An overview of Pocket-based drug design

Pablo Varas Pardo[a, b], Eugenia Ulzurrun[b, c], Nuria Campillo[c], David Quesada[a], and David Ríos[b]

[a]Aitenea Biotech
[b]Instituto de Ciencias Matemáticas (ICMAT-CSIC)
[c]Centro de Investigaciones Biológicas Margarita Salas (CIB)

June 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 its size, the so-called 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 [4–6].

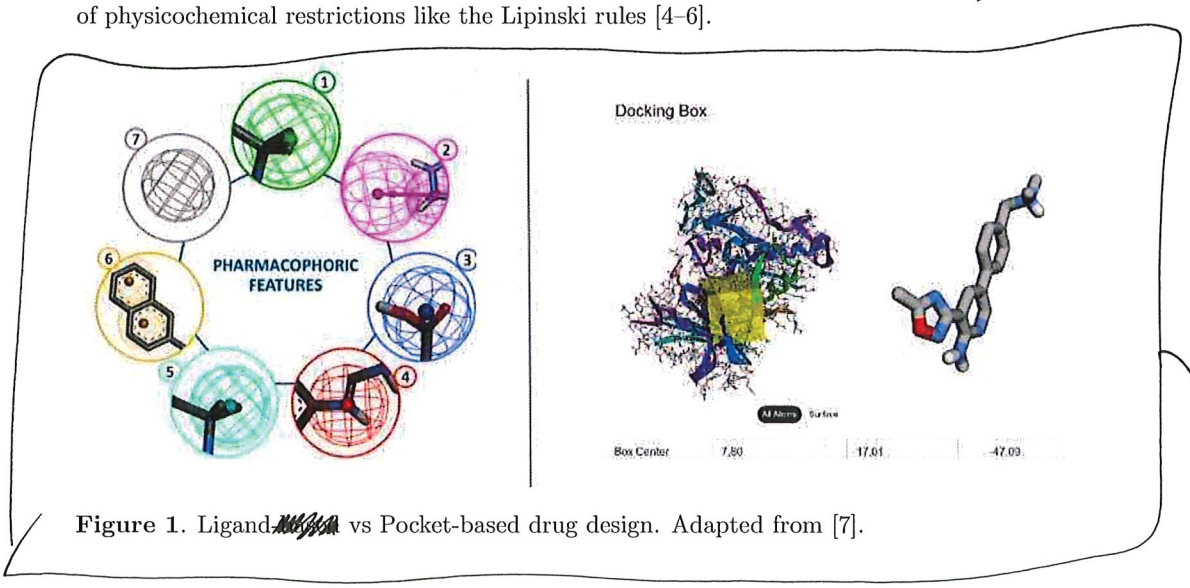


Figure 1. Ligand-based vs Pocket-based drug design. Adapted from [7].

The vast search space poses a significant challenge when 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, where 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 optimally configured to the target site and making them especially useful to create new drugs that inhibit certain proteins involved in the appearance of diseases.

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.

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]

The main contributions of this article are:

- To create a common framework of current pocket-based molecular generation methods.
- The benchmarking of these methods to assess which are the most promising in pocket-based drug design, creating a reference framework to evaluate 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).

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

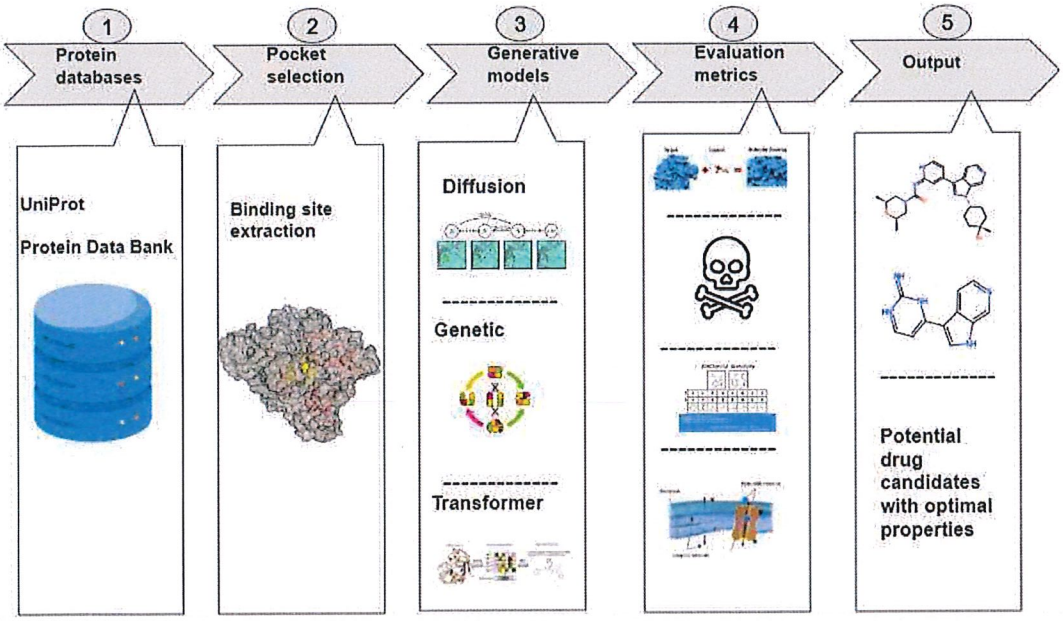


Figure 2. Computational Drug Discovery Pipeline.

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, WITH THE APPENDIX DISPLAYING KEY ASPECTS OF ALGORITHMS IMPLEMENTING THEM.

2.1 Transformer based models

Transformer-based algorithms are gaining popularity in drug discovery because they can capture complex relations within 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. Its 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 an encoder, and, starting from the initially calculated growth point, the decoder builds the molecule by iteratively joining molecular groups produced. DrugGPT [12] employs the GPT-2 model and Byte Pair Encoding (BPE) tokenizer, which can represent a huge number of compounds using a limited vocabulary, 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 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 Difussion based models

Diffusion models have attracted special interest in recent 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, this model can generate new molecules by sampling from a normal distribution and denoising this data until a new compound 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].

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

Graph neural networks have received special attention in recent years in the field of drug discovery. These models are very important for data processing 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.

- 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 molecules generated by the different models, the metrics that will be ~~used~~ CONSIDERED are:

- 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~~ EMPLOYED TO ASSESS 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 the ability of a compound to become an effective oral drug in humans [25].

- Lipophylicity (LogP). It ASSESSES ~~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 THE generated molecules ~~using fingerprints~~ through the formula

  A and B

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

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~~ V 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].

V SPECIFICALLY, THE FEATURES USUALLY EMPLOYED

- Molecular Weight. Its analysis is crucial for evaluating the molecule's atomic composition and to determine drug 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 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. Toxicity classes.

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

4 CASE STUDY

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 Experimental section

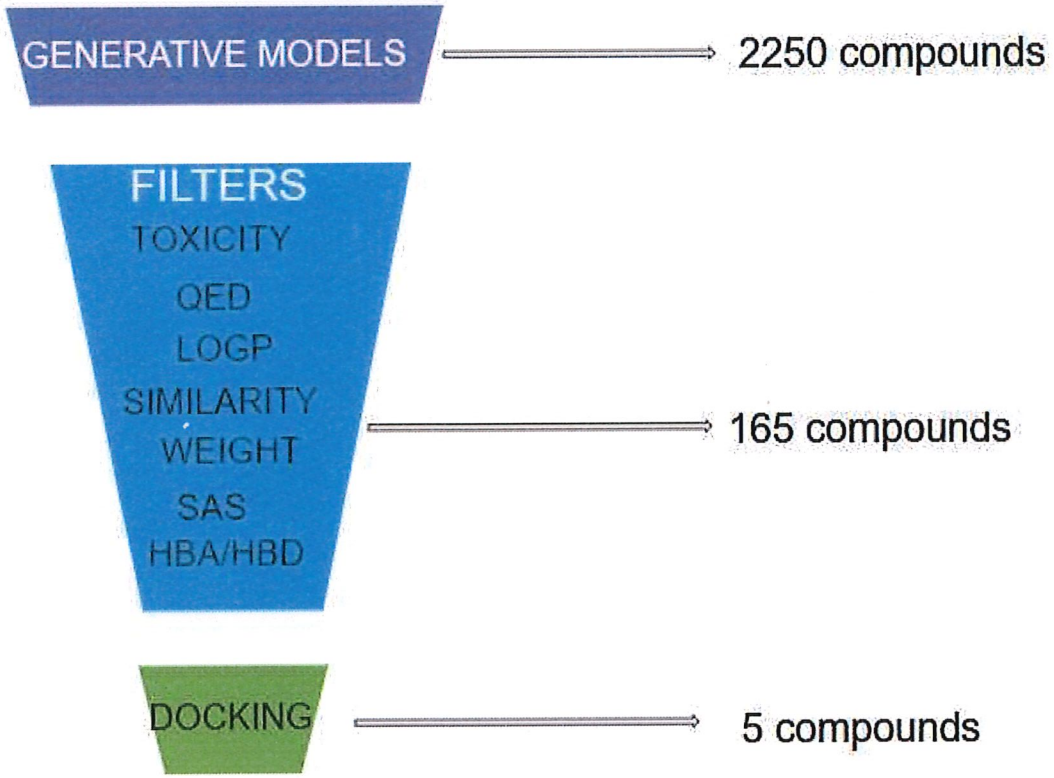


Figure 3. Candidates Selection Protocol.

This section explains the process of obtaining compounds with optimal properties as shown in Figure 3. To begin with, we obtained the DYRK1A 3D structure from the Prtoein Data Bank (PDB code: 6EIF). We extracted the mass centre of the crystallographic ligand B5T of chain A by taking the average of the x, y, and z coordinates, which were 7.80, 17.01, and -47.09 respectively. Molecules were then generated around a box of 15 Å from this point.

4.1.1 Selected generative models

The models used to develop new ligands against THE 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 were 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 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

These models were run 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.

4.1.2 Filters

The evaluation metrics described in Section 3 were then applied. The parameters used to filter the molecules are shown in [illegible]. For toxicity prediction, the Chemprop algorithm [32] 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.

4.1.3 Docking studies

Eugenia

- Ligand Preparation. The conversion from SMILES to SD format was carried out using the structconvert tool available in the Schrödinger suite [33]. Ligand preparation was performed utilizing the LigPrep tool included in the Maestro package [34, 35]. Progressive levels were generated, encompassing possible ionization states at physiological pH and potential tautomers. Final energy minimization was [illegible] using the OPLS4 force field, with default settings applied for stereoisomers.

- Protein Preparation. Human DYRK1A (PDB code 6EIF, [36]) was prepared for subsequent computational analyses using the Protein Preparation Wizard (Sastry, G.M., Protein Preparation Wizard Release 2023-2) integrated within Maestro (Maestro Release 2023-2). The preparation protocol included preprocessing steps such as bond order assignment and structural adjustments carried out using Prime (Jacobson, M. P., Prime Release 2023-2). Protonation and metal charge states for cofactors and metals at pH 7 ± 2 were generated using Epik (Johnston, R. C., Epik Release 2023-2). The hydrogen-bonding network was optimized, and residue protonation states at pH 7 were calculated using PROPKA (Olsson, M. H.). Water

molecules beyond a 5 Å radius from protein residues were excluded, and a final restrained minimization was performed using the OPLS4 force field.

- Ligand Docking. The centroid of the crystallized ligand in the catalytic pocket was used as the grid center. During grid generation, a van der Waals radius scaling factor of 1.0 and a partial charge cutoff of 0.25 were applied. Docking was performed using the Glide extra precision (XP) mode available in the Schrödinger software suite (Yang, Y., Friesner, R. A., Halgren, T. A., Glide Release 2023-2), without applying any constraints. Default parameters were employed for ligand setting, including flexible ligand sampling and the incorporation of Epik state penalties into the docking score. The final step involved post-docking minimization using default settings.

4.2 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 ~~shows~~ 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 ~~reflects~~ 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.

~~This~~ 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.

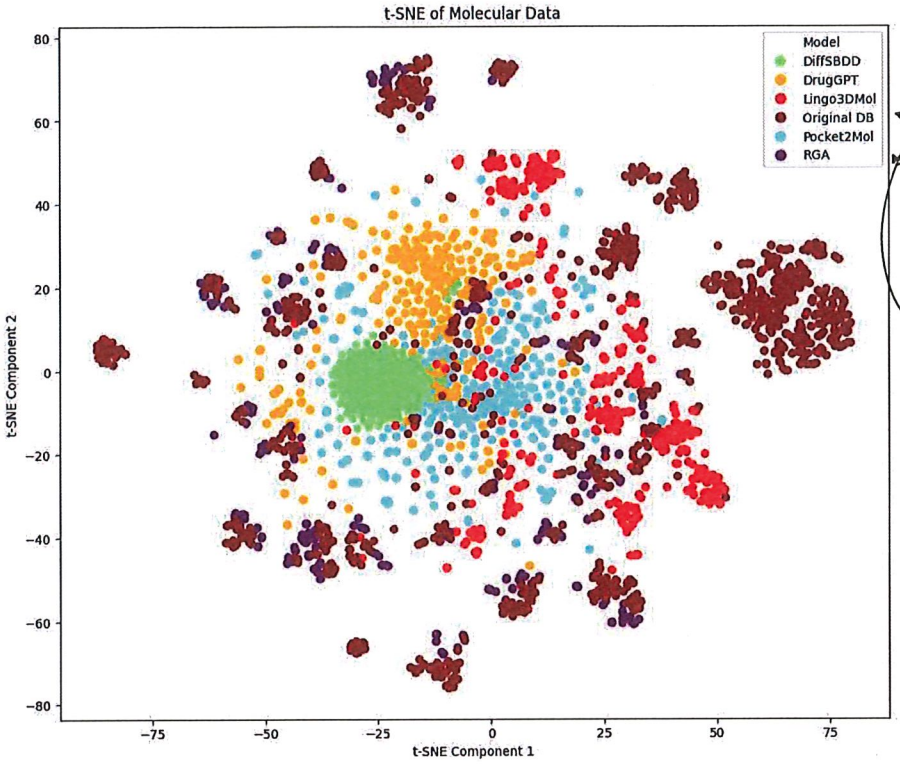


Figure 4. Diversity Comparison [illegible] OF PROPOSED GENERATING MODELS

[illegible] Table 3 showcases the algorithmic performance [illegible] 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. The docking value for all molecules was obtained using Smina, provided by PyScreener [37], allowing efficient and flexible computation of docking scores.

Pharmacological activity was calculated using Morgan fingerprints to process the initial SMILES and a Gaussian process to obtain the pChEMBL value.

Table 3. Model performance evaluation

	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

↓ 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 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 whether 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, 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.

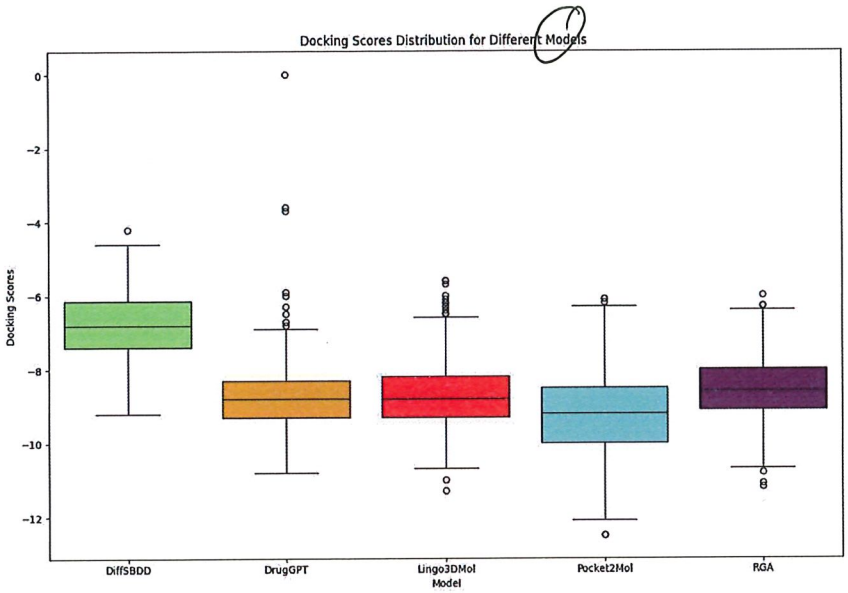


Figure 5. Docking Score Comparison by Model.

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.

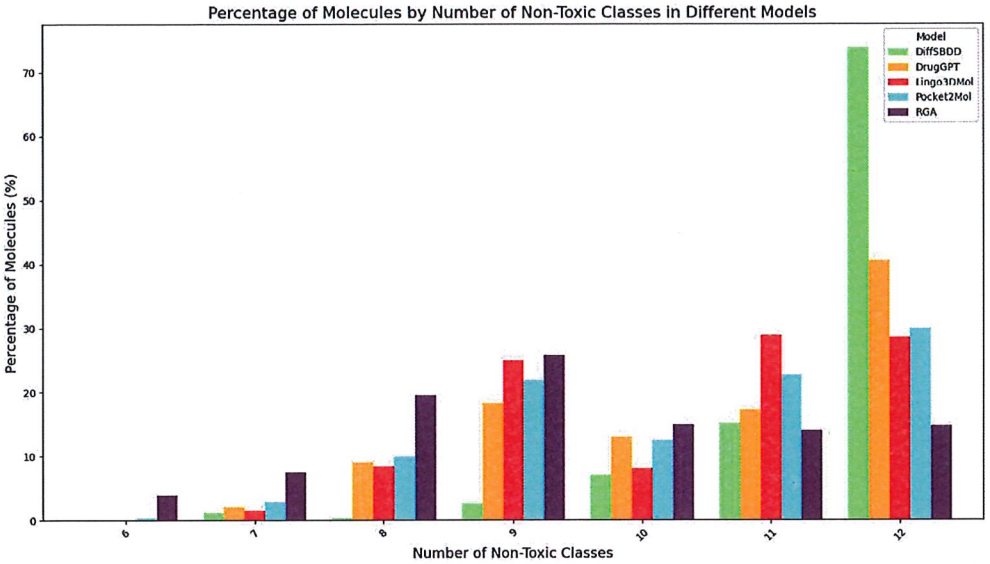


Figure 6. Toxicity Comparison by Model.

Tabla resultados Eugenia

+ COMMENT

decir de porcentaje de molecules queue than pasado el filtro y el mejor algoritmo

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.

HAS PROVIDED

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.

SAME ORDER

MORE

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~~ is 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.

POR QUÉ ELIGE, AQUÍ, ESO

SYNTHESISE + ASSESS ??

FUTURE TOPICS.
MORE CONCLUSIONS

ASSOCIATED CONTENT

Data Availability Statement

The DYRK1A crystal structure is retrievable from the Protein Data Bank `https://www.rcsb.org/structure/6eif`. The GitHub repository containing the source code and tools used in this article can be accessed at `https://github.com/pvaras8/pocketdrugdesign`.

Supporting Information

The Supporting Information is available free of charge at

AUTHOR INFORMATION

Corresponding Author

David Quesada – Aitenea Biotech; orcid.org/0000-0002-7280-904X
David Ríos Insúa – Instituto de Ciencias Matemáticas (CSIC), 28049 Madrid, Spain; orcid.org/0000-0002-5748-9658

Author

Pablo Varas Pardo – Aitenea Biotech & Instituto de Ciencias Matemáticas (CSIC), 28049 Madrid, Spain; orcid.org/0009-0006-1115-4824; Email: pablo.varas@icmat.es
Eugenia Ulzurrun – Centro de Investigaciones Biológicas Margarita Salas (CSIC), 28040 Madrid, Spain;
Nuria E. Campillo – Centro de Investigaciones Biológicas Margarita Salas (CSIC), 28040 Madrid, Spain; orcid.org/0000-0002-9948-2665

Notes

The authors declare no competing financial interest(s)

ACKNOWLEDGEMENTS

References

(1) Zhu, H. *Annual review of pharmacology and toxicology* **2020**.
(2) Vogt, M. *Expert Opinion on Drug Discovery* **2021**, *17*, 297–304.
(3) Polishchuk, P. G.; Madzhidov, T. I.; Varnek, A. *Journal of computer-aided molecular design* **2013**, *27*, 675–679.
(4) Henze, H. R.; Blair, C. M. *Journal of the American Chemical Society* **1931**, *53*, 3077–3085.
(5) Weininger, D. *Encyclopedia of Computational Chemistry* **2002**, *1*.
(6) Blair, C. M.; Henze, H. R. *Journal of the American Chemical Society* **1932**, *54*, 1538–1545.

(7) Giordano, D.; Biancaniello, C.; Argenio, M. A.; Facchiano, A. *Pharmaceuticals* **2022**, *15*, 646.

(8) Kaushik, A. C.; Kumar, A.; Bharadwaj, S.; Chaudhary, R.; Sahi, S. *Bioinformatics Techniques for Drug Discovery: Applications for Complex Diseases* **2018**, 11–19.

(9) Imrie, F.; Hadfield, T. E.; Bradley, A. R.; Deane, C. M. *Chemical science* **2021**, *12*, 14577–14589.

(10) Feng, W.; Wang, L.; Lin, Z.; Zhu, Y.; Wang, H.; Dong, J.; Bai, R.; Wang, H.; Zhou, J.; Peng, W., et al. *Nature Machine Intelligence* **2024**, 1–12.

(11) Özçelik, R.; van Tilborg, D.; Jiménez-Luna, J.; Grisoni, F. *ChemBioChem* **2023**, *24*, e202200776.

(12) Li, Y.; Gao, C.; Song, X.; Wang, X.; Xu, Y.; Han, S. *bioRxiv* **2023**, 2023–06.

(13) Wu, K.; Xia, Y.; Fan, Y.; Deng, P.; Liu, H.; Wu, L.; Xie, S.; Wang, T.; Qin, T.; Liu, T.-Y. *arXiv preprint arXiv:2209.06158* **2022**.

(14) Ang, D.; Rakovski, C.; Atamian, H. S. *Pharmaceuticals* **2024**, *17*, 161.

(15) Guo, Z.; Liu, J.; Wang, Y.; Chen, M.; Wang, D.; Xu, D.; Cheng, J. *Nature reviews bioengineering* **2024**, *2*, 136–154.

(16) Schneuing, A.; Du, Y.; Harris, C.; Jamasb, A.; Igashov, I.; Du, W.; Blundell, T.; Lió, P.; Gomes, C.; Welling, M., et al. *arXiv preprint arXiv:2210.13695* **2022**.

(17) Guan, J.; Qian, W. W.; Peng, X.; Su, Y.; Peng, J.; Ma, J. *arXiv preprint arXiv:2303.03543* **2023**.

(18) Terfloth, L.; Gasteiger, J. *Drug Discovery Today* **2001**, *6*, 102–108.

(19) Spiegel, J. O.; Durrant, J. D. *Journal of cheminformatics* **2020**, *12*, 1–16.

(20) Fu, T.; Gao, W.; Coley, C.; Sun, J. *Advances in Neural Information Processing Systems* **2022**, *35*, 12325–12338.

(21) Sanchez-Lengeling, B.; Reif, E.; Pearce, A.; Wiltschko, A. B. *Distill* **2021**, *6*, e33.

(22) Peng, X.; Luo, S.; Guan, J.; Xie, Q.; Peng, J.; Ma, J. In *International Conference on Machine Learning*, 2022, pp 17644–17655.

(23) Tang, X.; Dai, H.; Knight, E.; Wu, F.; Li, Y.; Li, T.; Gerstein, M. *arXiv preprint arXiv:2402.08703* **2024**.

(24) Chichester, C.; Digles, D.; Siebes, R.; Loizou, A.; Groth, P.; Harland, L. *Drug discovery today* **2015**, *20*, 399–405.

(25) Guan, L.; Yang, H.; Cai, Y.; Sun, L.; Di, P.; Li, W.; Liu, G.; Tang, Y. *Medchemcomm* **2019**, *10*, 148–157.

(26) Raevsky, O. A. *Mini reviews in medicinal chemistry* **2004**, *4*, 1041–1052.

(27) Bajusz, D.; Rácz, A.; Héberger, K. *Journal of cheminformatics* **2015**, *7*, 1–13.

(28) Bonnet, P. *European journal of medicinal chemistry* **2012**, *54*, 679–689.

(29) Garralaga, M. P.; Lomba, L.; Zuriaga, E.; Santander, S.; Giner, B. *Applied Sciences* **2022**, *12*, 11710.

(30) De Souza, M. M.; Cenci, A. R.; Teixeira, K. F.; Machado, V.; Mendes Schuler, M. C. G.; Gon, A. E.; Paula Dalmagro, A.; André Cazarin, C.; Gomes Ferreira, L. L.; de Oliveira, A. S., et al. *Current Medicinal Chemistry* **2023**, *30*, 669–688.

(31) Stotani, S.; Giordanetto, F.; Medda, F. *Future medicinal chemistry* **2016**, *8*, 681–696.

(32) Heid, E.; Greenman, K. P.; Chung, Y.; Li, S.-C.; Graff, D. E.; Vermeire, F. H.; Wu, H.; Green, W. H.; McGill, C. J. *Journal of Chemical Information and Modeling* **2023**, *64*, 9–17.

(33) Schrödinger Software Release 2023-2 distribution, 2023.

(34) Schrödinger Release 2023-2: LigPrep, Schrödinger, LLC, New York, NY, 2023, 2023.

(35) Schrödinger Release 2023-2: Maestro, Schrödinger, LLC, New York, NY, 2023, 2023.

(36) Falke, H.; Chaikuad, A.; Becker, A.; Loaéc, N.; Lozach, O.; Abu Jhaisha, S.; Becker, W.; Jones, P. G.; Preu, L.; Baumann, K., et al. *Journal of medicinal chemistry* **2015**, *58*, 3131–3143.

(37) Graff, D. E.; Coley, C. W. *arXiv preprint arXiv:2112.10575* **2021**.

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^*

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

EXPLICAR INPUTS ?, EN MAIN TEXT.

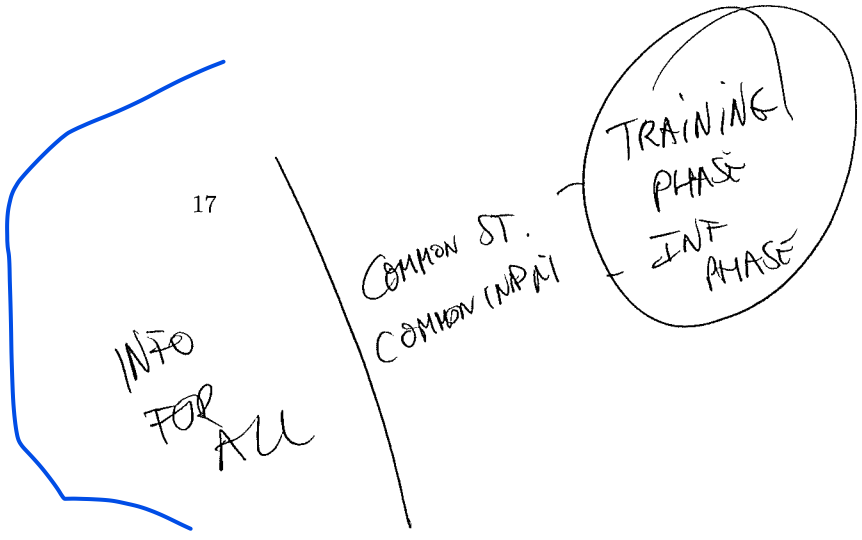
PONER LOS ALGOS. COMO APARECEN EN EL TEXTO PRINCIPAL EN EL ORAL

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^*

Initialize tokenizer $\mathcal{T}$ using Byte Pair Encoding (BPE);
Train tokenizer $\mathcal{T}$ on protein and ligand sequences to optimize vocabulary;
Initialize DrugGPT model $\mathcal{M}$ with parameters θ;
// Training Phase
for *each epoch* **do**
  for *each protein-ligand pair* (P, L) **do**
    Tokenize (P, L) to $(T_P, T_L) = \mathcal{T}(P, L)$;
    Prepare input sequence $X = T_P + T_L$;
    Train $\mathcal{M}$ to predict sequence $\hat{T}_L$ similar to T_L;
    Update $\mathcal{M}$ parameters θ to minimize prediction loss;
  end
  Validate model performance on unseen data;
end
// Inference Phase
Tokenize protein sequence $T_P^* = \mathcal{T}(P^*)$;
Start generation with $Y^* = T_P^*$;
while *not end of sequence* **do**
  Predict next token $\hat{y}$ using $\mathcal{M}$;
  Append $\hat{y}$ to Y^*;
  if $\hat{y} ==$ *end* **then**
    break;
  end
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
Initialize model parameters θ;
Pre-train model on large-scale molecule data to optimize θ;
Fine-tune model on specific protein-ligand dataset;
// Generation Phase
Input target pocket P^*; **for** *each atom in pocket* P^* **do**
  Compute input features $f_i = \text{Encoder}(P_i^*; \theta)$ for each atom P_i^* in pocket P^*;
end
Initialize molecule L^* as an empty set of atoms;
Select starting position in pocket based on NCI/Anchor prediction;
while *molecule not complete* **do**
  Predict next atom's FSMILES token and corresponding coordinates using the current state of L^*;
  $L^* \leftarrow L^* \cup \{\text{new atom}\}$;
  Update model state for next prediction;
end
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^*

Initialize model M with parameters θ;
Train the model on dataset $\{(P_i, L_i)\}$ to learn molecular configurations;
// Training Phase
for *each pocket* P_i *and ligand* L_i *in training data* **do**
    Represent P_i and L_i as graphs G_{P_i} and G_{L_i};
    Mask random parts of L_i to simulate incomplete data;
    Use M to predict missing parts of L_i based on G_{P_i};
    Update θ by minimizing loss between predicted and actual configurations of L_i;
end
// Generation Phase
for *each new protein pocket* P^* **do**
    Initialize ligand graph $G^0_{L^*}$ with minimal structure;
    while *not full ligand generated* **do**
        Use M to predict next atom or group to add to $G^t_{L^*}$;
        Update $G^{t+1}_{L^*}$ with new atom or group;
        If model predicts end of generation, terminate;
    end
    Convert final graph G_{L^*} to molecular structure L^*;
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

```
Initialize population M(0) randomly;
for generation t = 1 to T do
    for each molecule m ∈ M(t − 1) do
        Select parents m1, m2 from M(t − 1) based on fitness scores;
        Generate offspring o by crossover of m1, m2;
        Mutate offspring o to produce mutants o′;
        Evaluate binding affinity of o′ using docking simulations;
    end
    Select top N offspring to form new population M(t);
    if convergence or t = T then
        break;
    end
    Optionally refine neural model with reinforcement learning using performance feedback;
end
```

B Candidates Selection Protocol

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

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

- **QED Filter**: The QED filter ensures that the quantitative estimate of drug-likeness (QED) for molecule G exceeds a predefined threshold $QED_{\text{threshold}}$. IN OUR CASE, WE ADOPT THE THRESHOLD TO BE 0.5, THUS CHECKING $QED(G) > 0.5$.

- **LogP Filter**: The LogP filter ensures that the octanol-water partition coefficient (LogP) for molecule G falls within a predefined range $[\text{LogP}_{\min}, \text{LogP}_{\max}]$. IN OUR CASE THE RANGE IS [2,4] SO THAT WE CHECK $2 \leq \text{LogP}(G) \leq 4$.

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

$$s_D(G) < 0.30$$

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

$$300 \leq MW(G) \leq 500$$

WHY 5

- **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}

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

SENSITIVITY ANALYSIS.