Beyond Replacement: Cooperative Roles of GP and LLM in Bayesian Optimization

Chi-Chang Lee

Abstract

Bayesian Optimization (BO) is a powerful framework for optimizing expensive black-box functions. Recent advancements in Large Language Models (LLMs) have shown potential in enhancing BO through contextual initialization and surrogate modeling. However, challenges such as reliance on domain-specific knowledge and sensitivity to ambiguous prompts persist. To address these limitations, we propose BO-SCULPT, a collaborative framework that integrates Gaussian Process (GP)-based modeling with LLMs. Leveraging GP-based predictions as guiding instructions, BO-SCULPT employs a standardized GP-guided prompting strategy to minimize dependency on semantic instructions, enabling robust optimization without additional task-specific information. Experimental results on standard benchmarks and controlled case studies demonstrate that BO-SCULPT outperforms traditional GP-based BO and achieves competitive performance with LLM-based methods relying on semantic prompts. These findings highlight its scalability, reliability, and potential for broader optimization applications. Code is available at https://github.com/ChangLee0903/BO-SCULPT.

1 Introduction

Bayesian Optimization (BO) is a widely used method for efficiently optimizing expensive black-box functions in fields like robotics [1], experimental design, drug discovery, and hyperparameter tuning. By constructing surrogate models to approximate the objective function and iteratively proposing promising candidates using acquisition functions, BO minimizes the need for costly evaluations. However, BO's efficiency heavily depends on the quality of the surrogate model and the chosen prior distributions, which, if misaligned with the true data distribution, can lead to suboptimal performance.

Recent advancements in Large Language Models (LLMs), pre-trained on vast Internet-scale datasets, have demonstrated exceptional generalization from sparse data, making them promising for BO tasks. Specifically, LLAMBO [2] explored LLMs for contextual initialization (warmstarting) and surrogate modeling in BO, achieving initial success in hyperparameter tuning without predefined priors. Despite its promise, LLAMBO encounters two key challenges:

1. **Dependency on Extensive Domain-Specific Knowledge**: *The current framework requires detailed task-specific information, such as feature attributes and task descriptions, as prompt contexts.* For instance, in hyperparameter optimization, prompts need to include training dataset details and model setup specifications to encode prior knowledge for exploration bonuses. This reliance restricts scalability to broader domains, especially in high-dimensional scenarios or complex tasks that are difficult to describe concisely in natural language.
2. **Language Ambiguity and Lack of Robustness**: Unlike conventional BO methods such as Gaussian Process (GP)-based approaches [3], in-context learning (ICL) [4, 5] heavily depends on prompt content, making it difficult to quantify the influence of prompts on the agent's behavior. *Natural language instructions are prone to ambiguity, resulting in inconsistent effectiveness.* Moreover, LLMs are easily misled by initial samples, often producing trivial or random solutions.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

While LLAMBO performs well on standard benchmarks, its effectiveness lacks theoretical support without prior knowledge, undermining robustness and convergence reliability.

These challenges highlight the need for further refinement to enhance the scalability and robustness of LLM-based BO frameworks. Instead of heavily relying on domain-specific knowledge and empirical results, a simple GP model can act as an "automatic instructor," concisely delivering task information by predicting the mean and variance of given samples. This forms a unified prompting strategy applicable across diverse tasks without language ambiguity, significantly reducing the effort required for prompt design. By simplifying the original formulation, this approach shifts the focus to designing a collaborative learning scheme between the LLM and the instructor BO model. Intuitively, the instructor BO model serves as both a "planner" and a "regularizer," converting task samples into a "standardized" instruction format. This eliminates the dependency on domain-specific prompts while effectively addressing uncertainties, paving the way for a more generalizable and robust framework.

These challenges highlight the need for further refinement to enhance the scalability and robustness of LLM-based BO frameworks. Instead of heavily relying on domain-specific knowledge, a simple GP model can act as an "automatic instructor," concisely delivering task information by predicting the mean and variance of given samples. This forms a unified prompting strategy applicable across diverse tasks without language ambiguity, significantly reducing the effort required for prompt design. By simplifying the original formulation, this approach shifts the focus to designing a collaborative learning scheme between the LLM and the instructor BO model. Intuitively, the instructor BO model serves as both a "regularizer" and a "planner," converting task samples into a "standardized" instruction format.

In this paper, we propose a Bayesian Optimization framework with Scalable Collaborative Unification of LLM and GP Techniques (**BO-SCULPT**). The GP model serves as both a sampling filter and a prompting provider, supporting ICL within the LLM. By leveraging a constrained optimization formulation, the GP model acts as an instructor to guide and enforce robust ICL optimization, reducing reliance on domain-specific prompts. This integration ensures convergence, simplifies prompt design, and enhances performance consistency by harnessing the complementary strengths of GP and LLM-based BO.

We evaluated our method on a subset of the Bayesmark benchmark and conducted ablation studies to assess its robustness in addressing local maxima and sensitivity to initialization. Results show that our framework outperforms standalone GP and LLAMBO models without semantic information, while achieving comparable performance to LLAMBO with semantic prompts. These findings demonstrate its potential to streamline prompt design while maintaining robust performance across diverse tasks.

2 Preliminaries

Bayesian Optimization (BO) is a sequential strategy for globally optimizing black-box objective functions without assuming specific functional forms. BO iteratively constructs a surrogate model to approximate the objective function and proposes candidate points for evaluation. In this work, BO is framed as a one-step decision-making process, treating the BO model as a policy π that generates h, enabling explicit comparisons between algorithms. The goal is to identify a policy π such that $h \sim \pi$ maximizes f:

$$\pi^* = \arg\max_{\pi} f(\pi),$$

where f is a costly black-box function with inaccessible gradients. BO relies on two key components to approximate f:

- A **surrogate model**, $p_t^\pi(s|h; \mathcal{D}_t)$, which estimates $f(h)$ and its uncertainty.
- A **candidate sampler**, $p_t^\pi(h|\mathcal{D}_t)$, which proposes new inputs by balancing exploration and exploitation.

Together, these components define the policy π, enabling iterative updates to generate promising candidates.

Gaussian Process Regression (GP) [3]: GP regression is a popular choice for BO due to its ability to model both the objective function and associated uncertainties. At each iteration t, the GP models

$f(h)$ as a Gaussian distribution parameterized by a mean $\mu_t(h)$ and a standard deviation $\sigma_t(h)$:

$$p_t^{\text{GP}}(s|h;\mathcal{D}_t) = \mathcal{N}(\mu_t(h), \sigma_t^2(h)).$$

BO leverages acquisition functions, such as Expected Improvement (q-EI) [6, 7], to estimate the potential for improvement. Candidate points $\{\tilde{h}_i\}_{i=1}^q$ are sampled from $p_t^{\text{EI}}(h|\mathcal{D}_t)$, and the best candidate is evaluated to update the model.

LLAMBO [2]: LLAMBO integrates LLMs into BO by leveraging their contextual understanding and prior knowledge. Unlike traditional BO methods relying on predefined priors, LLAMBO dynamically incorporates optimization history as few-shot examples through in-context learning (ICL). LLAMBO enhances two core components:

- **Candidate Sampling:** LLAMBO generates K candidate points $\{\tilde{h}_k\}_{k=1}^K$ conditionally sampled based on a target objective value s': $\tilde{h}_k \sim p_t^{\text{LLM}}(h|s';\mathcal{D}_t)$.
- **Surrogate Modeling:** The surrogate model $p_t^{\text{LLM}}(s|h;\mathcal{D}_t)$ approximates f using $\mathcal{D}_t$. LLAMBO introduces two approaches: a *discriminative* method for regression with uncertainty estimates and a *generative* method scoring predictions via binary classification.

A key limitation of LLAMBO lies in its reliance on task-specific instructions (*Task Instructions*), which restricts scalability and adaptability across diverse problem domains. These instructions demand substantial domain knowledge and meticulous prompt engineering, making the approach impractical for tasks with ambiguous or poorly defined contexts. To overcome this limitation, we aim to propose a simplified and task-agnostic framework that streamlines prompt design while enhancing usability and generalizability across a wider range of optimization tasks.

3 Method: GP-BO as an Automatic Instructor and Regularizer

3.1 Key Insight: Leveraging ICL with Constraints

Natural language instructions can be ambiguous, leading to inconsistent optimization performance. Additionally, crafting task-specific prompts often requires significant domain knowledge and effort. To address these limitations while utilizing the contextual understanding capabilities of LLMs, we propose incorporating a lightweight GP-BO model as a baseline constraint and optimization guide. This approach formalizes the optimization process as:

$$\pi_{t+1}^{\text{LLM}} \leftarrow \arg\max_{\pi^{\text{LLM}}} f(\pi^{\text{LLM}}) \quad \text{subject to} \quad f(\pi^{\text{LLM}}) \geq f(\pi_t^{\text{GP}}), \tag{1}$$

where the GP-BO model serves two key roles:

- **Stabilizing Optimization:** Under theoretical assumptions, the GP model provides a error bound, ensuring convergence to at least a local maximum [8, 7, 9, 10].
- **Mitigating Ambiguity:** GP predictions can provide unified but informative guidance to reduce reliance on ambiguous language instructions, guiding LLM decisions more concisely and effectively.

This collaboration harnesses the robustness of GP-BO to enhance LLMs' exploratory capabilities. However, implementing this optimization is non-trivial due to the dynamic interplay between π^{GP} and π^{LLM}, both of which evolve iteratively. Detailed solutions to these challenges are provided in Sections 3.2.

3.2 Algorithm: Bayesian Optimization with Scalable Collaborative Unification of LLM and GP Techniques (BO-SCULPT)

The proposed algorithm addresses the constrained optimization problem in Eq. 1, combining GP-BO and LLMs into a unified framework:

1. GP-Guided Prompting: The GP model provides concise statistical summaries to guide LLMs in candidate selection. At iteration t, for a given sample h, the GP model estimates:

- The mean $\mu_t(h)$ and standard deviation $\sigma_t(h)$, which define the predicted range of the objective function $[\mu_t(h) - \sigma_t(h), \mu_t(h) + \sigma_t(h)]$.
- Statistical trends from observations, represented as $\mathcal{O}_t^{\text{GP}} = \{(h, s, \mu_t(h), \sigma_t(h))\}$, summarizing $f(h)$ across the dataset $\mathcal{D}_t$.

These summaries are translated into concise task instruction prompts $\mathcal{I}_t^{\text{GP}}$, enabling the LLM to balance exploration and exploitation effectively when generating candidate solutions. For the complete content of $\mathcal{I}_t^{\text{GP}}$, please refer to Appendix A.

2. Threshold Filtering: To ensure robust performance, this strategy imposes a hard constraint by retaining only LLM-generated candidates h satisfying:

$$\mu_t(h) + \sigma_t(h) \geq \max_{h \in \mathcal{H}} \mu_t(h).$$

This filtering ensures adherence to the GP-BO model's theoretical convergence guarantees while mitigating the risk of suboptimal solutions [8, 7, 9, 10]. For detailed derivations of the convergence analysis, please refer to Appendix B.

The algorithm iteratively integrates GP-guided insights and LLM's generative capabilities to propose and evaluate candidates, as detailed below.

Algorithm 1 Bayesian Optimization with Scalable Collaborative Unification of LLM and GP Techniques (BO-SCULPT)

Require: Objective function $f(h)$, initial dataset $\mathcal{D}_0 = \{(h_i, f(h_i))\}_{i=1}^n$, GP-BO model π^{GP}, LLM agent π^{LLM}, number of iterations T
Ensure: Optimized solution h^*
Initialize: Fit GP-BO model π^{GP} with $\mathcal{D}_0$
for $t = 1$ to T **do**
  GP-Guided Prompting:
    Use π^{GP} to compute:
      (a) Predictions $\mathcal{H}_t^{\text{GP}} = \{(h, \mu_t(h), \sigma_t(h))\}_h$ for query samples h
      (b) Observed trends $\mathcal{O}_t^{\text{GP}} = \{(h, s, \mu_t(h), \sigma_t(h)) \mid (h, s) \in \mathcal{D}_t\}$
      (c) Prompt instructions $\mathcal{I}_t^{\text{GP}}$ summarizing $\mathcal{H}_t^{\text{GP}}$ and $\mathcal{O}_t^{\text{GP}}$
  LLM Candidate Generation:
    Generate candidates $\tilde{\mathcal{H}}_t^{\text{LLM}}$ using:

$$\tilde{\mathcal{H}}_t^{\text{LLM}} \leftarrow \tilde{h}_k \sim p_t^{\text{LLM}}(h|s'; \mathcal{D}_t, \mathcal{I}_t^{\text{GP}})$$

  Threshold Filtering:
    Filter candidates:

$$\mathcal{H}_t^{\text{LLM}} = \{h \in \tilde{\mathcal{H}}_t^{\text{LLM}} \mid \mu_t(h) + \sigma_t(h) \geq \max_{h \in \mathcal{H}} \mu_t(h)\}$$

  Sample Evaluation:
    Merge candidates $\mathcal{H}_t = \mathcal{H}_t^{\text{GP}} \cup \mathcal{H}_t^{\text{LLM}}$
    Select h' and evaluate $f(h')$:

$$\mathcal{D}_{t+1} \leftarrow \mathcal{D}_t \cup \{(h', f(h'))\}$$

  Model Updates:
    Update π^{GP} and π^{LLM} using $\mathcal{D}_{t+1}$
end for
Return: $h^* = \arg\max_{h \in \mathcal{D}_T} f(h)$

4 Experiments

We evaluate the effectiveness of BO-SCULPT by leveraging the predictions of the GP-BO model as instructions, without relying on domain-specific guidance. The evaluation consists of two parts: **benchmark results** and **case studies**.

1. **Benchmark Results:** The goal is to compare the performance of BO-SCULPT on standard datasets (where the objective function and global maximum are unknown) against LLAMBO, which is trained using semantic instructions.
2. **Case Studies:** This involves designing specific scenarios with a known objective function (where the global maximum is explicitly known). The aim is to observe the behavior of the BO model in these controlled environments to enable deeper analysis and insights.

4.1 Benchmark results

The experiments are conducted on a subset of Bayesmark using two datasets, *digits* and *wine*, and three models: *MLP_SGD*, *AdaBoost*, and *RandomForest*. Each setup is run with 5 different random seeds, where each seed includes the same initial samples to ensure consistency.

Evaluation Metric: Normalized Regret To measure performance, we adopt simple regret as the primary metric. Normalized regret focuses on the best-observed performance after T iterations of the tuning process. It evaluates how close the method gets to the global optimum f^*, and is defined as:

$$R_T = \frac{f^* - \max_{i \in \{1,\ldots,T\}} f(h_i)}{f^* - f_{\text{min}}}$$

where:

- f^*: True global optimum of the objective function.
- f_{min}: Observed minimum value of the objective function across all methods.
- $f(h_i)$: Observed value of the objective function at the i-th hyperparameter configuration h_i.
- T: Total number of iterations.

Baselines We implement **BO-SCULPT** based on LLAMBO [2] and compare it with the following baselines:

- **LLAMBO-WIS (with semantic instruction):** The original LLAMBO trained using the best setups, with task-specific semantic instructions. Prompts for this setup are detailed in their paper. We use the discriminative version of the surrogate model for comparison.
- **LLAMBO-WOS (without semantic instruction):** This baseline removes access to all task-specific information, including model types, dataset details, and configuration descriptions.
- **GP-BO:** A standard GP-BO implementation using the qEI acquisition function [6]. Details of this baseline are provided in the submitted codebase.

In addition, we conduct an ablation study by removing the Threshold Filtering mechanism in the collaborative GP-based instructor to evaluate whether the LLM agent can independently make decisions purely based on GP-Guided Prompting. This variant is referred to as **BO-SCULPT-WOF (without filtering)**.

To ensure no additional semantic information interferes, other LLM methods, besides the original LLMABO with full context, omit all domain-specific information. For example, in the prompt, dataset-related details are excluded, and placeholders like {model}, {task}, and {metric} are replaced with generic terms such as "a model," "a task," and "a metric." Additionally, feature names such as "max_depth," "max_features," and "min_impurity_decrease" are anonymized as "X0," "X1," and "X2." For the detailed prompt examples for surrogate and candidate sampler models, please refer to Appendix A.

Results The results presented in Figure 1 highlight the comparative performance of various Bayesian Optimization (BO) models. Specifically, LLM-based BO models trained without domain-specific guidance (LLAMBO-WOS) generally underperform compared to those incorporating prior knowledge, such as predefined distributions (GP-BO) or semantic instructions (LLAMBO-WIS). Both BO-SCULPT and BO-SCULPT-WOF demonstrate significant improvements over GP-BO and LLAMBO-WOS. This performance advantage arises from their ability to effectively utilize GP-guided prompting, thereby overcoming limitations associated with discrepancies between real-world data and predefined GP distributions.

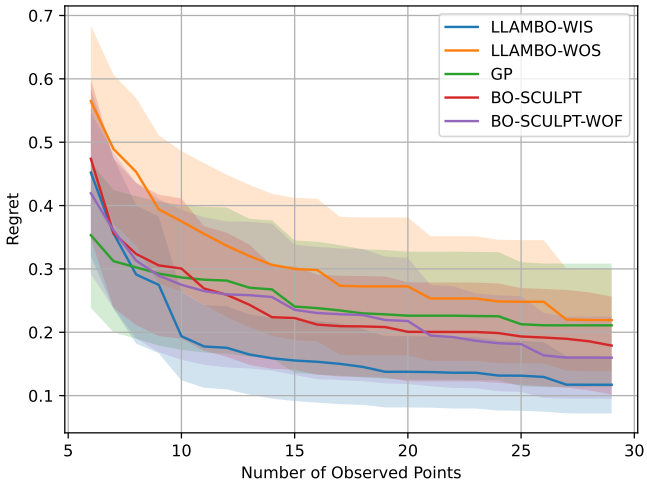

Figure 1: Performance of BO-SCULPT evaluated on the Bayesmark subset.

Consequently, BO-SCULPT and BO-SCULPT-WOF deliver robust optimization, maintaining a strong balance between exploration and exploitation during the optimization process. Interestingly, BO-SCULPT-WOF slightly outperforms BO-SCULPT when the number of observations exceeds 20. This suggests that GP-Guided Prompting is a pivotal factor in driving performance and highlights the LLMs' ability to effectively interpret and act on instructions derived from GP-based models.

Furthermore, the overlapping confidence intervals between BO-SCULPT, BO-SCULPT-WOF, and LLAMBO-WIS (approximately 50% overlap) indicate that BO-SCULPT achieves comparable performance to LLAMBO-WIS in certain scenarios. This finding underscores BO-SCULPT's scalability, its potential for further refinement, and its versatility in addressing a broad range of optimization tasks.

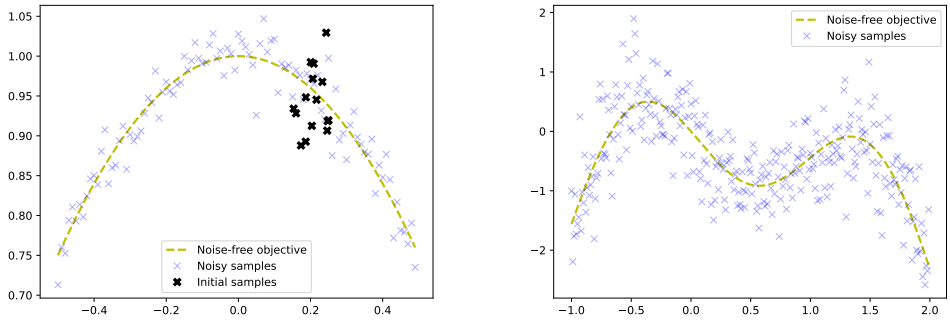

Figure 2: **(a)** Case I: one-dimensional smooth and concave function with 15 noisy initial samples. **(b)** Case II: one-dimensional smooth function with two maxima, which are at -0.36 and 1.33, respectively.

4.2 Case Studies

In this section, we utilize prior knowledge of the function's structure and the position of the global maximum to assess the performance of the models more effectively. Since these case studies focus on objective function optimization rather than hyperparameter tuning, semantic information is unavailable, and comparisons do not include LLAMBO-WIS. Specifically, we measure the minimum distance between the observed position of the maximum sample and the true global maximum. To ensure consistency and reproducibility, all experiments are repeated with 5 different random seeds.

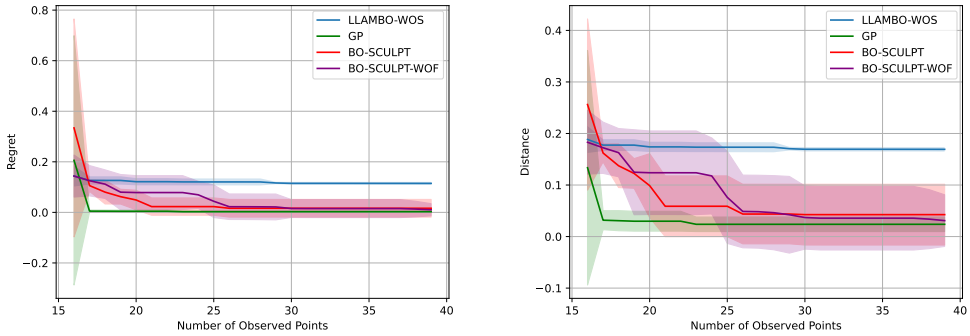

Figure 3: **(a)** Regret curves evaluated on Case I. **(b)** Distance between current observed maximum and the global maximum evaluated on Case I.

4.2.1 Case I: Can a LLM-based BO model be misled by initial samples?

Setup: To evaluate the influence of initial sample placement, we constructed a one-dimensional smooth function with 15 noisy initial samples positioned between 0.15 and 0.25, as shown in Figure 2(a). Noise of a defined scale was introduced at each iteration. Compared to other experiments, the use of 15 initial samples represents a significantly larger starting set. This setup allows us to investigate the impact of initial sample quantity on the behavior of LLM-based BO models. The global maximum, located at 0, should be readily attainable if the BO model possesses sufficient exploration and denoising capabilities.

Results: Figure 3 shows that the performance in terms of regret curve or the distance to the global maximum is significantly worse for LLAMBO-WOS compared to GP and BO-SCULPT. It is noteworthy that the regret and distance curves for LLAMBO-WOS remain nearly horizontal after the initial phase, indicating that it quickly stops exploring, even though the global maximum is close. This suggests that LLAMBO-WOS is heavily influenced by the initial samples and lacks the ability to escape local regions effectively. In contrast, GP successfully converges to the global maximum within a few iterations, demonstrating its strong exploration and optimization capabilities. BO-SCULPT and BO-SCULPT-WOF exhibit a similar trend, leveraging the guidance from GP-BO to ensure robust optimization and achieve competitive results. The collaboration between BO-SCULPT and the GP-BO instructor likely contributes to its resilience in overcoming the challenges posed by noisy initialization. Notably, BO-SCULPT-WOF's performance suggests that LLMs are capable of capturing and independently executing exploration strategies derived from GP-based prompting, even without stringent filtering rules.

Moreover, the shaded confidence intervals in Figure 3 reveal critical differences in model behavior. GP and BO-SCULPT display broader intervals during the early stages, reflecting variability in their exploratory phases. However, their intervals narrow significantly as the models stabilize at lower regret values, indicating reliable convergence to the global maximum. In contrast, LLAMBO-WOS maintains a narrower confidence interval throughout, but its consistently suboptimal performance highlights its limited exploration and adaptability. These observations underscore the importance of robust exploration mechanisms and effective denoising capabilities, particularly in scenarios involving large initial sample sets. The superior performance of GP and BO-SCULPT emphasizes their ability to overcome the limitations imposed by noisy initial conditions, achieving reliable convergence to the global maximum.

4.2.2 Case II: Can a LLM-based BO model converge to the global maximum instead of a local maximum?

Setup: To validate the ability of different models to distinguish between local and global maxima, we constructed a one-dimensional smooth function with two maxima, as illustrated in Figure 2(b), and introduced noise of a defined scale at each iteration. The global maximum is located at -0.36, while a local maximum is present at 1.33.

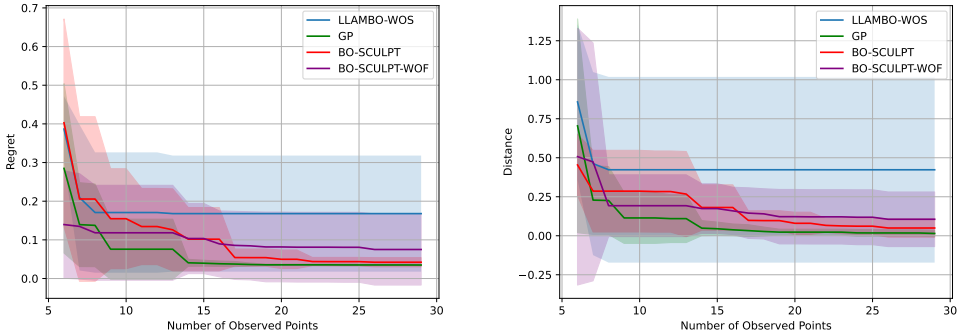

Figure 4: **(a)** Regret curves evaluated on Case II. **(b)** Distance between current observed maximum and the global maximum evaluated on Case II.

Results: Figures 4(a) and 4(b) highlight the performance differences among the models. The distance curves reveal that LLAMBO-WOS exhibits significant variability, failing to converge to the global maximum in some instances. This inconsistency underscores its limited exploration and optimization capacity. In contrast, the GP-BO model demonstrates robust performance, consistently converging to the global maximum across all seeds. BO-SCULPT mirrors the trends of GP-BO, showcasing comparable exploration and denoising capabilities. These results further affirm that incorporating GP-guided prompting enhances the model's ability to navigate complex optimization landscapes. Overall, the consistent performance of GP-BO and BO-SCULPT highlights their reliability in scenarios with multiple optima, while LLAMBO-WOS's limitations indicate room for improvement in its exploration mechanisms.

5 Conclusion & Discussion

In this paper, we introduced BO-SCULPT, a scalable and robust framework that combines GP-based Bayesian Optimization (BO) with the generative capabilities of LLMs. Our approach addresses key challenges faced by existing LLM-based BO frameworks, such as reliance on domain-specific prompts and susceptibility to noisy initializations. By incorporating GP-guided prompting and threshold filtering, BO-SCULPT delivers consistent performance across diverse optimization tasks without requiring extensive task-specific information. Through comprehensive experiments, we demonstrated that BO-SCULPT effectively handles noisy initial samples and multiple local optima, showcasing its robustness and adaptability. Its comparable performance to semantic instruction-driven LLMs underscores the framework's ability to bridge the gap between conventional GP-based methods and modern LLM-enhanced BO strategies.

Looking forward, LLMs present transformative opportunities to extend BO into domains characterized by high-dimensional and complex data, such as multi-modal media. LLMs possess the potential to capture, understand, and summarize these intricate features, transforming raw data into informative low-dimensional representations that conventional BO methods can process. In turn, BO can provide structured guidance and feedback, enabling iterative refinement and exploration. This collaborative synergy could unlock optimization capabilities for tasks previously infeasible with traditional BO strategies. If we expand our perspective even further, the integration of LLMs and BO could revolutionize the learning paradigm. By generalizing the optimization of models through BO, the reliance on supervised learning could diminish, paving the way for more autonomous, adaptive, and efficient learning systems. The vision of optimizing raw data directly without intermediate training steps could redefine the landscape of machine learning and its applications.

6 Related Works

Bayesian Optimization. Bayesian Optimization (BO) has been widely utilized for the optimization of expensive black-box functions. Traditional methods, such as GP-based BO, leverage the flexibility

and analytical tractability of GPs to model surrogate functions [11]. Recent works have extended these approaches with deep kernel GPs [12] and neural network-based surrogates to enhance scalability in high-dimensional spaces [13]. Acquisition functions such as Expected Improvement (EI) and Upper Confidence Bound (UCB) have also been explored for efficient candidate selection [7]. Additionally, Tree-structured Parzen Estimator (TPE) methods offer generative surrogate models for more diverse applications [14]. While these approaches are effective, they rely heavily on predefined priors, which may not align with the true data distribution, limiting their adaptability to diverse tasks.

Transfer Learning for BO. Recent advances in transfer learning have aimed to improve BO performance across related domains. Multitask GPs, for instance, leverage shared structures between tasks to transfer optimization knowledge [15]. Meta-learning frameworks such as neural processes and Transformers have also been employed to generalize BO across heterogeneous tasks [16, 17]. Despite their success, these methods often require significant task-specific information or inductive biases, limiting their applicability to broader domains. In contrast, our proposed framework avoids task-specific assumptions, utilizing GP-guided prompting and the generalization capabilities of LLMs to achieve scalability and robustness.

LLMs and Optimization. The integration of LLMs into optimization tasks has garnered recent attention. LLMs have been employed for prompt-based optimization [2], genetic algorithm operations [18], and surrogate modeling in domains like molecular optimization [19]. However, many existing methods rely on semantic prompts that require extensive domain knowledge and are prone to ambiguities. Unlike these methods, BO-SCULPT minimizes the dependency on semantic prompts by introducing GP-guided instruction, ensuring robust performance even in the absence of task-specific details. Additionally, our approach demonstrates the potential for LLMs to bridge gaps in high-dimensional, noisy, or sparse-data settings by complementing traditional BO methods.

Collaborative Optimization Frameworks. The synergy between different optimization paradigms, such as GP and LLM-based methods, is an emerging area of research. Collaborative frameworks that combine complementary strengths have been explored in areas like hybrid optimization [20] and reinforcement learning with Bayesian inference [21]. Our work builds on this by systematically unifying GP-based BO and LLMs into a scalable framework, providing robust performance across diverse tasks while simplifying prompt design.

Constrained Bayesian Optimization: Constrained Bayesian Optimization extends traditional BO frameworks to handle problems where the feasible region is defined by explicit or implicit constraints. Methods like constrained GP-based BO [22] leverage surrogate models for both the objective function and the constraints, optimizing acquisition functions within the feasible region. Recent advancements explore integrating high-dimensional data and non-linear constraints into the optimization process. For instance, methods incorporating deep learning into BO enable the handling of complex constraint representations [23]. These approaches are particularly relevant in real-world scenarios, such as hyperparameter tuning [3] and engineering design [24], where constraint satisfaction is critical. The collaborative framework introduced in this work can naturally incorporate constraints through GP-guided prompting, enabling a more versatile optimization process.

References

[1] Daniel J. Lizotte, Tao Wang, Michael Bowling, and Dale Schuurmans. Automatic gait optimization with gaussian process regression. In *International Joint Conference on Artificial Intelligence*, 2007. URL https://api.semanticscholar.org/CorpusID:10441616.

[2] Tennison Liu, Nicolás Astorga, Nabeel Seedat, and Mihaela van der Schaar. Large language models to enhance bayesian optimization. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=OOxotBmGol.

[3] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in Neural Information Processing Systems*, volume 25, pages 2951–2959, 2012.

[4] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models

are few-shot learners. *Advances in Neural Information Processing Systems (NeurIPS)*, 33: 1877–1901, 2020.

[5] Sang Michael Xie, Aditi Raghunathan, Percy Liang, and Tengyu Ma. An explanation of in-context learning as implicit bayesian inference. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=RdJVFCHjUMI.

[6] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global Optimization*, 13(4):455–492, 1998.

[7] Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. In *arXiv preprint arXiv:1012.2599*, 2010.

[8] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias W Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2010.

[9] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104 (1):148–175, 2016.

[10] Kaiwen Wu, Kyurae Kim, Roman Garnett, and Jacob R. Gardner. The behavior and convergence of local bayesian optimization. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. URL https://openreview.net/forum?id=9KtX12YmA7.

[11] Carl Edward Rasmussen and Christopher KI Williams. *Gaussian Processes for Machine Learning*. MIT Press, 2006.

[12] Andrew G Wilson, Zhiting Hu, Ruslan Salakhutdinov, and Eric P Xing. Deep kernel learning. In *Artificial Intelligence and Statistics (AISTATS)*, pages 370–378, 2016.

[13] Jasper Snoek, Olivier Rippel, Kevin Swersky, Ryan Kiros, Nadathur Satish, Narayanan Sundaram, Mostofa Ali Patwary, Mr Prabhat, and Ryan P Adams. Scalable bayesian optimization using deep neural networks. In *International Conference on Machine Learning (ICML)*, pages 2171–2180, 2015.

[14] James Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems (NIPS)*, volume 24, 2011.

[15] Kevin Swersky, Jasper Snoek, and Ryan P Adams. Multi-task bayesian optimization. In *Advances in Neural Information Processing Systems (NIPS)*, volume 26, 2013.

[16] Juho Lee, Minseop Lee, Jungtaek Lee, Changhoon Kim, Sangho Min, Saehoon Choi, Jinwoo Shin, and Kyunghyun Song. Meta-learning with differentiable convex optimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10657–10665, 2019.

[17] Matthias Feurer, Aaron Klein, Katharina Eggensperger, Jost Tobias Springenberg, Manuel Blum, and Frank Hutter. Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 2962–2970, 2015.

[18] Xinghao Chen and Julia Ling. Using machine learning to enhance bayesian optimization for engineering design. *Structural and Multidisciplinary Optimization*, 63:331–352, 2021.

[19] Aditi Gupta, Biswajit Pathak, and Masahiro Ehara. Artificial intelligence–driven molecular design: Tools and strategies. *Nature Reviews Chemistry*, 6:231–250, 2022.

[20] Simon Schweighofer, Maximilian Tesch, Dieter Stoll, Martin Falk, and Christof Weinhardt. Hybrid bayesian optimization with a generative model of uncertainty. In *Proceedings of the International Conference on Machine Learning (ICML)*, pages 10234–10248, 2022.

[21] Sayak Ray Chowdhury and Aditya Gopalan. Bayesian optimization with unknown constraints. *Journal of Machine Learning Research*, 21(114):1–60, 2019.

[22] Jacob Gardner, Matt Kusner, Zhixiang Xu, Kilian Weinberger, and John P Cunningham. Bayesian optimization with inequality constraints. In *Advances in Neural Information Processing Systems*, volume 27, pages 934–942, 2014.

[23] David Eriksson, Michael Pearce, Jacob R Gardner, Ryan D Turner, and Matthias Poloczek. Scalable constrained bayesian optimization. *Proceedings of the International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2021.

[24] Michael A Gelbart, Jasper Snoek, and Ryan P Adams. Bayesian optimization with unknown constraints. In *Proceedings of the Thirtieth Conference on Uncertainty in Artificial Intelligence*, pages 250–259, 2014.

A Complete Prompt Details

Each performance (S) is variable through sampling and estimated in terms of mean (S_avg), minimum (S_min), and maximum (S_max). Here are some collected samples:
- X0: {configuration number}, X1: {configuration number}, X2: {configuration number}..., S: {observed performance}, S_avg: $\{\mu(h)\}$, S_min: $\{\mu(h) - \sigma(h)\}$, S_max: $\{\mu(h) + \sigma(h)\}$
- X0: {configuration number}, X1: {configuration number}, X2: {configuration number}..., S: {observed performance}, S_avg: $\{\mu(h)\}$, S_min: $\{\mu(h) - \sigma(h)\}$, S_max: $\{\mu(h) + \sigma(h)\}$
...

Here are some predictions of uncollected options without observed performances:
- X0: {configuration number}, X1: {configuration number}, X2: {configuration number}..., S_avg: $\{\mu(h)\}$, S_min: $\{\mu(h) - \sigma(h)\}$, S_max: $\{\mu(h) + \sigma(h)\}$
- X0: {configuration number}, X1: {configuration number}, X2: {configuration number}..., S_avg: $\{\mu(h)\}$, S_min: $\{\mu(h) - \sigma(h)\}$, S_max: $\{\mu(h) + \sigma(h)\}$
...

Figure 5: Prompt from GP-BO model's instruction $\mathcal{I}_t^{\text{GP}}$.

The following are hyperparameter configurations for a model and the corresponding performance measured in a metric. Your response should only contain the predicted a metric in the format ## performance ##.
Hyperparameter configuration: X0 is 100, X1 is 2.6584 Performance: ## 0.937192 ##
Hyperparameter configuration: X0 is 31, X1 is 0.0096 Performance: ## 0.929557 ##
Hyperparameter configuration: X0 is 45, X1 is 0.2232 Performance: ## 0.922167 ##
Hyperparameter configuration: X0 is 100, X1 is 4.6003 Performance: ## 0.929557 ##
Hyperparameter configuration: X0 is 94, X1 is 1.7043 Performance: ## 0.861330 ##

Figure 6: Prompt example for discriminative surrogate model.

The following are examples of performance of a model measured in a metric and the corresponding model hyperparameter configurations. The allowable ranges for the hyperparameters are:
- X0: [10, 100] (int)
- X1: [0.0001, 10.0000] (float, precise to 4 decimals)
Recommend a configuration that can achieve the target performance of 0.975000. Do not recommend values at the minimum or maximum of allowable range, do not recommend rounded values. Recommend values with highest possible precision, as requested by the allowed ranges. Your response must only contain the predicted configuration, in the format ## configuration ##.
Performance: 0.922906 Hyperparameter configuration: ## X0: 100, X1: 2.5157 ##
Performance: 0.626847 Hyperparameter configuration: ## X0: 33, X1: 0.0001 ##
Performance: 0.936700 Hyperparameter configuration: ## X0: 100, X1: 2.2150 ##
Performance: 0.886946 Hyperparameter configuration: ## X0: 100, X1: 5.0000 ##
Performance: 0.975000 Hyperparameter configuration:

Figure 7: Prompt example for candidate sampling.

B Theoretical Analysis: Convergence Guarantee

1. Decomposition of Regret: The cumulative regret R_T is defined as:

$$R_T = \sum_{t=1}^{T} \Delta_t = \sum_{t=1}^{T} \left(f(h^*) - f(h_t) \right),$$

where h^* is the global maximum, h_t is the selected point at iteration t, and $\Delta_t = f(h^*) - f(h_t)$ is the instantaneous regret.

2. GP Approximation Error: From Srinivas et al. (2010) [8], the posterior error in predicting $f(h)$ is bounded with high probability:

$$|f(h) - \mu_t(h)| \leq \beta_t^{\frac{1}{2}} \sigma_t(h), \quad \forall h \in \mathcal{H},$$

where:

- $\mu_t(h)$: Posterior mean at iteration t,
- $\sigma_t(h)$: Posterior standard deviation,
- β_t: Confidence parameter, typically $\beta_t = O(\log t)$.

3. Substituting Bounds into Δ_t: The instantaneous regret Δ_t is expressed as:

$$\Delta_t = f(h^*) - f(h_t).$$

Adding and subtracting $\mu_t(h^*)$ and $\mu_t(h_t)$, and applying the triangle inequality:

$$\Delta_t \leq |f(h^*) - \mu_t(h^*)| + |\mu_t(h^*) - \mu_t(h_t)| + |\mu_t(h_t) - f(h_t)|.$$

Using the GP error bound:

$$|f(h) - \mu_t(h)| \leq \beta_t^{\frac{1}{2}} \sigma_t(h),$$

we have:

$$\Delta_t \leq \beta_t^{\frac{1}{2}} \sigma_t(h^*) + |\mu_t(h^*) - \mu_t(h_t)| + \beta_t^{\frac{1}{2}} \sigma_t(h_t).$$

4. Incorporating Threshold Filtering: The Threshold Filtering strategy 3.2 ensures that the selected point h_t satisfies:

$$\mu_t(h_t) + \sigma_t(h_t) \geq \max_{h \in \mathcal{H}} \mu_t(h) \geq \mu_t(h^*),$$

where h^* is the point that maximizes the true objective function $f(h)$, i.e., $h^* = \arg\max_{h \in \mathcal{H}} f(h)$. Importantly, h^* may not maximize the posterior mean $\mu_t(h)$ due to the approximation nature of the Gaussian Process model.

The term $\max_{h \in \mathcal{H}} \mu_t(h)$ represents the maximum of the posterior mean, which can be computed in closed form from the GP model. Threshold Filtering ensures that the selected point h_t satisfies:

$$\mu_t(h_t) \geq \mu_t(h^*) - \sigma_t(h_t),$$

where $\sigma_t(h_t)$ accounts for the uncertainty at h_t. Substituting this relationship into the regret decomposition:

$$|\mu_t(h^*) - \mu_t(h_t)| \leq \sigma_t(h_t).$$

The instantaneous regret Δ_t can then be bounded as:

$$\Delta_t \leq \beta_t^{\frac{1}{2}} \sigma_t(h^*) + \sigma_t(h_t) + \beta_t^{\frac{1}{2}} \sigma_t(h_t).$$

Simplifying terms involving $\sigma_t(h_t)$:

$$\Delta_t \leq \beta_t^{\frac{1}{2}} \sigma_t(h^*) + (1 + \beta_t^{\frac{1}{2}}) \sigma_t(h_t).$$

5. Cumulative Regret: The cumulative regret R_T over T iterations is:

$$R_T = \sum_{t=1}^{T} \Delta_t.$$

Substituting the bound for Δ_t:

$$R_T \leq \sum_{t=1}^{T} \left[\beta_t^{\frac{1}{2}} \sigma_t(h^*) + (1 + \beta_t^{\frac{1}{2}}) \sigma_t(h_t) \right].$$

Using results from Gaussian Process optimization:

- The uncertainty at the global maximizer h^* satisfies:

$$\sum_{t=1}^{T} \sigma_t(h^*) \leq \sqrt{T\gamma_T},$$

  where γ_T is the maximum information gain over T iterations.
- Similarly, for the selected points h_t:

$$\sum_{t=1}^{T} \sigma_t(h_t) \leq \sqrt{T\gamma_T}.$$

- The confidence parameter β_t grows as $O(\log t)$, so $\beta_t^{\frac{1}{2}} = O(\sqrt{\log t})$.

Substituting these bounds into the cumulative regret:

$$R_T \leq \sqrt{T\gamma_T} \cdot O(\sqrt{\log T}) + \sqrt{T\gamma_T} \cdot (1 + O(\sqrt{\log T})).$$

6. Final Bound: Combining terms and simplifying:

$$R_T = O(\sqrt{T\gamma_T \log T}),$$

where $\gamma_T = O(\log T)$ for common kernels like the squared exponential or Matérn.

This result guarantees sublinear cumulative regret, ensuring the convergence of the optimization process as $T \to \infty$.