AutoLead: An LLM-Guided Bayesian Optimization Framework for Multi-Objective Lead Optimization

Yiming Zhang[1,2], **Jun Jin Choong**[1], **and Keisuke Ozawa**[2,✉]

[1]Department of Computational Biology and Medical Sciences, Graduate School of Frontier Sciences, The University of Tokyo, Japan
[2]SB Intuitions, Japan

The process of lead optimization in drug discovery is a complex, multi-objective challenge that remains a major bottleneck in the development of new therapeutics. Traditional approaches often struggle to efficiently explore the vast chemical space while simultaneously optimizing multiple, and sometimes conflicting, molecular properties. In this work, we present AutoLead, a novel framework that integrates Large Language Models (LLMs) with multi-objective Bayesian optimization to tackle this challenge. By leveraging the chemical reasoning capabilities of LLMs, AutoLead effectively guides the search for novel drug-like molecules that satisfy multiple objectives. We evaluate our approach on two molecule optimization tasks, achieving state-of-the-art results. Furthermore, we introduce a new benchmark dataset designed around a more realistic lead optimization scenario, where the task is to modify compounds that violate Lipinski's Rule of Five to simultaneously meet all criteria and improve their QED score. Through extensive experiments and a detailed case study, we demonstrate the potential of combining LLMs with black-box optimization techniques for more efficient and practical drug discovery.

Drug discovery | Lead optimization | Bioinformatics | Large Language Models | Bayesian optimization

Correspondence: ***keisuke.ozawa@sbintuitions.co.jp***

Introduction

Lead optimization is a critical stage in drug discovery, where chemists iteratively modify hit compounds to improve their pharmacokinetic properties while ensuring efficacy and safety (1, 2). Unlike early-stage virtual screening, lead optimization involves complex, multi-objective trade-offs—such as simultaneously improving solubility, permeability, and drug-likeness—under strict medicinal chemistry constraints (3, 4). Traditional approaches often rely on expert heuristics or rule-based systems to make localized edits (5), which typically operate in oversimplified settings and fail to scale in the face of combinatorial chemical space and tightly coupled objectives (3, 6).

Recent advancements in large language models (LLMs) have opened promising avenues for pipelining molecular optimization (6–9). This has led to two main strategies. One involves developing domain-specific LLMs that are pre-trained on vast scientific corpora to enhance their chemical reasoning (10–12). Specifically for molecular optimization, DrugAssist (13) achieved leading results by instruction-tuning an LLM on a large dataset for multi-property optimization. These works highlight the potential of LLM-guided optimization but often lean on specialized models or extensive fine-tuning.

While these approaches highlight the power of LLMs in capturing chemical reasoning and facilitating property-guided edits, they also exhibit limitations, particularly their reliance on large-scale instruction-finetuning datasets tailored for molecule editing. The other adapts powerful general-purpose models for chemical tasks. ChatDrug (14) introduced a conversational framework to iteratively edit molecules through prompt design and dialogue. However, ChatDrug relies on an external molecular database for retrieval. Moreover, the tasks and datasets used there primarily focus on single-objective optimization or loosely coupled dual-property scenarios, falling short of the complex, tightly constrained multi-objective requirements encountered in realistic lead optimization workflows.

To overcome these limitations, we introduce a new framework—**AutoLead**—that combines the strengths of LLMs with the principled decision-making capabilities of Bayesian Optimization (BO), without requiring any external retrieval database or domain-specific training data. In our approach, the LLM first proposes a diverse set of candidate molecules by leveraging its inherent knowledge of chemical structure and function. These initial proposals serve as seeds for BO, which then guides the optimization process by selectively exploring or exploiting candidates based on historical performance and uncertainty estimates modeled by Gaussian Processes (GPs) (15, 16). This integration allows AutoLead to dynamically alternate between LLM-driven exploration and GP-guided exploitation, enabling efficient navigation of the chemical space under multi-objective constraints.

In practical drug discovery, lead optimization often begins with modifying hit compounds that violate key drug-like properties, such as Lipinski's Rule of Five (17, 18), to simultaneously satisfy multiple pharmacokinetic constraints and improve drug-likeness metrics like QED. To evaluate our method in this realistic setting, we compile a new benchmark dataset derived from the HiQBind (19) dataset, explicitly designed for lead optimization where the objective is to transform molecules that fail one or more Lipinski criteria into candidates that satisfy all rules while enhancing their QED scores.

Our key contributions are summarized as follows:

- We propose AutoLead that integrates general-purpose LLMs with multi-objective BO specifically designed to generate and refine molecules under realistic pharmacokinetic constraints, without target-specific tuning or retrieval databases.
- We demonstrate that AutoLead achieves state-of-the-

art performance on three benchmarks—ChatDrug-200, DrugAssist-500, and our newly introduced LipinskiFix-1000—highlighting its broad applicability and effectiveness in complex multi-objective settings.

- We construct LipinskiFix-1000, a challenging benchmark dataset derived from HiQBind (19), designed to evaluate lead optimization methods on tasks that require repairing non-compliant molecules while improving overall drug-likeness.

Together, these contributions showcase how combining LLMs' generative flexibility with BO's principled exploration can yield a powerful and data-efficient solution to real-world lead optimization.

Related Work

Traditional and Hybrid Optimization Methods. Molecular optimization has evolved from traditional generative models—such as Recurrent Neural Networks (RNNs) (20), Variational Autoencoders (VAEs) (21), and Transformers (22) or graph-based networks (23, 24)—and BO used for searching predefined libraries (25–27). While generative models often struggle with multi-objective constraints, conventional BO is not suited for de novo structural editing. More recently, hybrid methods have emerged that combine LLMs with BO to improve performance (28–31). However, these frameworks require predefined numerical feature spaces, which limits their applicability to the molecular optimization tasks we aim to address.

LLMs in Drug Discovery and Molecule Editing. LLMs are now widely applied to chemistry tasks like molecular Q&A (32) and bioactivity prediction (33). For molecule editing, this has led to systems that adapt general-purpose LLMs conversationally (ChatDrug) and that further requires property-dependent fine-tuning (DrugAssist and $\mathrm{GeLLM^4O-C}$ (12)), and the development of domain-specific models like MolT5 (34) and BioT5 (35) for deeper understanding of chemistry. Despite their specialized ability, these approaches rely on large-scale instruction-tuning, retrieval from external databases (14), or specialized layers (36), and frequently address simplified tasks that fall short of real-world, multi-constraint lead optimization challenges.

LLMs for Population-based Molecular Design. A distinct line of work integrates LLMs into population-based evolutionary algorithms (37) for open-ended molecular generation, starting from large pools like ZINC250K (38) (e.g., MOLLEO (39) and MOLLM (40)). This paradigm, focused on discovering entirely new candidates, fundamentally differs from our approach. Our AutoLead instead performs direct, constraint-aware editing on a single starting molecule. This process more closely aligns with practical lead optimization workflows, where improving known, promising scaffolds is often paramount.

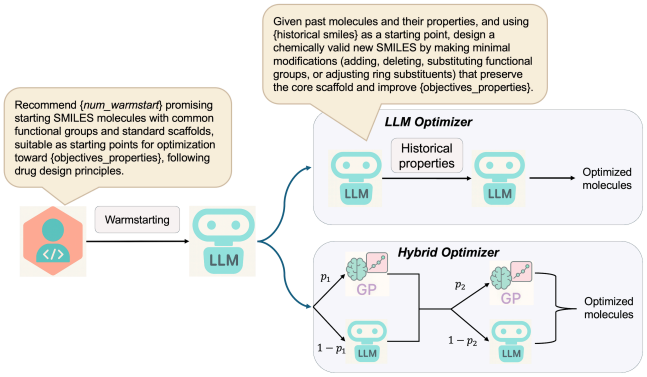

Fig. 1. Overview of the AutoLead framework. The system combines an LLM-based optimizer that leverages historical molecular properties and a hybrid optimizer integrating both LLM and GP components. Warmstarting uses the LLM to propose initial molecules with standard scaffolds. Subsequent optimization balances LLM chemical reasoning with uncertainty-aware BO to progressively refine molecules toward multi-objective drug-like criteria.

Method

A. Overview. We tackle molecular lead optimization, where given an initial molecule, the goal is to generate a sequence of improved molecules by optimizing multiple drug-relevant properties such as drug-likeness (QED), lipophilicity (logP), and Lipinski compliance, addressing the challenge of multi-objective optimization that existing methods often struggle with.

AutoLead addresses this problem via a hybrid *LLM-in-the-loop Bayesian optimization* framework (Figure 1) that couples the generative chemical reasoning capabilities of general-purpose LLMs with uncertainty-aware optimization using a GP surrogate model. We also provide a mapping strategy between SMILES strings and feature vectors to mitigate the difficulty of incorporating BO into our hybrid scheme.

B. Problem Formulation. Let $\mathcal{S}$ denote the set of valid SMILES (41) strings and let

$$f : \mathcal{S} \longrightarrow \mathbb{R}^d$$

be a property evaluator that returns a vector of target properties such as QED, logP, tPSA, number of hydrogen bond donors/acceptors and molecular weight. We aim to solve the optimization problem:

$$s^* = \arg\max_{s \in \mathcal{S}} U(f(s)), \tag{1}$$

where $U(\cdot)$ aggregates the d objectives, and is by default taken as a uniform average: $U(f(s)) = \frac{1}{d}\sum_{i=1}^{d} f_i(s)$.

C. Warmstarting. To ensure an effective start, we employ a warmstarting strategy using an LLM to generate initial candidate molecules. Given a reference molecule and desired optimization objectives, the LLM is prompted with normalized descriptors and asked to propose chemically valid, drug-like molecules that retain the core scaffold while potentially improving the target properties. This process generates an initial pool of promising candidates that are structurally similar yet diverse, enabling more efficient optimization.

D. Surrogate Objective and LLM Decoding. Since BO cannot directly optimize over discrete SMILES strings, we introduce a continuous descriptor space:

$$\phi : \mathcal{S} \longrightarrow \mathbb{R}^p,$$

where $\phi(s)$ maps a molecule s to a p-dimensional vector of normalized RDKit (42) descriptors. To ensure all values fall within the range $[0,1]$ for GP-based Bayesian Optimization, each descriptor is divided by a fixed normalization constant (shown in parentheses). Specifically, we use 10 molecular properties: molecular weight (500 Da), logP (5.0), polar surface area (200 Å^2), number of hydrogen bond donors (10), number of hydrogen bond acceptors (10), number of rotatable bonds (20), ring count (10), number of aromatic rings (5), number of saturated rings (5), and sp^3 carbon fraction (which is inherently bounded within $[0,1]$).

This 10-dimensional vector provides a pharmacologically meaningful and normalized space for optimization. It is important to note that f is a property evaluator (e.g., computing logP), whereas ϕ is used to represent molecules for optimization.

Incorporating BO requires the inverse mapping ϕ^{-1} but this is generally intractable. We then define an LLM-driven decoder $\tilde{\phi}^{-1}$ to alternate the inverse mapping. Specifically, given a point $x \in [0,1]^p$ in the descriptor space, we verbalize it as a sentence (e.g., "molecular weight: 310 Da, logP: 3.5, ...") and prompt the LLM to generate a corresponding SMILES string:

$$g(x) := U(f(\tilde{\phi}^{-1}(x))).$$

Optimization is thus approximately realized by:

$$\max_{x \in [0,1]^p} g(x),$$

where BO is performed in the continuous descriptor space and $\tilde{\phi}^{-1}$ translates optimized vectors back to molecules. The generated molecules are then evaluated by f to verify whether they satisfy the optimization goals.

E. Bayesian Surrogate Modeling. As in (29), given observations up to step $(t-1)$, we define the dataset as:

$$\mathcal{D}_{t-1} = \{(x_i, u_i)\}_{i=1}^{t-1}, \qquad \text{where } u_i = g(x_i).$$

We place a GP prior over the utility function $U(x)$ and obtain the following posterior distribution:

$$U(x) \mid \mathcal{D}_{t-1} \sim \mathcal{GP}\big(\mu_{t-1}(x), \sigma^2_{t-1}(x)\big),$$

where the predictive mean and variance are given by:

$$\mu_{t-1}(x) = k_{t-1}(x)^\top (K + \lambda^2 I)^{-1} y, \tag{2}$$

$$\sigma^2_{t-1}(x) = k(x,x) - k_{t-1}(x)^\top (K + \lambda^2 I)^{-1} k_{t-1}(x). \tag{3}$$

Here, $K \in \mathbb{R}^{(t-1)\times(t-1)}$ is the Gram matrix of the training inputs, $\mathcal{D}_{t-1}$, where each entry is defined as $K_{ij} = k(x_i, x_j)$ for $i,j \in \{1,\ldots,t-1\}$. The vector $k_{t-1}(x) = [k(x,x_1),\ldots,k(x,x_{t-1})]^\top$ contains the covariances between the test point x and the training inputs. The vector $y = [u_1,\ldots,u_{t-1}]^\top$ stores the observed utility values. We use a radial basis function kernel $k(\cdot,\cdot)$ as the covariance function.

F. LLM-in-the-Loop Optimization Strategy. We employ a transient hybrid strategy (29) to gradually transition from LLM-based exploration to BO-based exploitation. The probability of using BO at round t is:

$$p_t = \min\left(\frac{t^\alpha}{T}, 1\right), \quad \alpha = 2,$$

where T is the total iteration budget. Let $z_t \sim \text{Bernoulli}(p_t)$. If $z_t = 1$ and $|\mathcal{D}_{t-1}| \geq \tau$, we invoke BO using our LLM decoding; otherwise, the LLM generates the molecule:

$$s_t = \begin{cases} \tilde{\phi}^{-1}(x_t), & \text{if } z_t = 1 \text{ and } |\mathcal{D}_{t-1}| \geq \tau, \\ G_{\text{LLM}}(x_t), & \text{otherwise.} \end{cases} \tag{4}$$

Here, $\tilde{\phi}^{-1}(x_t)$ denotes molecule generation via the BO proposal obtained by maximizing the acquisition function in descriptor space, while G_{LLM} denotes a direct molecule generation by the LLM. We set $\tau = 3$ so that BO starts only after at least three data points have been collected.

Acquisition Function. The BO step maximizes the UCB acquisition function:

$$\alpha_{\text{UCB}}(x) = \mu_{t-1}(x) + \beta_t \sigma_{t-1}(x),$$

where

$$\beta_t = 2\log\left(\frac{(t+1)\cdot p \cdot \pi^2}{0.6}\right),$$

and p denotes the feature dimension, as also implemented in (29).

G. Overall Optimization Loop. Algorithm 1 outlines the overall optimization procedure of AutoLead. Starting from an initial molecule s_0, the algorithm iteratively alternates between LLM-based exploration and BO-guided exploitation, controlled by a Bernoulli variable. At each step, a new candidate is generated, evaluated, and added to the historical optimization trajectory. The surrogate model and Pareto front are continuously updated, and the final output is a set of non-dominated molecules after T iterations.

Algorithm 1 AutoLead

```
Input: initial SMILES s_0, budget T
D_0 ← {(φ(s_0), g(φ(s_0)))}
for t = 1 to T do
    Fit GP on D_{t-1}; sample z_t ~ Bernoulli(p_t)
    if z_t = 1 and |D_{t-1}| ≥ τ then
        x_t ← arg max_x α_UCB(x)
        s_t ← φ̃^{-1}(x_t)
    else
        s_t ← G_LLM
    end if
    u_t ← g(φ(s_t))
    D_t ← D_{t-1} ∪ {(φ(s_t), u_t)}
    Update Pareto front
end for
return non-dominated molecules
```

H. Multi-Objective Handling.

1. **Weighted Utility:** $U(f(s)) = \sum_{i=1}^{d} w_i f_i(s)$, with default $w_i = 1/d$.

2. **Pareto Tracking:** maintain a Pareto front of non-dominated solutions (43).

3. **Early Stopping:** terminate the process if the optimization is finished before the maximum iteration budget T is reached.

Experiments

Datasets. We evaluate AutoLead on multiple molecular optimization datasets covering diverse optimization scenarios, from single-objective property optimization to complex multi-objective lead optimization tasks.

LipinskiFix-1000. We further introduce LipinskiFix-1000, a benchmark dataset designed to evaluate molecular optimization strategies on realistic drug-like compounds. It consists of 1,000 unique ligands extracted from the high-quality HiQBind (19) database, each bound to experimentally validated protein targets. These molecules were selected for violating at least one of Lipinski's Rule of Five criteria, reflecting common challenges in early drug discovery. To ensure meaningful optimization, we applied additional quality constraints on molecular weight, lipophilicity, and QED scores, while limiting structural complexity. Each task targets restoring Lipinski compliance as the primary objective and maximizing QED as a secondary goal, providing a rigorous testbed for multi-objective molecular design.

Dataset construction. We start with the high-quality HiQBind database, which provides experimentally validated protein–ligand complexes with curated binding affinities. We filter for ligands that:

- have complete property annotations (MW, LogP, HBD, HBA, QED),
- contain at most 50 heavy atoms to ensure manageable complexity,
- violate at least one Lipinski rule (MW $>$ 500, LogP $>$ 5, HBD $>$ 5, or HBA $>$ 10),
- and additionally satisfy quality checks (QED $\geq$ 0.1, MW $\leq$ 800, LogP $\leq$ 8) to maintain feasible optimization targets.

We remove duplicates based on SMILES strings and random sample 1,000 ligands to form LipinskiFix-1000. Each entry is paired with its protein context and annotated with detailed property values, violated rules, and explicit primary (Lipinski compliance) and secondary (QED maximization) optimization objectives.

Molecular Property Optimization Dataset. We evaluate AutoLead on two standard molecular property optimization benchmarks from ChatDrug and DrugAssist that give diverse optimization scenarios.

ChatDrug-200. We use 200 molecules curated from the ChatDrug framework, which systematically investigates conversational drug editing with LLMs. These molecules cover diverse single- and multi-objective optimization tasks, such as adjusting LogP, QED, permeability (mesured by TPSA), and modifying hydrogen bond acceptor/donor counts, under loose and strict improvement thresholds Δ.

DrugAssist-500. We additionally evaluate on 500 molecules curated from the DrugAssist benchmark, designed to test interactive multi-property optimization. Tasks include simultaneous optimization of properties like BBBP, hERG inhibition, solubility, and QED, often requiring target values to lie within specified ranges while preserving structural similarity. Since RDKit can directly compute BBBP and hERG inhibition, we focus on solubility, QED, and acceptor/donor tasks for optimization under both loose and strict improvement threshold (Δ) constraints.

Evaluation Metrics. We adopt the evaluation metrics used in (13, 14) to assess AutoLead's performance across various optimization scenarios:

Success Rate:. Percentage of molecules successfully optimized to meet specified thresholds

Valid Rate:. Percentage of chemically valid molecules.
Valid rates per dataset are reported in the appendix.

Experimental Setup. To ensure the reliability of our results, all methods were evaluated over 3 independent runs for each task. We report the mean and standard deviation across these runs.

Specifications for AutoLead. We evaluate three variants of AutoLead to specifically demonstrate how the hybrid strategy improves performance:

- **LLMO (LLM Optimizer)**: An LLM-guided approach that leverages the large language model throughout all iterations to propose candidate molecules based on learned chemical priors.
- **BO (Bayesian Optimizer)**: A special case of the hybrid strategy where BO is used in every round ($z_t \equiv 1$), allowing the surrogate model to fully guide the optimization from start to finish.
- **HO (Hybrid Optimizer)**: A transient strategy that combines LLM exploration and Gaussian Process-based Bayesian Optimization (BO). It initially relies on the LLM to gather diverse molecules and switches to BO once a minimum dataset size ($\tau = 3$ points) is reached, allowing exploitation of the surrogate model (29).

Table 1. Success rate (%) on 16 ChatDrug-200 single-objective small-molecule editing. Δ denotes the required improvement threshold for each property. Evaluation is the hit ratio of the property change. The best results are marked in **bold**.

Property	Δ	Random	PCA	High Variance	GS-Mutate	MoleculeSTM-SMILES	ChatDrug		AutoLead		
							GPT-3.5-turbo	GPT-4o	LLMO	BO	HO
logP-	0	35.33 ± 1.31	33.80 ± 3.63	33.52 ± 3.75	52.00 ± 0.41	61.87 ± 2.67	22.8 ± 0.62	71.7 ± 1.70	88.3 ± 1.26	**89.8 ± 0.76**	87.7 ± 1.53
	0.5	11.04 ± 2.40	10.66 ± 3.24	10.86 ± 2.56	14.67 ± 0.62	49.02 ± 1.84	11.7 ± 0.94	62.2 ± 1.18	**78.5 ± 0.50**	**78.5 ± 0.87**	78.2 ± 1.15
logP+	0	43.36 ± 3.06	39.36 ± 2.55	42.89 ± 2.36	47.50 ± 0.41	52.71 ± 1.67	54.2 ± 0.24	78.8 ± 0.85	**91.5 ± 1.32**	90.7 ± 0.76	90.5 ± 1.00
	0.5	19.75 ± 1.56	15.12 ± 2.93	18.22 ± 3.20	12.50 ± 0.82	30.47 ± 3.26	7.5 ± 0.41	35.2 ± 0.85	**84.7 ± 1.26**	84.5 ± 0.00	83.0 ± 2.29
QED+	0	38.06 ± 2.57	33.99 ± 3.72	36.20 ± 4.34	28.00 ± 0.71	36.52 ± 2.46	18.2 ± 0.62	36.8 ± 1.31	42.0 ± 0.87	49.7 ± 2.31	**59.3 ± 2.08**
	0.1	5.27 ± 0.24	3.97 ± 0.10	4.44 ± 0.58	6.33 ± 0.82	8.81 ± 0.82	4.0 ± 0.41	8.3 ± 0.24	3.7 ± 0.58	8.5 ± 1.32	**11.5 ± 2.50**
QED-	0	36.96 ± 2.25	35.17 ± 2.61	39.99 ± 0.57	71.33 ± 0.85	58.59 ± 1.01	43.8 ± 2.72	58.0 ± 1.08	85.2 ± 2.52	85.5 ± 1.80	**86.5 ± 0.87**
	0.1	6.16 ± 1.87	5.26 ± 0.95	7.56 ± 0.29	27.67 ± 3.79	37.56 ± 1.76	8.7 ± 0.24	10.3 ± 0.85	60.5 ± 1.50	66.0 ± 1.32	**67.3 ± 1.26**
tPSA-	0	25.23 ± 2.13	21.36 ± 0.79	21.98 ± 3.77	22.00 ± 0.82	**57.74 ± 0.60**	6.8 ± 1.03	40.8 ± 2.01	39.0 ± 2.63	41.8 ± 1.26	40.5 ± 1.80
	10	17.41 ± 1.43	14.52 ± 0.80	14.66 ± 2.13	6.17 ± 0.62	**47.51 ± 1.88**	5.0 ± 0.41	28.3 ± 0.24	30.0 ± 2.60	30.0 ± 0.50	27.5 ± 0.50
tPSA+	0	16.79 ± 2.54	15.48 ± 2.40	17.10 ± 1.14	28.83 ± 1.25	34.13 ± 0.59	12.5 ± 0.41	83.8 ± 0.47	91.0 ± 0.54	89.3 ± 0.76	**91.5 ± 0.50**
	10	11.02 ± 0.71	10.62 ± 1.86	12.01 ± 1.01	15.17 ± 1.03	26.48 ± 0.97	11.8 ± 1.31	82.0 ± 1.63	90.5 ± 1.32	**90.8 ± 0.58**	90.3 ± 1.04
HBA+	0	12.64 ± 1.64	10.85 ± 2.29	11.78 ± 0.15	21.17 ± 3.09	54.01 ± 5.26	18.7 ± 1.70	74.8 ± 1.25	89.0 ± 0.38	79.0 ± 1.32	**89.3 ± 0.29**
	1	0.69 ± 0.01	0.90 ± 0.84	0.67 ± 0.01	1.83 ± 0.47	27.33 ± 2.62	1.3 ± 0.24	6.5 ± 0.50	79.2 ± 0.29	78.2 ± 1.15	**80.7 ± 1.04**
HBD+	0	2.97 ± 0.61	3.97 ± 0.55	6.23 ± 0.66	19.50 ± 2.86	28.55 ± 0.76	27.2 ± 2.01	77.8 ± 1.55	74.7 ± 12.77	83.5 ± 1.00	**86.5 ± 0.00**
	1	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	1.33 ± 0.24	7.69 ± 0.56	1.0 ± 0.00	7.2 ± 0.25	**72.5 ± 1.32**	71.2 ± 0.58	71.5 ± 0.50

Table 2. Success rate (%) on 12 ChatDrug-200 multi-objective small-molecule editing. Δ denotes the required improvement threshold for each property. Evaluation is the hit ratio of the property change. The best results are marked in **bold**.

Property	Δ	Random	PCA	High Variance	GS-Mutate	MoleculeSTM-SMILES	ChatDrug		AutoLead		
							GPT-3.5-turbo	GPT-4o	LLMO	BO	HO
logP- & HBA+	0–0	9.88 ± 1.03	8.64 ± 2.06	9.09 ± 1.25	14.00 ± 2.48	27.87 ± 3.86	19.8 ± 0.62	66.6 ± 0.00	86.5 ± 1.29	87.7 ± 0.76	**88.3 ± 0.58**
	0.5–1	0.23 ± 0.33	0.45 ± 0.64	0.22 ± 0.31	0.67 ± 0.62	8.80 ± 0.04	1.2 ± 0.24	54.9 ± 0.34	72.5 ± 0.23	72.3 ± 2.02	**75.2 ± 2.27**
logP+ & HBA+	0–0	2.99 ± 0.38	2.00 ± 0.58	2.45 ± 0.67	7.17 ± 0.85	8.55 ± 2.75	1.2 ± 0.47	4.3 ± 0.62	23.0 ± 4.36	30.7 ± 1.26	**33.3 ± 0.29**
	0.5–1	0.45 ± 0.32	0.00 ± 0.00	0.22 ± 0.31	0.17 ± 0.24	2.93 ± 0.30	0.0 ± 0.00	0.2 ± 0.24	1.7 ± 0.29	18.2 ± 0.76	**20.0 ± 0.00**
logP- & HBD+	0–0	2.28 ± 1.15	2.23 ± 1.16	4.44 ± 0.58	13.83 ± 2.95	33.51 ± 4.08	23.0 ± 1.41	79.8 ± 0.62	90.5 ± 0.00	90.3 ± 0.29	**92.0 ± 1.00**
	0.5–1	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	9.98 ± 1.03	0.7 ± 0.24	3.3 ± 0.24	75.8 ± 2.75	75.7 ± 0.29	**76.8 ± 1.76**
logP+ & HBD+	0–0	0.69 ± 0.58	1.96 ± 0.87	1.79 ± 0.66	5.67 ± 0.62	17.03 ± 2.75	3.3 ± 0.47	2.8 ± 0.24	36.2 ± 2.84	42.5 ± 1.80	**51.7 ± 1.61**
	0.5–1	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	0.00 ± 0.00	2.59 ± 1.14	0.0 ± 0.00	0.0 ± 0.00	0.8 ± 0.58	4.8 ± 1.04	**10.0 ± 4.50**
logP- & tPSA-	0–0	5.06 ± 1.21	3.53 ± 0.38	4.88 ± 2.21	8.17 ± 1.03	**35.69 ± 3.19**	0.8 ± 0.24	10.0 ± 0.71	5.0 ± 0.50	8.0 ± 0.50	8.3 ± 2.47
	0.5–10	1.16 ± 0.68	0.67 ± 0.55	0.66 ± 0.54	0.00 ± 0.00	**19.15 ± 0.73**	0.8 ± 0.24	2.8 ± 0.24	0.2 ± 0.29	1.3 ± 0.58	1.7 ± 0.76
logP- & tPSA+	0–0	12.17 ± 1.05	10.43 ± 2.88	13.08 ± 2.28	19.83 ± 2.46	44.35 ± 0.68	13.7 ± 1.25	75.8 ± 1.25	**92.0 ± 1.73**	91.5 ± 0.50	91.8 ± 1.61
	0.5–10	6.20 ± 0.64	6.23 ± 2.31	6.67 ± 0.53	4.83 ± 0.85	28.67 ± 2.22	12.3 ± 1.25	66.0 ± 0.71	82.0 ± 1.50	82.7 ± 0.29	**83.2 ± 0.58**

For both variants, we set the maximum number of optimization iterations T to 10 for each molecule. The process stops early if a molecule satisfying the target property threshold is found, or continues until the iteration limit is reached.

Baselines Methods. The baselines (44) are based on MegaMolBART (45), a pretrained auto-regressive model. Baselines include Random, PCA, High-Variance, GS-Mutate (46), and MoleculeSTM (44) with SMILES and ChatDrug with GPT-3.5 Turbo and GPT-4o (47) as the backbone LLMs. For the ChatDrug baselines, we set the number of conversation rounds to 10, and reran their implementation to be precisely aligned with the evaluation metrics, as also noted in (13). For the other baselines, we adopted the values reported in (14), if available.

Performance on the DrugAssist-500 dataset. Results on the DrugAssist-500 dataset (Table 3) further confirm the generalizability and effectiveness of our proposed hybrid method, AutoLead-HO, in interactive multi-property optimization scenarios. The AutoLead framework, including its LLMO and BO variants, demonstrates superior performance by achieving the top success rate across most of the tasks. This outperformance is especially pronounced in the more challenging strict settings. For instance, in the complex multi-property task of decreasing logP while increasing HBA+ with a stringent threshold ($\Delta = 1$), our hybrid approach HO achieves a success rate of 72.0%, a dramatic improvement over the 0.8% from ChatDrug. In another difficult task, increasing the number of hydrogen bond donors (HBD+) with a strict threshold of $\Delta = 1$, our BO variant leads with a 70.7% success rate, again showcasing a substantial leap from the 4.2% achieved by ChatDrug with GPT-4o.

Results

We evaluate AutoLead on a variety of single-objective and multi-objective molecular optimization tasks using datasets from ChatDrug, DrugAssist and our newly introduced LipinskiFix-1000 datasets. We compare three versions of our method—pure LLM optimization (AutoLead-LLMO), pure Bayesian optimization (AutoLead-BO), and our hybrid strategy (AutoLead-HO)—against state-of-the-art baselines.

Performance on the ChatDrug-200 dataset. Our proposed hybrid optimization method, AutoLead-HO, demonstrates a commanding performance on the ChatDrug-200 benchmark, significantly outperforming both existing state-of-the-art methods and its own ablation variants, AutoLead-LLMO and AutoLead-BO.

Table 3. Success rate (%) on DrugAssist-500 10 small molecule editing. Δ denotes the required improvement threshold for each property. Evaluation is the hit ratio of the property change. The best results are marked in **bold**.

Property	Δ	Random	PCA	High Variance	GS-Mutate	MoleculeSTM-SMILES	ChatDrug		AutoLead		
							GPT-3.5-turbo	GPT-4o	LLMO	BO	HO
logP-	0	39.20 ± 3.15	42.00 ± 3.40	39.60 ± 0.86	50.67 ± 0.41	53.67 ± 6.03	28.2 ± 0.33	66.6 ± 0.00	86.3 ± 1.29	86.6 ± 1.29	**87.7 ± 0.23**
	0.5	25.67 ± 4.06	27.20 ± 2.41	25.53 ± 1.84	13.27 ± 0.66	44.20 ± 5.66	18.7 ± 0.25	54.9 ± 0.34	73.5 ± 0.23	72.6 ± 0.23	**75.2 ± 2.27**
QED+	0	44.73 ± 2.21	46.47 ± 3.85	44.33 ± 3.40	30.40 ± 0.91	53.80 ± 0.00	16.0 ± 0.33	28.8 ± 0.49	36.1 ± 1.32	26.1 ± 16.52	**62.3 ± 1.75**
	0.1	25.20 ± 3.99	25.67 ± 2.99	22.93 ± 2.47	6.00 ± 0.43	**37.27 ± 6.88**	6.8 ± 0.28	10.1 ± 0.34	15.1 ± 0.40	10.1 ± 0.50	24.3 ± 0.92
HBA+	0	58.13 ± 1.91	58.60 ± 2.55	57.80 ± 1.13	**92.40 ± 1.56**	63.80 ± 0.00	24.1 ± 0.50	68.5 ± 1.23	49.5 ± 32.97	51.5 ± 32.97	86.3 ± 0.42
	1	13.13 ± 0.84	13.27 ± 0.84	12.53 ± 1.67	23.93 ± 1.43	15.20 ± 8.77	0.4 ± 0.16	2.1 ± 0.25	**79.8 ± 0.35**	77.8 ± 0.35	77.1 ± 1.33
HBD+	0	58.00 ± 2.57	58.40 ± 2.47	57.67 ± 0.41	**86.80 ± 1.56**	68.60 ± 1.98	28.6 ± 0.59	68.5 ± 0.52	84.1 ± 1.03	83.2 ± 1.01	85.7 ± 0.42
	1	8.00 ± 1.88	8.20 ± 1.88	8.33 ± 0.57	18.60 ± 0.33	8.13 ± 1.04	0.3 ± 0.09	4.2 ± 0.28	68.6 ± 0.87	**70.7 ± 0.60**	70.2 ± 0.40
logP- & HBA+	0	33.20 ± 1.88	35.00 ± 1.56	34.40 ± 0.59	49.33 ± 0.47	38.07 ± 3.49	26.8 ± 0.28	63.5 ± 1.16	**84.1 ± 0.95**	82.0 ± 0.60	75.0 ± 0.35
	1	7.93 ± 0.68	9.27 ± 1.09	8.47 ± 1.15	6.73 ± 0.66	11.93 ± 1.32	0.2 ± 0.16	0.8 ± 0.28	70.6 ± 0.53	71.0 ± 0.60	**72.0 ± 0.60**

Table 4. Success rate (%) on the LipinskiFix-1000 dataset. The evaluation metrics is the success rate (percentage of molecules that satisfy both Lipinski's Rule of Five and achieve QED ≥ 0.7). The best results are marked in **bold**.

Method	Success Rate (%)
Random	11.50 ± 3.65
PCA	11.90 ± 4.04
High Variance	12.30 ± 4.95
GS-Mutate	1.50 ± 0.23
MoleculeSTM-SMILES	13.90 ± 2.55
ChatDrug (GPT-3.5-turbo)	3.87 ± 0.06
ChatDrug (GPT-4o)	3.83 ± 0.29
AutoLead-LLMO	2.50 ± 0.21
AutoLead-BO	20.30 ± 0.85
AutoLead-HO	**28.90 ± 1.66**

In the single-objective setting (Table 1), AutoLead-HO consistently achieves top success rates, especially under the most stringent test conditions. For instance, on challenging tasks such as increasing the hydrogen bond acceptor count (HBA+) or optimizing for QED+, our method significantly outperforms all baselines, including the strong ChatDrug with GPT-4o model. This highlights the robustness of our hybrid strategy in focused optimization scenarios.

In the more demanding 12 multi-objective tasks (Table 2), the superiority of our hybrid approach is further confirmed. The AutoLead framework achieved the best performance on all tasks except for `logP- & tPSA-`, with AutoLead-HO consistently delivering the top performance in the most complex scenarios. While the MoleculeSTM-SMILES baseline is strong in tasks involving tPSA reduction, our method dominates elsewhere. For instance, when simultaneously decreasing logP and increasing hydrogen bond donors (`logP- & HBD+`) under strict thresholds ($\Delta = 0.5 - 1$), AutoLead-HO's success rate of **76.8%** is dramatically higher than ChatDrug's **3.3%**. Likewise, for increasing both logP and HBA+ ($\Delta = 0.5 - 1$), our hybrid method achieves a **20.0%** success rate where the previous state-of-the-art only achieves **0.2%**.

Performance on the LipinskiFix-1000 Dataset. Finally, we evaluate AutoLead on our newly introduced LipinskiFix-1000 dataset, which is specifically designed to mirror realistic scenarios in drug development. This benchmark focuses on lead optimization by requiring molecules to restore compliance with Lipinski's Rule of Five, while also improving the Quantitative Estimation of Drug-likeness (QED). The results, summarized in Table 4, highlight the strong performance of AutoLead in this challenging task.

Despite the fact that these tasks do not explicitly vary Δ as in the synthetic ChatDrug or DrugAssist settings, they still present a stringent multi-objective challenge. This is because they demand the simultaneous satisfaction of multiple pharmacokinetic constraints—such as Lipinski's Rule of Five—while also improving QED. These tasks thus offer a more realistic representation of the complexity of drug optimization in early-stage drug discovery.

AutoLead, particularly with its hybrid optimization strategy (HO), shows significant advantages in transforming non-compliant hits into viable leads, surpassing all other methods. The success rate for AutoLead-HO reached 28.9%, a considerable improvement over previous approaches like MoleculeSTM and ChatDrug, which achieve only 13.9% and 3.87%, respectively. This underscores the practical utility of the hybrid approach, which combines LLM-driven exploration with Bayesian optimization to navigate the complex, multi-dimensional chemical space.

The results on the LipinskiFix-1000 dataset confirm the framework's ability to handle real-world multi-objective challenges, demonstrating its potential for guiding the development of drug-like molecules. By improving both pharmacokinetic properties and QED, AutoLead, specifically its HO variant, provides a strong foundation for advancing practical lead optimization pipelines.

Comparative Analysis of Optimization Strategies Our results clearly demonstrate that the hybrid strategy (HO), which dynamically combines LLM-driven exploration with BO-guided exploitation, achieves the best or comparable performance to the LLM Optimizer (LLMO) across all tasks. In the early stages, the LLM excels at proposing chemically meaningful edits, rapidly exploring diverse regions of chemical space as similarly observed in hybrid frameworks like BORA (31). As the process progresses, the Bayesian surrogate increasingly focuses on promising areas, refining candidates with greater precision through uncertainty-aware acquisition (48, 49).

This collaborative dynamic is particularly advantageous in multi-objective optimization, where neither LLMs nor BO alone reliably achieve simultaneous satisfaction of all property constraints. Notably, the feature-to-SMILES inverse map-

ping (ϕ^{-1}) remains a challenging step; improving its fidelity would better translate optimized feature space points back into valid molecular structures. Exploring optimization trajectories directly in the descriptor space—highlighting how BO navigates trade-offs over iterations—will be an important direction for future work to provide deeper insights into the interplay between exploration and exploitation.

Visualization of Successful Lead Optimizations. To further illustrate the practical effectiveness of AutoLead, Figure 2 showcases representative examples of successful lead optimizations. These examples demonstrate AutoLead's ability to perform complex edits, transforming initial molecules into more favorable candidates by improving key drug-like properties. Notably, we observe significant increases in QED alongside improvements in Lipinski's Rule of Five compliance. For instance, several transformations show substantial reductions in MW and logP, as well as increases in HBD and HBA counts to fall within desirable ranges. The modifications, such as the strategic introduction of fluorine atoms or subtle structural alterations like ring contractions and functional group removals, are chemically sensible and preserve the core structural motifs of the original molecules. This highlights AutoLead's capability to generate realistic and relevant chemical transformations aligned with medicinal chemistry principles during early-stage lead optimization.

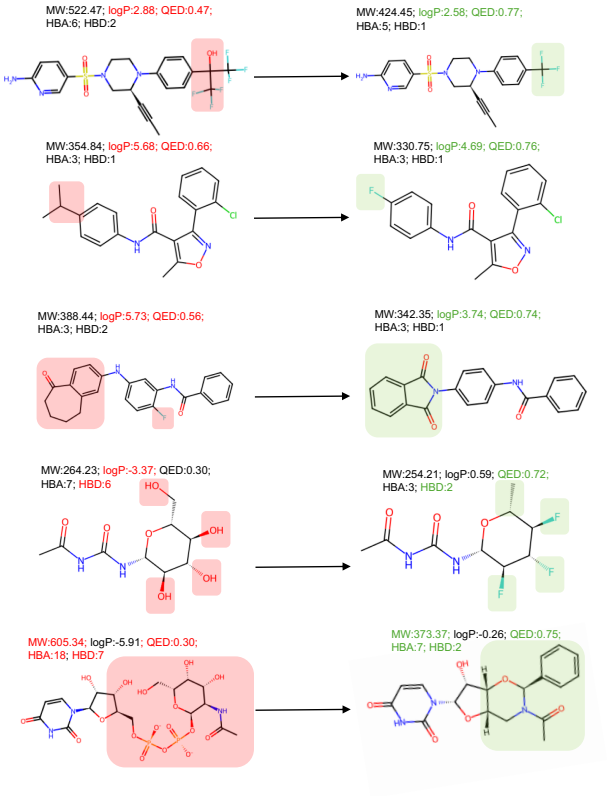

Fig. 2. Examples of successful molecule refinement by AutoLead on the LipinskiFix-1000 dataset. For each original (left) and optimized (right) pair, key properties are displayed. Green indicates a favorable change, while red flags the initial property that failed to meet drug-likeness criteria.

Conclusions

In this work, we introduce AutoLead, a novel framework that integrates the chemical reasoning of LLMs with the principled exploration of multi-objective Bayesian optimization to address realistic lead optimization challenges. Unlike prior approaches that rely on narrow objectives or handcrafted features, AutoLead combines the intuitive generative power of LLMs with the uncertainty-aware search of Gaussian Processes to efficiently explore complex molecular spaces.

Across three diverse benchmarks—ChatDrug-200, DrugAssist-500, and the newly introduced LipinskiFix-1000, which simulates a realistic drug-likeness evaluation scenario—AutoLead consistently outperforms existing methods. In particular, our hybrid strategy excels on challenging multi-property tasks, surpassing both LLM-only and BO-only baselines. These results underscore the potential of coupling LLMs with black-box optimization for molecular design, paving the way for more robust and scalable solutions in computational drug discovery.

Reference

1. Gary W Caldwell, Zhengyin Yan, Weimin Tang, Malini Dasgupta, and Becki Hasting. Adme optimization and toxicity assessment in early-and late-phase drug discovery. *Current topics in medicinal chemistry*, 9(11):965–980, 2009.
2. Alexander S Rich, Yvonne H Chan, Benjamin Birnbaum, Kamran Haider, Joshua Haimson, Michael Hale, Yongxin Han, William Hickman, Klaus P Hoeflich, Daniel Ortwine, et al. Machine learning adme models in practice: Four guidelines from a successful lead optimization case study. *ACS Medicinal Chemistry Letters*, 15(8):1169–1173, 2024.
3. Robin Winter, Floriane Montanari, Andreas Steffen, Hans Briem, Frank Noé, and Djork-Arné Clevert. Efficient multi-objective molecular optimization in a continuous latent space. *Chemical science*, 10(34):8016–8024, 2019.
4. Peter J Eddershaw, Alan P Beresford, and Martin K Bayliss. Adme/pk as part of a rational approach to drug discovery. *Drug discovery today*, 5(9):409–414, 2000.
5. Regine S Bohacek, Colin McMartin, and Wayne C Guida. The art and practice of structure-based drug design: a molecular modeling perspective. *Medicinal research reviews*, 16(1): 3–50, 1996.
6. Jiajun Yu, Yizhen Zheng, Huan Yee Koh, Shirui Pan, Tianyue Wang, and Haishuai Wang. Collaborative expert llms guided multi-objective molecular optimization. *arXiv preprint arXiv:2503.03503*, 2025.
7. Kun Li, Zhennan Wu, Shoupeng Wang, and Wenbin Hu. Drugpilot: Llm-based parameterized reasoning agent for drug discovery. *arXiv preprint arXiv:2505.13940*, 2025.
8. Ali Essam Ghareeb, Benjamin Chang, Ludovico Mitchener, Angela Yiu, Caralyn J Szostkiewicz, Jon M Laurent, Muhammed T Razzak, Andrew D White, Michaela M Hinks, and Samuel G Rodriques. Robin: A multi-agent system for automating scientific discovery. *arXiv preprint arXiv:2505.13400*, 2025.
9. Bowen Gao, Yanwen Huang, Yiqiao Liu, Wenxuan Xie, Wei-Ying Ma, Ya-Qin Zhang, and Yanyan Lan. Pharmagents: Building a virtual pharma with large language model agents. *arXiv preprint arXiv:2503.22164*, 2025.
10. Di Zhang, Wei Liu, Qian Tan, Jingdan Chen, Hang Yan, Yuliang Yan, Jiatong Li, Weiran Huang, Xiangyu Yue, Wanli Ouyang, et al. Chemllm: A chemical large language model. *arXiv preprint arXiv:2402.06852*, 2024.
11. Qiang Zhang, Keyan Ding, Tianwen Lv, Xinda Wang, Qingyu Yin, Yiwen Zhang, Jing Yu, Yuhao Wang, Xiaotong Li, Zhuoyi Xiang, et al. Scientific large language models: A survey on biological & chemical domains. *ACM Computing Surveys*, 57(6):1–38, 2025.
12. Vishal Dey, Xiao Hu, and Xia Ning. Large language models for controllable multi-property multi-objective molecule optimization. *arXiv preprint arXiv:2505.23987*, 2025.
13. Geyan Ye, Xibao Cai, Houtim Lai, Xing Wang, Junhong Huang, Longyue Wang, Wei Liu, and Xiangxiang Zeng. Drugassist: A large language model for molecule optimization. *Briefings in Bioinformatics*, 26(1):bbae693, 2025.
14. Shengchao Liu, Jiongxiao Wang, Yijin Yang, Chengpeng Wang, Ling Liu, Hongyu Guo, and Chaowei Xiao. Conversational drug editing using retrieval and domain feedback. In *The twelfth international conference on learning representations*, 2024.
15. Christopher KI Williams and Carl Edward Rasmussen. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.
16. Robert B Gramacy. *Surrogates: Gaussian process modeling, design, and optimization for the applied sciences*. Chapman and Hall/CRC, 2020.
17. Christopher A Lipinski, Franco Lombardo, Beryl W Dominy, and Paul J Feeney. Experimental and computational approaches to estimate solubility and permeability in drug discovery and development settings. *Advanced drug delivery reviews*, 64:4–17, 2012.
18. Christopher A Lipinski. Lead-and drug-like compounds: the rule-of-five revolution. *Drug discovery today: Technologies*, 1(4):337–341, 2004.
19. Yingze Wang, Kunyang Sun, Jie Li, Xingyi Guan, Oufan Zhang, Dorian Bagni, Yang Zhang, Heather A Carlson, and Teresa Head-Gordon. A workflow to create a high-quality protein–ligand binding dataset for training, validation, and prediction tasks. *Digital Discovery*, 4(5):

1209–1220, 2025.
20. David E. Rumelhart and James L. McClelland. Learning Internal Representations by Error Propagation. In *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations*, pages 318–362. MIT Press, 1987. ISBN 978-0-262-29140-8.
21. Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.
22. Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, page 5998–6008. Curran Associates, Inc., 2017.
23. Wengong Jin, Regina Barzilay, and Tommi S. Jaakkola. Junction tree variational autoencoder for molecular graph generation. In Jennifer G. Dy and Andreas Krause, editors, *ICML*, volume 80 of *Proceedings of Machine Learning Research*, pages 2328–2337. PMLR, 2018.
24. Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Hierarchical generation of molecular graphs using structural motifs. In *International conference on machine learning*, pages 4839–4848. PMLR, 2020.
25. Ksenia Korovina, Sailun Xu, Kirthevasan Kandasamy, Willie Neiswanger, Barnabas Poczos, Jeff Schneider, and Eric Xing. Chembo: Bayesian optimization of small organic molecules with synthesizable recommendations. In *International Conference on Artificial Intelligence and Statistics*, pages 3393–3403. PMLR, 2020.
26. David E Graff, Eugene I Shakhnovich, and Connor W Coley. Accelerating high-throughput virtual screening through molecular pool-based active learning. *Chemical science*, 12(22): 7866–7881, 2021.
27. Tai Dang, Long-Hung Pham, Sang T Truong, Ari Glenn, Wendy Nguyen, Edward A Pham, Jeffrey S Glenn, Sanmi Koyejo, and Thang Luong. Preferential multi-objective bayesian optimization for drug discovery. *arXiv preprint arXiv:2503.16841*, 2025.
28. Chengrun Yang, Xuezhi Wang, Yifeng Lu, Hanxiao Liu, Quoc V Le, Denny Zhou, and Xinyun Chen. Large language models as optimizers. In *The Twelfth International Conference on Learning Representations*, 2023.
29. Chih-Yu Chang, Milad Azvar, Chinedum Emmanuel Okwudire, and Raed Al Kontar. Llinbo: Trustworthy llm-in-the-loop bayesian optimization. *CoRR*, 2025.
30. Tennison Liu, Nicolás Astorga, Nabeel Seedat, and Mihaela van der Schaar. Large language models to enhance bayesian optimization. *arXiv preprint arXiv:2402.03921*, 2024.
31. Abdoulatif Cissé, Xenophon Evangelopoulos, Vladimir V Gusev, and Andrew I Cooper. Language-based bayesian optimization research assistant (bora). *arXiv preprint arXiv:2501.16224*, 2025.
32. Zheni Zeng, Bangchen Yin, Shipeng Wang, Jiarui Liu, Cheng Yang, Haishen Yao, Xingzhi Sun, Maosong Sun, Guotong Xie, and Zhiyuan Liu. Chatmol: interactive molecular discovery with natural language. *Bioinformatics*, 40(9):btae534, 2024.
33. Nikita Janakarajan, Tim Erdmann, Sarath Swaminathan, Teodoro Laino, and Jannis Born. Language models in molecular discovery. In *Drug Development Supported by Informatics*, pages 121–141. Springer, 2024.
34. Carl Edwards, Tuan Lai, Kevin Ros, Garrett Honke, Kyunghyun Cho, and Heng Ji. Translation between molecules and natural language. *arXiv preprint arXiv:2204.11817*, 2022.
35. Qizhi Pei, Wei Zhang, Jinhua Zhu, Kehan Wu, Kaiyuan Gao, Lijun Wu, Yingce Xia, and Rui Yan. Biot5: Enriching cross-modal integration in biology with chemical knowledge and natural language associations. *arXiv preprint arXiv:2310.07276*, 2023.
36. Tung Nguyen and Aditya Grover. Lico: Large language models for in-context molecular optimization. *arXiv preprint arXiv:2406.18851*, 2024.
37. John H Holland. *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.
38. Teague Sterling and John J Irwin. Zinc 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 55(11):2324–2337, 2015.
39. Haorui Wang, Marta Skreta, Cher-Tian Ser, Wenhao Gao, Lingkai Kong, Felix Strieth-Kalthoff, Chenru Duan, Yuchen Zhuang, Yue Yu, Yanqiao Zhu, et al. Efficient evolutionary search over chemical space with large language models. *arXiv preprint arXiv:2406.16976*, 2024.
40. Nian Ran, Yue Wang, and Richard Allmendinger. Mollm: Multi-objective large language model for molecular design–optimizing with experts. *arXiv preprint arXiv:2502.12845*, 2025.
41. David Weininger. Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules. *Journal of chemical information and computer sciences*, 28(1):31–36, 1988.
42. RDKit. Rdkit: Open-source cheminformatics. https://www.rdkit.org, 2025.
43. Anabel Yong, Austin Tripp, Layla Hosseini-Gerami, and Brooks Paige. Bayesian optimization for molecules should be pareto-aware. *arXiv preprint arXiv:2507.13704*, 2025.
44. Shengchao Liu, Weili Nie, Chengpeng Wang, Jiarui Lu, Zhuoran Qiao, Ling Liu, Jian Tang, Chaowei Xiao, and Animashree Anandkumar. Multi-modal molecule structure–text model for text-based retrieval and editing. *Nature Machine Intelligence*, 5(12):1447–1457, 2023.
45. Ross Irwin, Spyridon Dimitriadis, Jiazhen He, and Esben Jannik Bjerrum. Chemformer: a pre-trained transformer for computational chemistry. *Machine Learning: Science and Technology*, 3(1):015022, 2022.
46. Jan H Jensen. A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space. *Chemical science*, 10(12):3567–3572, 2019.
47. Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*, 2024.
48. Jaewon Chu, Jinyoung Park, Seunghun Lee, and Hyunwoo J Kim. Inversion-based latent bayesian optimization. *Advances in Neural Information Processing Systems*, 37:68258–68286, 2024.
49. Ryan-Rhys Griffiths and José Miguel Hernández-Lobato. Constrained bayesian optimization for automatic chemical design using variational autoencoders. *Chemical Science*, 11(2): 577–586, 2020. ISSN 2041-6539. doi: 10.1039/c9sc04026a.

Supplementary Note 1: Prompt Design and Workflow

We employ a novel LLM-in-the-loop multi-objective drug optimization framework that integrates Large Language Models (LLMs) with Bayesian Optimization (BO). The core idea is to use LLMs to generate and refine molecular structures in response to optimization objectives and historical evaluations, while BO guides the search in chemical space.

To facilitate this interaction, we design a set of prompt templates tailored to the distinct stages of the optimization process. Each prompt is carefully crafted to ensure the LLM outputs are chemically valid, optimization-aware, and compatible with downstream evaluation.

Table 5 summarizes the prompt templates used throughout the AutoLead pipeline.

Table 5. Prompt templates used across different stages of AutoLead. Each prompt is conditioned on historical data and task objectives, guiding the LLM in molecule generation, transformation, or evaluation.

Phase	Prompt Template
Initialization & Molecule Optimization	The following are the drug molecules and their properties evaluated in the past: **[History String]**. Optimization objectives: **[Objectives Description]**. Based on the molecule **[Base Molecule SMILES]**, please design an improved new molecule. **IMPORTANT CONSTRAINTS:** 1. You can make the following types of modifications: – ADD functional groups (e.g., -OH, -NH2, -COOH, -CH3, -F, -Cl) – DELETE existing functional groups or side chains – SUBSTITUTE functional groups (e.g., replace -OH with -NH2, -CH3 with -CF3) – Modify ring systems (add/remove substituents on rings) 2. Preserve the core molecular scaffold and main structural framework. 3. Avoid completely changing the molecular backbone or creating entirely different structures. 4. Focus on single-step modifications rather than multiple simultaneous changes. 5. The new molecule should optimize the above objectives while maintaining chemical validity. 6. **PLEASE CAREFULLY CONSIDER THE OPTIMIZATION SUGGESTIONS above when designing the new molecule.** Please only return a valid SMILES string.
Feature Vector to SMILES Conversion	**[Historical Context]** Based on the above molecular history, please generate a valid SMILES molecule with the following characteristics: **[Feature Description]**. The molecule should be chemically valid and drug-like. Learn from the patterns in the previously generated molecules to create a structurally diverse yet optimized compound. Return only the SMILES string.
Candidate Evaluation and Scoring	The following are past evaluations of drug molecules: **[History String]**. Optimization objectives: **[Objectives Description]**. Based on the historical data, predict the composite score for the molecule: **[Candidate SMILES]**. Return only a numerical value between 0 and 1.

Initialization of Molecules.. Given an initial lead molecule, this stage aims to generate a diverse pool of chemically valid and drug-like candidates using the LLM. The goal is to warm-start the optimization process by producing sufficient data points for fitting the Gaussian Process surrogate model used in subsequent BO steps. The prompt guides the LLM to propose structurally consistent modifications that satisfy domain-specific constraints, such as scaffold preservation and functional group plausibility, without relying on prior optimization history.

Molecule Optimization.. At this stage, the LLM is guided to propose improved molecules based on a given base molecule and a set of optimization objectives (e.g., increase QED or logP). The prompt constrains the model to perform minimal and chemically plausible modifications, such as single-step substitutions or functional group edits, while preserving the core scaffold.

Feature Vector to SMILES Conversion.. In some scenarios, molecules are represented as abstract latent or feature vectors. This prompt converts such representations back into valid SMILES strings by leveraging learned structural patterns from previous generations. The goal is to decode high-quality molecules that exhibit the target features.

Candidate Evaluation and Scoring.. Finally, the LLM can act as a surrogate evaluator, predicting the composite score of a candidate molecule given historical data. This supports query-efficient BO by approximating molecular fitness without actual oracle evaluation.

Table 6. Valid rate (%) on DrugAssist-500 10 small-molecule editing tasks. Evaluation is the hit ratio of the property change. The second-best results are marked in **bold**.

Property	Δ	Random	PCA	High Variance	GS-Mutate	MoleculeSTM-SMILES	ChatDrug		AutoLead		
							GPT-3.5-turbo	GPT-4o	LLMO	BO	HO
logP-	0	70.21 ± 0.38	69.57 ± 0.83	69.48 ± 0.78	100.0 ± 0.00	73.87 ± 11.79	66.9 ± 1.05	82.7 ± 0.68	86.9 ± 1.22	**88.7 ± 0.42**	88.4 ± 0.35
	0.5					73.87 ± 11.79	64.1 ± 0.94	81.8 ± 0.43	86.5 ± 1.10	**87.8 ± 0.69**	87.3 ± 0.95
QED+	0					68.08 ± 4.30	69.5 ± 0.74	**78.1 ± 0.90**	26.9 ± 16.92	63.3 ± 2.04	63.3 ± 1.89
	0.1					68.08 ± 4.30	65.1 ± 0.52	**77.1 ± 0.47**	46.1 ± 0.58	63.6 ± 1.83	62.9 ± 1.63
HBA+	0					75.61 ± 9.65	67.8 ± 0.65	80.3 ± 0.41	50.7 ± 34.18	**89.7 ± 0.31**	88.5 ± 0.81
	1					75.61 ± 9.65	59.3 ± 0.81	64.9 ± 0.25	88.9 ± 0.12	**89.3 ± 0.61**	89.0 ± 1.06
HBD+	0					79.52 ± 4.92	76.2 ± 0.91	79.9 ± 0.68	87.5 ± 0.64	88.7 ± 0.31	**89.1 ± 0.12**
	1					79.52 ± 4.92	70.7 ± 0.41	75.7 ± 1.24	88.8 ± 0.40	88.7 ± 0.42	**89.0 ± 0.72**
logP- & HBA+	0					77.40 ± 5.60	63.5 ± 1.15	80.8 ± 0.91	90.5 ± 0.58	91.2 ± 0.20	**91.4 ± 0.35**
	1					77.40 ± 5.60	51.4 ± 0.65	69.1 ± 0.90	90.7 ± 0.31	90.5 ± 0.31	**90.8 ± 0.72**

Table 7. Valid rate (%) on ChatDrug-200 16 single-objective and 12 multi-objective small molecule editing tasks. Evaluation is the hit ratio of the property change. The second-best results are marked in **bold**.

Property	Δ	Random	PCA	High Variance	GS-Mutate	MoleculeSTM-SMILES	ChatDrug		AutoLead		
							GPT-3.5-turbo	GPT-4o	LLMO	BO	HO
logP-	0	**92.8 ± 0.75**	92.0 ± 0.00	92.7 ± 0.05	100.0 ± 0.00	82.6 ± 11.77	57.2 ± 1.93	84.0 ± 1.41	88.8 ± 1.61	90.2 ± 1.04	88.7 ± 1.53
	0.5					82.6 ± 11.77	52.2 ± 0.62	81.2 ± 1.55	88.7 ± 0.76	89.2 ± 1.44	89.2 ± 1.04
logP+	0					72.7 ± 13.03	66.2 ± 1.55	85.2 ± 0.85	91.5 ± 1.32	90.7 ± 0.76	90.5 ± 1.00
	0.5					72.7 ± 13.03	47.3 ± 1.55	80.3 ± 0.62	90.3 ± 0.76	91.2 ± 0.58	90.3 ± 1.53
QED+	0					78.7 ± 9.26	58.5 ± 1.41	77.5 ± 1.08	43.2 ± 0.58	61.8 ± 2.47	60.5 ± 2.60
	0.1					78.7 ± 9.26	56.0 ± 0.41	72.3 ± 0.85	41.8 ± 2.02	62.2 ± 0.76	61.2 ± 2.36
QED-	0					77.0 ± 8.62	72.5 ± 0.71	81.8 ± 1.43	86.0 ± 1.32	85.5 ± 1.80	86.7 ± 1.15
	0.1					77.0 ± 8.62	61.0 ± 1.08	76.8 ± 1.89	85.3 ± 2.75	85.3 ± 1.15	85.8 ± 0.76
tPSA-	0					82.4 ± 8.84	57.0 ± 0.71	81.8 ± 1.25	90.0 ± 0.50	89.8 ± 1.04	88.8 ± 0.29
	10					82.4 ± 8.84	52.7 ± 1.25	81.8 ± 2.49	87.5 ± 0.43	91.8 ± 0.76	90.3 ± 0.58
tPSA+	0					81.1 ± 7.65	61.2 ± 0.85	85.7 ± 0.24	91.5 ± 0.83	91.7 ± 1.53	92.0 ± 0.50
	10					81.1 ± 7.65	57.0 ± 2.86	84.7 ± 0.62	91.2 ± 1.15	91.5 ± 0.50	91.0 ± 0.87
HBA+	0					81.8 ± 12.13	53.3 ± 1.84	79.7 ± 0.85	90.2 ± 0.29	91.7 ± 1.53	90.7 ± 0.29
	1					81.8 ± 12.13	40.5 ± 0.71	63.8 ± 0.75	90.0 ± 0.96	91.3 ± 0.58	90.7 ± 0.76
HBD+	0					87.8 ± 6.69	58.5 ± 2.68	83.5 ± 1.47	87.8 ± 0.76	88.7 ± 0.31	88.2 ± 0.29
	1					87.8 ± 6.69	40.3 ± 0.62	72.5 ± 0.50	76.0 ± 3.31	91.3 ± 0.58	87.7 ± 0.58
logP- & HBA+	0–0	**92.8 ± 0.75**	92.0 ± 0.00	92.7 ± 0.05	100.0 ± 0.00	92.30 ± 5.52	52.2 ± 0.47	85.0 ± 1.87	92.7 ± 0.58	91.8 ± 0.29	92.2 ± 0.58
	0.5–1					92.30 ± 5.52	40.2 ± 1.25	74.5 ± 0.41	91.5 ± 0.00	90.8 ± 0.58	91.5 ± 0.50
logP+ & HBA+	0–0					**93.43 ± 4.20**	38.8 ± 1.43	59.2 ± 0.47	87.7 ± 0.76	88.0 ± 1.00	86.8 ± 1.04
	0.5–1					**93.43 ± 4.20**	36.3 ± 0.24	57.7 ± 0.94	87.5 ± 1.80	86.3 ± 0.58	86.7 ± 0.58
logP- & HBD+	0–0					92.97 ± 4.86	50.3 ± 0.62	86.0 ± 1.08	92.8 ± 0.58	93.2 ± 0.58	**93.7 ± 0.76**
	0.5–1					92.97 ± 4.86	46.3 ± 1.70	79.3 ± 1.25	92.7 ± 0.76	**93.2 ± 0.76**	92.7 ± 0.76
logP+ & HBD+	0–0					**94.37 ± 4.15**	44.3 ± 1.03	67.7 ± 0.85	85.2 ± 0.58	86.5 ± 0.87	83.7 ± 0.58
	0.5–1					**94.37 ± 4.15**	39.0 ± 1.22	63.8 ± 1.03	84.0 ± 0.87	84.3 ± 0.76	83.2 ± 0.35
logP- & tPSA-	0–0					91.53 ± 0.24	48.5 ± 1.47	78.2 ± 0.24	87.7 ± 0.29	89.5 ± 1.73	89.2 ± 0.58
	0.5–10					91.53 ± 0.24	42.7 ± 0.24	75.5 ± 2.48	87.0 ± 1.00	88.3 ± 1.04	87.7 ± 0.29
logP- & tPSA+	0–0					92.27 ± 1.46	48.3 ± 0.24	82.7 ± 1.25	**94.3 ± 1.53**	93.7 ± 0.58	93.8 ± 0.29
	0.5–10					92.27 ± 1.46	47.5 ± 1.87	81.0 ± 0.41	93.0 ± 1.73	93.2 ± 0.29	**93.7 ± 0.76**

Supplementary Note 2: Additional Results

We report additional evaluation results of AutoLead on three molecule editing benchmarks: **ChatDrug-200** (14), **DrugAssist-500** (13), and **LipinskiFix-1000**, under both single- and multi-objective settings. Across all tasks, we measure the *valid rate*, i.e., the hit ratio of generating molecules that satisfy the desired property changes.

Valid Rate Analysis Across Benchmarks. Across all evaluated benchmarks—DrugAssist-500 (Table 6), ChatDrug-200 (Table 7), and LipinskiFix-1000 (Table 8)—our AutoLead framework consistently demonstrates high chemical validity. The

Table 8. Valid rate (%) on the LipinskiFix-1000 benchmark.

Method	Valid Rate (%)
Random	48.0 ± 0.17
PCA	48.3 ± 0.72
High Variance	48.2 ± 0.37
GS-Mutate	100.0 ± 0.00
MoleculeSTM-SMILES	49.4 ± 3.69
ChatDrug (GPT-3.5-turbo)	70.0 ± 0.59
ChatDrug (GPT-4o)	79.3 ± 0.32
AutoLead-LLMO	93.7 ± 0.25
AutoLead-BO	**94.7 ± 0.17**
AutoLead-HO	93.6 ± 1.22

baselines represent different strategies. GS-Mutate (46), a genetic algorithm, is guaranteed to have a 100% valid rate due to an internal check that reverts to the original molecule if no valid mutation is found. Given it serves as a deterministic upper bound, we highlight the **second-best performing model** in our tables for a more informative comparison. In contrast, methods like Random, PCA, and High Variance are task-agnostic; they apply a fixed modification procedure in the latent space of a MegaMolBART (45) autoencoder regardless of the optimization goal, resulting in identical valid rates across all tasks.

Among the generative models capable of handling prompts, our AutoLead variants consistently achieve higher valid rates than the LLM-based ChatDrug baselines. On the ChatDrug-200 tasks, AutoLead maintains valid rates often exceeding 90%. This performance is substantially higher than that of ChatDrug (GPT-4o), whose rates are often below 85%, and particularly ChatDrug (GPT-3.5-turbo), which frequently scores below 55%. This trend continues on the DrugAssist-500 benchmark, where AutoLead's validity remains robust; for example, the HO variant is the second-best generative model, reaching 91.4% for the `logP-&HBA+` task. On the stricter LipinskiFix-1000 benchmark, AutoLead's strong generalization is evident, with the BO variant achieving the second-best result of 94.7%, substantially higher than ChatDrug-GPT-4o 79.3%. This consistent high performance in generating chemically valid molecules underscores the reliability of the AutoLead framework for practical drug design applications.

Dynamic Usage of Bayesian Optimization in HO. To understand the contribution of Bayesian Optimization (BO) within our hybrid optimization (HO) strategy, we analyze the proportion of steps where BO was selected during the molecular editing process. The results, summarized in Table 9, reveal that HO dynamically adjusts its reliance on BO based on task difficulty. For simpler property edits like modifying `logP+` ($\Delta = 0$), where LLM-based exploration is often sufficient, the BO ratio remains very low (0.17%). In contrast, for tasks requiring more precise, data-driven exploitation, the reliance on BO increases substantially. This is evident in challenging single-objective tasks like `QED+` (56.19%) and complex multi-objective cases such as `logP+&HBD+` (56.62%). This trend culminates in the LipinskiFix-1000 benchmark, which involves satisfying multiple strict constraints, where BO usage reaches its peak at 64.8%. These findings validate our hybrid design, which effectively balances broad, LLM-driven exploration with focused, BO-driven exploitation according to the complexity of the optimization task.

Impact of BO Integration and LLM Choice. Figure 3 provides a comparative analysis of success rates, decomposing the impact of our hybrid optimization strategy (HO vs. LLMO) and the choice of LLM backbone (GPT-3.5-turbo vs. GPT-4o). The results clearly demonstrate that integrating BO yields substantial performance gains. The HO strategy consistently outperforms the LLM-only (LLMO) approach, with the improvement being most pronounced when using GPT-4o, where it boosts the success rate by approximately 3% (as shown in the right plot). Importantly, this significant increase in success comes without sacrificing the high valid rate of generated molecules, which remains above 85% for all variants.

Furthermore, the analysis reveals a crucial synergy between the hybrid method and a more advanced LLM. While upgrading from GPT-3.5 to GPT-4o provides only a modest gain for the LLMO strategy (+0.6%), this benefit is magnified under the HO strategy, yielding a much larger improvement (+2.1%). This indicates that the structured exploration from BO is most effective when paired with a powerful LLM. We hypothesize this is because a more capable model like GPT-4o serves as a more effective inverse mapping function, $\tilde{\phi}^{-1}$, when decoding optimized latent space vectors back into valid, high-quality molecular structures. The synergy between BO's intelligent search and the LLM's superior generative capabilities is central to the success of our hybrid approach.

More Visualization of Lead Optimization. To further illustrate the effectiveness of AutoLead, Figure 4 showcases examples of successful molecule refinements across the ChatDrug-200 and DrugAssist-500 datasets. Each pair displays the original molecule on the left and the optimized molecule on the right, along with key pharmacokinetic properties. Red highlights mark properties or structural regions in the original molecules that violated drug-likeness or optimization criteria, while green highlights indicate improvements or favorable modifications introduced by AutoLead. These examples reveal that AutoLead can perform targeted, chemically meaningful modifications—such as adding, deleting, or substituting functional groups—that enhance properties like

Table 9. Bayesian-Optimization (BO) selection ratio of the Heuristic Oracle (HO) across various benchmarks and tasks. The tasks are grouped by dataset and objective type.

Task / Property	Constraint (Δ)	HO BO-ratio (%)
ChatDrug-200 – Single-Objective Tasks		
logP-	0 0.5	3.17 11.16
logP+	0 0.5	0.17 2.42
QED+	0 0.1	33.81 56.19
QED-	0 0.1	8.67 35.01
tPSA-	0 10	35.16 43.46
tPSA+	0 10	0.00 0.00
HBA+	0 1	1.63 11.50
HBD+	0 1	1.58 12.31
ChatDrug-200 – Multi-Objective Tasks		
logP- & HBA+	0–0 0.5–1	2.54 12.86
logP+ & HBA+	0–0 0.5–1	30.33 37.79
logP- & HBD+	0–0 0.5–1	1.46 14.03
logP+ & HBD+	0–0 0.5–1	42.10 56.62
logP- & tPSA-	0–0 0.5–10	59.86 61.78
logP- & tPSA+	0–0 0.5–10	1.81 4.56
DrugAssist-500 Tasks		
logP-	0 0.5	4.80 14.29
QED+	0 0.1	30.51 52.88
HBA+	0 1	2.30 13.59
HBD+	0 1	3.50 13.93
logP- & HBA+	0 1	17.34 19.20
LipinskiFix-1000 Dataset		
Overall task		64.80

Table 10. Success rate (%) of AutoLead-HO with different BO latent dimensions (2, 5, 10). Δ denotes the required improvement threshold for each property.

Dataset / Task	Δ	BO-dim=2	BO-dim=5	BO-dim=10
ChatDrug-200				
logP-	0	86.7 ± 0.5	90.2 ± 0.6	87.9 ± 1.1
	0.5	80.3 ± 0.2	79.7 ± 1.0	78.5 ± 1.0
logP+	0	90.5 ± 0.4	90.2 ± 0.5	90.8 ± 0.8
	0.5	82.8 ± 0.8	83.2 ± 0.2	83.6 ± 1.9
QED+	0	45.2 ± 1.2	44.5 ± 1.8	59.0 ± 1.6
	0.1	5.2 ± 0.2	4.5 ± 0.4	9.6 ± 3.7
QED-	0	86.2 ± 1.4	86.7 ± 0.6	86.4 ± 0.6
	0.1	68.7 ± 0.2	67.0 ± 1.1	67.5 ± 0.9
tPSA-	0	39.8 ± 0.6	41.3 ± 1.6	40.9 ± 1.4
	10	28.5 ± 0.4	27.0 ± 1.5	28.2 ± 1.3
tPSA+	0	90.7 ± 0.2	91.2 ± 0.2	91.4 ± 0.4
	10	91.2 ± 0.2	91.5 ± 0.7	90.2 ± 0.8
HBA+	0	89.8 ± 1.2	88.2 ± 0.9	89.5 ± 0.4
	1	78.7 ± 1.7	76.2 ± 1.2	80.6 ± 0.7
HBD+	0	86.5 ± 0.4	86.0 ± 0.4	86.1 ± 0.6
	1	67.7 ± 1.6	65.7 ± 1.6	71.9 ± 0.7
logP- & HBA+	0–0	90.7 ± 0.2	89.0 ± 0.4	88.2 ± 0.4
	0.5–1	71.8 ± 1.6	68.5 ± 1.6	74.1 ± 1.5
logP+ & HBA+	0–0	37.2 ± 1.0	37.0 ± 2.3	33.6 ± 0.5
	0.5–1	8.0 ± 0.8	8.0 ± 0.5	18.9 ± 1.9
logP- & HBD+	0–0	91.0 ± 0.8	90.5 ± 0.7	92.0 ± 0.7
	0.5–1	71.5 ± 2.5	69.7 ± 2.6	76.4 ± 1.5
logP+ & HBD+	0–0	42.2 ± 1.4	42.8 ± 0.5	51.0 ± 1.6
	0.5–1	1.3 ± 0.8	1.5 ± 0.4	11.8 ± 3.6
logP- & tPSA-	0–0	5.0 ± 0.4	3.7 ± 0.6	8.4 ± 1.7
	0.5–10	0.5 ± 0.0	0.7 ± 0.2	1.6 ± 0.5
logP- & tPSA+	0–0	93.0 ± 0.4	92.2 ± 0.5	91.8 ± 1.1
	0.5–10	84.0 ± 0.4	83.2 ± 0.6	83.2 ± 0.4
Average	-	69.6 ± 28.1	69.5 ± 28.1	$\mathbf{71.2 \pm 26.5}$
	-	52.9 ± 34.0	51.9 ± 33.7	$\mathbf{55.4 \pm 31.8}$
DrugAssist-500				
logP-	0	86.7 ± 1.1	86.5 ± 0.4	87.7 ± 0.2
	0.5	76.4 ± 1.1	76.2 ± 0.5	75.2 ± 1.9
QED+	0	46.8 ± 0.7	46.3 ± 0.5	62.3 ± 1.4
	0.1	11.2 ± 0.3	10.5 ± 0.4	24.3 ± 0.8
HBA+	0	87.1 ± 0.1	87.0 ± 0.7	86.3 ± 0.3
	1	76.6 ± 0.2	76.1 ± 0.5	77.1 ± 1.1
HBD+	0	84.3 ± 0.6	85.3 ± 0.4	85.7 ± 0.3
	1	66.8 ± 0.0	68.3 ± 0.2	70.2 ± 0.3
logP- & HBA+	0	71.9 ± 0.8	71.6 ± 0.3	75.0 ± 0.3
	1	68.1 ± 1.1	68.5 ± 1.2	72.0 ± 0.5
Average	-	75.4 ± 15.3	75.3 ± 15.6	$\mathbf{79.4 \pm 9.7}$
	-	59.8 ± 24.6	59.9 ± 24.9	$\mathbf{63.8 \pm 19.9}$
LipinskiFix-1000				
Lipinski + QED ≥ 0.7		2.90 ± 0.10	2.30 ± 0.36	$\mathbf{28.90 \pm 1.66}$

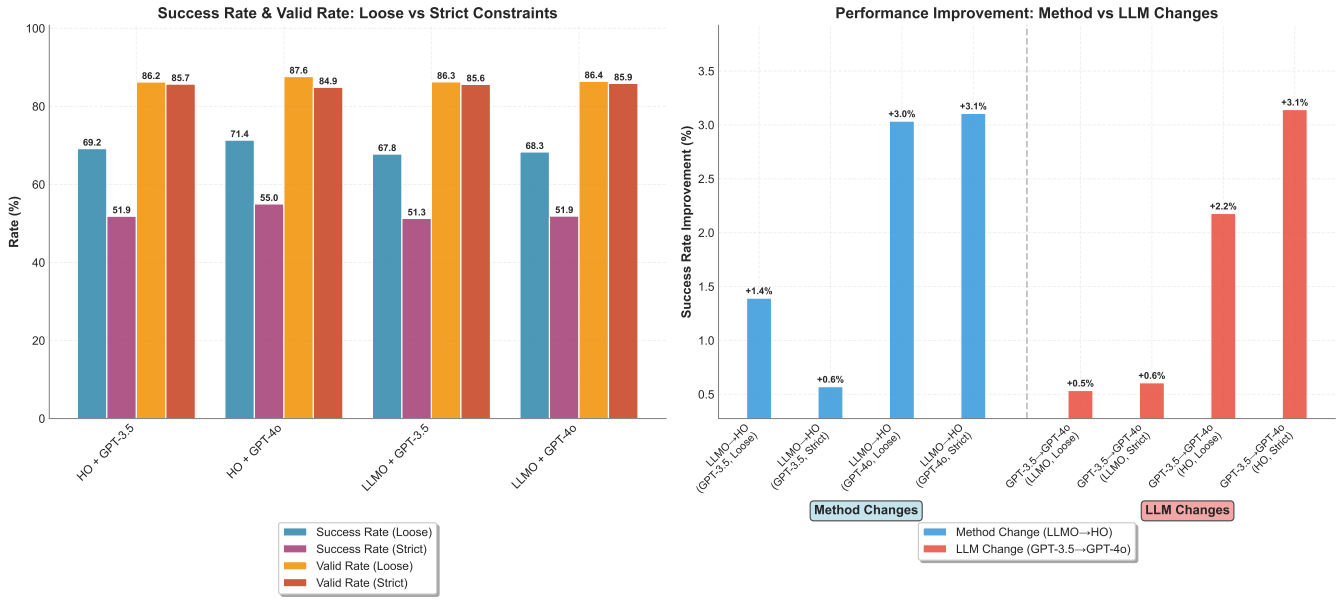

Fig. 3. Performance comparison on the ChatDrug-200 dataset. **Left:** Success rates and valid rates under both loose and strict constraint settings for different variants of HO and LLMO. **Right:** Success rate improvements attributable to method changes (LLMO → HO) versus LLM backbone changes (GPT-3.5 → GPT-4o). These results highlight the relative contribution of each factor to overall performance gains.

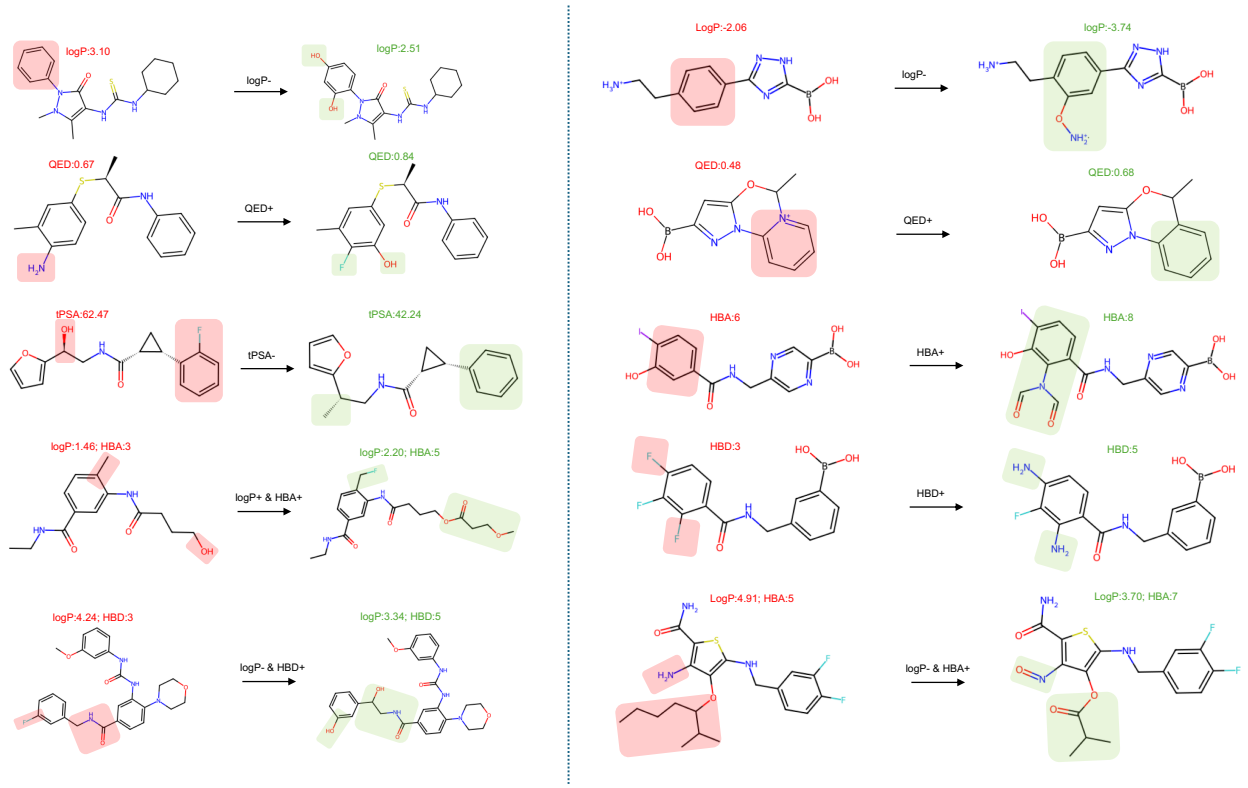

Fig. 4. Examples of molecule refinements generated by AutoLead across the ChatDrug-200 and DrugAssist-500 datasets. Each pair shows the original molecule (left) and the optimized molecule (right) with corresponding key property values. Red highlights indicate properties or structural regions in the initial molecule that violated optimization criteria, while green highlights show improvements or favorable modifications introduced by AutoLead.

logP, QED, tPSA, HBA, and HBD while preserving the core molecular scaffolds.

A Richer Feature Space is Crucial for Optimization Success. We also analyze our method, AutoLead-HO, by varying the dimensionality of the BO feature space to guide the molecular design process. The most comprehensive 10-dimension (10D) space included ten key molecular descriptors: Molecular Weight, LogP, Topological Polar Surface Area (TPSA), Hydrogen-Bond Donors, Hydrogen-Bond Acceptors, Rotatable Bonds, Ring Count, Aromatic Rings, Saturated Rings, and the fraction of sp^3 hybridized carbons. The 2-dimension and 5-dimension spaces were subsets, using only the first two and first five descriptors from this list, respectively. As shown in Table 10 While lower-dimensional models occasionally achieved a higher success rate on specific tasks (e.g., on the ChatDrug-200 dataset, the 5D model scored 90.2% on the `logP-` loose task, slightly surpassing the 10D model), the 10D model was consistently the top performer on average. Across all ChatDrug tasks, it achieved the highest average success rates for both loose (71.2%) and strict (55.4%) criteria, a trend that was mirrored on the DrugAssist-500 dataset. The advantage of a rich feature space was most dramatic on the LipinskiFix-1000 dataset, a significantly more challenging multi-objective benchmark. Here, the 10D model achieved a success rate of 28.9%, an order-of-magnitude improvement over the 2D (2.9%) and 5D (2.3%) models. This demonstrates that while a minimal feature set can be effective for simpler objectives, a comprehensive set of molecular descriptors is critical for navigating complex optimization landscapes and achieving robust performance.