Hit and Lead Discovery with Explorative RL and Fragment-based Molecule Generation

Anonymous Author(s)
Affiliation
Address
email

Abstract

Recently, utilizing reinforcement learning (RL) to generate molecules with desired properties has been highlighted as a promising strategy for drug design. Molecular docking program – a physical simulation that estimates protein-small molecule binding affinity – can be an ideal reward scoring function for RL, as it is a straightforward proxy of the therapeutic potential. Still, two imminent challenges exist for this task. First, the models often fail to generate chemically realistic and pharmacochemically acceptable molecules. Second, the docking score optimization is a difficult exploration problem that involves many local optima and less smooth surface with respect to molecular structure. To tackle these challenges, we propose a novel RL framework that generates pharmacochemically acceptable molecules with large docking scores. Our method – Fragment-based generative RL with Explorative Experience replay for Drug design (FREED) – constrains the generated molecules to a realistic and qualified chemical space and effectively explores the space to find drugs by coupling our fragment-based generation method and a novel error-prioritized experience replay (PER). We also show that our model performs well on both *de novo* and scaffold-based schemes. Our model produces molecules of higher quality compared to existing methods while achieving state-of-the-art performance on two of three targets in terms of the docking scores of the generated molecules. We further show with ablation studies that our method, predictive error-PER (FREED(PE)), significantly improves the model performance.

1 Introduction

Searching for “hits", the molecules with desired therapeutic potentials, is a critical task in drug discovery. Instead of screening a library of countless potential candidates in a brute-force manner, designing drugs with sample-efficient generative models has been highlighted as a promising strategy. While many generative models for drug design are trained on the distribution of known active compounds [1; 2; 3], such models tend to produce molecules that are highly similar to that of the training dataset [4], which discourages finding novel molecules.

In this light, reinforcement learning (RL) have been increasingly used for goal-directed molecular design, thanks to its exploration ability. Previous models have been assessed by using relatively simple objectives, such as cLogP and QED, or bioactivity scores predicted by auxiliary models.[5; 6**?** ; 7] However, it is inappropriate to use those scores as RL objectives for drug design, since high score in those simple physicochemical properties guarantee neither drug-likeness nor therapeutic potential, emphasizing the necessity of more relevant design objective in generative tasks [8; 9]. Molecular docking method, which is a physical simulation that estimates the protein-small molecule binding affinity – a key measure of drug efficacy – has been widely used to evaluate drug binding affinity.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

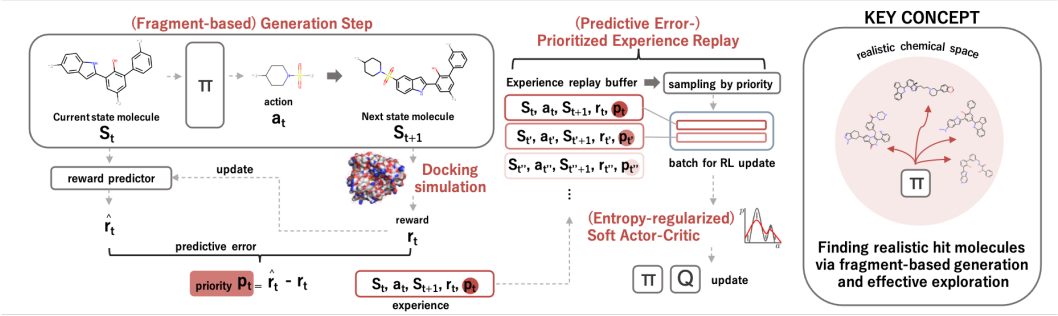

Figure 1: **Overview of our generative drug discovery method**. To find realistic 'hit' molecules which have high docking scores, we combine fragment-based generation method with SAC and PER. This figure illustrates our version of PER where the priority of the experience is defined by predictive error.

As docking simulations are more straightforward proxy of therapeutic potential, coupling RL with docking simulations would be a promising strategy.

While the simplistic scores (e.g., cLogP, QED) are computed by a sum of local fragments' score and are not a function of global molecular structure, making the generative tasks relatively simple, docking score optimization is a more demanding exploration problem for RL agents. A change in docking score is nonlinear with the molecule's local structural changes, and a large variance exists among the structures of high-scoring molecules, meaning that there exists many local optima [8].

In addition, many previous RL models often suffer from generating unreal and inappropriate structures. Docking score maximization alone is not sufficient to qualify molecules as drug candidates, as the molecules should satisfy strong structural constraints such as chemical realisticness and pharmacochemical suitability (e.g., less toxicity or reactivity). The low quality of generated molecules from such models is because a single improper addition of atoms and bonds can easily deteriorate the entire sanity of structures. In such cases, implicit guidance for molecules' quality such as jointly training the model with related design objectives (e.g., QED) and pretraining the models on druglike molecules might not completely prevent the model from producing those mistakes. Thus, generating molecules as a combination of appropriate fragments is considered as a promising solution [1; 10; 11] to explicitly restrict the generation space within realistic and qualified molecules. However, such a constraint in generation space would make the optimization problem of docking score even harder, urging a need of better exploration for RL agent.

In this respect, we introduce a new framework, **Fragment-based generative RL with Explorative Experience replay for Drug design (FREED)**, which encourages effective exploration while only allowing the generation of qualified molecules. Our model generates molecules by attaching a chemically realistic and pharmacochemically acceptable fragment unit on a given state of molecules at each step. Also, we enforce the model to form a new bond only on the attachment sites that are considered as appropriate at the fragment library preparation step. These strategies enable us to utilize chemistry prior knowledge and successfully constrain the molecule generation within the chemical space eligible for drug design. We also explore several explorative algorithms based on curiosity-driven learning and prioritized experience replay (PER) [12]. We devise an effective novel PER method which defines priority by the predictive error or uncertainty of the auxiliary reward predictor's outcome, in order to avoid the lack of robustness of previous methods. We provide an overall illustration of our framework in Figure 1.

Our main contributions can be summarized as follows:

- We propose a novel RL framework that can be readily utilized to design qualified molecules of high therapeutic potential.
- Our fragment-based generation method including connectivity-preserving fragmentation and augmentation allows our model to leverage Chemical prior knowledge.
- We propose novel explorative algorithms based on PER and show that they significantly improve the model performance.

2 Related Works

Docking as a reward function of RL. Studies on docking score optimization task has started very recently. Jeon et al.[13] developed MORLD, a atom-based generative model guided by MolDQN algorithm [14]. Olivecrona et al.[9] and Thomas et al.[15] utilized REINVENT[16], a simplified molecular-input line-entry system (SMILES)-based model generative model guided by improved REINFORCE algorithm[17] to generate hit molecules.

SMILES-based and atom-based generation methods. SMILES-based methods are infeasible for scaffold-based tasks[1] since molecular structures can substantially change through the sequential extension of the SMILES strings. Also, as explained in the Introduction, atom-based generation methods inherently suffer from unrealistic generated molecules. Thus, we focus our discussion on motif-based generation methods.

Motif-based molecular generation methods. Our work is the first application of RL with fragment-based molecular generation method. However, a number of previous works [1; 10; 11] have investigated similar motif-based molecule generation based on the variational autoencoders (VAE). JT-VAE and HierVAE [1; 10] decompose and reconstruct molecules into tree structure of motifs. These models might not be compatible for scaffold-based generations, since a latent vector from their encoders depends on a motif-wise decomposition order, which is an irrelevant information for docking score that may bias subsequent generation[11]. Maziarz et al.[11] also propose a VAE-based model which encodes a given scaffold with graph neural networks (GNNs) and decodes the motif-adding actions to produce an extended molecule. While Maziarz et al.'s motif-adding actions resemble our generation steps, we additionally introduce connectivity-preserving fragmentation and augmentation procedure which helps our model to generate molecules of better quality.

Coupling RL policy network with fragment-based generation holds general advantages compared to VAE-based methods. For example, our framework does not need to be trained on reconstruction task which might restrict the diversity of generated molecules.

Since generating synthesizable molecules is an important topic of molecular generative model research, some works attempt to generate molecules via forward synthesis actions. Gottipati et al. [18] and Bradshaw et al. [19] augment the current state molecule by choosing a building block and a reaction to apply. However, such forward synthesis-based methods do not allow their models to decide where to add next building blocks on the current molecular state. For docking score optimization, it would be important to let the model choose the relative location of each fragment in a molecule, since docking score is not a mere linear combination of fragment-contributions but a result of 3-dimensional interactions with the surrounding. We make our model compatible to docking score optimization task by allowing the model to choose where to attach a new fragment.

RL algorithms on hard-exploration problem. We view that there are two main approaches to achieve efficient exploration on a high level. First one is to introduce the "curiosity" or exploration bonus as intrinsic reward [20; 21; 22] for loss optimization. Bellemare et al.[23] proposed pseudo-count based exploration bonus as intrinsic reward, and Pathak et al. [24] defined curiosity as a distance between a neural network output given state-action and the next state. However, those "curiosity-driven", intrinsic reward-based models still fail to fully solve problems like Montezuma's Revenge[25], and the failures are explained as result of detachment and derailment in Ecoffet et al.[26]. The other approach of solving hard exploration problem is an efficient use of experiences[27]. Oh et al.[28] took experiences only where the actual reward is larger than agent's estimated value, and performed prioritized sampling based on the difference between the two values. However, such approach is cannot directly be applied to action-value based off-policy RL[29]. Moreover, prioritized sampling based on agent's value estimate (i.e., value function or Q function) is susceptible to outliers and spikes, which may lead to destabilizing the agent itself[30].

3 Methods

3.1 Generation method

Generation steps. The key concept of our method is to generate high quality molecules that binds well to a given target protein. In order to achieve this goal, we devise a fragment-based

[1] A drug design scheme where the drug candidates are designed from the given scaffold.

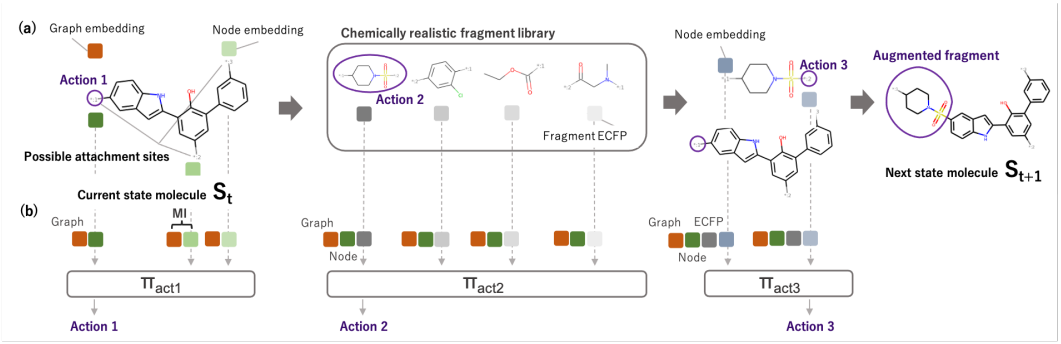

Figure 2: **Overview of our generation method (a) and policy network (b)**. Small colored squares represent a graph embedding of molecule, node embedding of each attachment site, and ECFP representation of each fragment. The graph or node embeddings, and ECFP representations from the chosen actions are autoregressively passed onto next action's policy network.

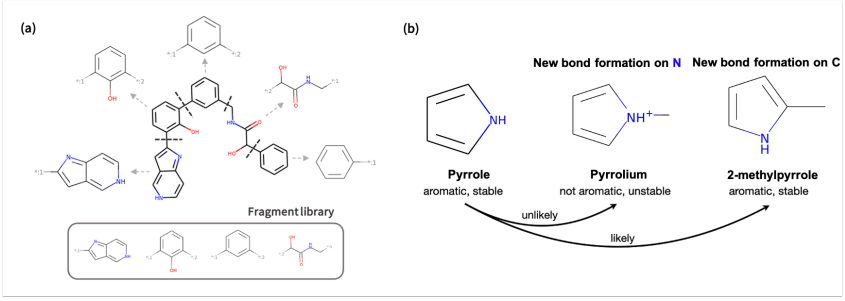

Figure 3: **Advantages of connectivity-preserving fragmentation. (a)** Substructure connectivity information is preserved as attachment sites in the fragmentation procedure. **(b)** Attaching a new bond to the nitrogen of a pyrrole ring would result in unstable molecule, breaking the aromaticity. Thus, pyrrole rings within an existing molecule are connected to other fragments by its carbon atoms, and preserving such an information would help us construct chemically realistic molecules.

generation method in which molecules are designed as an assemble of chemically realistic and pharmacochemically acceptable fragments. Given a molecular state, our model chooses "where to attach a new fragment", "what fragment to attach", and "where on a new fragment to form a new chemical bond". Note that the action "where to form a new bond" makes our model compatible to docking score optimization since the scores depend on the three-dimensional arrangement of molecular substructures.

Preserving fragment connectivity information in molecule generation. Another important feature of our method is harnessing the predefined connectivity information of the fragments and initial molecules. This feature allows our model to leverage chemists' expert knowledge in several ways. The connectivity information arises in fragmentation procedure, in which we follows CReM [31], when the algorithm decomposes any arbitrary molecules into fragments while preserving the fragments' attachment sites as shown in Figure 3**(a)**. Start and end points of new bonds are restricted to the attachment sites of given fragments and molecules. This restriction contributes to the chemical realisticness since the stability of the molecule depends on where the fragment is attached on as illustrated in Figure 3**(b)**. Also, we can utilize our prior knowledge of protein-ligand interaction by rationally assigning the attachment sites of the initial molecule (scaffold), which have been widely harnessed by medicinal chemists in usual scaffold-based scenario.

Policy network. Our generation method is coupled with the policy networks which guide the model to generate hit molecules. We provide an overall illustration of our policy network in Figure 2. Our state embedding network and policy network are designed as Markov models so that generation steps can take any arbitrary molecule as a current state and predict a next state, making the model more plausible for scaffold-based generation.

In our framework, each state vector s_t represents a molecular structure at the t-th step, and each step is a sequence of three actions that defines an augmentation of a fragment. A current state molecule is represented as an undirected graph G following Hwang et al. [32], which consists of a node feature matrix, an adjacency matrix, and an attachment site masking vector. A graph convolutional network (GCN) [33]-based network provides node embeddings, H, which are then sum-aggregated to produce a graph embedding, h_g. The sum-aggregation is a multi-set information for the entire graph, which is required for the docking score optimization task. Our policy network is an autoregressive process where Action 3 depends on Action 1 & 2 and Action 2 depends on Action 1.

$$p^{\text{act1}} = \pi_{\text{act1}}(Z^{\text{1st}}), Z^{\text{1st}} = \text{MI}(h_g, H_{\text{att}}) \tag{1}$$

$$p^{\text{act2}} = \pi_{\text{act2}}(Z^{\text{2nd}}), Z^{\text{2nd}} = \text{MI}(z^{\text{1st}}_{\text{act1}}, \text{ECFP}(h_{g_{\text{cand}}})) \tag{2}$$

$$p^{\text{act3}} = \pi_{\text{act3}}(Z^{\text{3rd}}), Z^{\text{3rd}} = \text{MI}(z^{\text{2nd}}_{\text{act2}}, U_{\text{cand}}) \tag{3}$$

where H_{att} refers to the node embeddings of attachment sites. For the first step of our policy network, π_{act1} takes the multiplicative interaction (MI) [34] of the node embedding of each attachment site and the graph embedding of the molecule, Z^{1st}, as inputs and predicts which attachment site should be chosen as Action 1. Since graph embeddings and node embeddings are defined in a heterogeneous space, we apply MI to fuse those two information. Given Action 1, π_{act2} takes the MI of $z^{\text{1st}}_{\text{act1}}$, (act1)-th vector of Z^{1st}, and the RDKit [35] ECFP (Extended Connectivity Fingerprint) representation of each candidate fragment, $h_{g_{\text{cand}}}$ as inputs, and predicts which fragment should be chosen as Action 2. By taking Z^{1st} as one of the inputs, π_{act2} reflects on the context of the current state and Action 1. Finally, given Action 1 and 2, π_{act3} takes the MI of $z^{\text{2nd}}_{\text{act2}}$ and the node embeddings of the chosen fragment's attachment sites, U_{cand}, as inputs and predicts which fragment attachment site should be chosen as Action 3. Each of the three policy networks $\pi_{\text{act1}}, \pi_{\text{act2}}$ and π_{act3} consists of three fully-connected layers with ReLU activations followed by a softmax layer to predict the probability of each action. In order to make gradient flow possible while sampling discrete actions from the probabilities, we implement the gumbel-softmax reparameterization trick[36].

3.2 Explorative RL for the discovery of novel molecules

Soft actor-critic. We employ *soft actor-critic* (SAC), an off-policy actor-critic RL algorithm based on maximum entropy reinforcement learning [37; 38], which is known to explore the space of state-action pairs more effectively than predecessors. Assuming that our docking score optimization task requires more effective exploration, we chose SAC as our RL baseline algorithm.

$$\pi^* = \arg\max_{\pi} \sum_t \mathbb{E}_{(s_t,a_t)\sim\rho_\pi}[r(s_t,a_t) + \alpha\mathcal{H}(\pi(\cdot|s_t))] \tag{4}$$

$$\begin{aligned}\mathbb{E}_{s_t\sim\mathcal{D}}[\alpha\log\pi(a|s)] = \sum[&\alpha\pi(a_t^{\text{act1}}, a_t^{\text{act2}}, a_t^{\text{act3}}|s_t) \\ &\times(\log\pi(a_t^{\text{act1}}|s_t) + \log\pi(a_t^{\text{act2}}|s_t, a_t^{\text{act1}}) + \log\pi(a_t^{\text{act3}}|s_t, a_t^{\text{act1}}, a_t^{\text{act2}}))]\end{aligned} \tag{5}$$

SAC aims to attain optimal policy such that satisfies (4) where α is the temperature parameter balancing between exploration and exploitation of the agent, $\mathcal{H}(\pi(\cdot|s_t))$ is entropy of action probabilities given s_t, and ρ_π is state-action transition distributions created from π. As we define act1, act2 and act3 autoregressively, the entropy regularization term defined in [39] $\mathbb{E}_{s_t\sim\mathcal{D}}[\alpha\log\pi(a|s)]$ is decomposed into (5).

Explorative algorithms. With the limitations of curiosity-driven approaches in hard-exploration problems[26] and the instability of agent's value estimate as a priority predictor of PER[30] considered, we use prioritized sampling-based approach with auxiliary priority predictor network to solve hard exploration problem. Specifically, we define the priority of an experience in the replay buffer as the predictive error or Bayesian uncertainty of output from the priority predictor. While PER with TD error from agent's estimated value was introduced in *Schaul et al.*[12], we are the first to use the other two new priority estimate functions with PER. In the experiments section, we show that PER with our priority estimate functions perform better than the previous methods.

1) Prediction error and Bayesian uncertainty We train an auxiliary neural network that predicts a reward of a given state. Then, we use the predictive error of the reward predictor as a priority of the experience. Such an use of predictive error has been introduced in curiosity-driven work [40], but our work is the first to apply this to PER.

Another way of measuring the novelty of a state is to use Bayesian uncertainty. Following Kendall et al.[41], we train an auxiliary neural network with MC dropout[42] that predicts the mean and variance of a reward of a given state. We use predicted standard deviation as priority of the experience.

2) Structure and procedure We define an auxiliary neural network module to estimate priority of a given transition which is a by-product from reward prediction of a given transition. This priority predictor consists of a graph encoder and fully-connected layers that estimate docking score for a given state. Following Thiede et al. [40], we define priority estimator in the state space, not in state-action space. In case of using Bayesian uncertainty as priority, every layers in the network are defined as MC dropout layers, and the network estimates mean and variance of the reward.

For each step, priority estimator outputs priority of given transition. In case of using predictive error as priority, we use Q value estimated by agent for given transition instead of docking score if it is not the end of episode, since docking scores are not provided every step. Priority estimator is optimized for every docking step, where we only optimize the estimator based on transitions that have docking score. When updating RL model, transitions are sampled with priority defined respectively. After updating policy and Q-function networks with prioritized sampling and importance sampling, priority value of sampled transitions are recalculated and updated to replay buffer.

4 Results and Analysis

4.1 Quantitative metrics

In this section, we introduce quantitative metric scores we used to assess our model. For every metric, we repeated every experiment five times with five different random seeds and report the mean and the standard deviation of the scores. Also, we calculated the scores when 3,000 molecules are generated and used to update the model during training.

Quality score. We report three widely used pharmacochemical filter scores – Glaxo[43], SureChEMBL[44], PAINS[45] – as quality scores. The quality scores are defined as a ratio of accepted, valid molecules to total generated molecules, as the filters reject the compounds that contain functional groups inappropriate for drugs (i.e., toxic, reactive groups). The higher the quality scores, the higher the probability of the molecule will be an acceptable drug. We also report the ratio of valid molecules to total generated molecules (validity) and the ratio of unique molecules among valid generated molecules (uniqueness).

Hit ratio. We report the hit ratio to compare the model's ability to produce as many unique hit molecules in a given length of iterations, where we define *Hits* as molecules whose docking scores are greater than the median of known active molecules' docking scores. We define hit ratio as a ratio of unique hit molecules to total generated molecules.

Top 5% score. We report the average score of top 5%-scored generated molecules to compare the model's ability to produce molecules with better docking scores.

4.2 Quantitative performance benchmark

In this section, we compare quality of the generated molecules and the model performance with two baseline models, MORLD and REINVENT, which are the models utilized for docking score optimization task in previous works [13; 9; 15]. We trained the models with respect to three carefully chosen protein targets fa7, parp1, and 5ht1b. The choice of protein targets and the specifics of experimental settings are described in Appendix. For the experiments in this section, fragment library (n=66) was constructed with pharmacochemically acceptable fragments.

Quality scores of generated molecules. In real-world drug design, the generated molecules should be acceptable by pharmacochemical criteria. We gave our generated molecules the pharmacochemical eligibility as we excluded from the fragment library the functional groups that are considered inappropriate by pharmacochemical filters. Such an explicit constraint is not applicable to atom-based (MORLD) or SMILES-based (REINVENT) methods. As shown in Table 1, our model generated mostly acceptable molecules while the other models showed poor rate of acceptable molecules,

Table 1: Quality scores of our model FREED(PE) – our framework with predictive error-PER, REINVENT, and MORLD. We trained our model and two baseline models with the target fa7, and computed quality scores of the first 3,000 molecules generated during training for each model. Standard deviation is given in brackets.

	Glaxo	SureChEMBL	PAINS	validity	uniqueness
MORLD	0.561 (.009)	0.131 (.013)	0.805 (.013)	**1.000** (.000)	**1.000** (.000)
REINVENT	0.773 (.023)	0.667 (.030)	0.769 (.022)	0.813 (.024)	0.988 (.008)
Ours: FREED(PE)	**0.996** (.001)	**0.808** (.049)	**0.991** (.002)	**1.000** (.000)	0.723 (.135)

confirming our approach's advantage in drug design. Such a trend was consistent for all three targets. See Appendix for visual inspection of the quality score results.

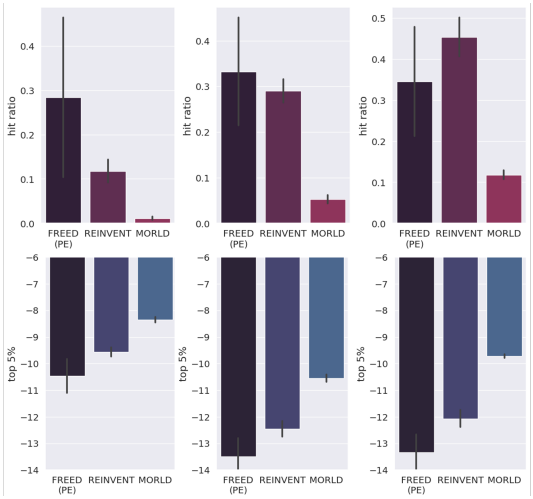

Figure 4: Hit ratio and top 5% score of our model FREED(PE), REINVENT, and MORLD. Standard deviation given as error bars. Larger the absolute value of docking score, higher the therapeutic potential of the molecule.

We assume that the low uniqueness of our model is due to the small size of fragment library (n=66), because when our model was tested with a larger library (n=91) which didn't exclude the inappropriate fragments, the uniqueness was 0.914 (.162) with similar hit ratio and top 5% scores.

Docking score optimization result. Figure 4 shows the hit ratios and top 5% scores of the generative models. Our model outperforms the other generative models MORLD and REINVENT in terms of both hit ratio and top 5% score, except for the hit ratio of the 5ht1b case. Such results show that our model's performance is superior or at least competitive to existing baselines, while our model exhibits many more practical advantages such as generating acceptable molecules, integrating chemist's expert knowledge by connectivity-preserving fragmentation and augmentation, and the feasibility in both *de novo* and scaffold-based generation.

4.3 Ablation studies: explorative algorithms

We also perform ablation studies of our algorithm to investigate the effect of our explorative algorithms. For this experiment, we used the larger fragment library (n=91) which includes some pharmacochemically inappropriate fragments. In Figure 5, we observe that all SAC models with explorative algorithm performed better compared to vanilla SAC model, while the PPO model showed the worst performance. FREED with both PE and BU outperformed curiosity driven models with PE and BU, showing effectiveness of our methods in our task compared to curiosity-driven explorations. Moreover, our predictive error-PER method outperformed TD error-PER method, where we conjecture that leveraging an auxiliary priority predictor network makes PER training more robust than using internal value estimate functions (Q function).

4.4 Case study on drug design

In this section, we show the practicality of our framework on *de novo* and scaffold-based drug design. We test FREED(PE), our framework with predictive error-PER, with our large fragment library (n=91).

***De novo* scenario.** Figure 6 **(a)** shows the distribution of the generated molecules before ("random") and after ("FREED(PE)") training the policy network. Our model was able to effectively generate molecules that have higher docking scores compared to the known active molecules. Figure 7 **(i)**

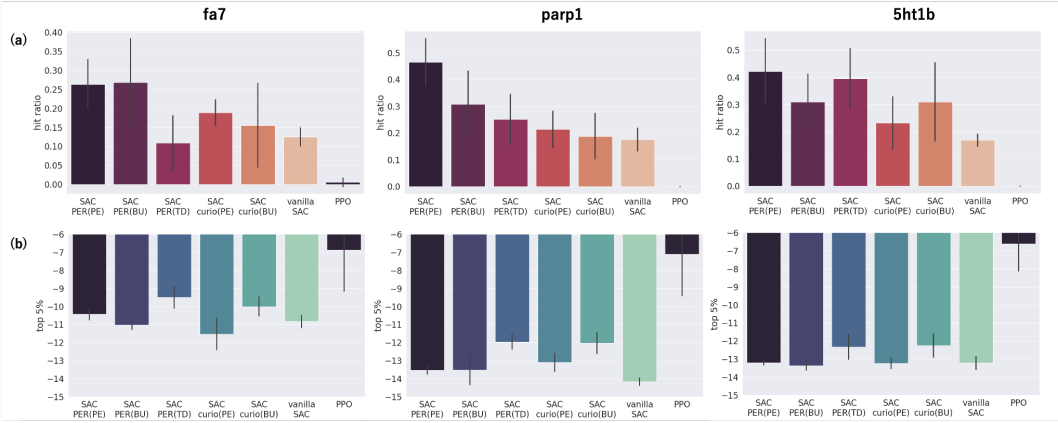

Figure 5: **Hit ratio and top 5% score of ablation studies.** Models can be categorized into whether using {PER, curiosity driven exploration(curio)}, and whether using {predictive error from predictor(PE), Bayesian uncertainty(BE), and TD error from agent(TD)} as priority or intrinsic reward, where FREED uses PER and PE or BU. Standard deviation given as error bars.

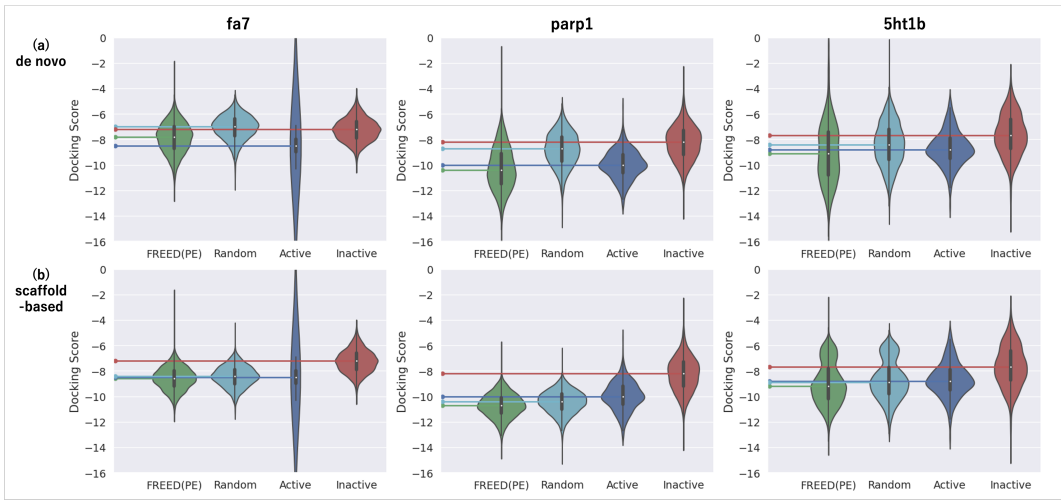

Figure 6: **Docking score distribution of the generated molecules.** Duplicate molecules were removed after gathering 3,000 molecules each from five random seed experiments. “Random" molecules are generated by our fragment-based generation method without training the policy network. “FREED(PE)" molecules are generated by the fragment-based generation method while training the policy network. We also plot known “Active" and “Inactive" molecules from DUD-E (fa7, parp1) or ChEMBL (5ht1b) datasets for comparison. Colored horizontal lines indicate the median of the corresponding distribution. **(a)** *de novo* scenario **(b)** scaffold-based scenario

shows the structure of each target’s optimized molecules. We can see that optimized molecules are chemically realistic.

Scaffold-based scenario. We validate our model on scaffold-based scenario, where we attempt to improve docking scores by adding fragments to an initial scaffold molecule. Figure 6 *(b)* shows the distribution of the optimized molecules before (“random") and after (“FREED(PE)") training the policy network, with a scaffold of each target as initial molecule.

Figure 6 (b) highlights our model’s ability to optimize given scaffold to have higher binding affinity with the target. Surprisingly, in Figure 6 (b), even the molecules randomly optimized with our fragment-based generation algorithm show high docking scores when given the proper scaffold. This result implies the importance of scaffold in hit discovery and highlights our generative method’s ability to effectively span the chemical space around the scaffold. Figure 7 **(ii)** shows the structure of

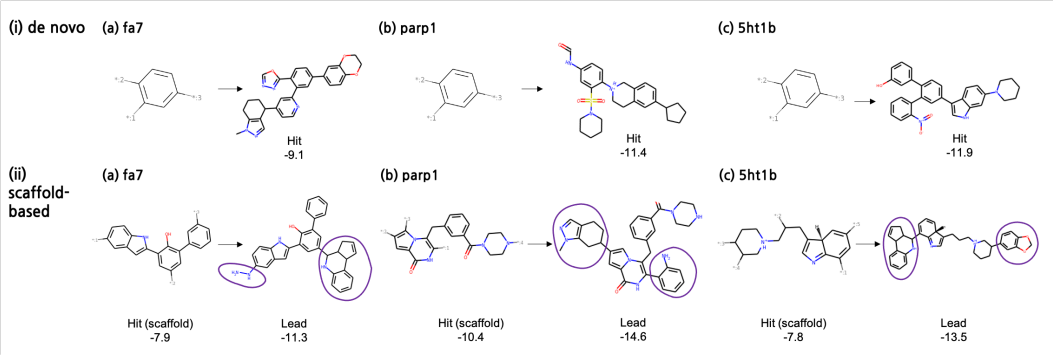

Figure 7: **Generated samples and their docking scores with our method, for *de novo* (i) and scaffold-based scenario (ii).** For each target, one of the high-scoring generated molecules is displayed with the initial molecule (benzene ring or scaffold). Purple line highlights the fragments augmented by the model in scaffold-based generation.

each target's scaffold and corresponding optimized molecules. We can see that optimized molecules are chemically realistic and the scaffold structures are well preserved.

It is notable that our framework doesn't have to be specific for *de novo* or scaffold-based scenarios, except for the initial molecule and number of fragments to be added. Since our model is fully Markovic, whether the initial molecule is a benzene ring or a scaffold does not affect the model's training.

5 Conclusion

In this work, we developed FREED, a novel RL framework for real-world drug design that couples a fragment-based molecular generation strategy with a highly explorative RL algorithm to generate qualified hit molecules. Our model generates pharmacochemically acceptable molecules with high docking scores, significantly outperforming previous docking RL approaches.

Limitations and future work. Our method does not explicitly account for synthesizability of generated molecules. We believe forward synthesis-based methods can be complementary to ours. It would be able to combine our method and forward synthesis-based method as follows: given a current state molecule, select a new reaction site with the policy network similar to our Action 2 network, select a reaction, and select a building block which can proceed the selected reaction with the reaction sites. In this way, we would be able to explicitly take synthesizability into account while providing the appropriate model inductive bias for docking score optimization. We leave such an improvement as a future work.

6 Broader Impact

Using data-driven approaches have been increasingly gained attention for drug discovery, but gathering pharmaceutical data from real-world experiments requires enormously expensive time and cost. Thus, simulation tools such as molecular docking are playing critical role in virtual screening to prioritize real-world experimental candidates. Our algorithm is standing for computationally efficient design of realistic drug candidates, instead of virtual screening in a brute force manner, thus we expect that our models will be utilized by computational chemists in drug development industry. Furthermore, using physical simulation tools as a reward function of RL is a powerful yet largely unexplored research area in drug discovery. We hope that our work provides insight on efficient and effective molecular design by combining simulation engines with machine learning algorithms.
Negative societal impacts. If used maliciously, our framework can be utilized to generate harmful compounds such as biochemical weapons. Thus, conscious use of the AI model is required.

References

[1] W. Jin, R. Barzilay, and T. Jaakkola, "Junction tree variational autoencoder for molecular graph generation," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2323–2332.

[2] A. Zhavoronkov, Y. A. Ivanenkov, A. Aliper, M. S. Veselov, V. A. Aladinskiy, A. V. Aladinskaya, V. A. Terentiev, D. A. Polykovskiy, M. D. Kuznetsov, A. Asadulaev *et al.*, "Deep learning enables rapid identification of potent ddr1 kinase inhibitors," *Nature biotechnology*, vol. 37, no. 9, pp. 1038–1040, 2019.

[3] N. De Cao and T. Kipf, "Molgan: An implicit generative model for small molecular graphs," *arXiv preprint arXiv:1805.11973*, 2018.

[4] W. P. Walters and M. Murcko, "Assessing the impact of generative ai on medicinal chemistry," *Nature biotechnology*, vol. 38, no. 2, pp. 143–145, 2020.

[5] Z. Zhou, S. Kearnes, L. Li, R. N. Zare, and P. Riley, "Optimization of molecules via deep reinforcement learning," *Scientific reports*, vol. 9, no. 1, pp. 1–10, 2019.

[6] J. You, B. Liu, R. Ying, V. Pande, and J. Leskovec, "Graph convolutional policy network for goal-directed molecular graph generation," *arXiv preprint arXiv:1806.02473*, 2018.

[7] T. Blaschke, O. Engkvist, J. Bajorath, and H. Chen, "Memory-assisted reinforcement learning for diverse molecular de novo design," *Journal of cheminformatics*, vol. 12, no. 1, pp. 1–17, 2020.

[8] W. C. Coley, "Defining and exploring chemical spaces," *Trends in Chemistry*, vol. 3, pp. 133–145, 2021.

[9] T. Cieplinski, T. Danel, S. Podlewska, and S. Jastrzębski, "We should at least be able to design molecules that dock well," *arXiv preprint arXiv:2006.16955v3*, 2020.

[10] W. Jin, R. Barzilay, and T. Jaakkola, "Hierarchical generation of molecular graphs using structural motifs," *Proceedings of the 37th International Conference on Machine Learning, PMLR*, no. 119, pp. 4839–4848, 2020.

[11] K. Maziarz, H. Jackson-Flux, P. Cameron, F. Sirockin, N. Schneider, N. Stiefl, and M. Brockschmidt, "Learning to extend molecular scaffolds with structural motifs," *arXiv preprint arXiv:2103.03864v1*, 2021.

[12] T. Schaul, J. Quan, I. Antonoglou, and D. Silver, "Prioritized experience replay," *arXiv preprint arXiv:1511.05952*, 2015.

[13] W. Jeon and D. Kim, "Autonomous molecule generation using reinforcement learning and docking to develop potential novel inhibitors," *Scientific Reports*, vol. 10, no. 22104, 2020.

[14] Z. Zhou, S. Kearnes, L. Li, R. N. Zare, and P. Riley, "Optimization of molecules via deep reinforcement learning," *Scientific Reports*, no. 10752, 2019.

[15] M. Thomas, R. Smith, N. M. O'Boyle, C. de Graaf, and A. Bender, "Comparison of structure- and ligand-based scoring functions for deep generative models: a gpcr case study," *ChemRxiv preprint*, 2021.

[16] M. Olivecrona, T. Blaschke, O. Engkvist, and H. Chen, "Molecular de-novo design through deep reinforcement learning," *Journal of Cheminformatics*, vol. 9, no. 48, 2017.

[17] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine Learning*, vol. 8, pp. 229—-256, 1992.

[18] S. K. Gottipati, B. Sattarov, P. Y. Niu, Sufeng, H. Wei, S. Liu, Y. Bengio *et al.*, "Learning to navigate the synthetically accessible chemical space using reinforcement learning," 2020.

[19] J. Bradshaw, B. Paige, J. M. Kusner, M. H. S. Segler, and J. M. Hernández-Lobato, "Barking up the right tree: an approach to search over molecule synthesis dags," *34th Conference on Neural Information Processing Systems*, 2020.

[20] R. M. Ryan and E. L. Deci, "Intrinsic and extrinsic motivations: Classic definitions and new directions," *Contemporary educational psychology*, vol. 25, no. 1, pp. 54–67, 2000.

[21] P.-Y. Oudeyer and F. Kaplan, "What is intrinsic motivation? a typology of computational approaches," *Frontiers in neurorobotics*, vol. 1, p. 6, 2009.

[22] A. G. Barto, "Intrinsic motivation and reinforcement learning," in *Intrinsically motivated learning in natural and artificial systems*. Springer, 2013, pp. 17–47.

[23] M. G. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, "Unifying count-based exploration and intrinsic motivation," *arXiv preprint arXiv:1606.01868*, 2016.

[24] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *International Conference on Machine Learning*. PMLR, 2017, pp. 2778–2787.

[25] A. A. Taïga, W. Fedus, M. C. Machado, A. Courville, and M. G. Bellemare, "Benchmarking bonus-based exploration methods on the arcade learning environment," *arXiv preprint arXiv:1908.02388*, 2019.

[26] A. Ecoffet, J. Huizinga, J. Lehman, K. O. Stanley, and J. Clune, "Go-explore: a new approach for hard-exploration problems," *arXiv preprint arXiv:1901.10995*, 2019.

[27] T. L. Paine, C. Gulcehre, B. Shahriari, M. Denil, M. Hoffman, H. Soyer, R. Tanburn, S. Kapturowski, N. Rabinowitz, D. Williams *et al.*, "Making efficient use of demonstrations to solve hard exploration problems," *arXiv preprint arXiv:1909.01387*, 2019.

[28] J. Oh, Y. Guo, S. Singh, and H. Lee, "Self-imitation learning," in *International Conference on Machine Learning*. PMLR, 2018, pp. 3878–3887.

[29] J. Ferret, O. Pietquin, and M. Geist, "Self-imitation advantage learning," *arXiv preprint arXiv:2012.11989*, 2020.

[30] S. Lee, J. Lee, and I. Hasuo, "Predictive per: Balancing priority and diversity towards stable deep reinforcement learning," *arXiv preprint arXiv:2011.13093*, 2020.

[31] P. Polishchuk, "Crem: chemically reasonable mutations framework for structure generation," *Journal of Cheminformatics*, vol. 12, no. 28, 2020.

[32] D. Hwang, S. Yang, Y. Kwon, K. H. Lee, G. Lee, H. Jo, S. Yoon, and S. Ryu, "Comprehensive study on molecular supervised learning with graph neural networks," *Journal of Chemical Information and Modeling*, vol. 60, no. 12, pp. 5936–5945, 2020.

[33] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[34] S. M. Jayakumar, W. M. Czarnecki, J. Menick, J. Schwarz, J. Rae, S. Osindero, Y. W. Teh, T. Harley, and R. Pascanu, "Multiplicative interactions and where to find them," in *International Conference on Learning Representations*, 2019.

[35] G. Landrum, "Rdkit: Open-source cheminformatics software," 2016. [Online]. Available: https://github.com/rdkit/rdkit/releases/tag/Release_2016_09_4

[36] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," *arXiv preprint arXiv:1611.01144*, 2016.

[37] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *International Conference on Machine Learning*. PMLR, 2018, pp. 1861–1870.

[38] T. Haarnoja, A. Zhou, K. Hartikainen, G. Tucker, S. Ha, J. Tan, V. Kumar, H. Zhu, A. Gupta, P. Abbeel *et al.*, "Soft actor-critic algorithms and applications," *arXiv preprint arXiv:1812.05905*, 2018.

[39] P. Christodoulou, "Soft actor-critic for discrete action settings," *arXiv preprint arXiv:1910.07207*, 2019.

[40] L. A. Thiede, M. Krenn, A. Nigam, and A. Aspuru-Guzik, "Curiosity in exploring chemical space: Intrinsic rewards for deep molecular reinforcement learning," *arXiv preprint arXiv:2012.11293*, 2020.

[41] A. Kendall and Y. Gal, "What uncertainties do we need in bayesian deep learning for computer vision?" *arXiv preprint arXiv:1703.04977*, 2017.

[42] Y. Gal and Z. Ghahramani, "Dropout as a bayesian approximation: Representing model uncertainty in deep learning," in *international conference on machine learning*. PMLR, 2016, pp. 1050–1059.

[43] S. J. Lane, D. S. Eggleston, K. A. Brinded, J. C. Hollerton, N. L. Taylor, and S. A. Readshaw, "Defining and maintaining a high quality screening collection: the gsk experience," *Drug discovery today*, vol. 11, no. 5-6, pp. 267–272, 2006.

[44] G. Papadatos, M. Davies, N. Dedman, J. Chambers, A. Gaulton, J. Siddle, R. Koks, S. A. Irvine, J. Pettersson, N. Goncharoff *et al.*, "Surechembl: a large-scale, chemically annotated patent document database," *Nucleic acids research*, vol. 44, no. D1, pp. D1220–D1228, 2016.

[45] J. B. Baell and G. A. Holloway, "New substructure filters for removal of pan assay interference compounds (pains) from screening libraries and for their exclusion in bioassays," *Journal of medicinal chemistry*, vol. 53, no. 7, pp. 2719–2740, 2010.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section **??**.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See bullets in introduction.
   (b) Did you describe the limitations of your work? [Yes] We have declared limitations of our work in conclusion and left it as future work.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Broader Impact.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See source code URL.

(b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Experiment settings on Appendix.

(c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We ran every experiements on 5 different seeds, showing averaged performances with standard deviation in both plots and tables.

(d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Computational resources on Appendix.

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

(a) If your work uses existing assets, did you cite the creators? [Yes]

(b) Did you mention the license of the assets? [Yes] License is mentioned at the footnote when asset is first mentioned.

(c) Did you include any new assets either in the supplemental material or as a URL? [No]

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]