ReMol: LLM-guided Molecular Optimization with Reinforcement Learning

Ziqing Wang
ziqingwang2029@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Kaize Ding
kaize.ding@u.northwestern.edu
Northwestern University
Evanston, Illinois, USA

Abstract

Effective lead molecule optimization is pivotal in drug discovery, requiring property enhancement while preserving structural similarity. While Large Language Models (LLMs) show promise, their application is often hindered by inefficiencies and substantial data requirements for specialized tasks. We introduce **ReMol**, an LLM-guided framework for efficient molecular optimization using Reinforcement Learning (RL). ReMol employs a two-stage post-training strategy: initial supervised fine-tuning on limited high-similarity molecule pairs equips the LLM for valid, local edits, followed by an RL phase that refines its ability to generate optimized molecules iteratively. This approach enables ReMol to achieve strong performance with significantly reduced training data. Notably, our RL stage is designed for efficiency, allowing the model to learn effective optimization policies with remarkably few oracle property evaluations during training. Our preliminary experiments show that ReMol can maintain high success rates even under scarce evaluation budgets, highlighting its practical efficiency and adaptability for real-world lead optimization.

CCS Concepts

• **Computing methodologies** → **Reinforcement learning**; *Neural networks*; • **Applied computing** → *Computational biology*; Chemical informatics.

Keywords

molecular optimization, reinforcement learning, large language models, drug discovery, computational chemistry

ACM Reference Format:

Ziqing Wang and Kaize Ding. 2018. ReMol: LLM-guided Molecular Optimization with Reinforcement Learning. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation email (Conference acronym 'XX)*. ACM, New York, NY, USA, 3 pages. https://doi.org/XXXXXXX.XXXXXXX

1 Introduction

The discovery and design of novel molecules are critical drivers of progress in fields ranging from pharmaceuticals to materials science [23]. However, identifying molecules with specific desired properties from the vast chemical space (up to 10^{60} molecules [21]) is inherently complex. Traditional methods, relying on iterative synthesis and experimentation [22], are notoriously slow and resource-intensive, particularly in drug discovery, where costs can exceed $1 billion per approved drug [28].

To accelerate this process, various computational techniques have been developed, including Bayesian Optimization [24], Reinforcement Learning (RL) [19], and other machine learning models [5, 7, 14]. While these methods show promise, a significant challenge, especially in lead optimization, is performing fine-grained modifications that simultaneously improve multiple properties while strictly adhering to structural similarity constraints. Many approaches struggle with controlled, local editing, often relying on generic rules or random chance [25], which limits their practical application and can lead to computational inefficiencies when navigating the extensive chemical search space [3].

Recent research has increasingly shifted towards deep generative models, such as Variational Autoencoders, Autoregressive models, and Diffusion models [6, 12], trained via supervised learning, RL, or contrastive learning strategies [17, 20, 30]. More recently, Large Language Models (LLMs) have emerged as powerful tools, possessing general chemical knowledge gleaned from vast text corpora [4, 27]. This has spurred their application in various chemistry tasks, including molecular property prediction and generation [9, 10, 16].

Despite their potential, directly applying general-purpose LLMs to the nuanced task of lead optimization presents significant hurdles. Standard LLMs are not inherently trained for the controlled, similarity-preserving edits crucial for lead refinement. Current LLM-based methods often depend on extensive prompt engineering [10, 18], struggle with precise numerical targets [1, 16], and can be inefficient when faced with expensive property evaluations. Our preliminary findings also indicate that standard RL techniques can suffer from reward hacking or become computationally prohibitive under realistic oracle budget constraints.

To address these limitations, we introduce **ReMol**, an LLM-guided framework for efficient molecular optimization using Reinforcement Learning. Our primary contributions are:

- **Data-Efficient Supervised Fine-Tuning (SFT):** We implement SFT as the first stage, leveraging a limited set of high-similarity molecule pairs. This grounds the LLM in domain-specific, structure-preserving local editing principles with high data efficiency.
- **Oracle-Efficient Reinforcement Learning (RL):** We design a multi-turn RL strategy for the second stage, where the LLM iteratively edits molecules. This allows learning of effective optimization policies with remarkably few oracle property evaluations during training.

Conference acronym 'XX, Woodstock, NY

ACM ISBN 978-1-4503-XXXX-X/2018/06
https://doi.org/XXXXXXX.XXXXXXX

- **Robust Performance:** We demonstrate that ReMol achieves state-of-the-art performance. It maintains high success rates under scarce evaluation budgets at inference.

2 Proposed Approach – ReMol

2.1 Problem Formulation

Lead optimization aims to refine an initial lead molecule $m \in \mathcal{M}$ into an improved molecule $m' \in \mathcal{M}$, where $\mathcal{M}$ is the set of valid molecules. A critical constraint is maintaining structural similarity, quantified by $\text{sim}(m, m') \geq \gamma$, where $\text{sim}(\cdot,\cdot)$ is a similarity function and γ a predefined threshold. Concurrently, we seek to improve n target properties, evaluated by functions $\{F_i(\cdot) : \mathcal{M} \to \mathbb{R}\}_{i=1}^{n}$. We consider two optimization objectives:

(1) **Constrained Optimization:** m' must satisfy $F_i(m') \geq \delta_i$ for all target properties i, where δ_i are predefined thresholds.
(2) **Unconstrained Optimization:** m' aims to maximize a weighted sum of properties, i.e., $m' = \text{argmax}_m \sum_{i=1}^{n} w_i F_i(m)$, with w_i as weighting coefficients.

Property evaluation functions F_i are treated as expensive black-box oracles lacking gradient information. Thus, optimization must adhere to a total oracle evaluation budget $B \in \mathbb{N}$.

2.2 Similarity-Based Supervised Fine-Tuning

Standard LLMs often fail to generate molecules with high structural similarity to a lead, yielding modifications comparable to random sampling [25]. For instance, edits by models like BioT5 can result in Tanimoto similarities as low as 0.173, while even GPT-4 shows wide variance (0.165 - 0.433), underscoring the need for specialized training.

To address this, the first stage of ReMol is SFT. We fine-tune a base chemistry-aware LLM on a curated dataset of high-similarity molecule pairs (e.g., Tanimoto similarity > 0.7 from PubChem [15]). This SFT phase explicitly trains the LLM on patterns of valid, local chemical modifications, equipping it with the foundational capability to propose edits that maintain structural fidelity. This stage grounds the LLM in domain-specific editing principles crucial for effective lead optimization.

2.3 Multi-Turn Reinforcement Learning

Building upon the SFT-initialized LLM, the second stage employs RL to further hone its optimization capabilities. Lead optimization is an iterative process, naturally framed as a finite-horizon Markov Decision Process (MDP): $\mathcal{M} = \langle \mathcal{S}, \mathcal{A}, P, R, \gamma_{\text{disc}} \rangle$.

Observation space ($\mathcal{S}$): Each state s_t is an intermediate, chemically valid SMILES string. An episode starts from the lead molecule m_0 and ends upon generating an `[EOS]` token or reaching T edits.

Action space ($\mathcal{A}$): At step t, the LLM agent selects a token a_t from the SMILES vocabulary $\mathcal{V}$ such that $s_t{+}a_t$ remains a syntactically valid SMILES prefix.

Transition dynamics (P): Appending a valid character deterministically extends the prefix: $P(s_{t+1} \mid s_t, a_t) = \mathbf{1}\big[s_{t+1} = s_t{+}a_t\big]$.

Reward function (R): Intermediate states receive zero reward. At termination (after T edits or `[EOS]`), the generated molecule m_T is scored: $R(s_T, a_T) = \sum_{i=1}^{n} w_i\, F_i(m_T) - \lambda\, \max\{0,\ \gamma_{\text{sim}} - \text{sim}(m_0, m_T)\}$, where F_i are property predictors, w_i are weights, $\text{sim}(m_0, m_T)$ is similarity to the lead, γ_{sim} is the similarity threshold, and λ is a penalty coefficient. We use a discount factor $\gamma_{\text{disc}} = 1$.

The goal is to learn a policy $\pi(a|s)$ that maximizes the expected terminal reward within the oracle evaluation budget B. Unlike single-turn RL, lead optimization benefits from *trajectory-level reasoning*. Our RL approach evaluates the full molecule after edit sequences and assigns rewards considering both local edit quality and overall trajectory success in achieving the optimization goals. This allows the LLM to plan multi-step edit sequences effectively, manage the evaluation budget, and adhere to similarity constraints. We employ a policy gradient method (e.g., PPO-style updates) to optimize the LLM.

Table 1: Success rate (%) for QED $\in [0.9, 1.0]$ with similarity sim ≥ 0.4 under evaluation budgets $B \in \{500, 1000\}$.

Method	Success Rate (%)	
	$B = 500$	$B = 1000$
Reinvent [8]	0.25	0.63
Graph-GA [29]	1.38	3.25
QMO [11]	15.88	19.62
RetMol [25]	42.38	63.25
ReMol-1.5B (Ours)	53.13	62.50
ReMol-3B (Ours)	**56.25**	**68.75**

3 Experiments

We evaluated ReMol on single-objective molecular optimization, focusing on Quantitative Estimate of Drug-likeness (QED) [2]. The objective was to improve QED scores while maintaining high structural similarity.

Experimental Setup: We used 800 molecules with QED scores in [0.7, 0.8] from ZINC250k [13] as leads. Following [26], for each lead m, we aimed to generate m' such that: (1) $F_{\text{QED}}(m') \geq 0.9$, and (2) Tanimoto similarity $\text{sim}(m', m) \geq 0.4$. Evaluations were run with budgets $B \in \{500, 1000\}$ oracle queries. We compared against QMO [11].

Evaluation Metric and Results: We used Success Rate: the percentage of leads for which a valid molecule satisfying both QED and similarity constraints was found within budget. As shown in Table 1, ReMol significantly outperformed QMO. With $B = 500$, ReMol achieved a 53.13% success rate, increasing to 62.50% with $B = 1000$. This is a 42.88 percentage point improvement over QMO's 19.62% at $B = 1000$. These results highlight ReMol's ability to efficiently optimize molecules under strict constraints.

4 Conclusion

We introduced ReMol, a two-stage post-training framework (SFT + RL) that equips LLMs for effective lead molecule optimization. By first grounding the LLM in similarity-preserving local edits via SFT, and then refining its ability to perform iterative, goal-directed optimization via multi-turn RL, ReMol addresses key limitations of applying general LLMs to this domain. Our experiments show that ReMol significantly improves success rates in optimizing molecular properties while adhering to structural similarity and evaluation budget constraints. This approach offers a promising direction for leveraging LLMs in complex, constrained molecular design tasks.

References

[1] Microsoft Research AI4Science and Microsoft Azure Quantum. 2023. The impact of large language models on scientific discovery: a preliminary study using gpt-4. *arXiv preprint arXiv:2311.07361* (2023).

[2] G. Richard Bickerton, Gaia V. Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L. Hopkins. 2012. Quantifying the chemical beauty of drugs. *Nature Chemistry* 4, 2 (Jan. 2012), 90–98. doi:10.1038/nchem.1243

[3] R S Bohacek, C McMartin, and W C Guida. 1996. The art and practice of structure-based drug design: a molecular modeling perspective. *Med. Res. Rev.* 16, 1 (Jan. 1996), 3–50.

[4] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[5] Yuanqi Du, Tianfan Fu, Jimeng Sun, and Shengchao Liu. 2022. Molgensurvey: A systematic survey in machine learning models for molecule design. *arXiv preprint arXiv:2203.14500* (2022).

[6] Yuanqi Du, Arian R Jamasb, Jeff Guo, Tianfan Fu, Charles Harris, Yingheng Wang, Chenru Duan, Pietro Liò, Philippe Schwaller, and Tom L Blundell. 2024. Machine learning-aided generative molecular design. *Nat. Mach. Intell.* 6, 6 (June 2024), 589–604.

[7] Daniel C Elton, Zois Boukouvalas, Mark D Fuge, and Peter W Chung. 2019. Deep learning for molecular design—a review of the state of the art. *Molecular Systems Design & Engineering* 4, 4 (2019), 828–849.

[8] Fartash Faghri, Hadi Pouransari, Sachin Mehta, Mehrdad Farajtabar, Ali Farhadi, Mohammad Rastegari, and Oncel Tuzel. 2023. Reinforce Data, Multiply Impact: Improved Model Accuracy and Robustness with Dataset Reinforcement. *arXiv preprint arXiv:2303.08983* (2023).

[9] Daniel Flam-Shepherd and Alán Aspuru-Guzik. 2023. Language models can generate molecules, materials, and protein binding sites directly in three dimensions as xyz, cif, and pdb files. *arXiv preprint arXiv:2305.05708* (2023).

[10] Taicheng Guo, Bozhao Nan, Zhenwen Liang, Zhichun Guo, Nitesh Chawla, Olaf Wiest, Xiangliang Zhang, et al. 2023. What can large language models do in chemistry? a comprehensive benchmark on eight tasks. *Advances in Neural Information Processing Systems* 36 (2023), 59662–59688.

[11] Samuel C Hoffman, Vijil Chenthamarakshan, Kahini Wadhawan, Pin-Yu Chen, and Payel Das. 2022. Optimizing molecules using efficient queries from property evaluations. *Nature Machine Intelligence* 4, 1 (2022), 21–31.

[12] Emiel Hoogeboom, Victor Garcia Satorras, Clément Vignac, and Max Welling. 2022. Equivariant Diffusion for Molecule Generation in 3D. arXiv:2203.17003 [cs.LG] https://arxiv.org/abs/2203.17003

[13] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. 2018. Junction Tree Variational Autoencoder for Molecular Graph Generation. In *Proceedings of the 35th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer Dy and Andreas Krause (Eds.). PMLR, 2323–2332. https://proceedings.mlr.press/v80/jin18a.html

[14] Wengong Jin, Kevin Yang, Regina Barzilay, and Tommi Jaakkola. 2018. Learning multimodal graph-to-graph translation for molecular optimization. *arXiv preprint arXiv:1812.01070* (2018).

[15] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, Leonid Zaslavsky, Jian Zhang, and Evan E Bolton. 2024. PubChem 2025 update. *Nucleic Acids Research* 53, D1 (Nov. 2024), D1516–D1525. doi:10.1093/nar/gkae1059

[16] Agustinus Kristiadi, Felix Strieth-Kalthoff, Marta Skreta, Pascal Poupart, Alán Aspuru-Guzik, and Geoff Pleiss. 2024. A sober look at LLMs for material discovery: are they actually good for bayesian optimization over molecules?. In *Proceedings of the 41st International Conference on Machine Learning*. 25603–25622.

[17] Jaechang Lim, Seongok Ryu, Jin Woo Kim, and Woo Youn Kim. 2018. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *J. Cheminform.* 10, 1 (July 2018), 31.

[18] Tung Nguyen and Aditya Grover. 2024. Lico: Large language models for in-context molecular optimization. *arXiv preprint arXiv:2406.18851* (2024).

[19] Marcus Olivecrona, Thomas Blaschke, Ola Engkvist, and Hongming Chen. 2017. Molecular de-novo design through deep reinforcement learning. *Journal of cheminformatics* 9 (2017), 1–14.

[20] Yang Jeong Park, Hyungi Kim, Jeonghee Jo, and Sungroh Yoon. 2023. Deep contrastive learning of molecular conformation for efficient property prediction. *Nat. Comput. Sci.* 3, 12 (Dec. 2023), 1015–1022.

[21] P G Polishchuk, T I Madzhidov, and A Varnek. 2013. Estimation of the size of drug-like chemical space based on GDB-17 data. *J. Comput. Aided Mol. Des.* 27, 8 (Aug. 2013), 675–679.

[22] Jonathan M Stokes, Kevin Yang, Kyle Swanson, Wengong Jin, Andres Cubillos-Ruiz, Nina M Donghia, Craig R MacNair, Shawn French, Lindsey A Carfrae, Zohar Bloom-Ackermann, et al. 2020. A deep learning approach to antibiotic discovery. *Cell* 180, 4 (2020), 688–702.

[23] Gary Tom, Stefan P Schmid, Sterling G Baird, Yang Cao, Kourosh Darvish, Han Hao, Stanley Lo, Sergio Pablo-García, Ella M Rajaonson, Marta Skreta, et al. 2024. Self-driving laboratories for chemistry and materials science. *Chemical Reviews* 124, 16 (2024), 9633–9732.

[24] Austin Tripp, Gregor NC Simm, and José Miguel Hernández-Lobato. 2021. A fresh look at de novo molecular design benchmarks. In *NeurIPS 2021 AI for Science Workshop*.

[25] Haorui Wang, Marta Skreta, Yuanqi Du, Wenhao Gao, Lingkai Kong, Cher Tian Ser, Felix Strieth-Kalthoff, Chenru Duan, Yuchen Zhuang, Yue Yu, et al. 2024. Efficient Evolutionary Search over Chemical Space with Large Language Models. In *ICML 2024 AI for Science Workshop*.

[26] Zichao Wang, Weili Nie, Zhuoran Qiao, Chaowei Xiao, Richard Baraniuk, and Anima Anandkumar. 2023. Retrieval-based Controllable Molecule Generation. In *International conference on learning representations (ICLR)*.

[27] Andrew D White. 2023. The future of chemistry is language. *Nature Reviews Chemistry* 7, 7 (2023), 457–458.

[28] Olivier J Wouters, Martin McKee, and Jeroen Luyten. 2020. Estimated research and development investment needed to bring a new medicine to market, 2009-2018. *JAMA* 323, 9 (March 2020), 844–853.

[29] Jiaxuan You, Bowen Liu, Zhitao Ying, Vijay Pande, and Jure Leskovec. 2018. Graph convolutional policy network for goal-directed molecular graph generation. *Advances in neural information processing systems* 31 (2018).

[30] Zhenpeng Zhou, Steven Kearnes, Li Li, Richard N Zare, and Patrick Riley. 2019. Optimization of molecules via deep reinforcement learning. *Sci. Rep.* 9, 1 (July 2019), 10752.