-2.2	-3.3	-2.7
w/o prior	-3.5	-3.2	-4.1	-2.4	-2.2	-2.8	-4.1	-4.0	-4.1	-3.2	-3.1	-2.9
TravelDest												
MORPH	43.7	39.2	38.2	47.8	42.1	40.3	68.2	67.5	68.7	75.1	72.1	71.8
w/o acq.	-2.2	-1.8	-2.2	-1.3	-1.6	-1.5	-3.0	-3.7	-3.9	-2.2	-3.3	-3.4
w/o prior	-3.2	-1.8	-1.9	-2.5	-1.6	-1.4	-4.5	-4.1	-3.6	-4.5	-4.4	-3.6

D Additional Results and Robustness

D.1 Full ablation results

Table 8 breaks the ablation of §5.2 out by dataset and reports both nCG@k and CE-nCG@k, with values averaged over the four embedder×CE pairs. Removing either component lowers every cell.

D.2 Robustness to retriever and cross-encoder

Results are robust to the choice of retriever and cross-encoder. To test our method's robustness to embedding model and CE model, we use two different embedding models (Qwen3-Embedding-4B (Zhang et al., 2025) and multilingual-e5-large (Wang et al., 2024)), and two different CE models (Qwen3-Reranker-8B (Zhang et al., 2025) and zerank-2 (Pipitone et al., 2025)) and test all methods with all 4 combinations. Figure 7 (Appendix D.3) shows the per-dataset result at $B = 50$, $k = 100$: BoundaryMORPH maintains a gap over baselines with all embedder/CE combinations on every dataset, indicating that the result is not a property of the retriever and CE.

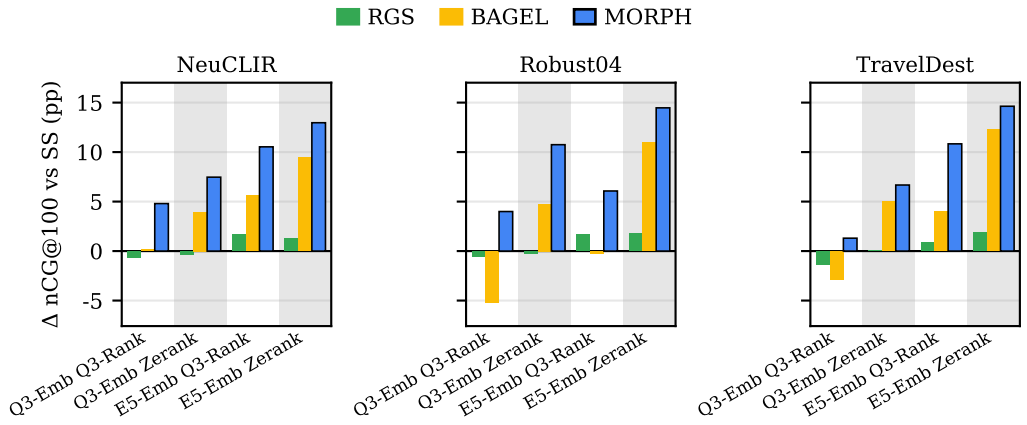

Figure 7: **The advantage is robust to choice of retriever and cross-encoder.** Per-dataset nCG@100 lift over single-stage rerank (percentage points) at $B = 50$, for all four embedder $\times$ CE combinations. BoundaryMORPH holds a gap over every baseline in all four, on every dataset.

Gains are largest with a weak retriever. Mean lift over SS runs from **+3.4 pp** with a strong retriever and strong CE to **+14.0 pp** with a weaker retriever; the advantage grows as retriever weakens, likely because there is simply more headroom between retriever and CE to take advantage of. Though in *absolute* terms, all methods improve when switching from the weaker e5 embedder to the stronger Qwen3 model, averaging $+10.6$ nCG@100 (B=50) for SS, an $+8.6$ for RGS, and $+4.9$ for BoundaryMORPH; a stronger prior universally helps.

D.3 Per-dataset embedder $\times$ CE results

nCG@100 lift over single-stage rerank (pp) at $B = 50$, per dataset, for all four embedder $\times$ CE combinations. Embedders: Qwen3-Emb = Qwen3-Embedding-4B, e5 = multilingual-e5-large. CEs: Qwen-Rerank = Qwen3-Reranker-8B, zerank = zerank-2.

D.4 Latency Analysis

Selectors that all call the same CE can be compared via B. We also compare wall clock latency (Figure 8, run on one A10, on NeuCLIR), and include an agentic retrieval loop in the comparison. We use a plain multi-hop retrieve-and-read loop, DSP (Khattab et al., 2023), (Khattab et al., 2024). The model (Qwen3.5-9B, running on 8 A10s) writes a search query, retrieves, reads and scores the results, records notes, and uses its notes to write another query (and this is repeated). Prompts and other details are in Appendix C.2 and Appendix D.5.

While such results are highly dependent on document length and hardware, figure 8 shows that CE budget scales each method's latency, justifying the use of B as a latency dial. BoundaryMORPH nearly Pareto-dominates, except at high B ($B >> k$). As B increases, selection matters less; there is budget enough to score every plausible document, so every reasonable allocator lands on a similar set. Thus, most methods converge at high B.

Due to the GP, BoundaryMORPH and BAGEL must be run sequentially, while SS can be batched. In practice we find that on 1 GPU, the variable document length leads to extra batch padding which *increases* latency for SS at batch sizes $\{2, 3, 8, 16, 32, 64\}$, so we report results on batch size of 1, which is the fastest possible setting for SS. In theory, with enough GPUs/memory, SS can be fully amortized, though based on the empirical results for our datasets, real-time speed would improve only from added GPUs, and not from batching on any given GPU. Thus, at least in theory, the latency of SS could be improved in an exact linear relationship with the number of GPUs.

Table 9: Median per-query latency by component. NeuCLIR, $N = 10{,}000$, capacity 50, $n = 85$. *algorithm* is the total non-CE time; *GP update* and *acquisition* are its components.

B	CE	CE %	algorithm	GP update	acquisition
25	4.66 s	95%	0.22 s	0.06 s	0.16 s
50	9.43 s	95%	0.51 s	0.17 s	0.35 s
100	19.45 s	93%	1.46 s	0.63 s	0.85 s
200	38.89 s	90%	4.56 s	2.73 s	1.78 s
500	94.39 s	74%	32.86 s	27.86 s	4.79 s
Totals: 4.88 s / 9.94 s / 20.91 s / 43.45 s / 127.25 s at $B = 25/50/100/200/500$.					

Notably, despite $8\times$ parallel compute, the agentic loop is Pareto-dominated. The whole agentic family lies below the budgeted frontier at every latency it reaches. Latency grows close to linearly in hops, because each hop pays for autoregressive decoding.

Origin of the gap. The information the GP exploits is already latent in the corpus embedding geometry and propagates through the kernel to documents never read. This agentic loop can only carry forward what it wrote down in natural language, in a bounded "notes", and pays decoding latency per token. We hypothesize that the gap is explained by the GP being much more efficient at propagating relevance information.

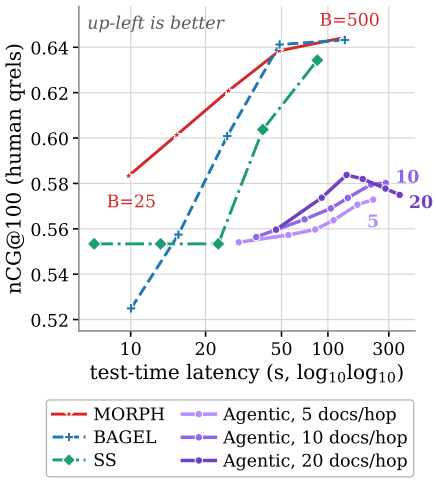

Figure 8: **Budget is a realistic proxy for latency.** We show nCG@100 against wall-clock latency per query on NeuCLIR, including an agentic loop (DSP). Latency scales with budget for all methods; with performance converging at high B. The agentic method is dominated due to decoding latency.

D.5 Latency setup

Hardware. One A10G (24 GB), bf16 with SDPA, left padding, maximum length 4,096. Document sequence lengths are p50 = 426, p95 = 1,217 tokens. Absolute seconds are hardware-specific; the ratios are the claim.

Per-component breakdown. NeuCLIR, $N = 10{,}000$, capacity 50, $n = 85$, medians (Table 9).

- CE dominates latency at every B, 74–95%; the share is highest at small B (95% at $B = 25, 50$) and falls only at $B = 500$ (74%), where GP update cost grows enough to compete.
- The algorithm's own cost (GP update + acquisition) is under 1.5 s for $B \leq 100$ and grows to 32.86 s at $B = 500$, driven almost entirely by the GP update.
- Per-call CE cost is $\sim$0.19 s, flat in B.

E Query Multimodality

Intuitively, diffuse queries can be *unimodal*, where relevant documents are relatively similar to one another, or *multimodal*, where relevant documents are relatively different from one another (Chen et al., 2025; Song et al., 2026; Liu et al., 2026). We expect multimodal queries to be especially difficult for max-seeking selectors, which concentrate calls on the primary mode, and define a geometric scalar, *multimodal mass*, to measure this.

E.1 Peak extraction and LLM audit

Peak extraction. We adapt the persistence-based clustering of Chazal et al. (2013), viewing each document's CE score as its *height* over the embedding kNN landscape. A *peak* is a group of close,

Table 10: Adjusted Rand index between the LLM's two-group partition and the geometric peak-of-origin labels, by slice.

Slice	n	mean ARI	median ARI
All pairs	496	0.635	1.000
Pairs including the main peak	190	0.486	0.437
Secondary–secondary pairs	306	0.726	1.000
Per-query mean	65	0.593	0.624

highly relevant documents, and its *prominence* measures how independent it is from other peaks: how far relevance must drop before it merges into a taller one. Concretely, on the top-$M = 1{,}000$ documents by CE score within each query's $N = 10{,}000$ pool, build a $k_{\mathrm{nn}} = 10$ OR-symmetrized cosine kNN graph. Sweep a descending CE-score threshold; each connected component of the super-level set carries a peak = its highest-scoring document, and merging components record 0-dimensional persistence (merge-tree prominence). Simplifying at $\tau = 0.20$ yields a hard partition of the top-M into peaks. Multimodal mass is the sum of mass-weighted prominence over all non-main peaks, so a unimodal landscape scores 0.

Peaks correspond to semantic distinctions. To test if peaks are semantically meaningful, we give an LLM documents drawn from the same peak and from different peaks, and ask it to partition them, scoring the result against the geometric partition by ARI: **mean 0.635, median 1.000** over **496 pairs from 65 multi-peak queries**, **85%** above chance and **63%** clean at ARI ≥ 0.5. The high success indicates that geometric multimodality is also a semantic feature.

LLM audit. For each query, keep peaks with ≥ 3 member documents; for every unordered pair, sample up to 8 documents from each peak's 0.5-core (falling back to the full basin when the core has < 3), shuffle the ≤ 16 together, and ask the judge to split them into two groups by *how* they are relevant. Score against the peak-of-origin labels by adjusted Rand index. One sample per pair, seeded deterministically. Judge: Claude Opus 4.5, temperature 0, max 1,200 tokens, strict-JSON two-group response; documents truncated to 1,500 characters. 496 pairs from 65 multi-peak NeuCLIR queries, parse success 496/496.

ARI histogram: 32 below -0.05, 42 at chance $[-0.05, 0.05]$, 46 in $(0.05, 0.2]$, 40 in $(0.2, 0.4]$, 47 in $(0.4, 0.6]$, 40 in $(0.6, 0.8]$, 249 in $(0.8, 1]$. 85.1% beat chance, 62.7% reach ARI ≥ 0.5, and half are separated perfectly. Correlation of ARI with pair size is -0.18.

The judge prompt is shown in Figure 9.

You are analyzing search results for a query. Below are {n} documents that are ALL relevant to the query. They are relevant in different ways — they emphasize different aspects, sub-topics, entities, or angles of the query.
QUERY: {query}
Your task: partition ALL {n} documents into exactly TWO groups based on *how* each document is relevant to the query (what aspect/angle/sub-topic of the query it addresses). The two groups need NOT be equal in size, but every document must be placed in exactly one group, and neither group may be empty. Base the split on the substantive way each document relates to the query, not on surface features like length or style.
Give each group a SHORT description (one sentence) of the way its documents are relevant to the query.
DOCUMENTS: {docs}
Respond with ONLY a JSON object, no other text, in exactly this form: {"group_1": {"description": "<one sentence>", "docs": [<doc numbers>]}, "group_2": {"description": "<one sentence>", "docs": [<doc numbers>]}}

Figure 9: Prompt for the LLM judge that partitions documents by relevance aspect (peak audit).