HybTrans: A Molecular Generation and Optimization Model Integrating Protein Pocket Sequence and Structural Information

Yinghao Chen, Hongde Liu, Xiao Sun*

School of Biological Science and Medical Engineering, Southeast University, Jiangning, Nanjing, Jiangsu, China

Abstract: Deep learning models for drug discovery offer broader chemical space exploration and generate more novel molecules compared to traditional methods. Deep learning models currently used in drug design can be divided into two main categories: those based on ligands and those based on protein pockets. However, there is a lack of models that integrate both protein pocket sequence and structural information for molecular generation. This study proposes HybTrans, a molecular generation and optimization model that integrates protein pocket sequence and structural information. The model includes a sequence encoding module, a structure encoding module, a small molecule Self-Referencing Embedded Strings (SELFIES) representation module, a fusion decoding module, and a reinforcement learning module. The evaluation results demonstrate that the molecules produced by HybTrans have an average diversity of 0.815 and a docking affinity score of -7.828 kcal/mol (Vina), surpassing the performance of similar models. Additionally, the generated molecules exhibit high drug-likeness (QED), synthesizability (SA), and Lipinski's rule of five scores (Lipinski). Ablation studies demonstrate the importance of key modules such as the fusion decoding module and the SELFIES representation module. A case study on the p21-activated kinase (PDB ID: 5i0b) binding pocket shows that HybTrans is capable of capturing the interactions between protein pockets and drug molecules, generating small molecules with high target affinity and excellent molecular properties.

1 Introduction

In the process of drug discovery and design, screening or designing candidate small molecules that bind to drug targets is crucial. It is estimated that the chemical space of small molecules contains between 10^{23} and 10^{60} combinations[1], making it extremely challenging and time-consuming to search for suitable small molecules in such a high-dimensional space. Traditional computer-aided drug design (CADD) methods, such as high-throughput screening[2] and virtual screening[3], search for candidate molecules within predefined compound libraries. However, these methods are limited in the chemical space they can explore and often fail to generate novel molecules[4]. In recent years, deep generative models have been widely applied in computer vision and natural language processing, and AI Drug Design (AIDD) has become a hot topic in drug discovery.

Deep learning-based molecular generation models can be categorized into sequence-to-sequence neural networks (Seq2seq), generative adversarial networks (GAN), and variational autoencoders (VAE)[5]. These models can be trained using preprocessed protein and small molecule data. For proteins, the sequence information typically refers to the primary structure, i.e., the amino acid sequence, while the structural information includes residue types, connectivity, and relative distances. For small molecules, the most commonly used representation method is the Simplified Molecular Input Line Entry System (SMILES) [6].

From medicinal chemistry perspective, molecular generation models can be divided into ligand-based and protein pocket-based models. Ligand-based models learn hidden chemical rules from existing small molecule structures and generate new molecules by sampling from the learned distribution. However, since these models do not consider protein pocket information, they cannot guarantee that the generated molecules will have good binding affinity for specific protein targets. Moreover, these models are highly dependent on target-specific ligand datasets, limiting their generalizability and making it difficult to generate reasonable candidate molecules for novel target proteins[7]. In contrast, protein pocket-based models use various representation methods to extract sequence or structural information from protein pockets, enabling the generation of molecules with high affinity for specific protein targets [8] .

Reinforcement learning for optimizing molecular properties is an essential module in molecular generation and optimization models. The goal of reinforcement learning is to train an agent (usually a pre-trained model in molecular generation tasks) to optimize target properties based on the objectives and reward functions provided by the environment. In molecular generation tasks, molecular property evaluation functions can be used to fine-tune the model within a reinforcement learning

* Corresponding author: xsun@seu.edu.cn

framework, thereby optimizing the properties of the generated molecules[9], such as drug-likeness (QED score), synthesizability (SA score), and Lipinski's rule of five score (Lipinski)[10].

In current protein pocket-based molecular generation models, the representation of protein pocket structures remains challenging. There is a lack of models that integrate both protein pocket sequence and structural information, which could help capture the essential features of protein pockets more comprehensively. Therefore, this study proposes HybTrans, a Transformer-based molecular generation and optimization model that integrates protein pocket sequence and structural information. The model uses the SELFIES representation for small molecules to generate candidate drug molecules with high pocket-ligand affinity and employs reinforcement learning to further enhance the drug-like characteristics of the generated molecules, ultimately producing small molecules with high target affinity and excellent molecular properties.

2 Materials and Methods

The formal description of a protein pocket-based molecular generation model is as follows: given a new protein pocket p, generate a set of new molecules {m_k, k=1, 2, …} that bind to the protein pocket p with high affinity and possess desirable physicochemical properties. Inspired by the Transformer model[11], this study treats this task as a translation process from proteins to small molecules. Based on this, we constructed HybTrans, a molecular generation and optimization model that integrates protein pocket sequence and structural information. The model consists of a sequence encoding module, a structure encoding module, a small molecule encoding module, a fusion decoding module, and a reinforcement learning module, as shown in Figure 1.

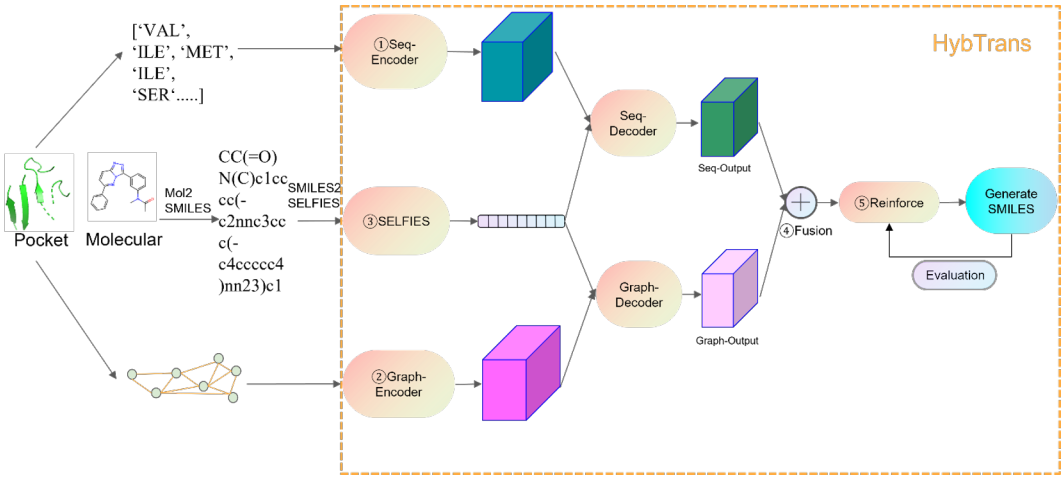

Figure 1. HybTrans model architecture diagram. The model mainly includes the following modules: (1) a sequence encoding module based on the standard Transformer encoder; (2) a structure encoding module that represents the 3D structure of the protein pocket as a graph, innovatively introducing random walk positional encoding and graph attention mechanisms into the graph encoder; (3) a small molecule SELFIES encoding module that uses self-referring embedded strings (SELFIES) to represent small molecules; (4) a fusion decoding module that combines the sequence and structure outputs from the encoder with the SELFIES representation of small molecules, decodes them through the Transformer decoder to obtain sequence and graph decoding outputs, and then uses an attention mechanism to generate the final SELFIES characters based on the vocabulary probability distribution; (5) a reinforcement learning module that uses policy gradient methods to optimize the physicochemical properties of the generated molecules.

2.1 Protein Pocket Representation

2.1.1 Protein Pocket Sequence Representation

This study represents the 20 types of amino acid residues that make up the protein pocket as a vocabulary and encodes the protein pocket amino acid sequence as a one-dimensional real-valued vector. To normalize the length of all data, the maximum length is set to the length of the longest sequence, and shorter sequences are padded with zeros at the end to establish the sequence structure representation of the protein pocket:

$$S_{norm} = (a_1, a_2, \ldots, a_i, \ldots, a_n, 0, 0, \ldots, 0), a_i \in \{1, 2, 3, \ldots, 20\} \quad (1)$$

2.1.2 Protein Pocket Structural Representation

For a protein pocket p, this study represents its structure as a graph $G = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the set of nodes, $\mathcal{V} = \{(n_i, p_i)\}_{i=1}^{n}$, with n_i representing a single amino acid residue node, $p_i \in \mathbb{R}^3$ representing the 3D coordinates of its α carbon atom, and n being the number of nodes. $\mathcal{E}$ represents the set of edges between nodes, $\mathcal{E} = \{e_{ij}, i, j = 1, 2, \ldots, n \,\&\, i \neq j\}$. In the graph $G = (\mathcal{V}, \mathcal{E})$, each node n_i is represented as a one-hot encoded vector x_i based on the 20 amino acid types, and edges e_{ij} are constructed using the k-nearest neighbors (KNN) algorithm. The KNN algorithm selects the k nearest

neighbor nodes $\{n_i \in \mathcal{N}_i\}$ based on the Euclidean distance $d_{ij} = p_i - p_{j2}$ between nodes i and j, where $\mathcal{N}_i$ is the neighborhood of node n_i, thus establishing the edge connections and representing them as an adjacency matrix[12, 13]. The above steps construct the structural graph representation of the protein pocket p:

$$G = (\mathcal{V}, \mathcal{E}) \tag{2}$$

2.2 Protein Pocket Sequence Encoding Module

The sequence encoding module is based on the standard Transformer encoder[11]. First, the input sequence S_{norm} is embedded and mapped to an N-dimensional space, where N is the hidden layer dimension of the encoder. The embedded representation is then added to the positional encoding matrix and fed into the Transformer encoder. After passing through 6 Transformer encoder layers, the final sequence encoding result is obtained:

$$s^{(L)} = SeqEncoder\left(S_{norm}\right) \tag{3}$$

2.3 Protein Pocket Structure Encoding Module

In the structure encoding module, this study innovatively introduces random walk positional encoding[14] and graph attention mechanisms[15] into the graph encoder. The graph encoder better captures the structural information of the protein pocket, including node and edge information. The protein pocket structure encoding module ultimately encodes the graph output:

$$g^{(L)} = GraphEncoder\left(G\right) \tag{4}$$

2.3.1 Positional Encoding

In graph neural networks, Laplacian positional encoding is commonly used as node positional encoding. However, Laplacian positional encoding has limitations: due to the multiplicity of Laplacian matrix eigenvalues, the eigenvectors may be unstable, and sometimes it is not possible to find enough convergent eigenvectors to satisfy the conditions when computing the eigenvectors of sparse matrices. As a result, this research adopts random walk positional encoding (RWPE)[14] for node positional encoding.

The random walk positional encoding vector h_i^{RWPE} is determined through a k-step random walk process:

$$h_i^{RWPE} = \left[RW_{ii}, RW_{ii}^2, RW_{ii}^3, \ldots, RW_{ii}^k \right] \in \mathbb{R}^k \tag{5}$$

Here, the random walk operator is represented by $RW_{ii} = AD^{-1}$, the adjacency matrix A of the graph G, and the degree matrix by D. This is a low-complexity random walk matrix algorithm that solely considers the transition probability from node *i* to itself.

2.3.2 Graph Attention Mechanism

In graph neural networks, graph data is often directly read out as a vector. If this representation is used in Transformer-based generative models, the readout vector is used as the initial sequence embedding input to the Transformer encoder. Consequently, the model is unable to produce valid molecules. This is because the process of reducing high-dimensional graph representations to one-dimensional vectors may lose feature information[16]. Therefore, this study designs a graph Transformer encoder based on graph self-attention mechanisms.

In the graph Transformer encoder layer of this study, for node *i*, its representation is updated by calculating the sum of attention scores with all its neighbor nodes *j*. The attention matrix $Attn_{ij}^{(l)}$ is represented as:

$$Attn_{ij}^{(l)} = softmax\left(\frac{q_i^{(l)} (k_j^{(l)})^T}{\sqrt{d_{kj}}} \right) v_j^{(l)} \tag{6}$$

where $q_i^{(l)}$ represents the q vector of node *i*, and $k_j^{(l)}$ and $v_j^{(l)}$ represent the k and v vectors of neighbor node j, respectively.

2.4 Small Molecule SELFIES Representation Module

The commonly used one-dimensional representation method for small molecules is SMILES. However, SMILES does not guarantee that the symbol sequence is valid in terms of molecular syntax and chemical rules, leading to a low validity rate of generated molecules when used in generative models. Most generated molecular sequences do not conform to common chemical rules. Therefore, this study introduces a new representation method: self-referring embedded strings (SELFIES)[17], which is a 100% valid representation. Every SELFIES string represents a chemically valid molecule. SELFIES avoids generating molecules that do not conform to chemical rules by mapping each chemical symbol to a specific bracket structure, effectively solving the problem of generating unbalanced brackets or ring identifiers during the generation process.

2.5 Fusion Decoding Module

In the fusion decoding module, this study optimizes the Transformer decoder. First, the embedded encoding and positional encoding of the small molecule SELFIES representation are added as input to the decoder. Then, the sequence decoding output $s^{(L)}$ from the sequence encoder and the graph decoding output $g^{(L)}$ from the graph encoder are passed through 6 Transformer decoder layers to obtain the sequence and graph outputs:

$$s_d^{(L)}, g_d^{(L)} = Decoder\left[s^{(L)}, g^{(L)} \right] \tag{7}$$

The sequence and graph decoding outputs are then combined using the attention mechanism:

$$z = Attention\left[s_d^{(L)}, g_d^{(L)} \right] \tag{8}$$

The fused decoding output probability distribution z is obtained. The core principle of the attention mechanism

fusion is weighted summation, enabling the model to allocate varying weights to the sequence and graph outputs at different positions, reflecting the relative importance of each type of output. Finally, the normalized SELFIES character generation probability distribution is obtained through softmax normalization, i.e., the predicted probability distribution of possible characters in the vocabulary at each position of the sequence.

$$output\_prob = Softmax\left(Linear\left(z\right)\right) \quad (9)$$

During model training, this study employs the cross-entropy loss function to quantify the discrepancy between predicted and actual results.

2.6 Reinforcement Learning Module

In this study, the Transformer-based molecular generation model initially only considers the information of the protein pocket and small molecules. By approximating the molecular generation problem as a machine translation problem from protein pockets to small molecules, the Transformer model learns the interaction relationship between protein pockets and small molecules. However, the generated small molecules may not perform well in terms of molecular properties. Therefore, in the reinforcement learning module, this study uses the existing policy gradient method[18] to optimize the molecular properties. Since the molecules generated by the fusion model perform well in terms of Lipinski's rule of five score, the reinforcement learning optimization target is set to the sum of drug-likeness (QED) and synthesizability (SA) scores.

2.7 Generation Strategy and Evaluation Metrics

This study adopts four generation strategies: greedy search, beam search, Top-k sampling[19], and Typical sampling[20]. The first two methods are search methods that select the generated characters based on the vocabulary probability distribution output by the model. The latter two are sampling methods that construct a sampling space and sample characters from it based on the probability distribution. In this study, the parameter k for the latter three methods is set to 5, meaning that the top five results with the highest probabilities from each method are selected as the generation results, which are then sorted based on Vina docking scores.

This study uses widely accepted evaluation metrics to assess the quality of the generated molecules, including: (1) Vina Score: the predicted binding affinity score between the small molecule and the protein pocket[21]; (2) Diversity: the dissimilarity (1 - Tanimoto similarity) between the generated molecules and reference molecules; (3) QED Score: a simple quantitative estimate of drug-likeness based on several ideal molecular properties; (4) SA Score: an evaluation of the ease of synthesis; (5) Lipinski Score: a measure of how many of Lipinski's rule of five the generated molecules satisfy, which is an empirical rule for assessing whether a molecule can become a drug. All metrics are calculated using Autodock Vina[21] and the Python RDKIT package[22].

3 Results and Discussion

3.1 Dataset and Parameter Settings

This study uses the CrossDocked2020 dataset[23] and follows the same filtering and splitting strategy as previous work[24] using MMseqs2 to split the data with 30% sequence identity. This results in a training dataset comprising 100,000 protein-ligand pairs and a test dataset of 100 protein-ligand pairs. In the training set, 99,900 pairs are used for training, and 100 pairs are used as the validation set. In the test set, 100 proteins are used for molecular generation experiments, and the performance of the generated molecules is evaluated.

This study uses the PyTorch deep learning framework[25] to implement the model code. The random walk step length is set to 8, generating an 8-dimensional graph positional encoding vector. The Q, K, and V vector lengths are all 64, the number of multi-head attention heads is 8, the fully connected layer dimension is 2048, and the number of encoder and decoder layers is 6, with the hidden layer dimension set to 512. The training hyperparameters are shown in Table 1.

Tabel 1. Hyperparameter settings

Hyperparameter	Value
Total training epochs	100
Batch size	4
Optimizer	Momentum SGD
Momentum value	0.99
Training learning rate	1×10^{-4}
Learning rate decay	Learning rate decay based on validation loss
Training weight decay coefficient	1×10^{-3}
Reinforcement learning rate	1×10^{-6}
Reinforcement learning weight decay coefficient	1×10^{-4}

3.2 Model Performance Comparison

This study compares the HybTrans model with recently published protein pocket-based molecular generation models: 3D-SBDD[26], Pocket2Mol[27], and DiffSBDD[24]. 3D-SBDD develops a 3D generative model to estimate the likelihood of atoms appearing in 3D space and uses an autoregressive sampling algorithm to generate new molecules from atomic point clouds. Pocket2Mol creates an E(3) equivariant neural network to produce 3D coordinates of small molecules in a continuous space. DiffSBDD designs an E(3) equivariant conditional diffusion model to generate new ligand molecules based on protein pocket conditions. For these three models, the default parameter values from the original papers are used for comparison. The comparison results are shown in Table 2.

Table 2. Model performance comparison results

	Vina↓	Diversity↑	QED↑	SA↑	Lipinski↑
Test Set	-7.158 ±2.10	/	0.476 ±0.20	0.655 ±0.13	4.340 ±1.14
3D-SBDD	-5.888 ±1.91	0.742± 0.19	0.502 ±0.17	0.675 ±0.14	4.787 ±0.51
Pocket2Mol	-7.288 ±2.53	0.312± 0.14	0.563 ±0.16	**0.689 ±0.13**	**4.902 ±0.42**
DiffSBDD	-7.333 ±2.56	0.758± 0.05	0.467 ±0.18	0.554 ±0.12	4.702 ±0.64
HybTrans	**-7.828 ±1.46**	**0.815± 0.13**	**0.573 ±0.16**	0.683 ±0.16	4.900 ±0.33

Note: Test Set represents the average metrics of the test set. ↑ indicates that the higher the value, the better, while ↓ indicates the opposite. Bold values represent the best performance among all models.

These findings indicate that the HybTrans model surpasses other models in molecular diversity and Vina docking affinity score (Vina Score), demonstrating its superior performance in generating molecules with high protein binding affinity and diversity. In terms of molecular property metrics, the HybTrans model generates molecules with higher drug-likeness scores (QED Score) compared to other models, while the synthesizability scores (SA Score) and Lipinski's rule of five scores are very close to the optimal values. Additionally, the average inference time of the HybTrans model is as low as 5.088 seconds per molecule, significantly lower than that of Pocket2Mol and 3D-SBDD models (25.035 seconds and 196.590 seconds per molecule, respectively), and not far from the inference time of the DiffSBDD model (1.603 seconds per molecule). This indicates that the HybTrans model has significant advantages in inference efficiency, and its generation process does not rely on GPU resources, further enhancing its practicality. Therefore, the model performance comparison results validate the superiority of the HybTrans model in molecular generation, molecular properties, and computational efficiency.

3.3 Ablation Study

In the ablation study, to thoroughly examine the influence of each module on the overall performance of the HybTrans model, this study constructs four variant models by gradually removing modules. Additionally, to better demonstrate the role of the small molecule SELFIES representation module, this study introduces the validity (valid) metric. Validity refers to the proportion of generated molecules that can be parsed by the Python RDKIT package as chemically valid molecules, reflecting the correctness of the generated molecules in terms of chemical rules. The results of the ablation study are illustrated in Figure 2.

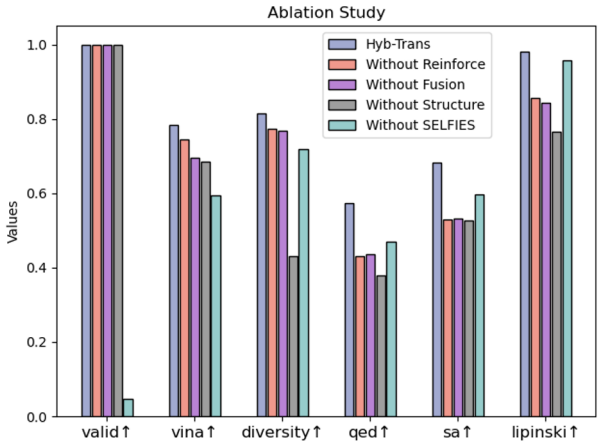

Figure 2. Ablation experiment results. To unify the values of different metrics within the same range, some metrics are preprocessed: Vina Score is divided by -10, Lipinski Score is divided by 5. Without Reinforce indicates the removal of the reinforcement learning module; Without Sequence indicates the removal of the protein pocket sequence decoding module, retaining only the protein pocket structure encoding module; Without Structure indicates the removal of the structure encoding module, retaining only the protein pocket sequence encoding module; Without SELFIES indicates the removal of the small molecule SELFIES representation module, using SMILES to represent small molecules.

The results show that the absence of the small molecule SELFIES representation module significantly reduces the validity of the generated molecules, indicating that the SELFIES representation module plays a crucial role in ensuring chemical rule correctness and improving model stability. Removing the reinforcement learning module significantly reduces the optimization effect on molecular properties, indicating that the reinforcement learning module is essential for generating high-quality molecules. The absence of the protein pocket structure encoding module greatly reduces the docking affinity score, and the absence of the protein pocket sequence encoding module also has a certain impact, indicating that the fusion of sequence and structural information is indispensable for representing the binding relationship between small molecules and protein pockets. In conclusion, the ablation study demonstrates that each component of the model contributes to the final model performance, including the reinforcement learning module, fusion decoding module, structure encoding module, and small molecule SELFIES representation module.

3.4 Case Study

This study selects the p21-activated kinase protein pocket (PID: 5I0B) from the test set for a case study. p21-activated kinases (PAKs) belong to the ste20 serine/threonine protein kinase family and are involved in regulating biological processes such as cytoskeleton reorganization, cell motility, cell proliferation, and cancerous transformation, which are closely related to tumorigenesis and metastasis. Studies have shown that PAK protein mutations are widely detected in various

tumor tissues, making them a new potential therapeutic target[28]. A case study on this protein can provide in-depth insights into the biological significance of the HybTrans model in protein pocket-based molecular generation and further validate the interpretability of its generation results[27]. After running the HybTrans model, the top five molecules with the highest pocket-ligand affinity scores (Vina Score) are selected for the case study, as shown in Figure 3.

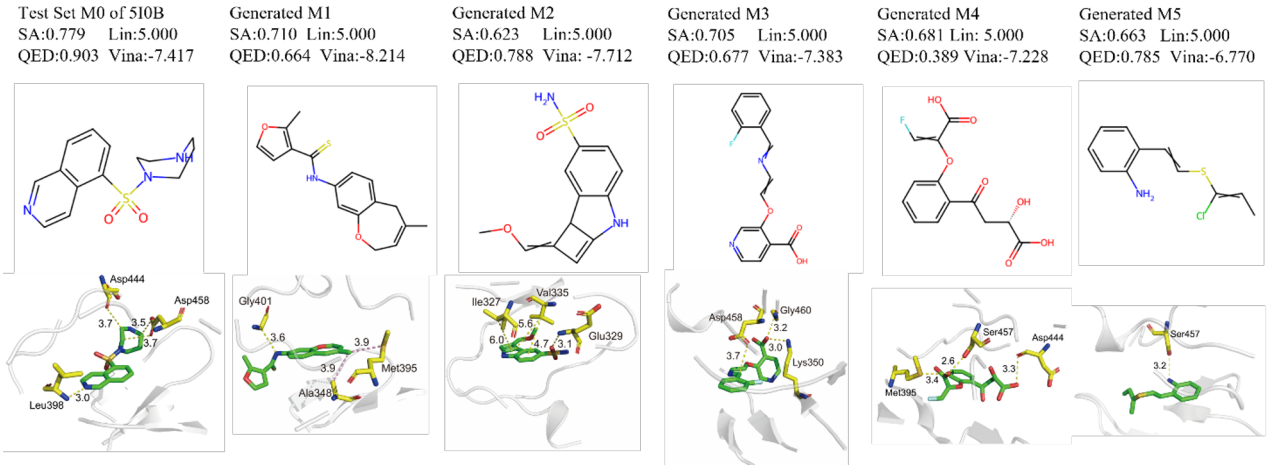

Figure 3. Molecular generation results for the p21-activated kinase protein pocket (PID: 5I0B). The reference molecule from the test set is located in the leftmost column, and the five generated molecules are arranged from left to right in descending order of Vina docking scores. To comprehensively evaluate the properties of the generated molecules, the binding affinity (Vina) and three key drug-like property metrics (QED, SA, and Lipinski scores) are annotated for each molecule.

Analysis of the Autodock Vina docking results between the generated molecules and the protein pocket shows that the reference molecule forms stable polar hydrogen bonds with three residues (Asp-444, Asp-458, and Leu-398) in the protein pocket. The molecules generated by HybTrans not only show the polar connections formed by the reference molecule (M3 with Asp-458 and M4 with Asp-444) but also form additional polar and non-polar connections with surrounding residues. For example, M1 forms polar connections with Gly-401, M2 with Ile327, Glu-329, and Val335, M3 with Lys-350 and Gly-460, M4 with Met-395 and Ser-457, and M5 with Ser-457. Additionally, M1 forms non-polar connections with Ala348 and Met395. These results indicate that the molecules generated by the model show more diverse interaction patterns within the binding pocket.

The case study results demonstrate that the HybTrans model can generate new molecules with both high binding affinity and excellent drug-like properties, fully showcasing the model's capability in generating high-quality molecules.

4 Conclusion

This study proposes HybTrans, a molecular generation and optimization model that integrates protein pocket sequence and structural information. The model consists of a sequence encoding module, a structure encoding module, a small molecule SELFIES encoding module, a fusion decoding module, and a reinforcement learning module. Comparative evaluation with the latest deep generative models shows that the HybTrans model outperforms similar models in terms of molecular diversity and docking affinity scores, while also generating molecules with excellent drug-like properties. Additionally, ablation studies and case studies further validate the scientific and practical value of the model. This study provides a creative and practical new approach for molecular generation models in the field of de novo drug design, significantly improving drug discovery efficiency.

References

1. Polishchuk, P.G., T.I. Madzhidov, and A.J.J.o.c.-a.m.d. Varnek, *Estimation of the size of drug-like chemical space based on GDB-17 data.* 2013. **27**: p. 675-679.
2. Macarron, R., et al., *Impact of high-throughput screening in biomedical research.* 2011. **10**(3): p. 188-195.
3. Schneider, G. and H.-J.J.D.d.t. Böhm, *Virtual screening and fast automated docking methods.* 2002. **7**: p. 64-70.
4. Li, J.-N., et al., *CProMG: controllable protein-oriented molecule generation with desired binding affinity and drug-like properties.* 2023. **39**(Supplement_1): p. i326-i336.
5. Pang, C., et al., *Deep generative models in de novo drug molecule generation.* Journal of Chemical Information and Modeling, 2023. **64**(7): p. 2174-2194.
6. Irwin, R., et al., *Chemformer: a pre-trained transformer for computational chemistry.* 2022. **3**(1): p. 015022.
7. Isert, C., K. Atz, and G. Schneider, *Structure-based drug design with geometric deep learning.* Current Opinion in Structural Biology, 2023. **79**: p. 102548.
8. Wang, L., et al., *Lingo3dmol: Generation of a pocket-*

based 3d molecule using a language model. 2023.

9. Li, C., et al. *Transformer-based Objective-reinforced Generative Adversarial Network to Generate Desired Molecules*. in *IJCAI*. 2022.
10. Du, Y., et al., *Machine learning-aided generative molecular design.* Nature Machine Intelligence, 2024. **6**(6): p. 589-604.
11. Zhang, S., et al., *Applications of transformer-based language models in bioinformatics: a survey.* Bioinformatics Advances, 2023. **3**(1): p. vbad001.
12. Rampásek, L., et al., *Recipe for a general, powerful, scalable graph transformer, 2022*. CoRR.
13. Veličković, P., *Everything is connected: Graph neural networks.* Current Opinion in Structural Biology, 2023. **79**: p. 102538.
14. Black, M., et al., *Comparing graph transformers via positional encodings.* arXiv preprint arXiv:2402.14202, 2024.
15. Müller, L., et al., *Attending to graph transformers.* arXiv preprint arXiv:2302.04181, 2023.
16. He, L., et al., *High-order graph attention network.* Information Sciences, 2023. **630**: p. 222-234.
17. Fang, Y., et al., *Domain-agnostic molecular generation with chemical feedback.* arXiv preprint arXiv:2301.11259, 2023.
18. Mazuz, E., et al., *Molecule generation using transformers and policy gradient reinforcement learning.* Scientific Reports, 2023. **13**(1): p. 8799.
19. Pandey, R., et al., *Generative AI-based text generation methods using pre-trained GPT-2 model.* arXiv preprint arXiv:2404.01786, 2024.
20. Meister, C., et al., *Locally typical sampling.* Transactions of the Association for Computational Linguistics, 2023. **11**: p. 102-121.
21. Che, X., Q. Liu, and L. Zhang, *An accurate and universal protein-small molecule batch docking solution using Autodock Vina.* Results in Engineering, 2023. **19**: p. 101335.
22. Landrum, G., *Rdkit: Open-source cheminformatics software.* 2016.
23. Francoeur, P.G., et al., *Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design.* 2020. **60**(9): p. 4200-4215.
24. Schneuing, A., et al., *Structure-based drug design with equivariant diffusion models.* 2022.
25. Paszke, A., et al., *Pytorch: An imperative style, high-performance deep learning library.* Advances in neural information processing systems, 2019. **32**.
26. Luo, S., et al., *A 3D generative model for structure-based drug design.* Advances in Neural Information Processing Systems, 2021. **34**: p. 6229-6239.
27. Peng, X., et al. *Pocket2mol: Efficient molecular sampling based on 3d protein pockets*. in *International Conference on Machine Learning*. 2022. PMLR.
28. Park, J.K., et al., *The discovery and the structural basis of an imidazo [4, 5-b] pyridine-based p21-activated kinase 4 inhibitor.* Bioorganic & medicinal chemistry letters, 2016. **26**(11): p. 2580-2583.