In silico drug design by Molecular Generative Model and Docking

Ma Dian[1,a)] Nobuaki Yasuo[2] Masakazu Sekijima[1]

Abstract: This study develops a new deep learning-based extendable multiple-objective molecular generator (MOMolGen). This generator integrates a recurrent neural network (RNN) to generate molecules and Pareto Multi-Objective Monte Carlo Tree Search (Pareto MOMCTS) to decide search direction. This generator is validated by generating compounds for specific target proteins with evaluation on the drug-like properties and docking score.

Keywords: molecule generative model, multiple objective,

1. Introduction

Developing a new drug to market is costly in terms of money and time. Drug development takes an average of 10-15 years with an approximate cost of 2.8 billion USD[1]. In the stage of screening and design, the size of the drug-like chemical space is approximate 10^{60}[2][3], and scientists cannot master such enormous search space with human experience.

Molecules with high pharmacological or biological activity towards the target have been selected in the stage of the lead compound screening[4]. Molecules with high affinity to target protein are called hit compounds, and hit compounds with other pharmacological properties are called lead compounds. Lead compounds will be optimized for drug candidates in a further stage of the drug design process.

The traditional chemical method in lead compound screening is high throughput screening (HTS)[5]. In HTS, only a few thousand samples can be tested in one time. It is a relatively small number for the search space, and testing on molecules needs synthesis is costly.

With the development of computer science, computer-aided drug discovery (CADD) is becoming more and more popular in pharmaceutical companies and research institutions[6][7]. In CADD, the search for lead compounds is more effective in a virtual simulation method. Furthermore, CADD searches lead compounds without compound synthesis, which is anticipated to cut costs[8].

In CADD, the screening method can apply to chemical database as virtual screening. However, virtual screening is hard to reach a full space search due to the enormous search space of drug-like compounds[2].

[1] School of Computing, Tokyo Institute of Technology
[2] Tokyo Tech Academy for Convergence of Materials and Informatics (TAC-MI), Tokyo Institute of Technology
[a)] ma.d.aa@m.titech.ac.jp

1.1 De novo drug design

Unlike virtual screening, de-novo drug design attempts to create structurally novel lead compounds with desired properties, which may be affinity to the target protein, molecular weight, or molecular stability[9][10]. De-novo drug design can avoid the problem of virtual screening that compounds beyond search space are unreachable.

Deep learning has played a noteworthy role in recently de-novo drug design research[11]. Deep learning has shown potential in multiple fields, such as computer vision[12], speech recognition[13], and audio signal processing[14]. The usage of neural networks with multiple hidden layers made an intuition that deep learning can learn higher-level abstractions of the input.

1.2 Generative model

Recently, Gomez-Bombarelli et al.[15] were firstly applied a neural network called variational autoencoder (VAE) on molecule generate. However, SMILES strings created by VAE have a low success rate; most of them have an invalid chemical structure; VAEs need repeated generation steps to obtain a molecule. Segler et al.[16] employ recurrent neural network (RNN) to generate molecules and achieve a high success rate of valid chemical structures. Their algorithm generates molecules randomly and chooses high-scoring molecules from them. It required a large scale of candidates to ensure including the desirable molecules; moreover, molecules generated by these models need further structural optimization to meet drug development requirements.

Sattarov et al.[17] and Grebner et al.[18] generate molecules with a generative model and select candidates according to the performance of docking simulation. However, the docking simulation in these models does not affect the generation process; thus, molecules produced from these models are not fully optimized in the docking simulation towards the target protein. Xu et al.[19] proposed a model with VAE and docking score, but it reached the optimal molecule in latent space of VAE with a long time search.

Yang et al.[20] integrate Monte Carlo tree search (MCTS) and

RNN to get a directional molecule generation called ChemTS. ChemTS builds a search tree according to the performance of generated molecules and directs the search process to a more promising structure to reduce the ineffective generation. Ma et al.[21] proposed SBMolGen, which enhances ChemTS with a more complex reward function. However, ChemTS focuses on optimizing a single reward that will lead the generation in an extreme direction where other essential properties are neglected. And SBMolGen has a sound performance in generating molecules with a good affinity towards target protein, but other evaluations, such as the QED score, are relatively poor.

To make the generative model more appropriate for realistic drug design tasks, multi-objective optimization has also been applied. Quentin et al.[22] has proposed multi-objective generative models based on RNN and a scalar reward function, which is a weighted sum of multiple rewards. Khemchandani et al.[23] has proposed a DeepGraphMolGen, which is based on grammar-based encoding of molecules into graphs and the graph convolutional neural network. Both studies use a scalar reward function that causes a loss of a degree of freedom in the search process by changing the multiple objective problem into a single objective problem.

2. Research Purpose

The purpose of this study is to develop a multiobjective molecule generator that is extensible using deep learning. For this purpose, we use utilizes an RNN to generate molecules and Pareto Multi-Objective Monte Carlo Tree Search (Pareto MOMCTS) to expand the search process into a multiple reward space. By docking simulation, MOMolGen generates molecules with a strong binding affinity toward the target protein. Moreover, other pharmaceutical properties involved in molecule evaluation, such as the QED score and logP, made generated molecules more practicable in the further drug design process. Furthermore, a vectorized reward keeps enough degree of freedom in the search process.

3. Methods

3.1 MOMolGen

The workflow of MOMolGen is illustrated in Figure 1. It consists of two parts, (1), Monte Carlo search tree to execute the search process and decide the head of a drug candidate, (2), generative RNN to complement the drug candidate, and followed by a series of evaluations which is extendable by drug design requirements.

The MOMolGen has four steps in every single search:

(1) Selection: Each node in the search tree have one character in SMILES vocabulary, which may represent an element or a structure. In the selection step, an expandable node is selected through tree policy, and the route from the root node to the selected node is the front part of the molecules been explored in the current search.

(2) Expansion: One node adds to the search tree as a child node of the selected node.

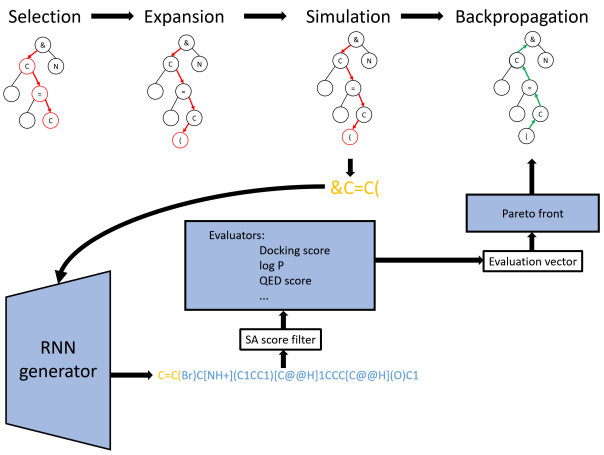

Fig. 1 MOMolGen

(3) Simulation: In the simulation stage, pre-trained RNN will act as default policy complement the molecule and send them to evaluators to get an evaluation vector, the new evaluation vector compared with the Pareto front to decide it is a dominated vector or not.

(4) Backpropagation: The value of the evaluation vector will pass to all parent nodes through the backpropagation route.

A whole search on a given target protein will repeat these four steps and stop when it reaches the computation limitation. Molecules with a nondominated reward vector on the Pareto front are promising drug candidates with property desired.

3.2 Tree policy

The tree policy chooses the nodes to explore in the search process grounds on the rewards of the nodes in selection stage. The cumulative rewards r_s of a state s are updated as:

$$r_s \leftarrow \frac{1}{n_s + 1}(n_s \times r_s + r_u)$$

where r_u is reward of a new evaluation, and n_s is the number of visit times of state s.

Simply selecting the nodes with a high cumulative reward will lose the balance between exploitation and exploration. The Upper Confidence Bound (UCB) $\bar{r}_s$ is used to control the balance:

$$\bar{r}_s = \left(r_{s;i} + \sqrt{c_i \ln\left(n_{parent}\right) / n_s} \right)_{i=1}^{d}$$

where c_i is exploration vs exploitation parameter for i-th component of reward vector.

A vector is hard to sort due to its high dimension. An Upper-bound $U(s)$ Using hyper-volume (HV) indicator of r_s with Pareto front P is:

$$U(s) = V(\bar{r}_s) = HV(P \cup \{\bar{r}_s\}; z)$$

where z is the reference point of hyper-volume indicator.

$U(s)$ provide a scalar evaluation of a node s. However, it keeps a constant value if $\bar{r}_s$ is dominated by any point in the Pareto front.

An approach to sort dominated point is called Pareto-rank. This method can sort every point into different layers, like the

Pareto front. Nodes in the same layer are nondominated by each other. This option requires the maintenance of all nodes, which is too computational cost. Instead, this research uses the distance of point perspective projection to pareto front:

$$W(s) = U(s) - |\bar{r}_s^p - \bar{r}_s|_2$$

The pseudocode of MOMolGen is illustrated:

Algorithm 1 The MOMolGen algorithm

MOMolGen
Input: computational limit
Output: search tree T, Pareto Front P.
Initialize $T_0 \leftarrow$ initial search tree, $v_0 \leftarrow$ root node, $P \leftarrow \emptyset$
while within computational limit **do**
  $v_s \leftarrow$ TreePolicy(v_0)
  $r_u \leftarrow$ DefaultPolicy(v_s)
  if r_u is not dominated by any points in P **then**
    Remove points dominated by r_u in P
    Add r_u to P
  end if
  while v_s is not root node **do**
    $r_s \leftarrow \frac{1}{n_s+1}(n_s \times r_s + r_u)$
    $n_s \leftarrow n_s + 1$
    $v_s \leftarrow$ parent of v_s
  end while
end while
return T

TreePolicy(v)
while v is nonterminal **do**
  if v is not a leaf node **then**
    $v = \underset{v' \in \text{children of } v}{\text{argmax}} W(v')$
  else
    return a new child node of v
  end if
end while
return v

DefaultPolicy v
extract a SMILES fragment S form path of v
while S is not terminal state **do**
  RNN generate a new character and append to S
end while
return reward for a complete SMILES r.

4. Experiments

4.1 Dataset

Dataset for RNN training is from ZINC. ZINC is a free public database for ligand discovery, including over twenty million molecules in biologically relevant representations. This research use about 250,000 ligand-like molecules represented in SMILES.

3D structure data of proteins for ligand generating is from the protein data bank; this research used four proteins to validate the molecular generation process. There are two kinases (cyclin-dependent kinase 2 (CDK2)[24] and epidermal growth factor receptor erbB1 (EGFR)[25]) and two G protein-coupled receptors (adenosine A2a receptor (AA2AR)[26] and beta-2 adrenergic receptor (ADRB2)[27]), details of proteins are listed in Table 1:

Table 1 Experiment environment

Protein	PDB ID	Actives	Decoys
CDK2	1H00	474	27830
EGFR	2RGP	542	17924
AA2AR	3EML	482	31498
ADRB2	3NY8	231	14993

Vocabulary of SMILES are listed in Table2

Table 2 SMILES vocabulary

Catagory	SMILES description
Atom	C,c,o,O,N,F,n,S,s,Br,I,P
Bonds	. - = # $: / \
Functional group	[C@@H], [O-], [C@H], [NH+], [C@], [nH] , [NH2+] , [C@@] , [N+] , [nH+] , [S@] , [N-] , [n-] , [OH+] , [NH-] , [P@@H] , [P@@] , [PH2] , [o+] , [CH2-] , [CH-] , [SH+] , [O+] , [S-], [S+] , [S@@+], [NH3+],[n+] , [S@@] ,[P@] , [P+] ,[PH] ,[s+] , [PH+] ,
Terminator	\ n

4.2 RNN training

This research uses an RNN model as a generator of ligand. RNN model is pre-trained and stays identical during the ligand search process.

Parameter of RNN training:

- Algorithm: Adaptive Moment Estimation (Adam)
- Learning rate: 0.01
- Batch size: 256
- Epoch: 100

4.3 Evaluations

This research chooses the Docking score, logP, QED score, and SAscore as evaluations for generated molecules.

The docking score evaluates the binding effect between molecules and target protein. The lower the docking score is, the better the molecule could combinate target protein. The reward function of the docking score is:

$$r_{docking} = -\frac{(dockingsocre - basescore) * 0.1}{1 + |(dockingsocre - basescore) * 0.1|}$$

where base score is different baseline for each protein, in this research is 0.

LogP evaluates the lipophilicity of generated molecules, which could estimate solubility, absorption, and membrane penetration. Different target proteins have a variety of best logP values. Therefore, the reward function of LogP is:

$$r_{logP} = max\left\{1 - (a * (logP - center\ value))^2,\ 0\right\}$$

where center value is the best logP value for different target protein, in this research is 1.4. And a is constant parameter.

QED score evaluates the drug-likeness of generated molecules from 0 to 1. The higher the number is, the molecule is more likely to become a drug, the reward function of the QED score is:

$$r_{QED} = QED\ score$$

SAscore evaluates the synthesis accessibility of generated molecules from 1 to 10. The greater number of SAscores represents a more difficult synthesis. This research uses the SAscore as a filter instead of the final evaluation. The threshold value of the SAscore is < 3.5.

5. Results

5.1 Molecule Generation

The exploration time of molecular generation against four target proteins is set to 120h. Approximately 35,000 molecules were generated for each target protein. Figure 2 3 4 shows the density distributions of Docking score, LogP, and QED score.

The docking score of generated molecules against four target proteins has a normal distribution. However, some molecules with a relatively high docking score are generated.

Distributions of logP of generated molecules concentrated in the choice center value, most of the generated molecules have a logP value in the range of -2 to 4.

A considerable number of molecules with a high QED score are generated in the QED score. Four distributions of QED score have a similar shape, the generative model shows portability on different proteins. Table 3 shows the numbers of generated molecules.

Table 3 Number of Generated Molecules

Protein	Total	Docking score < -10	logP *Center value* ± 0.3	QED score > 0.8
CDK2	36458	189	5166	9298
EGFR	50132	426	7159	11947
AA2AR	36506	496	5366	9536
ADRB2	35421	998	5058	8945

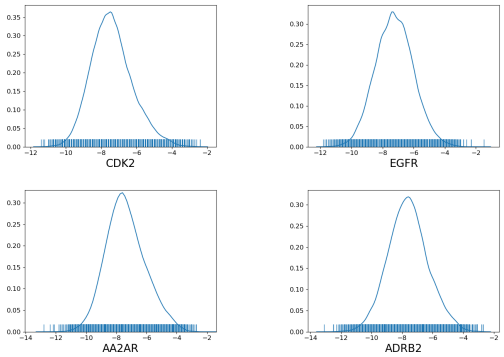

Fig. 2 Docking score of generated molecules

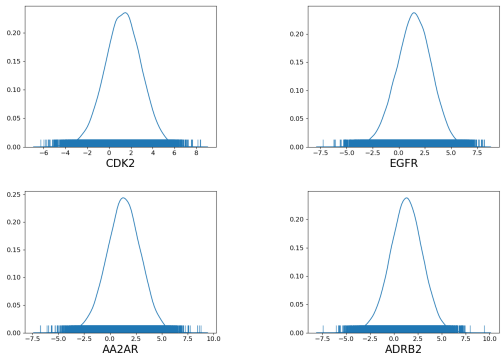

Fig. 3 LogP of generated molecules

The distribution of molecules in the reward space are illustrated in Figure 6, points are represented with colors from blue to red in the order of generation. Molecules generated in the early stage of generation are centralized in the center section. On the other hand, molecules generated in the later generation stage show a more widespread distribution in search space. The search tree builds in the generation process directs the search to a new region.

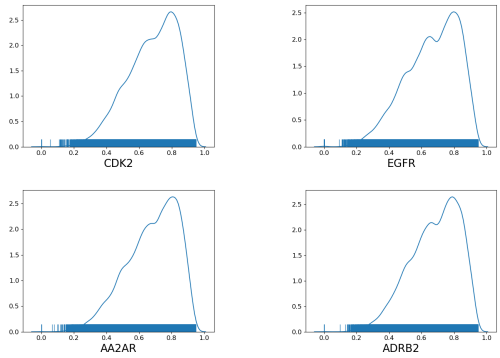

Fig. 4 QED score of generated molecules

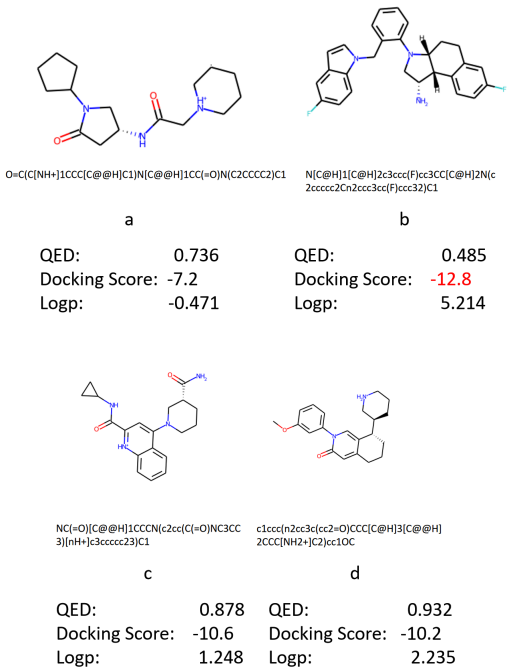

Fig. 5 Generated molecules for AA2AR

Figure 5 shows generated molecules towards AA2AR, molecule *a* is the first generated molecule, and molecules *b*, *c*, and *d* are molecules on the Pareto front at the end of the search. *b* is the molecule with the best docking score; however, the QED score is relatively low. *c* and *d* are two molecules with a balance performance on all evaluations.

5.2 Pareto Front Change

Projections of Pareto front change in the generation against AA2AR on the 2D reward plane are shown in Figure 7,Figure 8, and Figure 9. Pareto fronts on both the Docking-QED reward plane and Docking-LogP reward plane show a gradual move to the right-top during the search process, representing that MOMolGen changes the search directions to the more well performance branch in the Monto Carlo search tree.

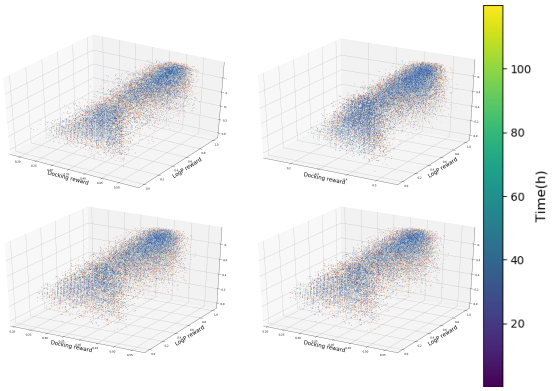

Fig. 6 Reward space of generated molecules

However, the Pareto front on the QED-LogP reward plane reaches right-top in the early search stage, which shows QED and logP are highly correlated to each other; multiple optimizations on this pair of evaluations are unnecessary. The selection of objectives needs to be more prudent in future work. A mistake like that is a waste of computational resources; moreover, it influences the performance of other optimization.

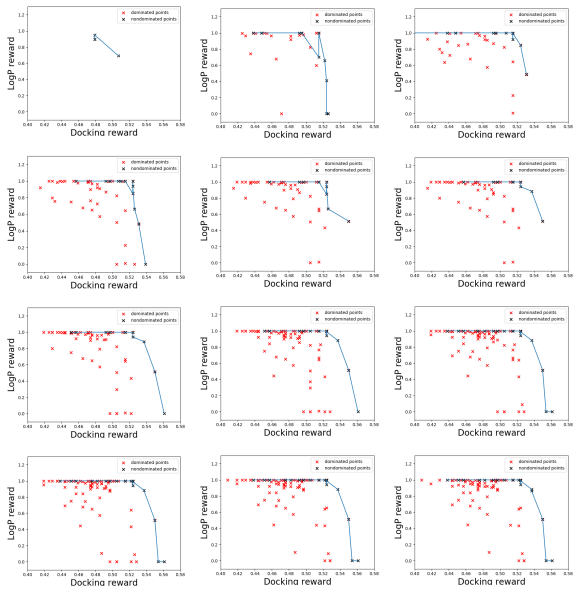

Fig. 7 Pareto fronts of AA2AR on the Docking-logP plane in every 10 hours

6. Conclusion

This research has developed MOMolGen, an extendable multiple-objective molecular generator model. MOMolGen is verified on two groups of proteins; it generates molecules with multiple desired pharmaceutical properties on test proteins. Compared with single objective molecular generator models, which neglect other pharmaceutical properties, MOMolGen has full consideration for multiple properties needed by drug design.

This research has validated the MOMolGen in multiple optimizations of docking score, QED score, and logP; however, the validation of MOMolGen extensibility on other pharmaceutical properties is needed.

Through this research have generated molecules with multiple

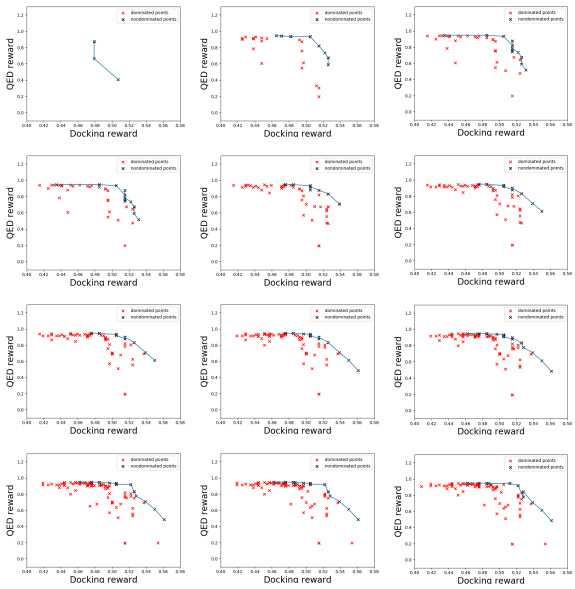

Fig. 8 Pareto fronts of AA2AR on the Docking-QED plane in every 10 hours

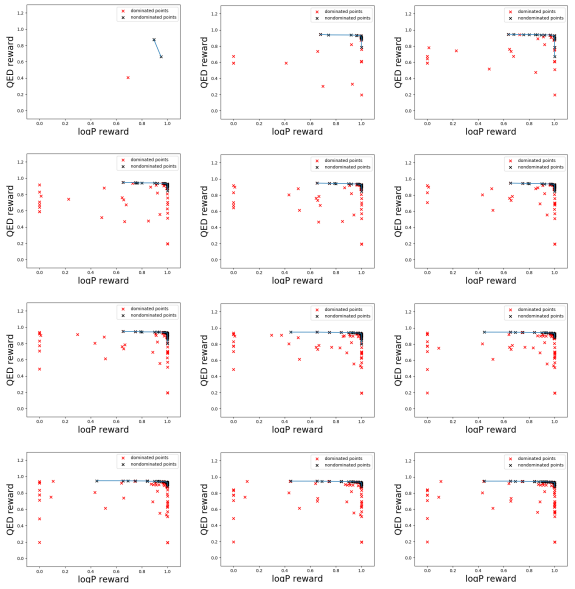

Fig. 9 Pareto fronts of AA2AR on the logP-QED plane in every 10 hours

desired properties, how to select drug candidates from these generated molecules is still challenging. Molecules on the Pareto front are better than other molecules in some aspects, but there lacks a method of comparison between these molecules. Furthermore, if only molecules on the Pareto front are considered, molecules distributed near the Pareto front in reward space are ignored, which still have an outstanding evaluation.

References

[1] Joseph A. DiMasi, Henry G. Grabowski, and Ronald W. Hansen. Innovation in the pharmaceutical industry: New estimates of R&D costs. *Journal of Health Economics*, 47:20–33, 2016.

[2] P. G. Polishchuk, T. I. Madzhidov, and A. Varnek. Estimation of the size of drug-like chemical space based on gdb-17 data. *Journal of Computer-Aided Molecular Design*, 27(8):675–679, 2013.

[3] Melvin J Yu. Druggable chemical space and enumerative combinatorics. *Journal of Cheminformatics*, 5(1), 2013.

[4] JP Hughes, S Rees, SB Kalindjian, and KL Philpott. Principles of early

drug discovery. *British Journal of Pharmacology*, 162(6):1239–1249, 2011.

[5] Jürgen Bajorath. Integration of virtual and high-throughput screening. *Nature Reviews Drug Discovery*, 1(11):882–894, 2002.

[6] Stephen J Haggarty, Thomas U Mayer, David T Miyamoto, Reza Fathi, Randall W King, Timothy J Mitchison, and Stuart L Schreiber. Dissecting cellular processes using small molecules: Identification of colchicine-like, taxol-like and other small molecules that perturb mitosis. *Chemistry & Biology*, 7(4):275–286, 2000.

[7] Kathleen Young, Stephen Lin, Lucy Sun, Eunice Lee, Mita Modi, Samuel Hellings, Morris Husbands, Brad Ozenberger, and Rodrigo Franco. Identification of a calcium channel modulator using a high throughput yeast two-hybrid screen. *Nature Biotechnology*, 16(10):946–950, 1998.

[8] Jian J. Tan, Xiao J. Cong, Li M. Hu, Cun X. Wang, Lee Jia, and Xing-Jie Liang. Therapeutic strategies underpinning the development of novel techniques for the treatment of hiv infection. *Drug Discovery Today*, 15(5):186–197, 2010.

[9] Hans-Joachim Bhm. The computer program ludi: A new method for the de novo design of enzyme inhibitors. *Journal of Computer-Aided Molecular Design*, 6(1):61–78, 1992.

[10] Jan A Hiss, Michael Reutlinger, Christian P Koch, Anna M Perna, Petra Schneider, Tiago Rodrigues, Sarah Haller, Gerd Folkers, Lutz Weber, Renato B Baleeiro, and et al. Combinatorial chemistry by ant colony optimization. *Future Medicinal Chemistry*, 6(3):267–280, 2014.

[11] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT Press, 2016.

[12] Athanasios Voulodimos, Nikolaos Doulamis, Anastasios Doulamis, and Eftychios Protopapadakis. Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 2018:1–13, 2018.

[13] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 6645–6649, 2013.

[14] Hendrik Purwins, Bo Li, Tuomas Virtanen, Jan Schlüter, Shuo-Yiin Chang, and Tara Sainath. Deep learning for audio signal processing. *IEEE Journal of Selected Topics in Signal Processing*, 13(2):206–219, 2019.

[15] Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, Alán Aspuru-Guzik, and et al. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, 2018.

[16] Marwin H. Segler, Thierry Kogej, Christian Tyrchan, and Mark P. Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Central Science*, 4(1):120–131, 2017.

[17] Boris Sattarov, Igor I. Baskin, Dragos Horvath, Gilles Marcou, Esben Jannik Bjerrum, and Alexandre Varnek. De novo molecular design by combining deep autoencoder recurrent neural networks with generative topographic mapping. *Journal of Chemical Information and Modeling*, 59(3):1182–1196, 2019.

[18] Christoph Grebner, Hans Matter, Alleyn T. Plowright, and Gerhard Hessler. Automated de novo design in medicinal chemistry: Which types of chemistry does a generative neural network learn? *Journal of Medicinal Chemistry*, 63(16):8809–8823, 2020.

[19] Ziqiao Xu, Orrette R. Wauchope, and Aaron T. Frank. Navigating chemical space by interfacing generative artificial intelligence and molecular docking. *Journal of Chemical Information and Modeling*, 61(11):5589–5600, 2021.

[20] Xiufeng Yang, Jinzhe Zhang, Kazuki Yoshizoe, Kei Terayama, and Koji Tsuda. Chemts: an efficient python library for de novo molecular generation. *Science and Technology of Advanced Materials*, 18(1):972–976, 2017. PMID: 29435094.

[21] Biao Ma, Kei Terayama, Shigeyuki Matsumoto, Yuta Isaka, Yoko Sasakura, Hiroaki Iwata, Mitsugu Araki, and Yasushi Okuno. Structure-based de novo molecular generator combined with artificial intelligence and docking simulations. *Journal of Chemical Information and Modeling*, 61(7):3304–3313, 2021.

[22] Robin Winter, Floriane Montanari, Andreas Steffen, Hans Briem, Frank Noe, and Djork-Arné Clevert. Efficient multi-objective molecular optimization in a continuous latent space. *Chemical Science*, 10(34):8016–8024, 2019.

[23] Yash Khemchandani, Stephen O'Hagan, Soumitra Samanta, Neil Swainston, Timothy J. Roberts, Danushka Bollegala, and Douglas B. Kell. Deepgraphmolgen, a multi-objective, computational strategy for generating molecules with desirable properties: A graph convolution and reinforcement learning approach. *Journal of Cheminformatics*, 12(1), 2020.

[24] John F. Beattie, Gloria A. Breault, Rebecca P.A. Ellston, Stephen Green, Philip J. Jewsbury, Catherine J. Midgley, Russell T. Naven, Claire A. Minshull, Richard A. Pauptit, Julie A. Tucker, and J.Elizabeth Pease. Cyclin-dependent kinase 4 inhibitors as a treatment for cancer. part 1: identification and optimisation of substituted 4,6-bis anilino pyrimidines. *Bioorganic & Medicinal Chemistry Letters*, 13(18):2955–2960, 2003. Recent Advances in the Design of Kinase Inhibitors as Novel Therapeutic Agents.

[25] Guozhang Xu, Marta C. Abad, Peter J. Connolly, Michael P. Neeper, Geoffrey T. Struble, Barry A. Springer, Stuart L. Emanuel, Niranjan Pandey, Robert H. Gruninger, Mary Adams, Sandra Moreno-Mazza, Angel R. Fuentes-Pesquera, and Steven A. Middleton. 4-amino-6-arylamino-pyrimidine-5-carbaldehyde hydrazones as potent erbb-2/egfr dual kinase inhibitors. *Bioorganic & Medicinal Chemistry Letters*, 18(16):4615–4619, 2008.

[26] Veli-Pekka Jaakola, Mark T. Griffith, Michael A. Hanson, Vadim Cherezov, Ellen Y. Chien, J. Robert Lane, Adriaan P. IJzerman, and Raymond C. Stevens. The 2.6 angstrom crystal structure of a human a 2a adenosine receptor bound to an antagonist. *Science*, 322(5905):1211–1217, 2008.

[27] Daniel Wacker, Gustavo Fenalti, Monica A. Brown, Vsevolod Katritch, Ruben Abagyan, Vadim Cherezov, and Raymond C. Stevens. Conserved binding mode of human β2 adrenergic receptor inverse agonists and antagonist revealed by x-ray crystallography. *Journal of the American Chemical Society*, 132(33):11443–11445, 2010.