

An overview of Pocket-based drug design

Pablo Varas Pardo

March 2024

1 Introduction

In recent years, advances in artificial intelligence (AI) techniques have greatly impacted the field of drug discovery (1). The rise of deep learning has allowed computational drug design to become increasingly a reality. One of the main advantages of using these algorithms is the possibility of exploring a chemical space that is immense(2). Researchers have tried to estimate the size of the chemical space, which refers to the number of possible compounds that could be synthesized. However, there is no scientific consensus on the exact number, with some estimating it to be between 10^{30} to 10^{60} (3). The discrepancy in the estimations is due to different criteria applied in the studies, such as the maximum size of the molecules, the types of atoms that compose them, and the presence of physicochemical restrictions like the Lipinski rules (4, 5, 6).

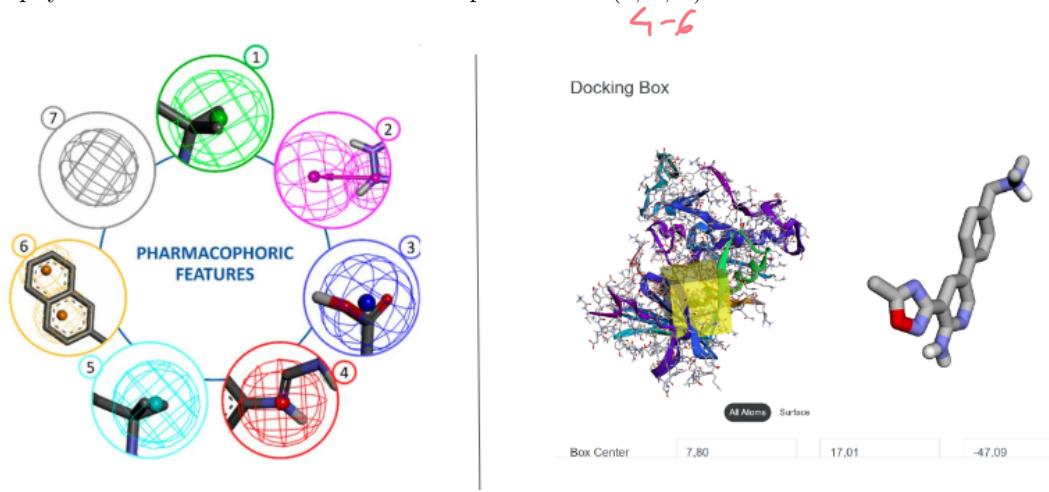


Figure 1: Ligand-based vs Pocket-based drug design. Adapted from (7)

The vast search space poses a significant challenge for optimizing or searching molecules with specific properties. As a result, the development of methods to navigate this space has gained considerable interest in recent years. Several techniques are now available to explore the huge chemical space and identify new potential drug candidates. As shown in Figure 1, we can divide existing methods into two families: ligand and pocket-based. Ligand-based methods use a set of high-affinity molecules to a target protein as input to create new ligands(8, 9). This is useful when

the research focuses on a narrow chemical space, so the corresponding algorithms generate new potential drug candidates around a singular point in the search space that represents a molecule.

On the other hand, pocket-based methods ~~are~~ used to generate appropriate ligands for protein pockets by utilizing detailed structural information of the target binding site (10). For this reason, pocket-based algorithms are highly effective when the three-dimensional structure of the protein is available, allowing for precise molecular docking and virtual screening to design molecules that fit within the structural constraints of the binding site. Such approaches can accurately model interactions between the ligand and the protein, simplifying the design of molecules that are optimally configured to the target site.

This article will focus on current generative models based on the pocket structure of proteins and the evaluation metrics of the molecules, corresponding to steps 3 and 4 of Figure 2. ~~This is~~ because they can generate completely novel molecules given a protein conformational structure, making them especially useful for creating new drugs that inhibit certain proteins involved in the appearance of diseases.

One of the main problems with pocket-based algorithms is the lack of experimental validation (11). Therefore, developing a protocol to select the best candidates for subsequent synthesis in the laboratory is essential ¹.

In summary, the main contributions of this article are:

- To create a common framework of current molecular generation methods based on ~~protein pockets~~ *pocket-based algorithms*
- The benchmarking of these methods to assess which are the most promising in pocket-based drug design, creating a reference framework for evaluating the additional algorithms.
- To develop a protocol to filter optimal compounds given a target protein.
- To validate experimentally pocket-based algorithms to generate potential drug candidates for DYRK1A protein, an enzyme directly implied in Alzheimer's Disease (AD).

¹All the code used in this article can be found in <https://github.com/pvaras8/pocketdrugdesign>.

+ We choose these type algorithm because....?

↓
Faster less & works are less

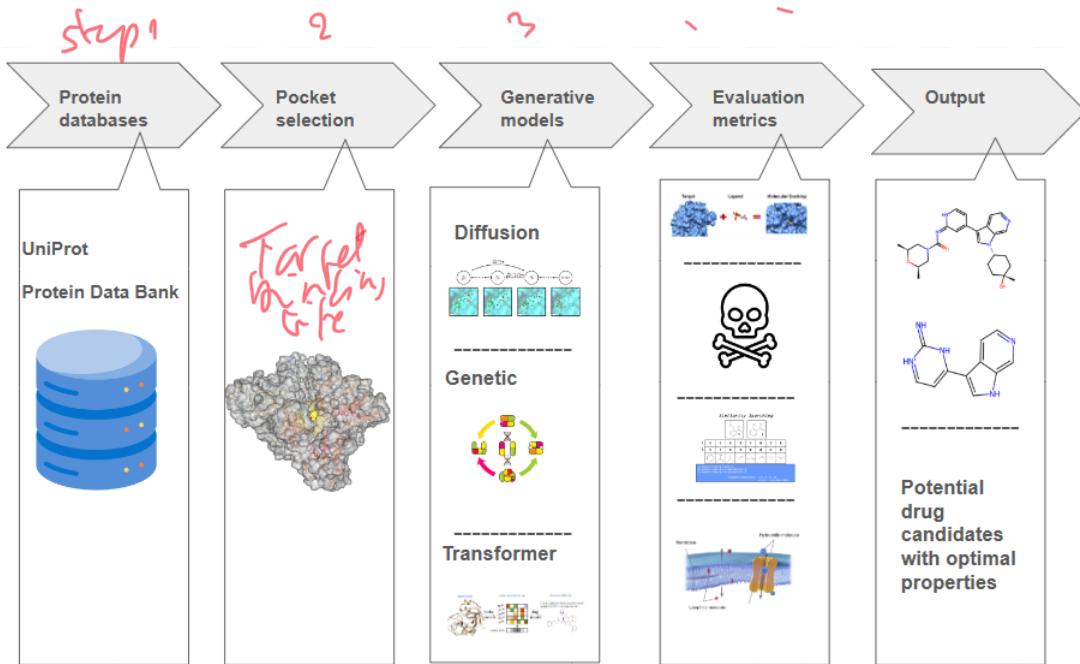


Figure 2: Computational Drug Discovery Pipeline

2 Pocket-based generative models

Protein pocket-based models are currently being used to generate new ligands and explore vast chemical space. This section examines key technical features of major models.

2.1 Transformer based models

Transformer-based algorithms are gaining popularity in drug discovery because they can capture complex relationships between training data through the attention mechanism and generate high-affinity ligands for a given protein using a sequence-based approach (11).

These models use a tokeniser to convert the SMILES code of the molecule and the protein's amino acid chain into tokens. The ligand and protein embeddings are then obtained and used to train the weights of the multi-attention layers. Once trained, the model can generate molecules related to a given amino acid sequence.

Following this approach, Lingo3DMol (10) proposes a new representation for SMILES called FSMILES, which fragments the SMILES into molecular groups according to certain rules. The protein pocket is then encoded through the encoder, and starting from the initially calculated growth point, the decoder builds the molecule by iteratively joining molecular groups produced by the decoder. DrugGPT (12) employs the GPT-2 model and the Byte Pair Encoding (BPE) tokeniser, which can represent a huge number of compounds using a limited vocabulary effectively, to explore the chemical space and discover new ligands for specific proteins.

TamGent (13) incorporates a variant of the Transformer encoder designed to process 3D geometric information of targets. Moreover, the study by Ang et al. (14) employs an Encoder-Decoder

in an effective
way?

Transformer combined with Reinforcement Learning through an Adaptive Monte Carlo Tree Search, emphasizing the generation of valid small molecules with desirable drug-like characteristics and binding affinities.

2.2 Diffusion based models

Diffusion models have attracted special interest in the last few years due to their capacity to generate novel compounds. These models create a Markov chain of progressive noising steps to add random Gaussian noise to real data until the original sample becomes unrecognizable. Consequently, a model is trained to reverse this process. Once trained, the model can generate new samples by sampling from a normal distribution and denoising this data until a new sample is created. In the field of pocket-based drug design, this denoising procedure is conditioned on the protein pocket for which the model will create new potential drug candidates (15).

?
atadir algo f,
no?

Compounds?
on or by.

One example is DiffSBDD (16), which has proved its capability to generate novel ligands with high predicted binding affinities to given protein pockets, highlighting its potential as a tool for molecules design in structure-based drug design. Moreover, TargetDiff (17) introduces an advanced diffusion model to generate molecules in the 3D space. This ensures that molecular generation is sensitive to the spatial conformation of protein targets.

2.3 Genetic algorithms

Genetic Algorithms (GA) implement heuristics inspired by natural evolutionary processes. They use mutation and/or crossover to explore the chemical space and maximize a target property, generally the docking value. Following this approach, after several generations, these algorithms can create novel ligands with high affinity given a target protein (18).

One example of GA for pocket-based drug design is AutoGrow4 (19), which generates novel drug-like molecules by dynamically applying a series of mutations and crossovers to an initial population of seed molecules. This process is further refined through a fitness function that ranks compounds based on their predicted binding affinities, continuously selecting the top performers for subsequent generations.

Another innovative approach, Reinforced Genetic Algorithm (RGA) (20), proposes a technique that relies on reinforcement learning to choose mutation and crossover operations. The goal is to maximize the docking value, which serves as the fitness function to optimize. The process starts with the selection of approximately 100 drug candidates from an initial database. These compounds are then subjected to mutation and crossover conditioned to the protein pocket to create new ones that improve the docking score.

2.4 Graph Neural Networks based models *in the data processing*

Graph neural networks have received special attention in recent years in the field of drug discovery. These models are very important when processing data based on structured graphs (21). In the field of drug discovery, molecules and proteins are represented as graphs denoted as $G = (V, E)$, where:

- V represents the nodes of the graph, each corresponding to an atom in the molecule or an amino acid protein.

* These factors are indicative

These factors are indicative of the ability of a compound to become a potential ~~hit~~ lead, an effective oral drug in humans

- E represents the edges of the graph, signifying the bonds between atoms or the sequential proximity between amino acids in proteins.

Following this approach, Pocket2Mol (22) introduces an equivariant generative network aiming to efficiently sample molecular structures based on the 3D structure of protein pockets. Its innovation lies in its dual-module design: a novel graph neural network capturing spatial and bonding relationships, and an efficient algorithm for conditional molecular sampling.

3 Main evaluation metrics

To create a benchmark for evaluating the different models the metrics that will be taken into account are:

yo crea que esto es
mejor en la parte exp-
rimental. (Filters)

- Virtual Docking. It predicts the interaction between a drug molecule and a target protein by simulating the binding affinity between the molecule and the amino acids (23). This helps to identify potential drug candidates by assessing how well they fit into the target site.
- Pharmacological activity. It measures the biological effects of a drug molecule on the body or specific cells or tissues. It evaluates the efficacy and the action mechanism of a compound in producing a therapeutic effect, such as inhibiting a disease-related enzyme or activating a receptor.

The continuous variable pChEMBL, defined as the $-\log_{10}$ of the molar concentration (IC_{50} , XC_{50} , EC_{50} , AC_{50} , Ki , Kd , or Potency), is associated with the pharmacological activity of a compound (24).

- Quantitative estimation of drug likeliness (QED). It measures the likelihood of a chemical compound to be a successful drug candidate based on its physicochemical properties. It quantifies drug-like properties as a single score, considering factors such as solubility, permeability, and molecular weight. (These factors are indicative of a compound's ability to be orally active in humans (25).)
- Lipophilicity (LogP). It measures the tendency of a compound to dissolve in fats, oils, and lipids over aqueous (water-based) solutions. It indicates the compound's ability to penetrate cell membranes, affecting its absorption, distribution, metabolism, and excretion properties. Often expressed as LogP, which is the logarithm of the partition coefficient between N-octanol and water (26).
- Molecular Diversity (Tanimoto Similarity). It assesses the structural variety among generated molecules and is calculated through the formula

$$\text{Tanimoto Similarity} = \frac{c}{a + b - c},$$

where a and b are the counts of features present in molecules A and B respectively, and c is the count of features common to both molecules. These features are represented by the Morgan Fingerprints, converting the molecular structure into vectors representing the presence or absence of certain chemical substructures in the molecule. A higher Tanimoto score indicates greater similarity (27).

- Molecular Weight. Its analysis is crucial for evaluating the molecule's atomic composition and to determine a drug's pharmacokinetics (26).
- Synthetic Accessibility Score (SAS). It is calculated as the sum of molecular fragment scores plus a complexity penalty based on the presence of certain molecular groups. This metric assesses the ease with which a chemical compound can be synthesized, providing crucial insights about the feasibility of its production on a larger scale (28).
- Toxicity. In the evaluation of potential drug candidates, assessing the toxicity profile is crucial to determine the safety and viability of molecules for further development. These toxicity classes include interactions with various nuclear receptors (e.g., androgenic, estrogenic), responses to environmental and endogenous stress signals (e.g., oxidative stress response, heat shock response), and effects on key cellular processes (e.g., DNA damage response) as shown in Table 1. Each class represents a distinct mechanism through which a compound might exhibit toxicological effects, providing a comprehensive view of its safety profile (29).

Table 1: Toxi . . .

Table 1: Toxicity classes.

Nuclear Receptor Panel (biomolecular targets)	Stress Response Panel
ER-LBD: estrogen receptor, luciferase	ARE: nuclear factor (erythroid-derived 2)-like 2 antioxidant responsive element
ER: estrogen receptor alpha aromatase	HSE: heat shock factor response element
AhR: aryl hydrocarbon receptor	ATAD5: genotoxicity indicated by ATAD5
AR: androgen receptor	MMP: mitochondrial membrane potential
AR-LBD: androgen receptor, luciferase	p53: DNA damage p53 pathway
PPAR: peroxisome proliferator-activated receptor gamma	

4 Case study

Alzheimer disease (AD)

In recent years, the DYRK1A enzyme has been identified as a promising target for therapeutic intervention in AD, as it is involved in multiple biological functions. Several studies have shown that DYRK1A undergoes alterations linked to the progression of AD, such as the phosphorylation of proteins like TAU and APP (30) (31). Therefore, DYRK1A is a highly promising enzyme as a therapeutic target for designing new drugs that could potentially be used to treat AD.

4.1 Selected Generative Models

The models used to develop new ligands against DYRK1A protein were DrugGPT, DiffSBDD, Lingo3DMol, Pocket2Mol and RGA. These models were chosen for their open-source nature, effectiveness and accessibility. As illustrated in Table 2, these models have been trained using different datasets, each with its unique size and composition of training compounds. The DrugGPT model utilizes a substantial dataset from 'jglaser/binding affinity' and 'ZINC 20', totalling 1.9 million and 2 billion compounds, respectively. In contrast, RGA, DiffSBDD and Pocket2Mol are trained on the

The flowchart used in this paper is shown in figure 3.

CrossDocked dataset, with DiffSBDD further supplemented by Binding MOAD with 41 thousand protein-ligand pairs for training and RGA with 250 thousand compounds of ZINC 15 database. By its way, Lingo3DMol is trained with 20 million commercially available compounds, and in its fine-tuning phase, uses a smaller subset of CrossDocked and DUD-E, focusing on precise adjustments with 11.8 thousand and 6.5 thousand compounds, respectively.

Table 2: Summary of models and databases used for training.

Model	Databases	Training Compounds
DiffSBDD	CrossDocked, Binding MOAD	100 K, 41 K
DrugGPT	jglaser/binding affinity, ZINC 20	1.9 M, 2 B
Lingo3DMol (fine-tuning phase)	CrossDocked, DUD-E	11.8 K, 6.5 K
Pocket2Mol	CrossDocked	22.5 M
RGA	CrossDocked, ZINC 15	22.5 M, 250 K

4.2 Experimental section

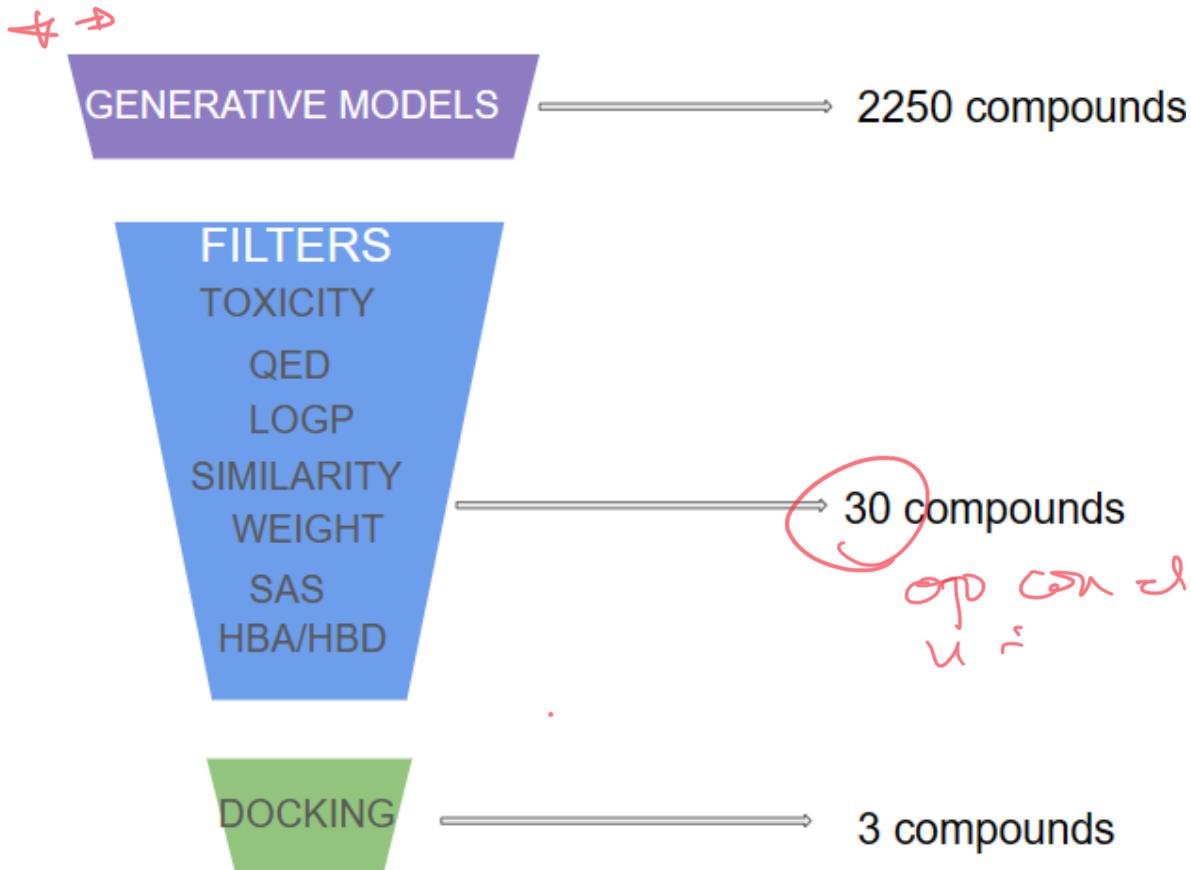


Figure 3: Candidates Selection Protocol

el sitio de unión se dirige como?

No se entiende bien

3D-structure of Prot - {PDB
Code...ref}

This section explains the process of obtaining compounds with optimal properties as shown in Figure 3. To begin with, we obtained the DYRK1A PDB from the Protein Data Bank (32). We extracted the binding site from the PDB by taking the average of the x, y, and z coordinates for the crystallographic ligand B5T of chain A, which were 7.80, 17.01, and -47.09 respectively. Next, we ran the five methods described in section 4.1 to obtain 2250 compounds for the given pocket. For DrugGPT and Lingo3DMol, the number of molecules generated can be chosen as a hyperparameter. For the other algorithms, the models were run 5 times to obtain a significant number of molecules for comparison.

The docking value for all molecules was obtained using Smina, provided by PyScreener (33), allowing efficient and flexible computation of docking scores.

Pharmacological activity was then calculated. We developed an algorithm using Morgan fingerprints to process the initial SMILES and a Gaussian process to obtain the pChEMBL value.

The evaluation metrics described in Section 3 were then applied. The parameters used to filter the molecules are shown in the Candidate Selection Protocol section. For toxicity prediction, the Chemprop algorithm (34) was used, which employs a directed message-passing neural network to predict molecular properties. The remaining properties were calculated using the RDKIT package. In the end, 30 molecules passed all the filters.

Eugenio

4.3 Results

This section provides an in-depth analysis of the performance of the molecular generation models. The evaluation includes the metrics described in Section 3, designed to provide a comprehensive view of each model's capabilities. Taken together, these metrics provide insight into the ability of the models to generate structurally innovative and diverse molecules that may be potential therapeutic agents. The results are critical to understanding the current state of the art in molecule generation algorithms and highlight strengths and areas for improvement for each model.

First, it is important to understand whether these models are exploring different parts of chemical space, or whether they are focusing on a narrow part. Figure 4 uses a t-distributed Stochastic Neighbour Embedding (t-SNE) plot to illustrate the molecular diversity generated by different computational models to help answer this question. Each point in the plot represents a molecule, with the proximity between points corresponding to their structural similarity.

The distinct clustering of points suggests that each model has a unique signature in terms of molecular generation, with some models, as indicated by the concentrated clusters, producing molecules with higher structural homogeneity. In contrast, the more dispersed clusters suggest models that generate a more structurally diverse set of molecules. The presence of distinct and well-defined clusters also implies that certain models may specialize in particular regions of the chemical space, potentially aligning with specialized drug discovery objectives. The original molecules are marked separately, serving as a baseline reference for the diversity introduced by each model. Overall, this figure highlights the importance of diversity in molecular design and the capacity of different models to explore the vast expanse of chemical space.

The RGA model produces molecules that are grouped in a molecular space quite similar to that of the original database. This is expected since the genetic algorithm forms molecules by molecular subgroups.

Similarly, the DIFSBDD molecules are grouped in a specific chemical space, similar to those of DrugGPT. On the other hand, both Pocket2Mol and Lingo3DMol molecules achieve greater

molecular diversity by exploring different areas of chemical space that are distant from the molecules in the original database.

This is related to Figure 6, where it is observed that DIFFSBDD produced a greater number of non-toxic molecules than the other models. As this model generates molecules over a narrow chemical space, likely these compounds do not have toxic molecular features.

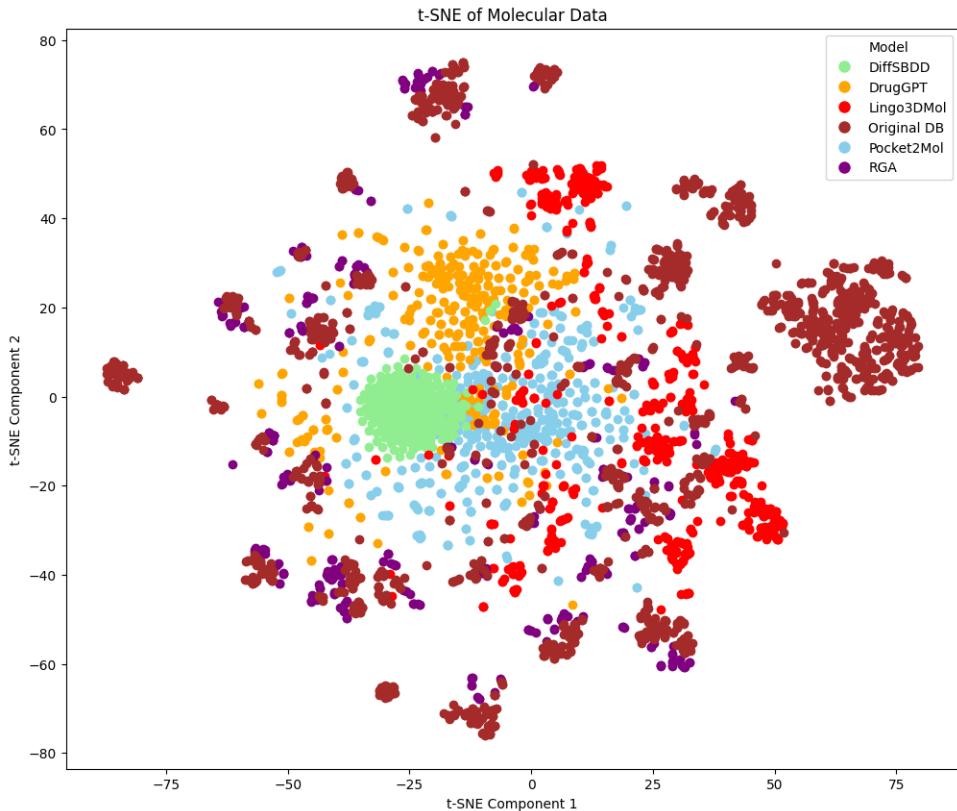


Figure 4: Diversity Comparison by Model

By its way, Table 3 showcases the algorithmic performance in different evaluation metrics. The results underscore the efficacy of Pocket2Mol in docking score averages and top-ranking molecules, suggesting a superior fit for potential drug candidates within the target protein's binding site. Meanwhile, RGA distinguishes itself in predicting binding affinities, hinting at its utility in identifying potent drug molecules.

¿ Cuáles usamos para los tres?
 Hechas que identifican ~~los~~ moléculas
 & cede otros.

Table 3: Models performance evaluation²

	DiffSBDD	DrugGPT	Lingo3DMol	Pocket2Mol	RGA
Docking (Mean ↓)	-6.77	-8.72	-8.71	-9.21	-8.53
Docking Top1 (↑)	-9.2	-10.80	-11.30	-12.51	-11.2
Docking Top10 (↓)	-8.77	-10.47	-10.64	-12.08	-10.61
pCHEMBL (Mean ↑)	5.87	5.99	6.08	5.96	6.18
pCHEMBL Top1 (↑)	6.83	7.08	8.03	7.13	9.01
pCHEMBL Top10 (↑)	6.59	6.86	7.69	6.89	8.22
logP (Mean 2-4)	0.43	3.98	2.46	2.68	2.80
QED (Mean ↑)	0.48	0.53	0.65	0.67	0.62
SAS (Mean ↓)	4.76	2.57	2.83	3.25	3.52
Validity (↑)	1	1	0.99	1	0.85
Molecular Weight (< 500)	268.67	418.80	335.16	309.73	314.41
Molecules evaluated	271	400	600	562	416

explicar

Figure 5 presents a visual comparison of the docking score distributions for the five molecular generation models. The median docking score, indicated by the horizontal line within each boxplot, serves as a robust indicator of the central tendency among the scores achieved by each model. To determine if there were significant differences in docking scores among the five models, we conducted a one-way ANOVA, which indicated a significant difference ($p < 0.001$). Given this result, we performed Dunn's post-hoc test with Bonferroni correction. The results showed significant differences in docking scores between most pairs of models, except between DrugGPT and Lingo3DMol. Notably, Pocket2Mol achieves the most favourable median docking score, suggestive of its adeptness in generating molecules that could form stable complexes with target proteins. The spread and range of scores, as denoted by the boxes and whiskers, also offer insights into the consistency and reliability of each model's predictions. For instance, while RGA shows a comparatively tight distribution signalling consistency in scoring, Lingo3DMol demonstrates a broader range, indicating greater variability in its docking score predictions. Outliers, represented by individual dots, highlight exceptional cases where molecules exhibit either particularly high or low docking scores compared to the typical range for the model output. Collectively, this graphical representation underscores the performance landscape of these models, guiding researchers to prudent selections.

² ↓ indicates better performance with lower values; ↑ indicates better performance with higher values. Top1 represents the best molecule, Top10 represents the top 10 molecules, and mean represents the mean for all the molecules.

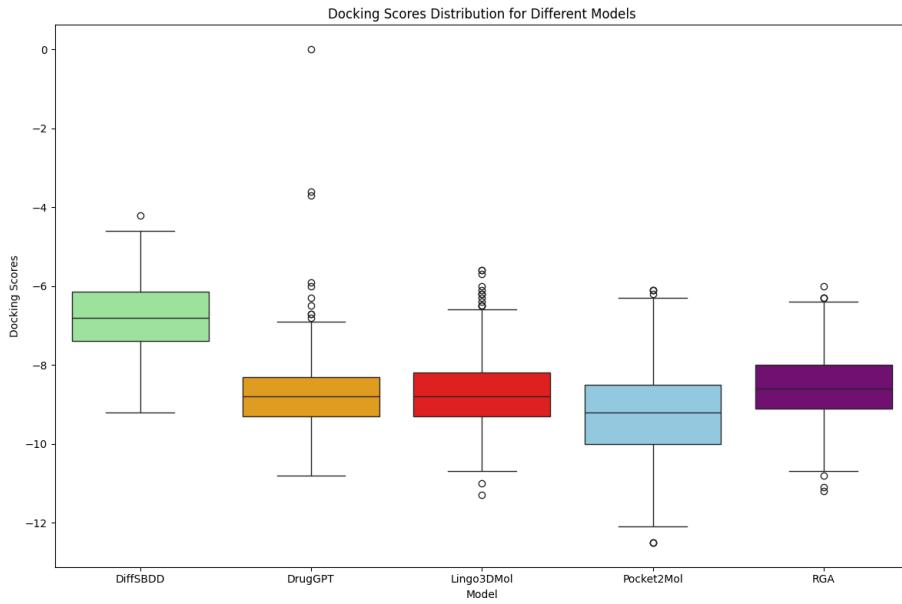


Figure 5: Docking Score Comparison by Model

The distribution of ring sizes across the generated molecules through the five models, as presented in Table 4, offers a glimpse into the structural diversity each algorithm brings to the table. The models display a varied propensity towards certain ring sizes, which could be indicative of their bias towards or against specific molecular frameworks. This diversity in structural motifs is essential in the exploration of the vast chemical space.

Table 4: Ring size distribution

Ring Size	DrugGPT	DiffSBDD	RGA	Pocket2Mol	Lingo3DMol
3	0.27%	36.71%	1.36%	0.05%	0%
4	0.27%	5.27%	0.41%	0%	0.28%
5	25.72%	21.94%	31.45%	23.96%	36.44%
6	72.61%	30.38%	66.78%	72.21%	63.14%
7	0.94%	4.22%	0%	2.37%	0.14%
8	0.07%	0.84%	0%	0.33%	0%
9	0%	0.42%	0%	0.09%	0%
≥ 10	0.20%	1.91%	0%	2.15%	0%

Figure 6 illustrates the percentage of molecules across non-toxic classes for different models. Models like DiffSBDD and DrugGPT demonstrate a substantial percentage of molecules falling into non-toxic classes, particularly in higher non-toxic class counts, suggesting a favourable safety profile for subsequent drug development stages.

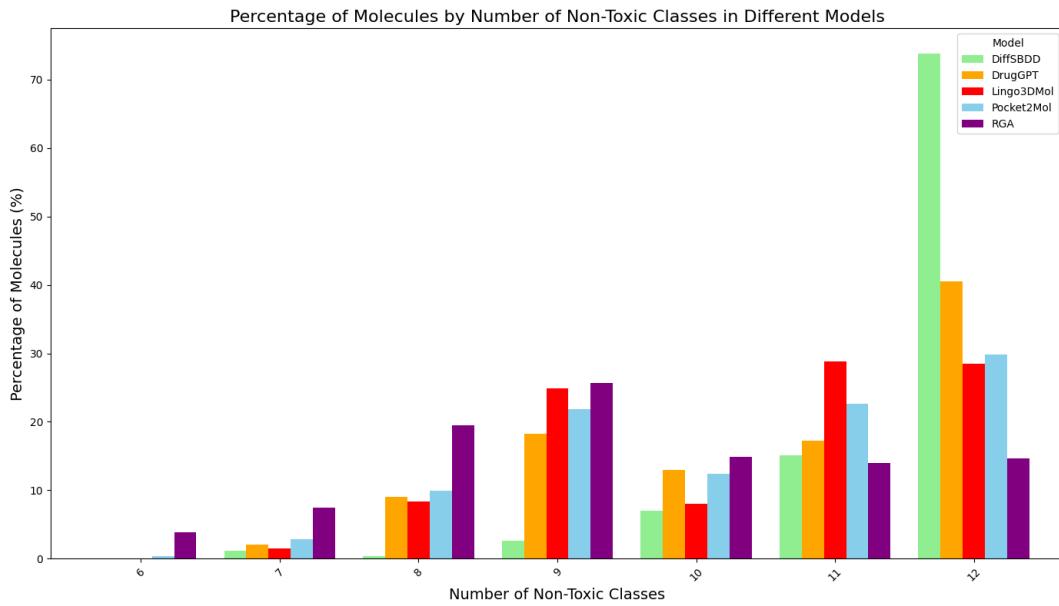


Figure 6: Toxicity Comparison by Model

Tabla resultados Eugenia

5 Conclusions

The use of artificial intelligence tools like diffusion models, genetic algorithms, or transformers in designing molecules has gained significant attention in recent years. This is due to their ability to generate new molecules related to a given protein.

This article provides an overview of primary pocket drug design models. It also evaluates the performance of five models in generating molecules against the DYRK1A protein, an enzyme associated with AD. The models evaluated for their ability to generate protein-related molecules included DiffSBDD, DrugGPT, Lingo3DMol, Pocket2Mol and RGA. However, there are many models under development. These algorithms proved to be highly effective in developing new molecules and exploring new areas of the vast chemical space. They created de novo molecules with a high affinity for the DYRK1A protein. The results of virtual docking were later corroborated with traditional docking tools.

A metrics framework has been proposed to assess the performance of protein pocket-based drug design models. This framework is particularly useful as the lack of experimental evaluation is one of the major challenges in the field. It is therefore important to have a filter-based framework to ensure that molecules are screened for suitability before chemists evaluate them in the laboratory. The code used to build this evaluation framework has been made available.

Finally, as running these models can be complicated, a simple code is supplied so that anyone can generate new molecules given the PDB of a protein and the coordinates of its protein pocket.

6 Code availability

The GitHub repository containing the source code and tools used in this article can be accessed at <https://github.com/pvaras8/pocketdrugdesign>.

7 Acknowledgments

References

- [1] H. Zhu, “Big data and artificial intelligence modeling for drug discovery.” *Annual review of pharmacology and toxicology*, 2020. [Online]. Available: <https://api.semanticscholar.org/CorpusID:202572317>
- [2] M. Vogt, “Using deep neural networks to explore chemical space,” *Expert Opinion on Drug Discovery*, vol. 17, pp. 297 – 304, 2021. [Online]. Available: <https://api.semanticscholar.org/CorpusID:245262420>
- [3] 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*, vol. 27, pp. 675–679, 2013.
- [4] H. R. Henze and C. M. Blair, “The number of isomeric hydrocarbons of the methane series,” *Journal of the American Chemical Society*, vol. 53, no. 8, pp. 3077–3085, 1931.
- [5] D. Weininger, “Combinatorics of small molecular structures,” *Encyclopedia of Computational Chemistry*, vol. 1, 2002.
- [6] C. M. Blair and H. R. Henze, “The number of stereoisomeric and non-stereoisomeric paraffin hydrocarbons,” *Journal of the American Chemical Society*, vol. 54, no. 4, pp. 1538–1545, 1932.
- [7] D. Giordano, C. Biancaniello, M. A. Argenio, and A. Facchiano, “Drug design by pharmacophore and virtual screening approach,” *Pharmaceuticals*, vol. 15, no. 5, p. 646, 2022.
- [8] A. C. Kaushik, A. Kumar, S. Bharadwaj, R. Chaudhary, S. Sahi, A. C. Kaushik, A. Kumar, S. Bharadwaj, R. Chaudhary, and S. Sahi, “Ligand-based approach for in-silico drug designing,” *Bioinformatics Techniques for Drug Discovery: Applications for Complex Diseases*, pp. 11–19, 2018.
- [9] F. Imrie, T. E. Hadfield, A. R. Bradley, and C. M. Deane, “Deep generative design with 3d pharmacophoric constraints,” *Chemical science*, vol. 12, no. 43, pp. 14 577–14 589, 2021.
- [10] W. Feng, L. Wang, Z. Lin, Y. Zhu, H. Wang, J. Dong, R. Bai, H. Wang, J. Zhou, W. Peng *et al.*, “Generation of 3d molecules in pockets via a language model,” *Nature Machine Intelligence*, pp. 1–12, 2024.
- [11] R. Özçelik, D. van Tilborg, J. Jiménez-Luna, and F. Grisoni, “Structure-based drug discovery with deep learning,” *ChemBioChem*, vol. 24, no. 13, p. e202200776, 2023.
- [12] Y. Li, C. Gao, X. Song, X. Wang, Y. Xu, and S. Han, “Druggpt: A gpt-based strategy for designing potential ligands targeting specific proteins,” *bioRxiv*, pp. 2023–06, 2023.
- [13] K. Wu, Y. Xia, Y. Fan, P. Deng, H. Liu, L. Wu, S. Xie, T. Wang, T. Qin, and T.-Y. Liu, “Tailoring molecules for protein pockets: a transformer-based generative solution for structured-based drug design,” *arXiv preprint arXiv:2209.06158*, 2022.
- [14] D. Ang, C. Rakovski, and H. S. Atamian, “De novo drug design using transformer-based machine translation and reinforcement learning of an adaptive monte carlo tree search,” *Pharmaceuticals*, vol. 17, no. 2, p. 161, 2024.
- [15] Z. Guo, J. Liu, Y. Wang, M. Chen, D. Wang, D. Xu, and J. Cheng, “Diffusion models in bioinformatics and computational biology,” *Nature reviews bioengineering*, vol. 2, no. 2, pp. 136–154, 2024.

- [16] A. Schneuing, Y. Du, C. Harris, A. Jamasb, I. Igashov, W. Du, T. Blundell, P. Lió, C. Gomes, M. Welling *et al.*, “Structure-based drug design with equivariant diffusion models,” *arXiv preprint arXiv:2210.13695*, 2022.
- [17] J. Guan, W. W. Qian, X. Peng, Y. Su, J. Peng, and J. Ma, “3d equivariant diffusion for target-aware molecule generation and affinity prediction,” *arXiv preprint arXiv:2303.03543*, 2023.
- [18] L. Terfloth and J. Gasteiger, “Neural networks and genetic algorithms in drug design,” *Drug Discovery Today*, vol. 6, pp. 102–108, 2001.
- [19] J. O. Spiegel and J. D. Durrant, “Autogrow4: an open-source genetic algorithm for de novo drug design and lead optimization,” *Journal of cheminformatics*, vol. 12, pp. 1–16, 2020.
- [20] T. Fu, W. Gao, C. Coley, and J. Sun, “Reinforced genetic algorithm for structure-based drug design,” *Advances in Neural Information Processing Systems*, vol. 35, pp. 12 325–12 338, 2022.
- [21] B. Sanchez-Lengeling, E. Reif, A. Pearce, and A. B. Wiltschko, “A gentle introduction to graph neural networks,” *Distill*, vol. 6, no. 9, p. e33, 2021.
- [22] X. Peng, S. Luo, J. Guan, Q. Xie, J. Peng, and J. Ma, “Pocket2mol: Efficient molecular sampling based on 3d protein pockets,” in *International Conference on Machine Learning*. PMLR, 2022, pp. 17 644–17 655.
- [23] X. Tang, H. Dai, E. Knight, F. Wu, Y. Li, T. Li, and M. Gerstein, “A survey of generative ai for de novo drug design: New frontiers in molecule and protein generation,” *arXiv preprint arXiv:2402.08703*, 2024.
- [24] C. Chichester, D. Digles, R. Siebes, A. Loizou, P. Groth, and L. Harland, “Drug discovery faqs: workflows for answering multidomain drug discovery questions,” *Drug discovery today*, vol. 20, no. 4, pp. 399–405, 2015.
- [25] L. Guan, H. Yang, Y. Cai, L. Sun, P. Di, W. Li, G. Liu, and Y. Tang, “Admet-score—a comprehensive scoring function for evaluation of chemical drug-likeness,” *Medchemcomm*, vol. 10, no. 1, pp. 148–157, 2019.
- [26] O. A. Raevsky, “Physicochemical descriptors in property-based drug design,” *Mini reviews in medicinal chemistry*, vol. 4, no. 10, pp. 1041–1052, 2004.
- [27] D. Bajusz, A. Rácz, and K. Héberger, “Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations?” *Journal of cheminformatics*, vol. 7, pp. 1–13, 2015.
- [28] P. Bonnet, “Is chemical synthetic accessibility computationally predictable for drug and lead-like molecules? a comparative assessment between medicinal and computational chemists,” *European journal of medicinal chemistry*, vol. 54, pp. 679–689, 2012.
- [29] M. P. Garralaga, L. Lomba, E. Zuriaga, S. Santander, and B. Giner, “Key properties for the toxicity classification of chemicals: A comparison of the reach regulation and scientific studies trends,” *Applied Sciences*, vol. 12, no. 22, p. 11710, 2022.
- [30] M. M. de Souza, A. R. Cenci, K. F. Teixeira, V. Machado, M. C. G. Mendes Schuler, A. E. Gon, A. Paula Dalmagro, C. André Cazarin, L. L. Gomes Ferreira, A. S. de Oliveira *et al.*, “Dyrk1a inhibitors and perspectives for the treatment of alzheimer’s disease,” *Current Medicinal Chemistry*, vol. 30, no. 6, pp. 669–688, 2023.
- [31] S. Stotani, F. Giordanetto, and F. Medda, “Dyrk1a inhibition as potential treatment for alzheimer’s disease,” *Future medicinal chemistry*, vol. 8, no. 6, pp. 681–696, 2016.
- [32] U. Rothweiler, “Dyrk1a in complex with xmd7-117,” *Protein Data Bank*, 2018, classification: TRANS-FERASE. Organism(s): Homo sapiens. Expression System: Escherichia coli. Mutation(s): No. Deposited: 2017-09-19. Released: 2018-08-29. URL: <https://www.rcsb.org/structure/6eif>.
- [33] D. E. Graff and C. W. Coley, “pyscreener: A python wrapper for computational docking software,” *arXiv preprint arXiv:2112.10575*, 2021.

- [34] E. Heid, K. P. Greenman, Y. Chung, S.-C. Li, D. E. Graff, F. H. Vermeire, H. Wu, W. H. Green, and C. J. McGill, “Chemprop: A machine learning package for chemical property prediction,” *Journal of Chemical Information and Modeling*, vol. 64, no. 1, pp. 9–17, 2023.

A1 DiffSBDD

Algorithm 1: Equivariant Diffusion Model for Structure-Based Drug Design (DiffSBDD)

Data: Set of protein-ligand pairs (P, L) for training, protein sequence P^* for inference
Result: Generate new ligands L^* optimized for binding to P^*

```

1 Initialize diffusion parameters  $\theta$ , noise levels  $\sigma_t$ , and set total timesteps  $T$ ;
// Training Phase
2 for each epoch do
3   for each protein pocket  $p \in P$  and corresponding ligand  $l \in L$  do
4     Represent  $p$  and  $l$  as point clouds  $z_p$  and  $z_l$ ;
5     Initialize diffusion process with  $z_{l,0} = z_l$  and apply noise to get  $z_{l,t}$ ;
6     for  $t = T$  downto 1 do
7       Predict noise  $\epsilon_\theta(z_{l,t}, t)$  using the model;
8       Perform denoising step to estimate  $z_{l,t-1} \sim p_\theta(z_{l,t-1} | z_{l,t})$ ;
9       Update  $\theta$  by minimizing loss between  $z_{l,t-1}$  and  $z_{l,t-1}^{predicted}$ ;
10    end
11  end
12 end
// Generation Phase
13 for each protein pocket  $p^* \in P^*$  do
14   Sample a new ligand point cloud  $z_{l^*,T} \sim \mathcal{N}(0, I)$ ;
15   for  $t = T$  down to 1 do
16     Refine  $z_{l^*,t-1} \sim p_\theta(z_{l^*,t-1} | z_{l^*,t}, p^*)$ ;
17   end
18   Convert the final point cloud  $z_{l^*,0}$  back to molecular structure  $L^*$ ;
19 end

```

A2 DrugGPT

Algorithm 2: DrugGPT for Ligand Design

Data: Set of protein-ligand pairs (P, L) for training, protein sequence P^* for inference
Result: Generate optimized ligand L^* for given protein P^*

```
1 Initialize tokenizer  $\mathcal{T}$  using Byte Pair Encoding (BPE);  
2 Train tokenizer  $\mathcal{T}$  on protein and ligand sequences to optimize vocabulary;  
3 Initialize DrugGPT model  $\mathcal{M}$  with parameters  $\theta$ ;  
    // Training Phase  
4 for each epoch do  
5   for each protein-ligand pair  $(P, L)$  do  
6     Tokenize  $(P, L)$  to  $(T_P, T_L) = \mathcal{T}(P, L)$ ;  
7     Prepare input sequence  $X = T_P + T_L$ ;  
8     Train  $\mathcal{M}$  to predict sequence  $\hat{T}_L$  similar to  $T_L$ ;  
9     Update  $\mathcal{M}$  parameters  $\theta$  to minimize prediction loss;  
10    end  
11    Validate model performance on unseen data;  
12 end  
    // Inference Phase  
13 Tokenize protein sequence  $T_P^* = \mathcal{T}(P^*)$ ;  
14 Start generation with  $Y^* = T_P^*$ ;  
15 while not end of sequence do  
16   Predict next token  $\hat{y}$  using  $\mathcal{M}$ ;  
17   Append  $\hat{y}$  to  $Y^*$ ;  
18   if  $\hat{y} == \text{end}$  then  
19     | break;  
20   end  
21 end
```

A3 Lingo3DMol

Algorithm 3: Generation of a 3D molecule using Lingo3DMol

Data: Set of protein-ligand pairs (P, L) for training, protein sequence P^* for inference
Result: Generate optimized ligand L^* for given protein P^*

// Training Phase

- 1 Initialize model parameters θ ;
- 2 Pre-train model on large-scale molecule data to optimize θ ;
- 3 Fine-tune model on specific protein-ligand dataset;

// Generation Phase

- 4 Input target pocket P^* ; **for** each atom in pocket P^* **do**
- 5 | Compute input features $f_i = \text{Encoder}(P_i^*; \theta)$ for each atom P_i^* in pocket P^* ;
- 6 **end**
- 7 Initialize molecule L^* as an empty set of atoms;
- 8 Select starting position in pocket based on NCI/Anchor prediction;
- 9 **while** molecule not complete **do**
- 10 | Predict next atom's FSMILES token and corresponding coordinates using the current state of L^* ;
- 11 | $L^* \leftarrow L^* \cup \{\text{new atom}\}$;
- 12 | Update model state for next prediction;
- 13 **end**
- 14 Convert FSMILES tokens and coordinates into a 3D molecular structure;

A4 Pocket2Mol

Algorithm 4: Efficient Molecular Sampling with Pocket2Mol

Data: Set of protein-ligand pairs (P, L) for training, protein sequence P^* for inference
Result: Generate optimized ligand L^* for given protein P^*

```
1 Initialize model  $M$  with parameters  $\theta$ ;  
2 Train the model on dataset  $\{(P_i, L_i)\}$  to learn molecular configurations;  
    // Training Phase  
3 for each pocket  $P_i$  and ligand  $L_i$  in training data do  
4     Represent  $P_i$  and  $L_i$  as graphs  $G_{P_i}$  and  $G_{L_i}$ ;  
5     Mask random parts of  $L_i$  to simulate incomplete data;  
6     Use  $M$  to predict missing parts of  $L_i$  based on  $G_{P_i}$ ;  
7     Update  $\theta$  by minimizing loss between predicted and actual configurations of  $L_i$ ;  
8 end  
    // Generation Phase  
9 for each new protein pocket  $P^*$  do  
10    Initialize ligand graph  $G_{L^*}^0$  with minimal structure;  
11    while not full ligand generated do  
12        Use  $M$  to predict next atom or group to add to  $G_{L^*}^t$ ;  
13        Update  $G_{L^*}^{t+1}$  with new atom or group;  
14        If model predicts end of generation, terminate;  
15    end  
16    Convert final graph  $G_{L^*}$  to molecular structure  $L^*$ ;  
17 end
```

A5 RGA

Algorithm 5: Reinforced Genetic Algorithm for Structure-Based Drug Design

Data: Set of protein-ligand pairs (P, L) for training, protein sequence P^* and Population of molecules \mathcal{M} for inference

Result: Optimized molecules with enhanced binding affinity

```

1 Initialize population  $\mathcal{M}(0)$  randomly;
2 for generation  $t = 1$  to  $T$  do
3   for each molecule  $m \in \mathcal{M}(t - 1)$  do
4     Select parents  $m_1, m_2$  from  $\mathcal{M}(t - 1)$  based on fitness scores;
5     Generate offspring  $o$  by crossover of  $m_1, m_2$ ;
6     Mutate offspring  $o$  to produce mutants  $o'$ ;
7     Evaluate binding affinity of  $o'$  using docking simulations;
8   end
9   Select top  $N$  offspring to form new population  $\mathcal{M}(t)$ ;
10  if convergence or  $t = T$  then
11    | break;
12  end
13  Optionally refine neural model with reinforcement learning using performance feedback;
14 end
```

B Candidates Selection Protocol

- **Toxicity Filter:** The toxicity filter ensures that a molecule G is classified as nontoxic according to 12 toxicity classes

$$\sum_{i=1}^{12} f_{\text{tox}}^{(i)}(G) = 0$$

- **QED Filter:** The QED filter ensures that the quantitative estimate of drug-likeness (QED) for molecule G exceeds a predefined threshold $QED_{\text{threshold}}$

$$QED(G) > 0.5$$

- **LogP Filter:** The LogP filter ensures that the octanol-water partition coefficient (LogP) for molecule G falls within a predefined range $[\text{LogP}_{\min}, \text{LogP}_{\max}]$

$$2 \leq \text{LogP}(G) \leq 4$$

- **Similarity Filter:** The similarity filter ensures that the similarity between a molecule G and a dataset D is below a predefined threshold $\text{similarity}_{\text{threshold}}$

$$s_D(G) < 0.25$$

- **Molecular Weight Filter:** The molecular weight filter ensures that the molecular weight of molecule G falls within a predefined range $[\text{MW}_{\min}, \text{MW}_{\max}]$

$$300 \leq \text{MW}(G) \leq 500$$

- **Synthetic Accessibility Filter:** The synthetic accessibility filter ensures that the synthetic accessibility score (SAS) for molecule G is below a predefined threshold $SAS_{\text{threshold}}$

$$SAS(G) < 5$$

- **HBA Filter:** The HBA filter ensures that the number of hydrogen bond acceptors (HBA) for molecule G is below a predefined threshold HBA_{max}

$$\text{HBA}(G) \leq 5$$

- **HBD Filter:** The HBD filter ensures that the number of hydrogen bond donors (HBD) for molecule G is below a predefined threshold HBD_{max}

$$\text{HBD}(G) \leq 10$$