Rethinking Druggability in the Evaluation of AI-driven Structure-based Drug Design

Anonymous Author(s)
Affiliation
Address
email

Abstract

Structure-based drug design harnesses three-dimensional structural information to guide ligand discovery and has seen rapid progress through machine learning. Yet the evaluation of AI-driven SBDD models has largely ignored **druggability**—the propensity of a binding pocket to accept a small, drug-like molecule. As a result, generative models may appear successful by creating compounds that dock well to pockets that are not feasible drug targets. We review SBDD benchmarks and druggability assessment methods, highlight pitfalls of current evaluation protocols, and propose a methodology to incorporate continuous druggability scores into the widely used CrossDocked2020 benchmark. By weighting generative scores according to pocket druggability and analysing performance across druggable and undruggable targets, our framework encourages models to focus on realistic therapeutic targets and reveals algorithmic biases.

1 Introduction

Structure-based drug design (SBDD) has become a cornerstone of modern drug discovery because it directly leverages the three-dimensional (3D) structure of a target to find ligands with complementary shape, electrostatics, and hydrophobicity. Compared with high-throughput screening, SBDD provides a more targeted and cost-efficient approach for lead generation. Reviews of the field note that SBDD is "becoming an essential tool for faster and more cost-efficient lead discovery" and that it is widely used in industry and academia [4]. The availability of high-resolution structures for thousands of proteins and the rapid advances in machine learning make it possible to automate key steps such as virtual screening, docking and ligand optimization. In recent years, **AI-driven *de novo*** design models have emerged that attempt to generate novel small molecules tailored to a given pocket, often relying on geometric deep learning to encode the pocket's shape and chemical environment. These models promise to accelerate drug discovery but require careful evaluation.

Druggability refers to the propensity of a protein binding site to bind drug-like small molecules with high affinity. A binding pocket may be druggable because of its size, depth and hydrophobicity; a "druggable pocket" is one where small drug-like molecules have been shown to bind [28]. Distinguishing *druggability* from related concepts is important: *ligandability* measures whether a site can bind any small molecule, whereas druggability implies the ability to modulate a target to achieve a therapeutic effect [12]. Only about a few thousand of the ~20,000 human proteins are considered druggable [5]. Druggability assessments guide target selection and ranking in early discovery; however, many current AI evaluation benchmarks ignore druggability and treat every pocket as equally suitable for drug discovery.

Ignoring druggability in evaluation has serious consequences. A generative model may achieve a favorable docking score simply by generating large, hydrophobic molecules that fill any pocket [25]. If some of the pockets in a benchmark are intrinsically undruggable, high docking scores for those pockets are meaningless and may encourage the design of compounds that are unlikely to be viable

Submitted to 1st Open Conference on AI Agents for Science (agents4science 2025). Do not distribute.

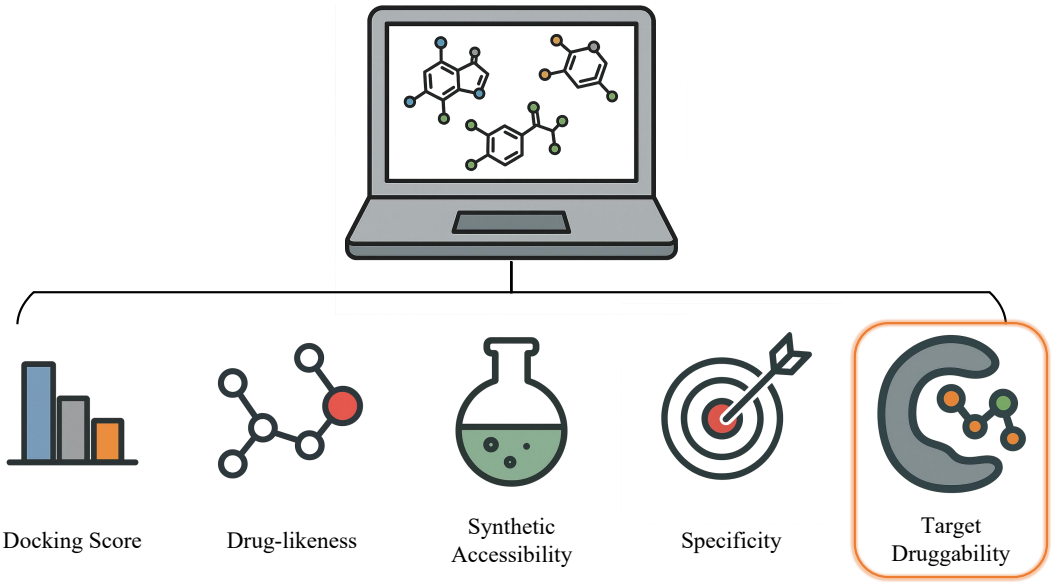

Figure 1: The evaluation of AI-driven structure-based drug design. We propose to incorporate **target druggability** into this evaluation framework.

drugs. Thus, there is a need to rethink evaluation protocols for structure-based generative models to account for the underlying druggability of targets. Recent perspectives also emphasize that the definition of druggability is itself evolving in the AI era. For instance, [3] highlights how machine learning can uncover new classes of druggable proteins by combining structural, sequence, and systems-level data. Classical definitions of druggability, based on geometric and physicochemical heuristics, are now being extended by AI approaches that leverage multi-modal biological knowledge, ranging from proteome-scale embeddings to network-based disease associations. This broadening of scope underscores why evaluation benchmarks that ignore druggability risk becoming detached from contemporary discovery practices.

In addition to these challenges, there is a growing awareness that modern SBDD must consider not only the ability to bind a target but also the broader pharmaceutical context. Molecules generated by AI models need to satisfy *druglikeness*, *synthetic accessibility*, and *specificity*. Experimental success stories—such as HIV protease inhibitors, kinase inhibitors and antibiotics identified through rational design—illustrate the potential of SBDD when the right targets are chosen [4]. Yet many clinical failures trace back to poor target selection or pockets that cannot be drugged.

To bridge the gap of aligning machine-learning evaluation with the realities of therapeutic discovery, we argue that generative models should not be judged solely on ligand-based metrics like docking scores in isolation; rather, these metrics need to reflect the underlying druggability of the target, as shown in Figure 1. To that end, we propose augmenting existing datasets like CrossDocked2020 [9] with continuous druggability scores and weighting generative performance accordingly. This refinement is expected to encourage models that prioritize genuinely tractable pockets and dissuade those that exploit bias in undruggable sites.

Looking ahead, embedding druggability into the AI-driven SBDD pipeline opens several opportunities. As more accurate pocket-scoring tools and experimental data become available, druggability-aware benchmarks could be expanded to cover a broader range of targets and conditions. The community could also explore models that jointly learn to predict druggability and generate ligands, ensuring that both target feasibility and molecular design evolve together. Ultimately, incorporating druggability should help steer generative algorithms toward compounds with a higher likelihood of clinical success, making this an important direction for future research and development.

2 Background

2.1 Structure-based Drug Design

Traditional SBDD workflows involve identifying a target protein, determining its 3D structure, locating a binding site and designing ligands iteratively. Computational techniques used in SBDD include structure-based virtual screening, molecular docking and molecular dynamics simulations [4]. Over the past decade, machine-learning models have been developed to predict binding affinities, model protein-ligand interactions and generate novel ligands [1, 15]. Deep learning models represent pockets as point clouds or graphs and learn features that capture spatial arrangements of chemical properties, including auto-regressive models [21, 13], diffusion models [10, 11] and flow models [22].

Evaluating these methods requires benchmarks that contain protein structures, binding pockets, and either known ligands or predictions derived from docking. CrossDocked2020 is currently the most widely used benchmark for pose prediction and generative design. It contains 13,839 unique ligands, 2,922 receptor pockets and ~22.6 million docked poses; about 41.9% of ligands have affinity data [9]. The dataset generates negative examples by cross-docking ligands into non-cognate pockets. For generative tasks of SBDD, benchmarks have been proposed to focus on docking scores (often from AutoDock Vina [23]): for example, the benchmark proposed by [8] uses the mean docking score for assessment. Besides, CBGBench [18] evaluates generative models within protein-ligand binding graphs, stressing relational reasoning. Tartarus [20] emphasizes realistic drug-design tasks, integrating pharmacokinetic constraints. Durian [19] provides a large-scale 3D molecular generation platform, enabling fair comparison across architectures. [27] recently questioned whether 3D methods consistently outperform 2D approaches, underscoring that methodological diversity remains essential. Collectively, these benchmarks demonstrate that while docking score remains the dominant evaluation criterion, there is growing interest in incorporating broader aspects of molecular feasibility—a trend our proposal seeks to extend by explicitly embedding druggability into evaluation.

2.2 Druggability and its Quantification

The concept of druggability emerged to prioritize proteins that can be modulated by small molecules. A druggable protein possesses a pocket whose shape and physicochemical properties complement drug-like molecules [5]. Several computational strategies exist to assess druggability:

- **Experience-based methods** rely on knowledge that members of the same family (e.g., GPCRs or kinases) have been successfully targeted by drugs. While useful, this approach may miss novel druggable proteins in uncharacterized families.
- **Ligand-based methods** infer druggability from known endogenous or synthetic ligands. The presence of a high-affinity ligand indicates that a suitable binding site exists, but this fails when no ligands are known.
- **Structure-based methods** analyze pocket geometry (size, depth, curvature), hydrophobicity and electrostatics. Geometry-based binding site predictors achieve ~74% success in identifying pockets [28]. Energy-based methods place probes around the pocket to estimate binding energies. Tools such as PockDrug [7], DrugPred [16], P2Rank [17] and DoGSiteScorer [24] combine these descriptors with machine-learning classifiers. These methods typically output continuous druggability scores or probabilities. Moreover, one-class learning approaches avoid explicitly defining the "non-druggable" class and learn the support of druggable pockets from positive examples [2].
- **Sequence-based methods** infer druggability from sequence motifs or protein-protein interaction networks. Machine-learning models using sequence features have been developed but often suffer from small training datasets and uncertain labels [12].
- **AI-driven methods.** In recent years, artificial intelligence has introduced a paradigm shift in druggability prediction. Unlike traditional structure- or sequence-based approaches, AI models integrate diverse feature sets, including 3D pocket descriptors, protein sequence embeddings, and systems biology context such as protein-protein interaction networks. For example, the DrugProtAI framework [12] applies robust ensemble learning and feature engineering to predict protein druggability with improved sensitivity and specificity, even

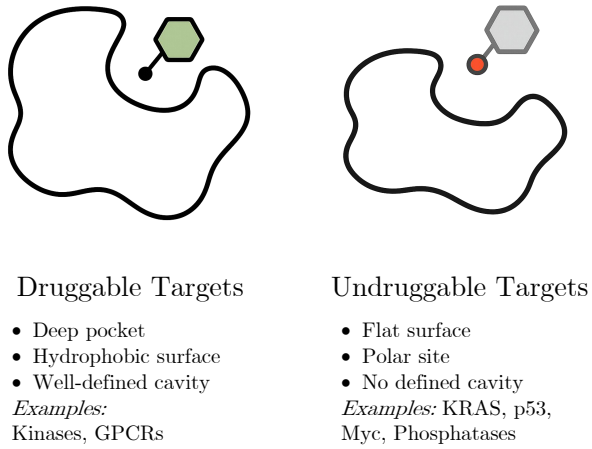

Figure 2: Druggable vs. undruggable targets.

when structural data are limited. Graph neural networks and transformer-based sequence encoders have also been applied to infer cryptic or allosteric binding sites that might escape conventional predictors. While these methods improve coverage of the "dark proteome," they bring new challenges in terms of interpretability and potential bias from noisy training datasets. Together, they suggest that druggability should no longer be viewed as a static property but as a dynamic prediction informed by AI across multiple biological scales.

Assessment tools highlight that druggable pockets tend to be large, deep and hydrophobic [5]. However, the lack of consensus on non-druggable examples and the dynamic nature of pockets complicate predictions [2].

Examples of druggable and undruggable targets. Understanding concrete examples helps illustrate why druggability matters. **Druggable targets** often belong to protein families with well-defined pockets that accommodate small molecules. Protein kinases are the classic example of druggable targets: they possess an ATP-binding pocket that is deep and conserved, and dozens of kinase inhibitors have been approved for clinical use. Indeed, reviews emphasize that kinases are a representative class of druggable targets [26]. G-protein-coupled receptors (GPCRs) form another large family of druggable proteins; their seven-transmembrane architecture presents extracellular binding pockets that are highly amenable to modulation, and many marketed drugs act on GPCRs.

By contrast, **undruggable targets** lack obvious pockets or have surfaces that are flat, polar, or involved in protein-protein interactions. A typical example is KRAS, a small GTPase that was long considered undruggable because its surface lacks a defined pocket and its shallow binding site has undesired polarity [26]. Although a covalent inhibitor (sotorasib) has recently been approved for a specific KRAS mutation, the general class of RAS proteins remains difficult to drug. Phosphatases—enzymes that remove phosphate groups—are structurally similar within families, making it challenging to achieve specificity; low specificity and associated side effects hinder drug discovery [26]. Transcription factors such as p53 and Myc regulate gene expression and are involved in numerous diseases. Their structural heterogeneity and lack of tractable binding sites mean that conventional small molecules cannot easily bind them. Finally, protein-protein interaction (PPI) interfaces with flat surfaces, such as those in the B-cell lymphoma-2 (Bcl-2) family and intrinsically disordered proteins, are also considered undruggable [26]. These examples underscore the diversity of undruggable targets and highlight the need for evaluation protocols that penalize models for focusing on pockets that are unlikely to yield drug-like modulators.

Importantly, the boundary between druggable and undruggable targets is increasingly fluid as AI-based analyzes uncover new opportunities. For instance, cryptic binding pockets in KRAS and Myc—once paradigmatic undruggable proteins—have been identified using machine-learning-guided structural mining and molecular dynamics simulations [12, 26]. Similarly, AI-driven discovery of covalent inhibitors and allosteric modulators has begun to shift long-standing assumptions about RAS proteins and transcription factors. Moreover, degrader strategies such as PROTACs, aided by AI in

linker and degrader design, provide avenues to target proteins previously considered inaccessible. These developments illustrate that undruggability is not absolute but context-dependent, and reinforce the need for evaluation frameworks that can adapt to changing definitions of target tractability.

Limitations of druggability metrics. Despite advances, druggability assessment remains imperfect. First, the absence of reliable negative datasets makes it difficult to robustly define "undruggable" pockets; many are simply untested rather than truly intractable [2]. Second, static crystal structures cannot capture conformational dynamics, such as cryptic or transient binding sites, which AI and enhanced sampling methods are only beginning to uncover. Third, existing predictors may overweight hydrophobicity, leading to false positives for shallow hydrophobic cavities. Finally, AI-driven methods such as DrugProtAI [11] rely on training data from known druggable targets, which may bias predictions against novel protein classes. These limitations caution against treating druggability as a binary label and motivate our proposal for probabilistic, continuously valued scores.

3 Methodology: Incorporating Druggability into SBDD Evaluation

Existing SBDD benchmarks evaluate a model's ability to rank or generate ligands based on docking scores or pose accuracy. These metrics implicitly assume all pockets are equally viable drug targets. To incorporate **druggability** into evaluation, we propose the following evaluation protocol, based on CrossDocked2020 [9]:

A. PDB-to-pocket mapping and preparation. CrossDocked provides receptor coordinates and pocket definitions derived from the Pocketome. For each pocket, we extract the coordinates of residues within a certain radius around the ligand binding site. Before analysis, we remove crystallographic waters and keep counterions consistent with the original CrossDocked protocol.

B. Druggability scoring. We compute a continuous druggability score for each pocket using a state-of-the-art predictor (e.g., PockDrug [7] or DrugPred [16]). These tools accept the pocket coordinates and return a probability that the pocket is druggable. If multiple models are available, we could average their predictions to reduce variance. Because not all pockets are known to be druggable or undruggable, using a probabilistic score rather than a binary label allows a smooth weighting.

In addition to established predictors like PockDrug and DrugPred, advanced AI-based druggability predictors are expected to be incorporated into the scoring step. Such models may capture not only static pocket geometry but also dynamic and functional determinants of druggability, including protein family patterns and disease associations. A practical strategy would be to compute both traditional structure-based scores and AI-predicted probabilities, then combine them—either through weighted averaging or multi-criteria optimization—when calculating the druggability scores. This hybrid approach allows benchmarks to remain grounded in physical chemistry while also reflecting AI-driven redefinitions of what constitutes a druggable pocket. As AI models evolve, their predictions could be dynamically updated, ensuring that benchmarks capture the expanding frontiers of tractable target space.

C. Reweighting of evaluation metrics. Let s_i denote the druggability score for pocket i, scaled to $[0, 1]$. In generative design tasks, models are typically evaluated only by ligand-based metrics such as the docking score. Based on this, we propose a **druggability-weighted docking score**:

$$\text{score}_{\text{weighted}} = \frac{\sum_i s_i \bar{D}_i}{\sum_i s_i}, \tag{1}$$

where $\bar{D}_i$ is the mean docking score of all generated ligands for pocket i. A higher s_i gives more weight to pockets that are more druggable. This weighting emphasizes generation of good binders for realistic targets and reduces the influence of undruggable pockets. Metrics of molecular diversity can, however, be reported separately and remain unweighted. [6, 14].

D. Thresholding and benchmark splits. To facilitate comparison with current benchmarks, we create subsets of CrossDocked2020 at different druggability thresholds (e.g., 0.2, 0.5, 0.8). The high-druggability subset includes pockets with $s_i > 0.5$ and represents realistic targets; the low-druggability subset can serve as negative controls or test a model's ability to avoid undruggable sites.

E. Calibration and validation. Because druggability predictors may themselves be biased, we recommend validating the reweighted benchmark using known drug-target pairs: evaluate whether pockets with high s_i correspond to proteins with approved drugs and adjust scoring functions accordingly. Further, one should test whether models that perform well under weighted metrics also yield compounds with favorable drug-likeness and high synthetic accessibility.

F. Analysis of model performance after weighting. After integrating druggability scores into the evaluation, researchers should analyze how different model classes perform across the druggability spectrum. For example, diffusion models conditioned on pocket geometry [11] may excel at generating ligands for highly druggable pockets because the latent space can capture well-defined cavities. Conversely, graph-based retrieval-augmented models or 1D/2D genetic algorithms might generate structurally diverse molecules that occasionally fill low-druggability or atypical pockets, leading to higher scores in the unweighted setting but being penalized under our weighting. Models that perform well at undruggable sites might be exploiting spurious correlations (e.g., generating large hydrophobic molecules), which could translate into poor specificity or toxicity. A comparative analysis can thus reveal the superiority of some methods at realistic targets (high druggability) and highlight the risks of overfitting to undruggable cavities. Such insights will guide future model development and help prioritise architectures that generalise across druggable targets while avoiding pathological behaviours.

4 Conclusion and Discussion

In summary, our work emphasizes that druggability is a critical variable missing from current SBDD evaluation protocols. By integrating druggability into CrossDocked2020, we aim to provide a more realistic assessment of generative models. Weighted metrics focus attention on pockets where medicinal chemistry is most likely to succeed and discourage the generation of large hydrophobic ligands that fill any cavity. Our approach also introduces heterogeneity into the benchmark, allowing researchers to compare model performance across different druggability regimes.

Several limitations should be acknowledged. Druggability predictors themselves rely on training datasets of known druggable pockets and may misclassify novel types of pockets; energy-based predictors may overestimate pockets that favor hydrophobic fragments. Moreover, weighting metrics by druggability may reduce the influence of novel but low-probability targets, potentially discouraging exploration of innovative chemistries. It is important to maintain separate analyzes for high- and low-druggability subsets rather than exclude the latter entirely.

Future work should explore joint learning of druggability and ligand design. Multitask models could simultaneously predict pocket druggability and generate ligands, allowing the model to allocate resources appropriately. Improved datasets with experimentally validated undruggable pockets would reduce reliance on proxies. Finally, evaluation should incorporate additional factors such as specificity, ADMET properties and synthetic accessibility. Nonetheless, integrating druggability into benchmark evaluation represents a practical step toward aligning AI-driven SBDD with real-world drug discovery.

Looking ahead, embedding AI-driven redefinitions of druggability into structure-based drug design promises to further align generative evaluation with therapeutic reality. Classical metrics based solely on pocket geometry risk excluding cryptic or context-dependent sites that AI now reveals as druggable. Conversely, AI methods risk overestimating tractability if benchmark design fails to enforce chemical realism. Thus, future SBDD benchmarks should adopt a hybrid paradigm: structural druggability scores for stability and interpretability, coupled with AI-derived predictions for sensitivity and discovery of novel opportunities. Multitask frameworks that co-train generative models to optimize both ligand fit and AI-predicted target feasibility represent an especially promising direction. Ultimately, this synthesis of structure-based heuristics with AI-derived insights could redefine not just how we evaluate generative models, but also how we conceptualize the druggable genome itself.

References

[1] Josh Abramson, Jonas Adler, Jack Dunger, Richard Evans, Tim Green, Alexander Pritzel, Olaf Ronneberger, Lindsay Willmore, Andrew J Ballard, Joshua Bambrick, et al. Accurate structure prediction of biomolecular interactions with alphafold 3. *Nature*, 630(8016):493–500, 2024.

[2] Riccardo Aguti, Erika Gardini, Martina Bertazzo, Sergio Decherchi, and Andrea Cavalli. Probabilistic pocket druggability prediction via one-class learning. *Frontiers in Pharmacology*, 13:870479, 2022.

[3] Karen Akinsanya, Mohammed AlQuraishi, Ann Boija, John Chodera, Anna Cichońska, Marzyeh Ghassemi, Martha Head, Wengong Jin, Warren A Kibbe, Nevan Krogan, et al. Redefining druggable targets with artificial intelligence. *Nature Biotechnology*, pages 1–3, 2025.

[4] Maria Batool, Bilal Ahmad, and Sangdun Choi. A structure-based drug discovery paradigm. *International journal of molecular sciences*, 20(11):2783, 2019.

[5] G Beis, AP Serafeim, and I Papasotiriou. Data-driven analysis and druggability assessment methods to accelerate the identification of novel cancer targets. *Computational and Structural Biotechnology Journal*, 21:46–57, 2023.

[6] Mostapha Benhenda. Chemgan challenge for drug discovery: can ai reproduce natural chemical diversity? *arXiv preprint arXiv:1708.08227*, 2017.

[7] Alexandre Borrel, Leslie Regad, Henri Xhaard, Michel Petitjean, and Anne-Claude Camproux. Pockdrug: A model for predicting pocket druggability that overcomes pocket estimation uncertainties. *Journal of chemical information and modeling*, 55(4):882–895, 2015.

[8] Tobiasz Cieplinski, Tomasz Danel, Sabina Podlewska, and Stanisław Jastrzebski. Generative models should at least be able to design molecules that dock well: A new benchmark. *Journal of Chemical Information and Modeling*, 63(11):3238–3247, 2023.

[9] Paul G Francoeur, Tomohide Masuda, Jocelyn Sunseri, Andrew Jia, Richard B Iovanisci, Ian Snyder, and David R Koes. Three-dimensional convolutional neural networks and a cross-docked data set for structure-based drug design. *Journal of chemical information and modeling*, 60(9):4200–4215, 2020.

[10] Jiaqi Guan, Wesley Wei Qian, Xingang Peng, Yufeng Su, Jian Peng, and Jianzhu Ma. 3d equivariant diffusion for target-aware molecule generation and affinity prediction. In *International Conference on Learning Representations (ICLR)*, 2023.

[11] Jiaqi Guan, Xiangxin Zhou, Yuwei Yang, Yu Bao, Jian Peng, Jianzhu Ma, Qiang Liu, Liang Wang, and Quanquan Gu. Decompdiff: diffusion models with decomposed priors for structure-based drug design. In *International Conference on Machine Learning (ICML)*, 2023.

[12] Ankit Halder, Sabyasachi Samantaray, Sahil Barbade, Aditya Gupta, and Sanjeeva Srivastava. Drugprotai: A machine learning–driven approach for predicting protein druggability through feature engineering and robust partition-based ensemble methods. *Briefings in Bioinformatics*, 26(4):bbaf330, 2025.

[13] Xiuyuan Hu, Guoqing Liu, Can Chen, Yang Zhao, Hao Zhang, and Xue Liu. 3dmolformer: A dual-channel framework for structure-based drug discovery. In *International Conference on Learning Representations (ICLR)*, 2025.

[14] Xiuyuan Hu, Guoqing Liu, Quanming Yao, Yang Zhao, and Hao Zhang. Hamiltonian diversity: effectively measuring molecular diversity by shortest hamiltonian circuits. *Journal of Cheminformatics*, 16(1):94, 2024.

[15] Xiuyuan Hu, Guoqing Liu, Yang Zhao, and Hao Zhang. De novo drug design using reinforcement learning with multiple gpt agents. *Advances in Neural Information Processing Systems*, 36, 2024.

[16] Agata Krasowski, Daniel Muthas, Aurijit Sarkar, Stefan Schmitt, and Ruth Brenk. Drug-pred: a structure-based approach to predict protein druggability developed using an extensive nonredundant data set. *Journal of chemical information and modeling*, 51(11):2829–2842, 2011.

[17] Radoslav Krivák and David Hoksza. P2rank: machine learning based tool for rapid and accurate prediction of ligand binding sites from protein structure. *Journal of cheminformatics*, 10(1):39, 2018.

[18] Haitao Lin, Guojiang Zhao, Odin Zhang, Yufei Huang, Lirong Wu, Zicheng Liu, Siyuan Li, Cheng Tan, Zhifeng Gao, and Stan Z Li. Cbgbench: fill in the blank of protein-molecule complex binding graph. *arXiv preprint arXiv:2406.10840*, 2024.

[19] Dou Nie, Huifeng Zhao, Odin Zhang, Gaoqi Weng, Hui Zhang, Jieyu Jin, Haitao Lin, Yufei Huang, Liwei Liu, Dan Li, et al. Durian: A comprehensive benchmark for structure-based 3d molecular generation. *Journal of Chemical Information and Modeling*, 65(1):173–186, 2024.

[20] AkshatKumar Nigam, Robert Pollice, Gary Tom, Kjell Jorner, John Willes, Luca Thiede, Anshul Kundaje, and Alán Aspuru-Guzik. Tartarus: A benchmarking platform for realistic and practical inverse molecular design. *Advances in Neural Information Processing Systems*, 36:3263–3306, 2023.

[21] Xingang Peng, Shitong Luo, Jiaqi Guan, Qi Xie, Jian Peng, and Jianzhu Ma. Pocket2mol: Efficient molecular sampling based on 3d protein pockets. In *International Conference on Machine Learning (ICML)*, 2022.

[22] Yanru Qu, Keyue Qiu, Yuxuan Song, Jingjing Gong, Jiawei Han, Mingyue Zheng, Hao Zhou, and Wei-Ying Ma. Molcraft: Structure-based drug design in continuous parameter space. In *International Conference on Machine Learning (ICML)*, 2024.

[23] Oleg Trott and Arthur J Olson. Autodock vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization, and multithreading. *Journal of computational chemistry*, 31(2):455–461, 2010.

[24] Andrea Volkamer, Daniel Kuhn, Friedrich Rippmann, and Matthias Rarey. Dogsitescorer: a web server for automatic binding site prediction, analysis and druggability assessment. *Bioinformatics*, 28(15):2074–2075, 2012.

[25] Jesse A Weller and Remo Rohs. Structure-based drug design with a deep hierarchical generative model. *Journal of Chemical Information and Modeling*, 64(16):6450–6463, 2024.

[26] Xin Xie, Tingting Yu, Xiang Li, Nan Zhang, Leonard J Foster, Cheng Peng, Wei Huang, and Gu He. Recent advances in targeting the "undruggable" proteins: from drug discovery to clinical trials. *Signal transduction and targeted therapy*, 8(1):335, 2023.

[27] Kangyu Zheng, Yingzhou Lu, Zaixi Zhang, Zhongwei Wan, Yao Ma, Marinka Zitnik, and Tianfan Fu. Structure-based drug design benchmark: do 3d methods really dominate? *arXiv preprint arXiv:2406.03403*, 2024.

[28] Xiliang Zheng, LinFeng Gan, Erkang Wang, and Jin Wang. Pocket-based drug design: exploring pocket space. *The AAPS journal*, 15(1):228–241, 2013.

Agents4Science AI Involvement Checklist

1. **Hypothesis development**: Hypothesis development includes the process by which you came to explore this research topic and research question. This can involve the background research performed by either researchers or by AI. This can also involve whether the idea was proposed by researchers or by AI.

   Answer: **[C]**

   Explanation: Humans presented the initial idea, while AI conducted background research and analysis.

2. **Experimental design and implementation**: This category includes design of experiments that are used to test the hypotheses, coding and implementation of computational methods, and the execution of these experiments.

   Answer: **[D]**

   Explanation: No computational experiments is conducted.

3. **Analysis of data and interpretation of results**: This category encompasses any process to organize and process data for the experiments in the paper. It also includes interpretations of the results of the study.

   Answer: **[D]**

   Explanation: No computational experiments is conducted.

4. **Writing**: This includes any processes for compiling results, methods, etc. into the final paper form. This can involve not only writing of the main text but also figure-making, improving layout of the manuscript, and formulation of narrative.

   Answer: **[C]**

   Explanation: Humans mainly did typesetting.

5. **Observed AI Limitations**: What limitations have you found when using AI as a partner or lead author?

   Description: Some hallucinations are observed, such as giving some incorrect examples with inexistent references.

Agents4Science Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include theoretical results.

- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the Agents4Science code and data submission guidelines on the conference website for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, or overall run with given experimental conditions).

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [NA]

   Justification:

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the Agents4Science Code of Ethics (see conference website)?

   Answer: [Yes]

   Justification:

   Guidelines:

   - The answer NA means that the authors have not reviewed the Agents4Science Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification:

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations, privacy considerations, and security considerations.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies.