# Bayesian Active Learning with Gaussian Processes Guided by LLM Relevance Scoring for Dense Passage Retrieval

Anonymous ACL submission

## Abstract

While Large Language Models (LLMs) exhibit exceptional zero-shot relevance modeling, their high computational cost necessitates framing passage retrieval as a budget-constrained global optimization problem. Existing approaches passively rely on first-stage dense retrievers, which leads to two limitations: (1) failing to retrieve relevant passages in semantically distinct clusters, and (2) failing to propagate relevance signals to the broader corpus. To address these limitations, we propose **B**ayesian **A**ctive Learning with **G**aussian Processes guid**E**d by **L**LM relevance scoring (**BAGEL**)[1], a novel framework that propagates sparse LLM relevance signals across the embedding space to guide global exploration. BAGEL models the multimodal relevance distribution across the entire embedding space with a query-specific Gaussian Process (GP) based on LLM relevance scores. Subsequently, it iteratively selects passages for scoring by strategically balancing the exploitation of high-confidence regions with the exploration of uncertain areas. Extensive experiments across four benchmark datasets and two LLM backbones demonstrate that BAGEL effectively explores and captures complex relevance distributions and matches or outperforms LLM reranking methods on all four datasets under the same LLM budget.

## 1 Introduction

While Large Language Models (LLMs) demonstrate a strong zero-shot ability in modeling complex query-passage relevance (Sachan et al., 2022; Sun et al., 2023), their high computational cost prohibits exhaustive inference over large corpora. Consequently, the passage retrieval task can be framed as a budget-constrained global optimization problem: identifying relevant passages within a massive search space under a strictly limited number of LLM inferences. Prevalent approaches largely adhere to the *LLM reranking* paradigm (Zhuang et al., 2024; Sun et al., 2023), where dense retrievers retrieve a top-$K$ candidate set based on computationally efficient vector similarity and then reorder the set with an LLM. However, as shown in Figure 1a, passively relying on this first-stage candidate set introduces two fundamental limitations:

**First**, the first-stage retriever imposes an upper bound on the recall. Relevant passages often reside in multiple, semantically distinct clusters scattered throughout the embedding space (Vikraman et al., 2021), forming a multimodal relevance distribution. Standard dense retrievers, which retrieve from a local neighborhood around the query embedding may fail to detect global structures or high-relevance clusters located far from the initial neighborhood of query.

**Second**, existing approaches fail to propagate relevance signals from previously scored passages to unseen passages, overlooking the underlying embedding space that connects them. By failing to utilize the semantic structure where the relevance of one passage often implies the relevance of its neighbors (Kurland, 2013), current approaches hinder efficient exploration of the global landscape.

To overcome these limitations, we propose Gaussian Process (GP)-based active learning as an effective framework for this budget-constrained task. GPs offer two intrinsic properties that directly address the aforementioned challenges:

(1) *Kernel-based Relevance Signal Propagation*: GPs naturally model correlations between data points via kernel functions, enabling the representation of multimodal, clustered relevance functions (cf. Figure 1b). They also allow us to interpolate relevance signals across the embedding space, effectively inferring the relevance of unobserved passages based on sparse LLM signals.

(2) *Uncertainty for Active Learning*: GPs provide a probabilistic posterior that includes both pre-

[1]Code is available at https://anonymous.4open.science/r/BAGEL-7BE0

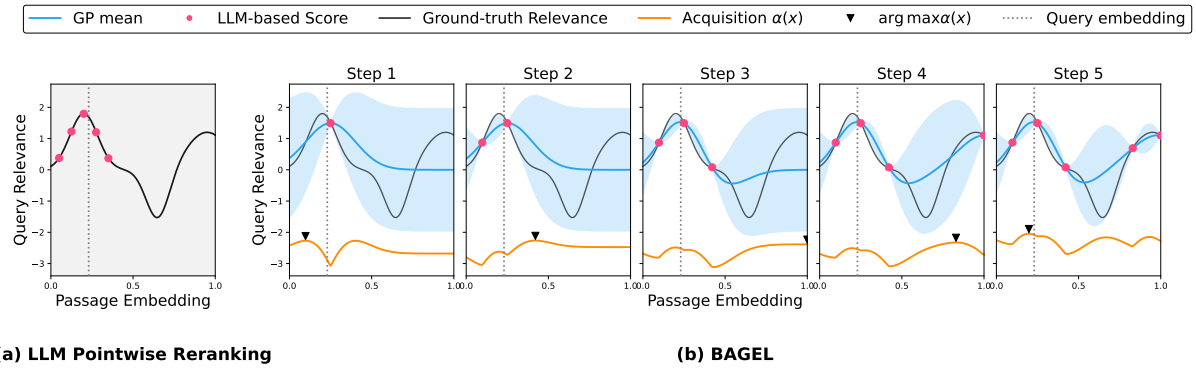


Figure 1: Comparison of passage selection strategies under a fixed LLM budget. The x-axis shows a 1D projection of passage embeddings, and the y-axis displays the estimated Gaussian Process posterior mean (blue line) and standard deviation (shaded region). (a) LLM Pointwise Reranking focuses on candidates (red dots) located nearby the query embedding (dashed line), often missing relevant passages in distant semantic clusters. (b) BAGEL actively explores the embedding space with active learning. By modeling the predictive mean (blue line) and uncertainty (blue shaded area) from observed scores (red dots), the acquisition function (orange line) guides the selection of the next passage (black ▼) to explore high uncertainty regions and/or exploit high expected relevance areas, uncovering relevant passages in diverse clusters.

dictive mean and variance (uncertainty). This enables Bayesian Active learning to effectively search the embedding space by balancing exploration (*e.g.,* via uncertainty) with exploitation.

Building on these insights, we propose **BAGEL** (**B**ayesian **A**ctive learning with **G**aussian Processes guid**E**d by **L**LM relevance scoring) by leveraging the advantages of GPs to actively navigate the embedding space. Figure 1b illustrates how BAGEL models relevance scores and actively selects the next passage for LLM scoring using both the GP predicted mean (blue line) and uncertainty (blue shaded area) as further elaborated below.

Specifically, BAGEL constructs a query-specific GP surrogate model of the LLM-based query-passage relevance function, defined over the dense passage embedding space. By conditioning on the observed LLM relevance scores, BAGEL extends relevance signals to unseen passages, capturing complex, multimodal relevance structures and producing a full ranking under a limited LLM budget.

Moreover, to explore multiple high relevance clusters (*i.e.,* modes), BAGEL actively selects additional passages (Steps 1–5 in Figure 1b) for LLM relevance scoring using an acquisition function. By jointly evaluating the GP's predicted relevance and its uncertainty, it balances *exploitation*: selecting passages predicted to be highly relevant, and *exploration*: selecting passages with high uncertainty. This strategy enables efficient navigation of the *entire* passage embedding space while making parsimonious use of expensive LLM relevance scoring.

The main contributions of our work are summarized as follows:

- **Gaussian Process-based Active Learning for Passage Exploration.** We propose a framework that integrates LLM-based relevance scoring with active learning driven by Gaussian Processes, enabling the strategic exploration of the dense passage embedding space.
- **Empirical Analysis of Kernel and Acquisition Functions.** We empirically show that stationary kernels (*e.g.,* RBF, Matérn) are effective for capturing multimodal relevance structures, while uncertainty-based acquisition functions play a critical role in guiding effective exploration.
- **Comprehensive Validation.** Evaluated on four distinct passage retrieval datasets, BAGEL significantly outperforms conventional LLM reranking baselines under the same budget, for example, improving NDCG@50 from 29.3 to 41.6 on the TravelDest dataset.

## 2 Preliminaries

### 2.1 Gaussian Processes

GPs are non-parametric Bayesian models that define a prior over functions, allowing probabilistic function modeling(Williams and Rasmussen, 1995, 2006). Letting $f : \mathbb{R}^d \to \mathbb{R}$, the prior over $f$ is specified as

$$f \sim \mathcal{GP}(0, k(\cdot, \cdot)), \tag{1}$$

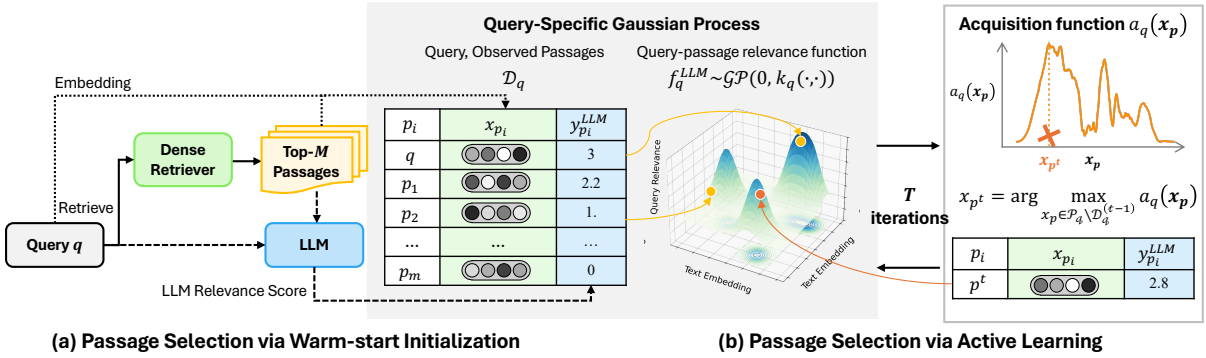


Figure 2: Overview of BAGEL. For each query, BAGEL defines a query-specific Gaussian Process (GP) using LLM-based relevance scores of selected passages in dense embedding space. The process begins with a (a) warm-start initialization by labeling the query itself and top-$M$ dense-retrieved passages. In the (b) active learning phase, an acquisition function combines the GP's predictive mean and uncertainty to iteratively select next passage.

where $k : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ is a chosen kernel function that aims to determine how similarity is measured and defines the function space the GP can represent. Consequently, the choice of kernel function influences properties such as smoothness and complexity of the GP. Given a set of inputs $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]^\top \in \mathbb{R}^{n \times d}$ with an observed output $\mathbf{f}(\mathbf{X}) = \mathbf{y} \in \mathbb{R}^n$, the posterior predictive distribution for a new input $\mathbf{x}_* \in \mathbb{R}^d$ follows a Gaussian distribution with mean and uncertainty:

$$\mu(\mathbf{x}_*) = \mathbf{k}_*^\top (\mathbf{K} + \alpha \mathbf{I})^{-1} \mathbf{y}, \tag{2}$$

$$\sigma^2(\mathbf{x}_*) = k(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{k}_*^\top (\mathbf{K} + \alpha \mathbf{I})^{-1} \mathbf{k}_*, \tag{3}$$

where $\mathbf{K} \in \mathbb{R}^{n \times n}$ is the kernel matrix with $\mathbf{K}_{ij} = k(\mathbf{x}_i, \mathbf{x}_j)$, $\mathbf{k}_* \in \mathbb{R}^n$ contains the kernel values between $\mathbf{x}_*$ and $\mathbf{x}$ (*i.e.,* $k(\mathbf{x}_*, \mathbf{x})$), and $\alpha > 0$ is a hyperparameter that accounts for observation noise. The value of $f(\mathbf{x}_*)$ is estimated by the posterior mean $\mu(\mathbf{x}_*)$, while $\sigma^2(\mathbf{x}_*)$ quantifies the epistemic uncertainty of the model.

## 2.2 Acquisition Function

To guide the sequential search process effectively, it is essential to utilize both the predictive mean and the associated uncertainty provided by the GP. This is achieved through an acquisition function, which maps the posterior distribution to a utility value of each candidate input:

$$a(\mathbf{x}) = \phi\big(\mu(\mathbf{x}),\, \sigma(\mathbf{x});\, \boldsymbol{\theta}\big). \tag{4}$$

Here, $\phi(\cdot)$ defines the specific strategy for combining the posterior mean and variance. By iteratively maximizing the acquisition function and updating the GP posterior, the process actively navigates the search space to converge toward the global optimum.

## 2.3 LLM Query-Passage Relevance Scoring

Recent work (Zhuang et al., 2024; Sachan et al., 2022; Qin et al., 2024; Shen et al., 2023) has proposed using LLMs to estimate query-passage relevance in a zero-shot setting. Following previous work (Zhuang et al., 2024), we formulate the relevance estimation as a constrained generation task where the output is restricted to a single token representing a relevance score (*e.g.,* the token "1"). Under this formulation, given a query $q$, a passage $p$, and an instruction prompt `prompt`, the LLM produces a vector of logits $z = [z_0, z_1, \ldots, z_{K-1}]$, where each component corresponds to a predefined integer relevance label $r_k \in \{0, 1, \ldots, K-1\}$.

$$\mathbf{z} = \mathrm{LLM}(\texttt{prompt}, q, p) \in \mathbb{R}^K \tag{5}$$

.

To derive a scalar relevance score from these logits, (Zhuang et al., 2024) introduces two variants for scoring. The first, *expected relevance* (ER), interprets the logits as a probability distribution via the softmax function and computes the expected relevance value:

$$S_{\mathrm{ER}}(q, p) = \sum_{k=0}^{K-1} \underbrace{\left( \frac{e^{z_k}}{\sum_{j=0}^{K-1} e^{z_j}} \right)}_{P(r_k|q,p)} r_k. \tag{6}$$

The second variant, *peak relevance* (PR), assigns the score corresponding to the relevance label with the highest logit:

$$S_{\mathrm{PR}}(q, p) = r_{\arg\max_k z_k}. \tag{7}$$

We will empirically examine the performance of the proposed method using both variants across datasets in Section 5.1.

## 3 BAGEL: Bayesian Active Learning with Gaussian Processes Guided by LLM Relevance Scoring

In this section, we present BAGEL, a passage retrieval framework that integrates dense retrieval, LLM relevance scoring, and GP–guided exploration. BAGEL employs a query-specific GP to capture the query-passage relevance distribution based on a set of passages with observed LLM relevance scores. These passages are iteratively selected through an active selection process that balances exploration and exploitation as derived from the GP posterior estimates. We begin by introducing the query-specific GP, followed by two phases of active passage selection: (i) a warm-start initialization phase and (ii) an exploration phase guided by an acquisition function.

### 3.1 Query-Specific Gaussian Process with LLM Relevance Score

We first define the query-specific GP for each query $q$, based on LLM relevance scores and extending the formulation in Section 2.1. The query-specific GP takes as input the dense encoder embeddings of the passage $p$, $\mathbf{x}_p \in \mathbb{R}^d$, and the query $q$, $\mathbf{x}_q \in \mathbb{R}^d$, along with their observed LLM relevance scores, to model the query-passage relevance function $f_q^{\text{LLM}}(\mathbf{x}_p)$. This model can then be used to estimate the relevance score for any passage $p_*$ with an unknown LLM relevance score.

Formally, for a query $q$, let

$$\mathcal{D}_q = \{(\mathbf{x}_{p_i},\, y_{p_i}^{\text{LLM}})\}_{i=1}^n$$

denote the set of observed passages with known LLM relevance scores, where $\mathbf{x}_{p_i} \in \mathbb{R}^d$ is the dense embedding of passage $p_i$. Let $S(\cdot,\cdot)$ be the LLM relevance scoring function defined in Section 2.3 (*e.g.,* $S_{\text{ER}}$ or $S_{\text{PR}}$). For each passage $p_i$, the score

$$y_{p_i}^{\text{LLM}} = S(q, p_i)$$

represents its LLM relevance score with respect to the query. The query-specific GP for the query-passage relevance function is then defined as

$$f_q^{\text{LLM}} \sim \mathcal{GP}\big(0,\, k(\cdot,\cdot)\big), \tag{8}$$

where $k$ is the kernel function chosen as discussed in Section 2.1.

While our framework supports any valid kernel function, we adopt the Radial Basis Function (RBF) kernel as the default choice in our experiments. The RBF kernel (Williams and Rasmussen, 2006) is a standard choice of GP and is defined as

$$k_{\text{RBF}}(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell^2}\right), \tag{9}$$

where the length-scale $\ell$ determines how quickly correlations decay with distance. This kernel assumes that inputs that are closer in the input space produce more strongly correlated outputs, allowing the GP to represent complex functions. Additionally, we examine Linear and Matérn kernels, detailed in Appendix A, with empirical comparisons provided in Section 5.3.

Building on the preliminaries, the posterior predictive distribution for a passage $p_*$ with dense embedding $\mathbf{x}_{p_*}$ follows a Gaussian distribution with mean and variance given by:

$$\mu_q(\mathbf{x}_{p_*}) = \mathbf{k}_*^\top \big(\mathbf{K} + \alpha\mathbf{I}\big)^{-1}\mathbf{y}^{\text{LLM}}, \tag{10}$$

$$\sigma_q^2(\mathbf{x}_{p_*}) = k(\mathbf{x}_{p_*}, \mathbf{x}_{p_*}) - \mathbf{k}_*^\top \big(\mathbf{K} + \alpha\mathbf{I}\big)^{-1}\mathbf{k}_*, \tag{11}$$

where $\mathbf{K} \in \mathbb{R}^{n\times n}$ now represents the kernel matrix between the dense embeddings of the selected passages in $\mathcal{D}_q$, and $\mathbf{k}_* \in \mathbb{R}^n$ is the kernel vector between the previously observed passages and the selected passage $\mathbf{x}_{p_*}$.

### 3.2 Passage Selection via Warm-start Initialization

Our warm-start strategy aims to initialize BAGEL with reliable, high-quality signals in embedding regions that are likely to contain relevant passages. This approach reduces early-stage uncertainty and mitigates the cold-start problem. To this end, we utilize the query itself and top-ranked passages from a dense retriever to guide the initial selection process.

We first treat the query itself as an observation with maximum relevance. BAGEL treats the query embedding $\mathbf{x}_q$ as an observation and assigns it a relevance score $y_q^{\text{LLM}}$ equal to the maximum possible value under the defined LLM relevance scoring function (*e.g.,* $S_{\text{ER}}$ or $S_{\text{PR}}$). This ensures that the GP receives a strong positive signal and guided to model the relationship between the query and passages. To augment this initial signal with promising passages, we retrieve the top-$M$ passages $\{p_1, p_2, \ldots, p_M\}$ using a dense retriever

given the query $q$. The resulting observed pairs $\{(\mathbf{x}_{p_i}, y_{p_i}^{\text{LLM}})\}_{i=1}^{M}$, together with the query embedding–score pair $(\mathbf{x}_q, y_q^{\text{LLM}})$, form the warm-start observation set:

$$\mathcal{D}_q^{(0)} = \{(\mathbf{x}_{p_i}, y_{p_i}^{\text{LLM}})\}_{i=1}^{M} \cup \{(\mathbf{x}_q, y_q)\}. \quad (12)$$

### 3.3 Passage Selection via Active Learning

After warm-start initialization, BAGEL iteratively selects new passages for LLM relevance scoring by applying the *acquisition function* introduced in Section 2.2. This phase enables efficient exploration of the entire passage embedding space while judiciously allocating expensive LLM relevance scoring.

Instead of generic inputs, we evaluate the utility of each candidate passage embedding $\mathbf{x}_p$ using the query-specific GP predictive mean $\mu_q(\mathbf{x}_p)$ and predictive standard deviation $\sigma_q(\mathbf{x}_p)$. While BAGEL is compatible with various strategies, we primarily utilize the Upper Confidence Bound (UCB) (Srinivas et al., 2010).

Specifically, the UCB acquisition function is defined as

$$a^{\text{UCB}}(\mathbf{x}) = \mu_q(\mathbf{x}) + \sqrt{\beta}\,\sigma_q(\mathbf{x}), \quad (13)$$

where $\beta > 0$ is a hyperparameter that balances exploration (high uncertainty) and exploitation (high predictive mean). Additional acquisition function formulations are provided in Appendix B, and empirical comparisons can be found in Section 5.3.

At iteration $t \in \{1, \ldots, T\}$, where $T$ denotes the remaining LLM budget, BAGEL selects the next passage $p^t$ to label by maximizing the acquisition score over the pool of unlabeled passages:

$$\mathbf{x}_{p^t} = \arg \max_{\mathbf{x}_p \in \mathcal{P}_q \setminus \mathcal{D}_q^{(t-1)}} a^{\text{UCB}}(\mathbf{x}_p). \quad (14)$$

We then obtain the LLM-based relevance score $y_{p^t}^{\text{LLM}} = S(q, p^t)$ and update the observation set:

$$\mathcal{D}_q^{(t)} = \mathcal{D}_q^{(t-1)} \cup \{(\mathbf{x}_{p^t}, y_{p^t}^{\text{LLM}})\}, \quad (15)$$

where $\mathcal{P}_q$ is the set of all candidate passages for query $q$ and $\mathcal{D}_q^{(t-1)}$ is the set of passages with observed relevance scores up to the previous iteration $t-1$. This process repeats for $T$ iterations. Finally, the updated GP estimates the relevance scores for all passages in $\mathcal{P}_q$. Notably, BAGEL supports anytime prediction; the query-specific GP can generate a ranking over all passages after any iteration $t$, making the method highly adaptable to online settings with varying budget constraints.

## 4 Experimental Setup

We address the following research questions:

**RQ1**: Does BAGEL outperform LLM reranking baselines under a fixed budget across different LLMs?

**RQ2**: How does BAGEL balance exploration and exploitation in the embedding space?

**RQ3**: How do components such as the kernel and the acquisition function impact the performance of BAGEL?

### 4.1 Baselines

We compare BAGEL against five representative baselines spanning traditional sparse retrieval (**BM25** (Robertson et al., 1995)), dense retrieval (**Dense Retriever** (Reimers and Gurevych, 2019)), neural reranking (**Cross Encoder** (Nogueira et al., 2019)), and LLM-based reranking (**Pointwise LLM** (Zhuang et al., 2024) and **Listwise LLM** (Sun et al., 2023)). We refer to Appendix C.1 for detailed descriptions of each baseline. We evaluate all methods using NDCG and Recall at cutoffs of $k = 10, 50$.

### 4.2 Datasets

We evaluate our method on four passage retrieval datasets, including Covid, NFCorpus, and Robust04 from the BEIR benchmark (Thakur et al., 2021), and TravelDest (Wen et al., 2024). The former two datasets serve as domain-specific benchmarks, while the latter two feature ambiguous queries. Dataset statistics and details are reported in Appendix C.2.

### 4.3 Implementation Details

For all reranking methods, we set a total budget of 50 LLM calls. For BAGEL this budget is split into $M = 25$ calls for warm-start initialization and $T = 25$ calls for active learning. We set the GP noise variance $\alpha$ and UCB scaling factor $\beta$ to 0.001 and 2, respectively. For the dense retriever backbone, we utilize `all-MiniLM-L6-v2` with a dimension of 384. Our experiments employ Qwen3-14B (Yang et al., 2025) and GPT-4o (Hurst et al., 2024) as LLM backbones. We conduct LLM query-passage relevance scoring based on the Umbrela (Upadhyay et al., 2024) template, assigning integer relevance labels from $\{0, 1, 2, 3\}$, with the full template provided in Table 4. Further implementation details on BAGEL and baselines are provided in Appendix C.3.

Table 1: Overall Performance with a fixed LLM budget of 50 calls per query. DR and CE denote the dense retriever and cross encoder baselines, respectively. List. and Point. indicate Listwise and Pointwise LLM. Bold denotes the best score within the same LLM backbone. R@k and N@k refer to Recall@k and NDCG@k, respectively.

| Dataset | LLM | Baseline | | | Qwen3-14B | | | | | GPT-4o | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Score/ Method | BM25 | DR | CE | List. | PR Point. | PR Ours | ER Point. | ER Ours | List. | PR Point. | PR Ours | ER Point. | ER Ours |
| Covid | N@10 | 50.4 | 57.0 | 66.5 | 71.8 | 74.2 | 76.6 | 76.5 | **77.2** | 73.2 | 71.9 | 70.4 | 74.6 | **74.7** |
| | N@50 | 42.8 | 48.7 | 51.1 | 52.3 | 52.9 | 61.4 | 53.6 | **63.6** | 52.6 | 52.4 | 57.7 | 52.8 | **62.1** |
| | R@10 | 1.7 | 2.0 | 2.3 | 2.4 | 2.5 | **2.6** | **2.6** | **2.6** | 2.5 | 2.5 | 2.5 | **2.6** | **2.6** |
| | R@50 | 7.3 | 7.7 | 7.7 | 7.7 | 7.7 | 9.5 | 7.7 | **9.7** | 7.7 | 7.7 | 9.2 | 7.7 | **9.9** |
| NFCorpus | N@10 | 31.5 | 31.2 | 33.7 | 34.7 | 37.5 | 37.7 | 37.5 | **38.4** | 38.4 | 37.8 | 38.9 | 39.1 | **40.6** |
| | N@50 | 27.7 | 29.0 | 30.7 | 30.9 | 32.7 | 32.8 | 32.7 | **33.6** | 33.0 | 32.8 | 32.8 | 33.5 | **35.9** |
| | R@10 | 15.1 | 15.3 | 15.5 | 16.8 | 17.7 | 18.0 | 17.5 | **18.3** | 18.1 | 17.9 | 18.2 | 18.2 | **19.0** |
| | R@50 | 21.4 | 25.3 | 25.3 | 25.3 | 25.3 | 25.9 | 25.3 | **26.4** | 25.3 | 25.3 | 24.5 | 25.3 | **27.3** |
| Robust04 | N@10 | 38.9 | 39.1 | 46.4 | 50.5 | 52.8 | 55.9 | 53.8 | **57.3** | 50.6 | 55.5 | 60.4 | 57.3 | **62.1** |
| | N@50 | 34.9 | 33.2 | 36.1 | 36.8 | 38.2 | 44.4 | 38.6 | **45.7** | 35.2 | 39.1 | 45.9 | 40.0 | **48.7** |
| | R@10 | 13.4 | 12.1 | 14.2 | 14.3 | 15.6 | 16.7 | 15.7 | **17.4** | 14.0 | 16.3 | 17.6 | 16.4 | **18.8** |
| | R@50 | 29.0 | 24.9 | 24.9 | 24.9 | 24.9 | 30.7 | 24.9 | **31.7** | 21.7 | 24.9 | 30.7 | 24.9 | **33.2** |
| TravelDest | N@10 | 21.1 | 22.3 | 43.1 | 48.6 | 45.8 | 49.8 | 48.3 | **51.0** | 53.3 | 48.9 | 57.0 | 55.1 | **58.5** |
| | N@50 | 17.2 | 21.6 | 26.4 | 27.7 | 27.2 | 37.4 | 27.9 | **38.0** | 28.8 | 28.2 | 40.2 | 29.3 | **41.6** |
| | R@10 | 1.0 | 0.9 | 1.7 | 2.1 | 2.1 | **2.5** | 2.2 | **2.5** | 2.3 | 2.0 | 2.7 | 2.3 | **2.9** |
| | R@50 | 3.4 | 3.9 | 3.9 | 3.9 | 3.9 | 6.7 | 3.9 | **6.8** | 3.9 | 3.9 | 6.7 | 3.9 | **6.9** |

## 5 Experimental Results

### 5.1 Overall Performance

To answer **RQ1**, Table 1 presents the results on four datasets under a fixed LLM budget of 50 calls per query. Across all four datasets and both LLM backbones, BAGEL consistently outperforms all baselines methods. Notably, in contrast to conventional rerankers confined to the initial candidate pool, the Recall@50 results demonstrate that BAGEL discovers relevant passages beyond this fixed set by actively exploring the embedding space. Furthermore, the widening performance gap at deeper cutoffs (*e.g.,* NDCG@50 vs. NDCG@10) reveals that while baselines saturate on top-ranked positives, BAGEL utilizes these discovered passages to maintain high relevance density even at lower ranks where baselines typically falter.

**GPT-4o vs. Qwen3-14B** Table 1 shows that GPT-4o not only boosts overall performance but also amplifies the gains of BAGEL over pointwise baselines compared to Qwen3-14B. We attribute this to GPT-4o's stronger ranking capability, which yields a relevance score distribution with lower noise. This high-quality supervision allows the Gaussian Process to learn a more stable posterior, thereby enhancing both mean estimation and uncertainty quantification for more effective exploration.

**ER vs. PR** Comparing the scoring functions, $S_{\mathrm{ER}}$ consistently yields larger gains over the pointwise LLM than $S_{\mathrm{PR}}$. This result stems from the inductive bias of GPs, which assumes a continuous, smooth target function. $S_{\mathrm{ER}}$ aligns with this assumption by providing smooth, real-valued scores via the probability-weighted average of relevance scores. In contrast, the coarse, discrete labels of $S_{\mathrm{PR}}$ result in step-like supervision, obscuring fine-grained relevance patterns and hindering effective posterior learning.

Based on these observations, subsequent experiments are conducted using $S_{\mathrm{ER}}$ with Qwen3-14B as the default backbone. Detailed analyses of varying LLM budgets and latency are provided in Appendix D and Appendix E, respectively.

### 5.2 Case Study

To better understand BAGEL's selection strategy for **RQ2**, Figure 3 shows UMAP (McInnes et al., 2018) projections of (i) the dense retriever's Top-50 passages, (ii) warm-start passages and passages selected during BAGEL's active learning, and (iii) Top-50 passages of BAGEL.

In Covid (Figure 3a), for the query *"what are the benefits and risks of re-opening schools in the midst of the COVID-19 pandemic?"*, the most relevant passages are concentrated near the query embedding, with a few located farther away. While the dense retriever overlooks a left-side cluster, BAGEL gradually explores high-uncertainty regions and uncovers these overlooked passages.

In TravelDest (Figure 3b), for the general query *"I want to capture stunning sunshine"*, the relevant passages are scattered across the entire embedding

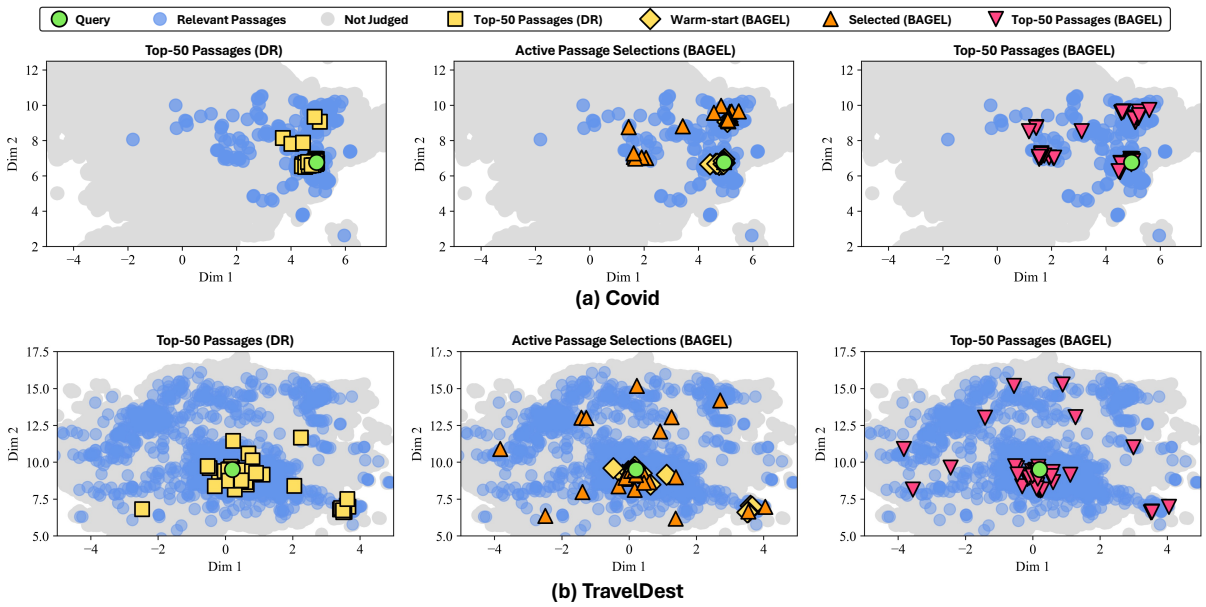


Figure 3: UMAP projections from Covid and TravelDest showing how BAGEL balances exploitation-exploration in two different relevance distributions: locally concentrated clusters ((a) Covid), and globally dispersed clusters ((b) TravelDest). Observe specifically that BAGEL (right) samples a broader selection of relevant passages (middle) compared to dense retrieval (left).

Table 2: Performance by different kernels, where stationary kernels excel by capturing score multimodality.

| Dataset | Kernel | N@10 | N@50 | R@10 | R@50 |
|---|---|---|---|---|---|
| Covid | Linear | 16.43 | 11.88 | 0.50 | 1.76 |
| | Matérn | 75.24 | 62.31 | 2.54 | 9.64 |
| | RBF | 77.24 | 63.64 | 2.60 | 9.66 |
| NFCorpus | Linear | 36.91 | 31.95 | 17.35 | 24.74 |
| | Matérn | 38.29 | 33.78 | 18.19 | 26.65 |
| | RBF | 38.40 | 33.65 | 18.31 | 26.42 |
| Robust04 | Linear | 25.47 | 18.52 | 7.29 | 11.72 |
| | Matérn | 57.71 | 46.23 | 17.32 | 32.26 |
| | RBF | 57.28 | 45.66 | 17.39 | 31.72 |
| TravelDest | Linear | 21.91 | 14.00 | 0.97 | 2.24 |
| | Matérn | 50.92 | 38.04 | 2.53 | 6.72 |
| | RBF | 50.96 | 38.02 | 2.52 | 6.82 |

space. The dense retriever, constrained by its implicit unimodal assumption, focuses near the query embedding and neglects other clusters. In contrast, BAGEL actively explores diverse regions with high uncertainty and retrieves relevant passages from multiple clusters, leading to more balanced coverage.

These visualizations illustrate how BAGEL adapts its exploitation–exploration balance to the underlying relevance structure. In relatively concentrated distributions like Covid, it tends to explore locally, whereas in dispersed multimodal distributions such as TravelDest, it explores globally to cover all major clusters.

### 5.3 Component Analysis

We next examine how components influence the performance of BAGEL across datasets, addressing **RQ3**. Additional analyses on hyperparameters (*e.g.*, $\alpha$, $\beta$) and the number of warm-start passages are provided in Appendix F.

**Kernel** To examine the impact of kernel choice, we compare the Linear and Matérn kernels in addition to the RBF kernel. Table 2 shows results across four datasets. We observe that RBF and Matérn perform similarly and consistently outperform the Linear kernel. Unlike Linear, *stationary kernels* (RBF and Matérn) effectively model complex, multimodal relevance landscapes by preserving local neighborhoods based on relative distance (Williams and Rasmussen, 2006). Consequently, RBF and Matérn achieve comparable performance and significantly surpass the Linear kernel. This performance gap between two kernel types indicates the multimodality in the relevance score distribution.

**Acquisition Function** To evaluate active passage selection strategies, Table 3 compares several acquisition functions with 25 warm-start passages and 25 active selection passages. As a result Random and Dense yield the weakest performance, as they fail to balance exploration and exploitation: Random selects passages in an arbitrary manner, whereas Dense restricts itself to local optima. In

Table 3: Performance by different acquisition functions, demonstrating the advantage of uncertainty-guided exploration over naive selection.

| Dataset | Acq. | N@10 | N@50 | R@10 | R@50 |
|---|---|---|---|---|---|
| Covid | Random | 76.0 | 60.4 | 2.5 | 9.1 |
| | Dense | 73.9 | 57.9 | 2.4 | 8.4 |
| | PI | 76.9 | 60.7 | 2.6 | 9.3 |
| | EI | 76.3 | 63.3 | 2.7 | 9.9 |
| | TS | 75.8 | 62.0 | 2.5 | 9.4 |
| | UCB | 77.2 | 63.6 | 2.6 | 9.7 |
| NFCorpus | Random | 37.5 | 33.8 | 18.1 | 28.3 |
| | Dense | 37.8 | 33.1 | 18.2 | 26.5 |
| | PI | 38.1 | 33.6 | 17.9 | 26.8 |
| | EI | 38.5 | 33.8 | 18.0 | 26.9 |
| | TS | 37.7 | 34.4 | 18.4 | 29.0 |
| | UCB | 38.4 | 33.6 | 18.3 | 26.4 |
| Robust04 | Random | 50.2 | 40.8 | 14.8 | 28.7 |
| | Dense | 54.3 | 41.6 | 15.7 | 28.5 |
| | PI | 57.1 | 45.8 | 16.9 | 32.0 |
| | EI | 57.1 | 45.9 | 17.0 | 32.2 |
| | TS | 50.4 | 41.5 | 14.6 | 29.2 |
| | UCB | 57.3 | 45.7 | 17.4 | 31.7 |
| TravelDest | Random | 44.7 | 34.1 | 2.1 | 5.8 |
| | Dense | 49.9 | 36.3 | 2.4 | 6.0 |
| | PI | 51.9 | 38.4 | 2.6 | 6.8 |
| | EI | 50.8 | 37.6 | 2.5 | 6.6 |
| | TS | 42.4 | 33.7 | 2.0 | 5.8 |
| | UCB | 51.0 | 38.0 | 2.5 | 6.8 |

contrast, Bayesian acquisition strategies (PI, EI, TS, and UCB) consistently outperform these baselines by leveraging the GP posterior to guide exploration. While differences among them are minor, they collectively demonstrate the advantage of uncertainty-guided selection.

# 6 Related Work

## 6.1 Passage Retrieval

Passage retrieval aims to identify relevant passages from corpus given a query (Manning et al., 2010). Traditional approaches include sparse lexical methods (Robertson et al., 1995) and dense embedding methods (Karpukhin et al., 2020). While scalable, these methods often struggle to capture complex semantic relationships. To address this, reranking paradigms (Khattab and Zaharia, 2020; Nogueira et al., 2019) employ sophisticated models to reorder candidates filtered by a lightweight retriever. However, this pipeline is bottlenecked by the recall of initial retriever; any relevant passage missed in the first stage remains inaccessible.

## 6.2 LLM-based Reranking

LLMs enhance semantic comprehension and relevance scoring (Hurst et al., 2024; Yang et al., 2025) through pointwise or listwise prompting. Pointwise methods score query-passage pairs independently, using generation likelihood (Sachan et al., 2022) or fine-grained labels (Zhuang et al., 2024). Conversely, listwise methods rank multiple passages simultaneously by generating a ranked list (Ma et al., 2023). For instance, RankGPT (Sun et al., 2023) orders passage identifiers using a sliding window. These approaches often match or exceed supervised baselines that rely on human-annotated relevance labels, particularly in zero-shot settings (Ma et al., 2023; Sun et al., 2023), but BAGEL outperforms both Pointwise and Listwise LLM reranking.

## 6.3 Gaussian Processes

Gaussian Processes (GPs) are non-parametric Bayesian models that define a distribution over functions, providing principled uncertainty quantification (Williams and Rasmussen, 2006, 1995). By modeling surrogate functions with well-calibrated uncertainty, GPs facilitate sample-efficient exploration via acquisition strategies (Srinivas et al., 2010). This makes them particularly effective in limited-supervision scenarios, such as hyperparameter optimization (Snoek et al., 2012) and active learning task (Di Fiore et al., 2023). Recent findings also show that GPs maintain robustness in high-dimensional settings (Xu et al., 2025). Despite their success in other domains, GPs remain largely unexplored in the context of passage retrieval.

# 7 Conclusion

In this paper, we introduce BAGEL, a framework that provides a novel integration of Gaussian Process-based Bayesian active learning with LLM relevance scoring. Unlike traditional reranking methods restricted to a static candidate set, BAGEL actively explores the dense embedding space, propagating relevance signals to discover semantically distinct clusters that are often overlooked. Our experiments demonstrate that by leveraging Bayesian uncertainty to guide the selection of passages, BAGEL significantly improves retrieval performance on all four datasets compared to LLM reranking baselines under fixed computational budgets. Overall BAGEL demonstrates the potential of Bayesian active learning combined with LLM-based relevance scoring to make effective and parsimonious use of a limited LLM-budget for retrieval.

## 8 Limitations

The limitations of this study can be primarily categorized into three aspects regarding data dependency, robustness, and scalability. First, the framework exhibits a strong dependence on the initial embedding quality, as the kernel function operates directly within the pre-trained dense retriever's latent space; consequently, suboptimal semantic mapping or kernel mismatches may hinder the effective propagation of relevance signals across the manifold. Second, the sensitivity to LLM scoring noise poses a potential risk of reinforcing hallucinations. If the LLM generates factually incorrect relevance scores, the Gaussian Process may confidentially propagate these errors, leading to a degraded retrieval model that prioritizes misinformation. Finally, scalability to web-scale corpora remains a challenge, as calculating uncertainty for every unobserved document in a billion-scale index is computationally prohibitive, necessitating future research into efficient pruning or hierarchical search strategies for real-world deployment.

## References


Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Francesco Di Fiore, Michela Nardelli, and Laura Mainini. 2023. Active learning and bayesian optimization: a unified perspective to learn with a goal. *arXiv preprint arXiv:2303.01560*.

Daniel Han, Michael Han, and Unsloth team. 2023. Unsloth. http://github.com/unslothai/unsloth.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 6769–6781. Association for Computational Linguistics.

Omar Khattab and Matei Zaharia. 2020. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 39–48. Association for Computing Machinery.

Oren Kurland. 2013. The cluster hypothesis in information retrieval. In *Proceedings of the 36th International ACM SIGIR Conference on Research and Development in Information Retrieval*, page 1126. Association for Computing Machinery.

Harold J Kushner. 1964. A new method of locating the maximum point of an arbitrary multipeak curve in the presence of noise. *Journal of Basic Engineering*, 86:97–106.

Jimmy Lin, Xueguang Ma, Sheng-Chieh Lin, Jheng-Hong Yang, Ronak Pradeep, and Rodrigo Nogueira. 2021. Pyserini: A Python toolkit for reproducible information retrieval research with sparse and dense representations. In *Proceedings of the 44th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR 2021)*, pages 2356–2362.

Xueguang Ma, Xinyu Zhang, Ronak Pradeep, and Jimmy Lin. 2023. Zero-shot listwise document reranking with a large language model. *arXiv preprint arXiv:2305.02156*.

Christopher D Manning, Prabhakar Raghavan, and Hinrich Schütze. 2010. Introduction to information retrieval. *Natural Language Engineering*, 16(1):100–103.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861.

Jonas Mockus. 1998. The application of bayesian methods for seeking the extremum. *Towards Global Optimization*, 2:117.

Rodrigo Nogueira, Wei Yang, Kyunghyun Cho, and Jimmy Lin. 2019. Multi-stage document ranking with bert. *arXiv preprint arXiv:1910.14424*.

Zhen Qin, Rolf Jagerman, Kai Hui, Honglei Zhuang, Junru Wu, Le Yan, Jiaming Shen, Tianqi Liu, Jialu Liu, Donald Metzler, Xuanhui Wang, and Michael Bendersky. 2024. Large language models are effective text rankers with pairwise ranking prompting. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 1504–1518. Association for Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, and 1 others. 1995. Okapi at trec-3. British Library Research and Development Department.

Devendra Sachan, Mike Lewis, Mandar Joshi, Armen Aghajanyan, Wen-tau Yih, Joelle Pineau, and Luke Zettlemoyer. 2022. Improving passage retrieval with zero-shot question generation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 3781–3797. Association for Computational Linguistics.

Tao Shen, Guodong Long, Xiubo Geng, Chongyang Tao, Tianyi Zhou, and Daxin Jiang. 2023. Large language models are strong zero-shot retriever. *arXiv preprint arXiv:2304.14233*.

Jasper Snoek, Hugo Larochelle, and Ryan P Adams. 2012. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 25, pages 2960–2968.

Niranjan Srinivas, Andreas Krause, Sham Kakade, and Matthias Seeger. 2010. Gaussian process optimization in the bandit setting: no regret and experimental design. In *Proceedings of the 27th International Conference on Machine Learning*, pages 1015–1022. Omnipress.

Ingo Steinwart. 2001. On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, 2:67–93.

Weiwei Sun, Lingyong Yan, Xinyu Ma, Shuaiqiang Wang, Pengjie Ren, Zhumin Chen, Dawei Yin, and Zhaochun Ren. 2023. Is chatgpt good at search? investigating large language models as re-ranking agents. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 14918–14937. Association for Computational Linguistics.

Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021. Beir: A heterogenous benchmark for zero-shot evaluation of information retrieval models. *arXiv preprint arXiv:2104.08663*.

William R Thompson. 1933. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294.

Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Nick Craswell, and Jimmy Lin. 2024. Umbrela: Umbrela is the (open-source reproduction of the) bing relevance assessor. *arXiv preprint arXiv:2406.06519*.

Lakshmi Vikraman, Ali Montazeralghaem, Helia Hashemi, W. Bruce Croft, and James Allan. 2021. Passage similarity and diversification in non-factoid question answering. In *Proceedings of the 2021 ACM SIGIR International Conference on Theory of Information Retrieval*, pages 271–280. Association for Computing Machinery.

Qianfeng Wen, Yifan Liu, Joshua Zhang, George Saad, Anton Korikov, Yury Sambale, and Scott Sanner. 2024. Elaborative subtopic query reformulation for broad and indirect queries in travel destination recommendation. *arXiv preprint arXiv:2410.01598*.

Christopher KI Williams and Carl Edward Rasmussen. 1995. Gaussian processes for regression. In *Advances in Neural Information Processing Systems*, volume 8, pages 514–520. MIT Press.

Christopher KI Williams and Carl Edward Rasmussen. 2006. *Gaussian processes for machine learning*, volume 2. MIT Press.

Zhitong Xu, Haitao Wang, Jeff M. Phillips, and Shandian Zhe. 2025. Standard gaussian process is all you need for high-dimensional bayesian optimization. In *The Thirteenth International Conference on Learning Representations*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Honglei Zhuang, Zhen Qin, Kai Hui, Junru Wu, Le Yan, Xuanhui Wang, and Michael Bendersky. 2024. Beyond yes and no: Improving zero-shot llm rankers via scoring fine-grained relevance labels. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 358–370. Association for Computational Linguistics.

| **Prompt** |
| --- |
| Given a query and a list of passages, you must provide a score on an integer scale of 0 to 3 with the following meanings: |
| 0 = represent that the passage has nothing to do with the query, |
| 1 = represents that the passage seems related to the query but does not answer it, |
| 2 = represents that the passage has some answer for the query, but the answer may be a bit unclear, or hidden amongst extraneous information and |
| 3 = represents that the passage is dedicated to the query and contains the exact answer. |
| Important Instruction: Assign category 1 if the passage is somewhat related to the topic but not completely, category 2 if passage presents something very important related to the entire topic but also has some extra information and category 3 if the passage only and entirely refers to the topic. If none of the above satisfies give it category 0. |
| Query: {**query**} |
| Passage: {**passage**} |
| Split this problem into steps: |
| Consider the underlying intent of the search. |
| Measure how well the content matches a likely intent of the query (M). |
| Measure how trustworthy the passage is (T). |
| Consider the aspects above and the relative importance of each, and decide on a final score (O). Final score must be an integer value only. |
| Do not provide any code or reasoning in result. Provide only the score without any explanation. |
| ##final score: |

Table 4: Prompt for LLM-based scoring in Section 2.3. Both **query** and **passage** are placeholders.

## A Kernel Function

The kernel function $k(\mathbf{x}, \mathbf{x}')$ defines the similarity between two inputs. We use the following kernels (Steinwart, 2001):

- **Linear (Dot Product) Kernel:**

$$k_{\text{lin}}(\mathbf{x}, \mathbf{x}') = \mathbf{x}^\top \mathbf{x}'$$

- **Matérn Kernel:**

$$k_{\text{Matérn}}(\mathbf{x}, \mathbf{x}') = \frac{2^{1-\nu}}{\Gamma(\nu)} \left( \frac{\sqrt{2\nu}\, \|\mathbf{x} - \mathbf{x}'\|}{\ell} \right)^{\nu} \times K_\nu \left( \frac{\sqrt{2\nu}\, \|\mathbf{x} - \mathbf{x}'\|}{\ell} \right)$$

- **Radial Basis Function (RBF) Kernel:**

$$k_{\text{RBF}}(\mathbf{x}, \mathbf{x}') = \exp\left( -\frac{\|\mathbf{x} - \mathbf{x}'\|^2}{2\ell^2} \right)$$

## B Acquisition Functions

Acquisition functions determine which passages to evaluate next. We employ two heuristic functions (Random, Dense) and four Bayesian functions (EI, PI, TS, UCB). Let $\mu(\mathbf{x})$ and $\sigma(\mathbf{x})$ denote the GP posterior mean and standard deviation for passage $\mathbf{x}$, $f^*$ the current best observed score, and $\xi \geq 0$ a parameter controlling exploration. $\Phi(\cdot)$ and $\phi(\cdot)$ as the standard normal CDF and PDF.

- **Random:** Selects passages uniformly at random without using any scoring function.

- **Dense:** Selects passages with the highest dense retriever scores, equivalent to selecting the top-$n$ passages from the initial warm-start set without active learning.

- **Probability of Improvement (PI)** (Kushner, 1964): Selects passages most likely to surpass $f^*$:

$$\alpha_{\text{PI}}(\mathbf{x}) = \Phi\left( \frac{\mu(\mathbf{x}) - f^* - \xi}{\sigma(\mathbf{x})} \right).$$

- **Expected Improvement (EI)** (Mockus, 1998): Selects passages expected to yield the greatest improvement over $f^*$:

$$\alpha_{\text{EI}}(\mathbf{x}) = (\mu(\mathbf{x}) - f^* - \xi)\, \Phi(Z) + \sigma(\mathbf{x})\, \phi(Z), \qquad Z = \frac{\mu(\mathbf{x}) - f^* - \xi}{\sigma(\mathbf{x})}.$$

- **Thompson Sampling (TS)** (Thompson, 1933): Draws a sample from the GP posterior and selects the passage with the highest sampled value:

$$\mathbf{x}_{\text{next}} = \arg\max_{\mathbf{x}} \tilde{f}(\mathbf{x}), \qquad \tilde{f} \sim \mathcal{GP}(\mu, k).$$

- **Upper Confidence Bound (UCB)** (Srinivas et al., 2010): Selects passages with the highest optimistic estimate of relevance:

$$\alpha_{\text{UCB}}(\mathbf{x}) = \mu(\mathbf{x}) + \sqrt{\beta}\, \sigma(\mathbf{x}),$$

where $\beta > 0$ controls the exploration–exploitation trade-off.

Table 5: Data statistics.

| Dataset | # Query | # Corpus | # Qrel |
|---|---|---|---|
| Covid | 50 | 171,332 | 55,853 |
| NFCorpus | 323 | 3,633 | 12,288 |
| Robust04 | 249 | 528,155 | 308,857 |
| TravelDest | 100 | 131,268 | 2,314,420 |

# C Additional Experimental Setup

## C.1 Baseline

We compare BAGEL against the following five baselines:

- **BM25** (Robertson et al., 1995): A traditional sparse retrieval method based on term frequency –inverse document frequency (TF–IDF) weighting.
- **Dense Retriever** (Reimers and Gurevych, 2019): A dual-encoder model that encodes queries and passages into dense vectors and retrieves based on vector similarity. This model serves both as the first-stage retriever for all reranking baselines and as the dense retrieval component for initializing BAGEL.
- **Cross Encoder** (Nogueira et al., 2019): A BERT-based reranker that jointly encodes a query–passage pair, leveraging token-level interactions to predict fine-grained relevance scores.
- **Pointwise LLM** (Zhuang et al., 2024): An LLM-based method that independently scores each query–passage pair. Our implementation follows the scoring variants described in Section 2.3. Note that BAGEL adopts this same approach to obtain LLM relevance scores during the active search process.
- **Listwise LLM** (Sun et al., 2023): An LLM-based reranking approach that inputs a list of candidate passages into the LLM simultaneously to generate a reordered list based on global context.

## C.2 Dataset Statistics

Table 5 reports the statistics of the datasets used in our experiments. TravelDest contains queries, passages, and cities, with each passage linked to a city in a many-to-one relationship. As it only provides query–city relevance annotations, we generate query–passage labels by prompting an LLM to assess the relevance between each query and passages from its ground-truth city. For this, we use Gemini-2.5-Flash (Comanici et al., 2025) with a modified binary version of the Umbrela prompt (Upadhyay et al., 2024). For each dataset, we list the number of queries, the size of the passage corpus, and the number of relevance annotations (qrels). These statistics are provided for completeness and reproducibility.

## C.3 Additional Implementation Details

All experiments were conducted using a single training run on an NVIDIA H100 GPU with a 40GB MIG partition.

**BAGEL** For BAGEL, The target values $y$ are standardized to have zero mean and unit variance prior to training. Before each acquisition function evaluation, the Gaussian Process hyperparameters are reoptimized from scratch. The kernel length scale $\ell$ is optimized using a learning rate of $0.01$ and constrained to the interval $[0.01, 2]$, with a uniform prior defined over the same range, while the output scale is optimized using a $\mathrm{Gamma}(2, 2)$ prior.

**Baselines** We implemented the retrieval baselines using standard open-source libraries: BM25 via Pyserini (Lin et al., 2021) and dense retrievers/cross-encoders via `sentence-transformers` (Reimers and Gurevych, 2019). To optimize computational efficiency during LLM inference, Qwen3-14B was loaded in 4-bit precision using the unsloth framework (Han et al., 2023). Regarding the LLM reranking baselines, pointwise LLM reranking is implemented identically to the query-passage relevance scoring of BAGEL. Full prompt templates used for pointwise scoring are detailed in Table 4. Listwise LLM ranking follows the RankGPT (Sun et al., 2023) with a sliding window size of 50.

# D Budget

To examine whether BAGEL utilizes the LLM budget more effectively than existing LLM reranking methods, we compare performance under varying LLM budgets, as illustrated in Figure 4. The number of warm-start passages is fixed at 25, with the remainder allocated to active learning. As shown, BAGEL consistently outperforms the Pointwise LLM baseline. Notably, BAGEL achieves comparable performance with significantly fewer calls, demonstrating that it maximizes budget utility by

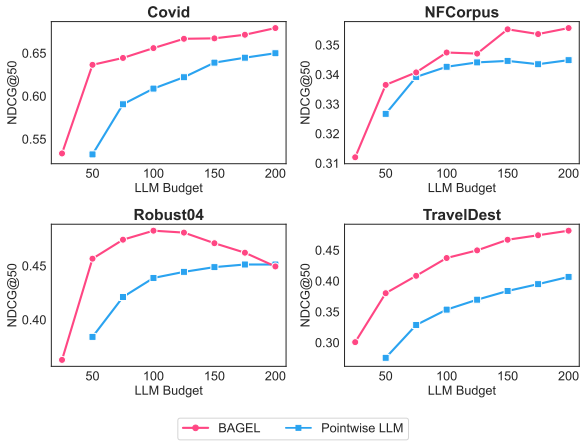


Figure 4: Performance by different values of budget.

Table 6: Comparison of average latency (seconds per query) between BAGELwith different acquisition functions. Thomp. denotes Thompson Sampling and COV, NFC, ROB, and TRAV stand for COVID, NFCorpus, Robust04, and TravelDest.

| Latency (sec) | | COV | NFC | ROB | TRAV |
|---|---|---|---|---|---|
| Pointwise LLM | | 4.77 | 4.95 | 6.72 | 4.19 |
| Ours | Random | 5.48 | 5.10 | 8.62 | 4.94 |
| | Greedy | 5.33 | 5.07 | 7.15 | 4.32 |
| | PI | 7.44 | 6.17 | 12.98 | 6.83 |
| | EI | 7.49 | 6.18 | 13.08 | 6.86 |
| | Thomp. | 7.37 | 6.20 | 12.97 | 6.88 |
| | UCB | 7.38 | 6.21 | 12.88 | 6.75 |

propagating relevance signals via the GP model and strategically balancing exploitation and exploration.

## E Latency

To assess computational efficiency, Table 6 compares the average latency per query of BAGEL against the LLM pointwise baseline. Although BAGEL requires additional time for GP updates, resulting in higher latency compared to the pointwise baseline, this overhead is acceptable given the substantial performance gains. Furthermore, both methods operate within the same constraint of 50 LLM calls, ensuring fair resource usage.

## F Additional Parameter Study

**Number of Warm-start Passages** Figure 5 shows how varying the number of warm-start passages affects performance, with a fixed total LLM budget of 50 and the remainder after the warm start allocated to active learning. We observe different

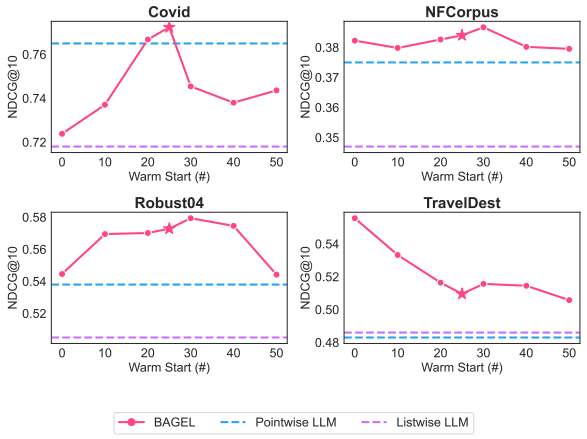


Figure 5: Effect of number of warm-start passages on performance. The star represents performance with default value ($M = 25$).

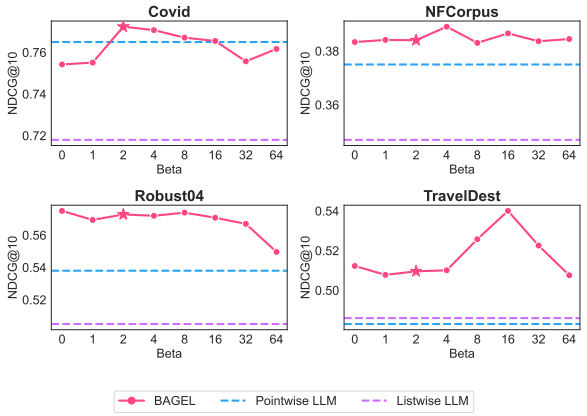


Figure 6: Performance by different values of beta. The blue dots represent performance with default parameter ($\beta = 2$)

trends across datasets. This discrepancy may be due to differences in the quality of the warm-start passages retrieved by the dense retriever. In general, however, performance tends to be low when there is no warm start at all, likely because navigating the high-dimensional embedding space without any initial guidance is inherently challenging.

**Beta** Figure 6 examines the effect of the parameter $\beta$, which controls the balance between exploitation and exploration in UCB. A larger $\beta$ favors selecting samples with higher uncertainty, while a smaller $\beta$ prioritizes those with higher predicted relevance. The optimal value of $\beta$ varies across datasets, highlighting the importance of dataset-specific tuning. In cases where $\beta = 0$ yields the best performance, the GP's posterior mean may already provide sufficiently accurate predictions, making pure exploitation effective.

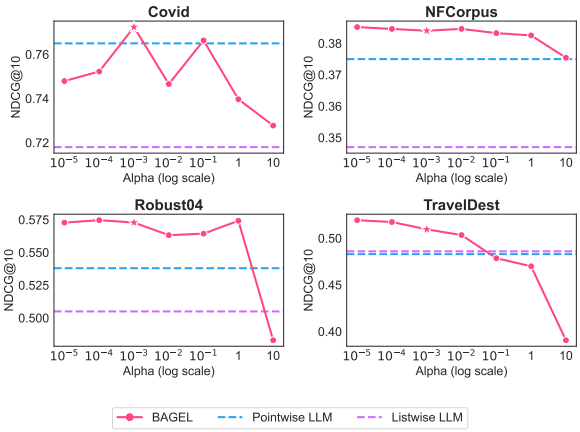


Figure 7: Performance by different values of alpha. The star represent performance with default parameter ($\alpha = 1e-3$)

**Alpha** Figure 7 shows that the optimal observation–noise parameter $\alpha$ differs by dataset, reflecting variations in LLM score noise. While larger $\alpha$ can help in noisier datasets, overly large values over-smooth the relevance function and hurt performance, making per-dataset tuning essential.