Supporting Information for LS-MolGen: Ligand-and-Structure Dual-driven Deep Reinforcement Learning for Target-specific Molecular Generation Improves Binding Affinity and Novelty

Song Li,[†] Chao Hu,[‡] Song Ke,[¶] Chenxing Yang,[¶] Jun Chen,[¶] Yi Xiong,[§] Hao Liu,[*,¶,‖] and Liang Hong[*,¶,†,⊥]

[†]*School of Physics and Astronomy, Shanghai Jiao Tong University, Shanghai, China*

[‡]*Department of Computer Science and Engineerings, East China University of Science and Technology, Shanghai, China*

[¶]*Shanghai Matwings Technology Co., Ltd., Shanghai, China*

[§]*School of Life Sciences and Biotechnology, Shanghai Jiao Tong University, Shanghai, China*

[‖]*Institute of Natural Sciences, Shanghai Jiao Tong University, Shanghai, China*

[⊥]*Shanghai Artificial Intelligence Laboratory, Shanghai, China*

E-mail: chaohao2010@sjtu.edu.cn; hongl3liang@sjtu.edu.cn

Abstract

Molecule generative models based on deep learning have attracted significant attention in de novo drug design. However, most current generative approaches are either only ligand-based or only structure-based, which do not leverage the complementary

knowledge from ligands and the structure of binding target. In this work, we proposed a new ligand and structure combined molecular generative model, LS-MolGen, that integrates representation learning, transfer learning, and reinforcement learning. Focus knowledge from transfer learning and special explore strategy in reinforcement learning enables LS-MolGen to generate novel and active molecules efficiently. The results of evaluation using EGFR and case study of inhibitor design for SARS-CoV-2 Mpro showed that LS-MolGen outperformed other state-of-the-art ligand-based or structure-based generative models and was capable of de novo designing promising compounds with novel scaffold and high binding affinity. Thus, we recommend that this proof-of-concept ligand-and-structure-based generative model will provide a promising new tool for target-specific molecular generation and drug design.

The hyper-parameters of Models and training

LS-MolGen

The model network in this work is the recurrent neural network (RNN), which contains six layers: one input layer, one embedding layer, three recurrent layers, and one output layer. In the embedding layer, each token was encoded into a 128-dimensional embedding vector. For a recurrent layer, a gated recurrent unit (GRU) was used as the recurrent cell with 512 hidden neurons.

In pre-training, we trained the prior model with batch size 128 and initial learning rate 0.001, and decayed the learning rate by a factor of 0.03. The number of epoch was 5 for the pre-training. In transfer learning, we trained the transfer model with batch size 16 and initial learning rate 0.001 and decayed learning rate by a factor of 0.03. The number of epochs was 500, and we halted the training session if the model was not improving for 20 epochs. In the reinforcement learning, we trained the agent model with batch size 128 and learning rate 0.0005. The number of RL steps is 3000, and we halted the exploration if the model

was not improving for 200 steps. An inception was used to randomly replay the explored good SMILES, of which the min similar value was set to 0.4 and the max size of scaffold bucket was one. 5000 was set as the max size of the agent memory in this work, because we generated 5000 molecules at a time. 32 parallel computing was used in RL loop to get docking scores.

Computational cost

All the experiments were conducted on Ubuntu Linux with an NVIDIA GeForce RTX 3090 GPU and more than 32 CPUs. we trained these seven models (i.e., AAE, VAE, LatentGAN, REINVENT, Pocket2Mol, MolDQN, SBMolGen) from scratch on our platform. The training times and generating times of these models were listed in Table S1. The training time includes the time used for pre-training (AAE, VAE, LatentGAN, REINVENT, LS-MolGen) or model training (Pocket2Mol, SBMolGen). For each model, we run the generating process 6 times independently and generated 5000 molecules at each time. The generating time refers to the average cost of 6 times. We noted that Pocket2Mol can not generate 5000 molecules at a time since the limitation of the model itself, so we only used 2000 molecules generated molecules for evaluations.

MM-GBSA

The MMGBSA module in Maestro was utilized to perform MM-GBSA calculations. MM-GBSA is a widely-used method for estimating the binding free energy of protein-ligand complexes. It involves calculating the energy difference between the bound and unbound states of the complex, which is comprised of molecular mechanics (MM) energy, solvation energy, and entropy contributions. The MMGBSA module in Maestro provides an efficient and user-friendly interface for performing these calculations, with options for selecting force fields, solvent models, and other parameters. The module also allows for the analysis of key interactions between the protein and ligand, such as hydrogen bonds and hydrophobic

contacts. Overall, the MMGBSA module in Maestro is a valuable tool for predicting and analyzing protein-ligand binding energies, and the results were listed in Table S3.

Supplementary Tables

Table 1: Comparison of the computational cost.

Model		Training time	Generating time
Ligand-based	AAE	7.0 hours	15.2 seconds
	VAE	7.0 hours	13.3 seconds
	LatentGAN	15.0 hours	55.7 seconds
Structure-based	REINVENT	4.0 hours	12.0 hours
	Pocket2Mol	2.0 days	4.2 hours
	MolDQN	-	2.0 days
Ligand-and-structure-based	SBMolGen	2.8 days	5.0 days
	LS-MolGen	4.0 hours	4.3 hours

Table 2: The evaluation metrics of molecules generated by LS-MolGen for Mpro inhibitors design. 5000 molecules were generated.

Model	Validity	Uniqueness	Diversity	Novelty	Recovery	Active rate	Success rate
LS-MolGen	1.000	0.999	0.986	0.974	0.017	0.839	0.827

Table 3: The results of MM-GBSA for compounds 1-6

Compound	MM-GBSA dG Bind (kcal/mol)
1	-43.98
2	-41.92
3	-53.53
4	-74.93
5	-47.07
6	-58.97

Supplementary Figures

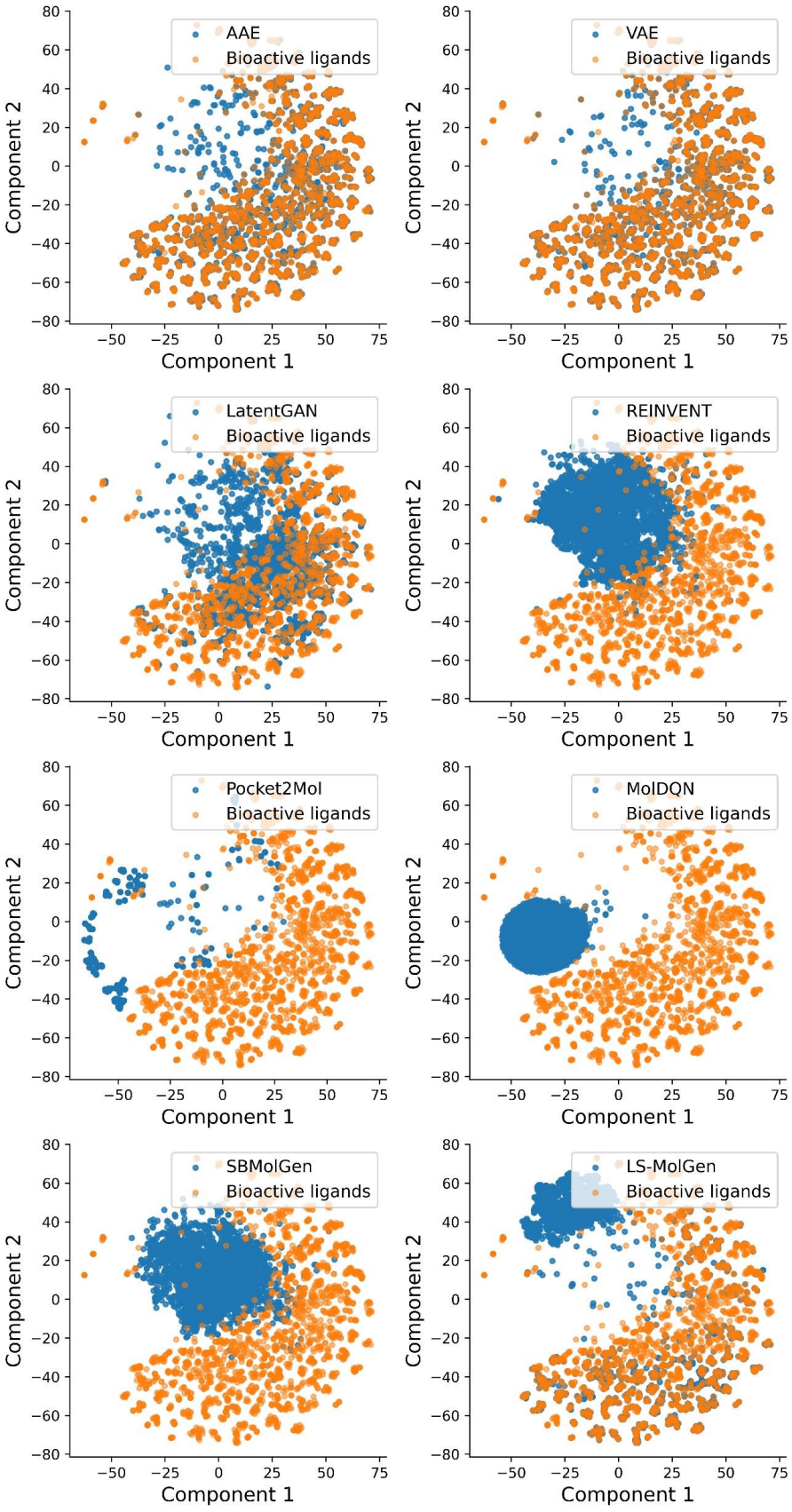

Figure 1: t-SNE Chemical space visualization of generated molecules for EGFR.

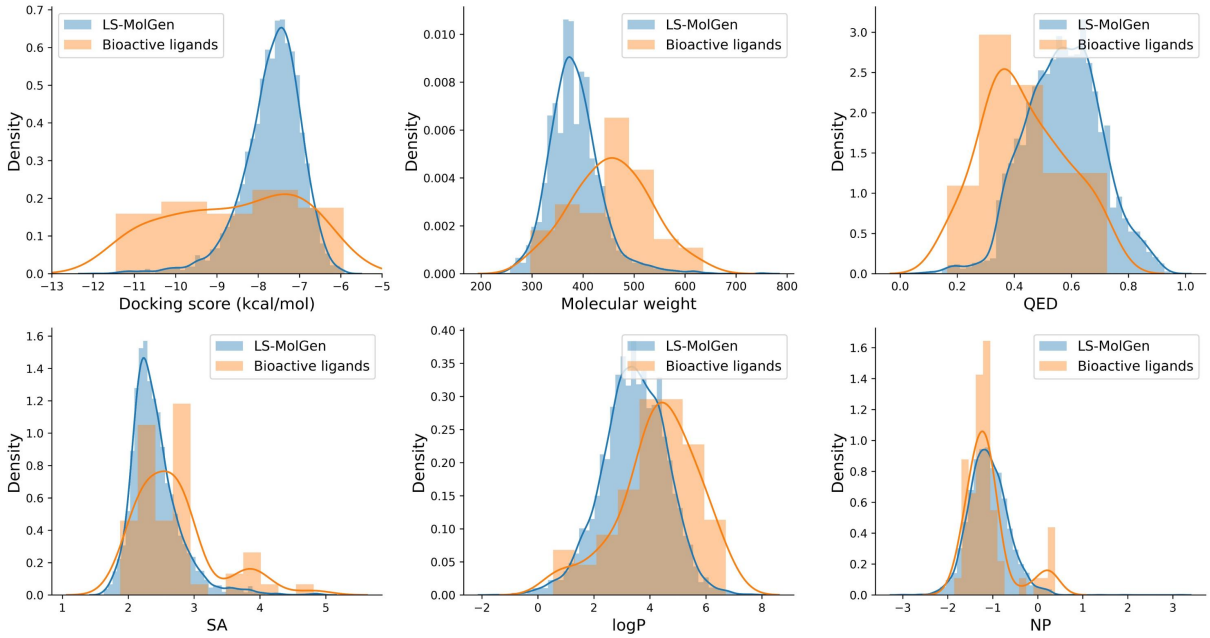

Figure 2: Distributions of properties of generated molecules for Mpro inhibitors design by LS-MolGen, including docking score, molecular weight, QED, SA score, logP, and NP score.