We Should At Least Be Able To Design Molecules That Dock Well

Tobiasz Cieplinski[1], Tomasz Danel[1], Sabina Smusz[1], Stanisław Jastrzębski[23]

Abstract

Designing compounds with desired properties is a key element of the drug discovery process. However, measuring progress in the field has been challenging due to lack of realistic retrospective benchmarks, and the large cost of prospective validation. To close this gap, we propose a benchmark based on docking, a popular computational method for assessing molecule binding to a protein. Concretely, the goal is to generate molecules that are scored highly by SMINA, a popular docking software. We observe that current state-of-the-art technique, graph-based generative models, fail to generate molecules with a high docking score. This suggest a limitation of the current incarnation of models for de novo drug design. Finally, we propose a simplified version of the benchmark based on an easier to optimize docking score, and show that the tested models are able to partially solve it. We release the benchmark as an easy to use package available at https://github.com/cieplinski-tobiasz/smina-docking-benchmark. We hope that our benchmark will become a stepping stone towards the goal of automatically generating promising drug candidates.

1 Introduction

Designing compounds with some desired chemical properties is the central challenge in the drug discovery process [Sliwoski et al., 2014, Elton et al., 2019]. De novo drug design is one of the most successful computational approach that involves generating new structures *from scratch*, which avoids enumerating explicitly the vast space of possible structures. Recently, deep learning has unlocked new progress in drug design. Promising results using deep generative models have been shown in generating soluble [Kusner et al., 2017a], bioactive [Segler et al., 2018], and drug-like [Jin et al., 2018b] molecules.

A key challenge in the field of drug design is the lack of realistic benchmarks [Elton et al., 2019]. Ideally, the proposed molecule should be tested in the wet lab for the desired property. In practice, a proxy is typically used. For example, octanol-water partition coefficient or bioactivity is calculated using a computational model [Segler et al., 2018, Kusner et al., 2017a]. These models are often too simplistic [Elton et al., 2019]. This is aptly summarized in Coley et al. [2019] as "The current evaluations for generative models do not reflect the complexity of real discovery problems.". In contrast to drug design, more realistic benchmarks have been used in the design of photovoltaics [Pyzer-Knapp et al.,

[1] Jagiellonian University, Poland
[2] New York University, USA
[3] Molecule.one, Poland

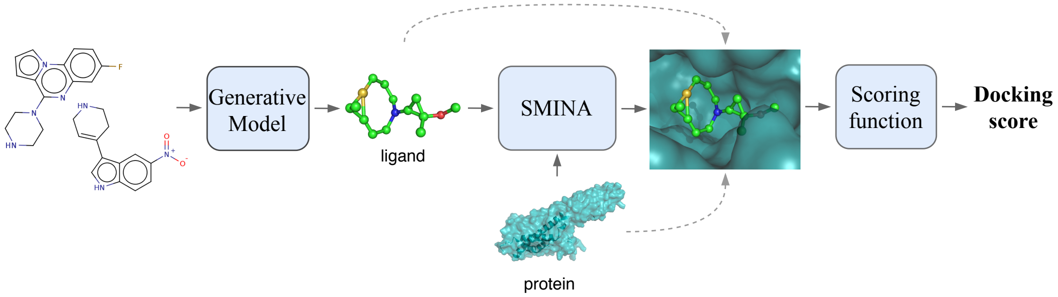

Figure 1: Visualization of the proposed docking-based benchmark for de-novo drug design methods. First, the proposed molecule (leftmost) is docked to the target's binding site using SMINA, a popular docking software. The final score is computed based on the ligand pose. In particular, the hardest version of the benchmark uses the original SMINA docking score as the evaluation of the proposed molecule.

2015] or molecules with certain excitation energies [Sumita et al., 2018], where a physical calculation was carried out to both train models, and evaluate generated compounds.

Our main contribution is a more realistic benchmark for de novo drug design, compared to previous work. We base our benchmark on docking, a popular computational method for assessing molecule binding to a protein. Concretely, the goal is to generate molecules that are scored highly by SMINA [Koes et al., 2013]. We picked Koes et al. [2013] due to its popularity and having a free license. While we focus on de novo drug design, our methodology can be extended to evaluate retrospectively other approaches to designing molecules. Code to reproduce results and evaluate new models is available online at https://github.com/cieplinski-tobiasz/smina-docking-benchmark.

Our second contribution is exposing limitation of currently popular de novo drug design methods for generating bioactive molecules. When trained using a few hundreds compounds, a typical training set size, the tested methods fail to generate active structures according to the docking software. This suggest we should exercise caution when applying them in drug discovery pipelines. We hope our benchmark will serve as a stepping stone to further improve these promising models.

The paper is organised as follows. We first discuss prior work and introduce our benchmark. Next, we use our benchmark to evaluate two popular models for de novo drug design. Finally, we analyse why the tested models fail on the most difficult version of the benchmark.

2 Docking-based benchmark

We begin by briefly discussing prior work and motivation. Next, we introduce our benchmark.

2.1 Why do we need yet another benchmark?

Standardized benchmarks are critical to measure progress in any field. Development of large-scale benchmarks such as the ImageNet was critical for the recent developments in artificial intelligence [Deng et al., 2009, Wang et al., 2018]. Many new methods for de novo drug design are conceived every year, which motivates the need for a systematic comparison [Schneider and Clark, 2019].

De novo drug design methods are typically evaluated using *proxy tasks* that circumvent the need to test the generated compounds experimentally [Jin et al., 2018a, You et al., 2018, Maziarka et al., 2020, Kusner et al., 2017b, Gómez-Bombarelli et al., 2016]. Optimizing the octanol-water partition coefficient (logP) is a common example. The logP coefficient is commonly computed using an atom-based method that involves summing contribution of individual atoms [Wildman and Crippen, 1999, Jin et al., 2018a], which is available in the RDKit package [Landrum, 2016]. Due to the fact that it is easy to optimize the atom-based method by producing unrealistic molecules [Brown et al., 2019], a version that heuristically penalizes hard to synthesize compounds is used in practice [Jin et al., 2018a]. This example illustrates the need to develop more realistic ways to benchmark these methods. The QED score [Bickerton et al., 2012] which is designed to capture *druglikeliness* of a compound, is another example. Finally, some papers use a model (e.g. a neural network) to predict bioactivity of the generated compounds Segler et al. [2018]. Similarly to logP, these two tasks are also possible to optimize by producing unrealistic molecules. This is aptly summarized in Coley et al. [2019] as

> “The current evaluations for generative models do not reflect the complexity of real discovery problems.”

Interestingly, besides the aforementioned proxy tasks, rarely more realistic proxy tasks are used in the context of evaluating de novo drug design methods. This is in contrast to evaluation of generative models for generating photovoltaics [Pyzer-Knapp et al., 2015] or molecules with certain excitation energies [Sumita et al., 2018]. One notable exception is Aumentado-Armstrong [2018] who try to generate compounds that are active according to the DrugScore [Neudert and Klebe, 2011], and then evaluate the generated compounds using rDock [Ruiz-Carmona et al., 2014]. This lack of the overall diversity and realism in the typically used evaluation methods motivates our benchmark.

2.2 Docking-based benchmark

Our docking-based benchmark is defined by the following three ingredients: (1) docking software that computes for a generated compound its pose in the binding site, (2) a function that scores the pose, (3) a training set of compounds with an already computed docking score.

The goal is to generate a given number of molecules that achieve the maximum possible docking score. For the sake of simplicity, we do not impose limits on the distance of the proposed compounds to training set. Thus a simple baseline is to return the training set. Finding similar compounds that have a higher docking score is already prohibitively challenging for current state-of-the-art methods. As the field progresses, our benchmark can be easily extended to account for the similarity between the generated compounds and the training set.

Finally, we would like to stress that the benchmark is not limited to de novo methods. The benchmark is applicable to any other approaches such as virtual screening. The only limitation required for a fair comparison is that docking is performed only on the supplied training set.

2.3 Instantiation

As a concrete instantiation of our docking-based benchmark, we use SMINA [Koes et al., 2013] due to its wide-spread use and being offered under a free license. To form the training set, we download from the ChEMBL [Gaulton et al., 2016] database X molecules tested against a popular drug target, the 5-HT1B receptor. This dataset contains X active and X inactive molecules.

We dock each molecule using default settings in SMINA to a manually selected binding site coordinate. The protein structure are downloaded from the PDB database, and prepared using a state-of-the-art chemist referred herein to as Sabina. The resulting protein structure is provided in our code repository.

Starting from the above, we define the following three variants of the benchmark. In the first variant, we use the default SMINA docking score. The goal is to propose molecules that achieve the smallest SMINA docking score, defined as follows:

$$\begin{aligned}\text{Docking score} = &-0.035579 \cdot \text{gauss}(o=0, w=0.5)\\ &-0.005156 \cdot \text{gauss}(o=3, w=2)\\ &+0.840245 \cdot \text{repulsion}(o=0)\\ &-0.035069 \cdot \text{hydrophobic}(g=0.5, b=1.5)\\ &-0.587439 \cdot \text{non\_dir\_h\_bond}(g=-0.7, b=0)\\ &+1.923 \cdot \text{num\_tors\_div}\end{aligned},$$

where all terms are computed based on the final docking pose. The first three terms measure the steric interaction between ligand and the protein. The fourth and the fifth term look for hydrophobic and hydrogen bonds between the ligand and the protein. The final term penalizes ligands with many torsional bonds. We include in App. A a detailed description of all the terms.

Next, we propose two simpler variants of the benchmark based on individual terms in the SMINA scoring function. In the *Repulsion* task, the goal is to only minimize the repulsion component, which is defined as:

$$\text{repulsion}(a_1, a_2) = \begin{cases} d_{\text{diff}}(a_1, a_2)^2, & d_{\text{diff}}(a_1, a_2) < 0 \\ 0, & \text{otherwise} \end{cases}$$

where $d_{\text{diff}}(a_1, a_2)$ is the distance between the atoms minus the sum of their van der Waals radii. Distance unit is Angstrom (10^{-10}m).

The third task, *Hydrogen Bond Task*, is to maximize the non_dir_h_bond term:

$$\text{non\_dir\_h\_bond}(a_1, a_2) = \begin{cases} 0, & (a_1, a_2) \text{ do not form hydrogen bond} \\ 1, & d_{\text{diff}}(a_1, a_2) < -0.7 \\ 0, & d_{\text{diff}}(a_1, a_2) \geq 0 \\ \frac{d_{\text{diff}}(a_1, a_2)}{-0.7} & \text{otherwise.} \end{cases}$$

To make the results more stable, instead of picking a single pose which achieved the best docking score, we average the result over the top 5 best poses. Finally, to make the benchmark more realistic, we filter the generated compounds using the Lipinski rule, and discard molecules with molecular weight lower than 100.

3 Results and discussion

In this section we evaluate two popular models for de novo drug design on our benchmark.

3.1 Models

We compare two popular models for de novo drug design. Chemical Variational Autoencoder (CVAE) [Gómez-Bombarelli et al., 2018] applies Variational Autoencoder [Kingma and Welling, 2013] by representing molecules as strings of characters (using SMILES encoding). This approach was later extended by Grammar Variational Autoencoder (GVAE) [Kusner et al., 2017a], which ensures that generated compounds are grammatically correct.

3.2 Experimental details

To generate active compounds, we follow a similar approach to in Gómez-Bombarelli et al. [2018] and Kusner et al. [2017a]. First, we fine-tune a given generative model for 5 epochs on the training set ligands, starting from weights made available by the authors[1]. All hyperparameters are set to default values used in Gómez-Bombarelli et al. [2018] and Kusner et al. [2017a]. Additionally, we use the provided scores to train a multilayer perceptron (MLP) to predict the target (e.g. SMINA scoring function) from the latent space representation of the molecule.

To generate compounds, we first take a random sample of the model latent space by sampling from standard Gaussian distribution with standard deviation of 1 and mean of 0. Starting from this point in the latent space, we take Z gradient steps to optimize the output of the MLP. Based on this approach we generate 250 compounds from the model.

All other experimental details, including hyperparameter values used in the experiments, can be found in Appendix B.

In the experiments, we compare to three baselines: (1) random compounds from ZiNC, (2) random unseen in training active compounds (Ki $<$ 100nm), (3) random unseen in training inactive compounds (Ki $>$ 1000nm).

3.3 Results

In this section, we use the above procedure to generate compounds for the three targets defined in Section 2.3. Table 1 summarizes the results on all three tasks. Below we make several observations.

First, we observe that CVAE and GVAE models fail to generate compounds that achieve higher scores than the three baselines. In particular, both models produce compounds that achieve on average docking scores below the mean in the training set (x). Even looking at the Top 1%, the achieved scores are below the mean. This is reminiscent of results in Gao and Coley [2020]. They show that de novo models tend to generate difficult to synthesize molecules, even if optimizing for a proxy of synthesability. Similarly, even though we optimize for docking score using the MLP, the models fail to optimize the actual score. We will delve deeper into this direction in Section 3.4.

The next two tasks are simpler. On the Repulsion task, both models improve upon the baselines. GVAE generates compounds with remarkably low repulsion that is an order

[1] Available at https://github.com/aspuru-guzik-group/chemical$_{v}ae/tree/master/models/zinc$ and at https://github.com/mkusner/grammarVAE/tree/master/pretrained.

of magnitude lower (x) than the best scores observed in the training set (x). This however should be interpreted with caution. It is possible to minimize repulsion by avoiding docking to the binding site, which might explain why Inactives tend to have lower repulsion than Actives. As such we suggest to treat the Repulsion task as a form of a unit test for the validity of the generating procedure.

The Hydrogen Bond task difficulty lies between the first two tasks. The mean score of molecules generated by GVAE is x larger than the mean score of Actives (x for Top 1%). Contrarily, CVAE Top 1% results are worse the ones in Inactives dataset. This suggests that maximizing the non_dir_h_bond term in Equation A is harder than minimizing the repulsion term, but is still an easier task than optimizing the whole docking score.

The promising results on the Repulsion and the Hydrogen Bond tasks show it is feasible to optimize individual components of the score. This suggests that solving SMINA Score task is an attainable goal.

Our results also suggest that generative models applied to de novo drug discovery pipelines might require substantial more data to generate realistic compounds than is typically available for training. Despite using over x compounds for training, the achieved docking scores are worse than in a random sample from the ZiNC dataset. Docking score is only a simple proxy of the actual binding affinity, and as such it should worry us that it is already challenging to optimize.

	5HT1B	5HT2B	ACM2	CYP2D6
ZiNC	-8.241 (-12.068)	-8.303 (-14.477)	-7.587 (-11.533)	-6.873 (-10.601)
Inactives	-7.707 (-11.306)	-8.375 (-11.212)	-6.971 (-10.451)	-6.992 (-10.76)
Actives	-8.727 (-12.294)	-8.527 (-14.38)	-8.156 (-11.532)	-6.866 (-8.869)
CVAE	-4.888 (-8.942)	x (x)	x (x)	-4.829 (-7.719)
GVAE	-4.681 (-7.507)	-4.139 (-6.983)	-5.156 (-7.869)	-5.425 (-7.590)

Table 1: Score Function Task

	5HT1B	5HT2B	ACM2	CYP2D6
ZiNC	3.438 (0.571)	2.559 (0.557)	5.200 (0.517)	5.716 (0.604)
Inactives	3.092 (0.548)	2.368 (0.616)	4.781 (0.601)	5.828 (0.787)
Actives	3.688 (1.417)	2.729 (0.785)	5.531 (0.911)	5.186 (0.995)
CVAE	x (x)	0.741 (0.034)	0.806 (0.076)	x (x)
GVAE	1.200 (0.039)	0.686 (0.017)	x (x)	x (x)

Table 2: Repulsion Task

	5HT1B	5HT2B	ACM2	CYP2D6
ZiNC	1.547 (4.983)	0.971 (3.379)	1.012 (5.805)	0.634 (4.533)
Inactives	1.258 (3.358)	0.945 (3.216)	0.948 (5.455)	0.689 (4.363)
Actives	1.698 (4.994)	1.095 (3.037)	1.124 (4.025)	0.475 (1.551)
CVAE	1.234 (5.059)	0.818 (3.585)	0.717 (4.241)	0.583 (4.620)
GVAE	2.581 (11.225)	2.111 (6.525)	2.771 (10.071)	2.339 (7.548)

Table 3: Hydrogen Bonding Task

Table 4: Benchmark results for the three tasks on 5-HT1B and 5-HT2B protein. The goal is to propose 250 compounds achieving the lowest mean SMINA docking score towards a given protein. Additionally, each method we report in parenthesis mean score for the top 1% of compounds. We observe that for Score Function Task CVAE and GVAE fail to outperform ZiNC, which samples 250 compounds from the ZiNC database.

3.4 Analysis

In this section, we investigate why do CVAE and GVAE models fail to generate compounds that achieve high docking scores.

The most natural hypothesis is that predicting docking score is difficult. Our procedure uses the gradient of the MLP to generate active compounds. If the model fails at predicting activity, it is reasonable to assume it also fails to guide the generating process towards active compounds.

Recall that both models failed to generate compounds that achieve better docking scores towards 5HT1B than the considered baselines. However, the MLP predicts that compounds are active towards 5HT1B. We show this in Table 5.

To quantify this effect, we measure the root mean squared error (MSE) between the predicted and the true docking score on three datasets: (1) a holdout test subset of the fine-tuning dataset (*Holdout*), (2) 100 random samples of latent space from Gaussian distribution

	5HT1B	5HT2B	ACM2	CYP2D6
CVAE	-60.071 (-58.792)	x (x)	x (x)	-515.886 (-377.572)
GVAE	-36.565 (-25.608)	-50.412 (-47.629)	-66.009 (-53.335)	-55.021 (-47.307)

Table 5: Predicted docking score towards 5HT1B by the MLP for the compounds presented in Table 1. Comparing to Table 1, we can observe that the predicted docking scores are two, three or even four orders of magnitude overestimating the true activity.

(*Random Gauss*), and (3) 250 molecules generated using gradient latent space optimization (*Generated*). We compare this to SMINA by redocking compounds and measuring the discrepancy between the two docking runs (*SMINA Rerun*).

Table 8 reports all the results, and Figure 2 shows the predicted docking score against the true docking score on the compounds generated by CVAE. We observe that on all subsets, excluding Holdout, the RMSE is two to even four orders of magnitude higher than RMSE between two docking runs (SMINA Rerun). This shows that using better models for predicting docking scores is a particularly promising avenue for improving results on the benchmark.

	MLP	SMINA Rerun
Holdout	x	x
Random Gauss	4.064	x
Generated	56.062	x

Table 6: CVAE

	MLP	SMINA Rerun
Holdout	x	x
Random Gauss	8.24	x
Generated	33.646	x

Table 7: GVAE

Table 8: The root mean square error between the predicted and the true docking scores for the generative models, compared to SMINA.

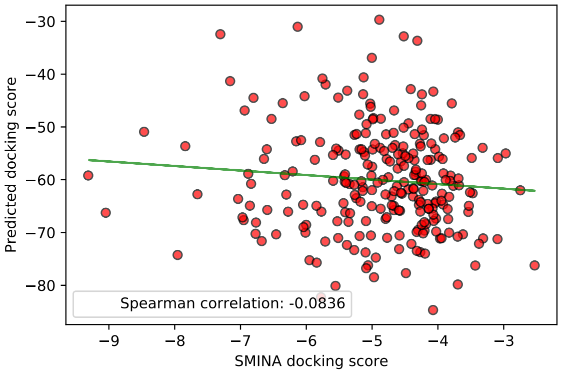

Figure 2: The predicted docking score (vertical axis) versus the true docking score (horizontal axis) for the 250 compounds generated by CVAE. Improving modeling of the docking score is a promising avenue for improving on the benchmark.

4 Conclusion

As concluded by Coley et al. [2019], “the current evaluations for generative models do not reflect the complexity of real discovery problems”. In this work, we proposed a new, more realistic, benchmark tailored to de novo drug design using docking score as the target to optimize. Code to evaluate new models is available at https://github.com/cieplinski-tobiasz/smina-docking-benchmark.

Our results also suggest that generative models applied to de novo drug discovery pipelines might require substantial more data to generate realistic compounds than is typically available for training. Despite using over 8000 compounds for training, the achieved docking scores are worse than in a random sample from the ZiNC dataset. Docking score is only a simple proxy of the actual binding affinity, and as such it should worry us that it is already challenging to optimize. As such, it might better reflect “the complexity of real discovery problems“.

On a more optimistic note, the tested models were able to optimize the number of hydrogen bonds to the binding site, which is a term in the SMINA scoring function. This suggests that producing compounds that optimize docking score based on the provided dataset is an attainable, albeit challenging, task.

References

Tristan Aumentado-Armstrong. Latent molecular optimization for targeted therapeutic design. *CoRR*, abs/1809.02032, 2018.

G. Richard Bickerton, Gaia V. Paolini, Jérémy Besnard, Sorel Muresan, and Andrew L. Hopkins. Quantifying the chemical beauty of drugs. *Nature Chemistry*, 4(2):90–98, 2012.

Nathan Brown, Marco Fiscato, Marwin H.S. Segler, and Alain C. Vaucher. Guacamol: Benchmarking models for de novo molecular design. *Journal of Chemical Information and Modeling*, 59(3):1096–1108, 2019.

Connor W Coley, Natalie S Eyke, and Klavs F. Jensen. Autonomous discovery in the chemical sciences part ii: Outlook. *Angewandte Chemie International Edition*, n/a(n/a), 2019.

J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *CVPR09*, 2009.

Daniel C. Elton, Zois Boukouvalas, Mark D. Fuge, and Peter W. Chung. Deep learning for molecular design—a review of the state of the art. *Mol. Syst. Des. Eng.*, 4:828–849, 2019.

Wenhao Gao and Connor W. Coley. The synthesizability of molecules proposed by generative models, 2020.

Anna Gaulton, Anne Hersey, Michał Nowotka, A. Patrícia Bento, Jon Chambers, David Mendez, Prudence Mutowo, Francis Atkinson, Louisa J. Bellis, Elena Cibrián-Uhalte, Mark Davies, Nathan Dedman, Anneli Karlsson, María Paula Magariños, John P. Overington, George Papadatos, Ines Smit, and Andrew R. Leach. The ChEMBL database in 2017. *Nucleic Acids Research*, 45(D1):D945–D954, 11 2016. ISSN 0305-1048. doi: 10.1093/nar/gkw1074. URL https://doi.org/10.1093/nar/gkw1074.

Rafael Gómez-Bombarelli, David Duvenaud, José Miguel Hernández-Lobato, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *CoRR*, abs/1610.02415, 2016.

Rafael Gómez-Bombarelli, Jennifer N. Wei, David Duvenaud, José Miguel Hernández-Lobato, Benjamín Sánchez-Lengeling, Dennis Sheberla, Jorge Aguilera-Iparraguirre, Timothy D. Hirzel, Ryan P. Adams, and Alán Aspuru-Guzik. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science*, 4(2):268–276, Jan 2018. ISSN 2374-7951.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation. In *International Conference on Machine Learning*, pages 2328–2337, 2018a.

Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Junction tree variational autoencoder for molecular graph generation, 2018b.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes, 2013. URL http://arxiv.org/abs/1312.6114. cite arxiv:1312.6114.

David Ryan Koes, Matthew P. Baumgartner, and Carlos J. Camacho. Lessons learned in empirical scoring with smina from the csar 2011 benchmarking exercise. *Journal of Chemical Information and Modeling*, 53(8):1893–1904, 2013. PMID: 23379370.

Matt J. Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder, 2017a.

Matt J. Kusner, Brooks Paige, and José Miguel Hernández-Lobato. Grammar variational autoencoder, 2017b.

Greg Landrum. Rdkit: Open-source cheminformatics software. 2016. URL https://github.com/rdkit/rdkit/releases/tag/Release_2016_09_4.

Łukasz Maziarka, Agnieszka Pocha, Jan Kaczmarczyk, Krzysztof Rataj, Tomasz Danel, and Michał Warchoł. Mol-cyclegan: a generative model for molecular optimization. *Journal of Cheminformatics*, 12(1):2, 2020.

Gerd Neudert and Gerhard Klebe. Dsx: A knowledge-based scoring function for the assessment of protein–ligand complexes. *Journal of Chemical Information and Modeling*, 51 (10):2731–2745, 10 2011.

Edward O. Pyzer-Knapp, Kewei Li, and Alan Aspuru-Guzik. Learning from the harvard clean energy project: The use of neural networks to accelerate materials discovery. *Advanced Functional Materials*, 25(41):6495–6502, 2015.

Sergio Ruiz-Carmona, Daniel Alvarez-Garcia, Nicolas Foloppe, A. Beatriz Garmendia-Doval, Szilveszter Juhos, Peter Schmidtke, Xavier Barril, Roderick E. Hubbard, and S. David Morley. rdock: A fast, versatile and open source program for docking ligands to proteins and nucleic acids. *PLOS Computational Biology*, 10(4):1–7, 04 2014.

Gisbert Schneider and David E. Clark. Automated de novo drug design: Are we nearly there yet? *Angewandte Chemie International Edition*, 58(32):10792–10803, 2019.

Marwin H. S. Segler, Thierry Kogej, Christian Tyrchan, and Mark P. Waller. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Central Science*, 4(1):120–131, 2018.

Gregory Sliwoski, Sandeepkumar Kothiwale, Jens Meiler, and Edward W. Lowe. Computational methods in drug discovery. *Pharmacological Reviews*, 66(1):334–395, 2014. ISSN 0031-6997.

Masato Sumita, Xiufeng Yang, Shinsuke Ishihara, Ryo Tamura, and Koji Tsuda. Hunting for Organic Molecules with Artificial Intelligence: Molecules Optimized for Desired Excitation Energies. 4 2018.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium, November 2018. Association for Computational Linguistics.

Scott A. Wildman and Gordon M. Crippen. Prediction of physicochemical parameters by atomic contributions. *Journal of Chemical Information and Computer Sciences*, 39(5): 868–873, 09 1999.

Jiaxuan You, Bowen Liu, Rex Ying, Vijay S. Pande, and Jure Leskovec. Graph convolutional policy network for goal-directed molecular graph generation. *CoRR*, abs/1806.02473, 2018.

A Default SMINA scoring function

We include the definitions of SMINA's default scoring function components and weights used for calculating docking score. a_1 and a_2 denote atoms, $d(a_1, a_2)$ is the distance between atoms, d_{opt} is the sum of their van der Waals radii and $d_{\text{diff}}(a_1, a_2) = d(a_1, a_2) - d_{\text{opt}}(a_1, a_2)$. Distance unit is Angstrom (10^{-10}m).

$$\begin{aligned}
\text{Docking score} = &- 0.035579 \cdot \text{gauss}(o = 0, w = 0.5) \\
&- 0.005156 \cdot \text{gauss}(o = 3, w = 2) \\
&+ 0.840245 \cdot \text{repulsion} \\
&- 0.035069 \cdot \text{hydrophobic} \\
&- 0.587439 \cdot \text{non\_dir\_h\_bond} \\
&+ 1.923 \cdot \text{num\_tors\_div}
\end{aligned}$$

$$\begin{aligned}
\text{gauss}(a_1, a_2) &= \exp\left(-\left(\frac{d_{\text{diff}}(a_1, a_2) - o}{w}\right)^2\right) \\
\text{repulsion}(a_1, a_2) &= \begin{cases} d_{\text{diff}}(a_1, a_2)^2, & d_{\text{diff}}(a_1, a_2) < 0 \\ 0, & \text{otherwise} \end{cases} \\
\text{hydrophobic}(a_1, a_2) &= \begin{cases} 0, & \textit{not\_hydrophobic}(a_1) \text{ or } \textit{not\_hydrophobic}(a_2) \\ 1, & d_{\text{diff}}(a_1, a_2) < 0.5 \\ 0, & d_{\text{diff}}(a_1, a_2) \geq 1.5 \\ 1.5 - d_{\text{diff}}(a_1, a_2), & \text{otherwise} \end{cases} \\
\text{non\_dir\_h\_bond}(a_1, a_2) &= \begin{cases} 0, & (a_1, a_2) \text{ do not form hydrogen bond} \\ 1, & d_{\text{diff}}(a_1, a_2) < -0.7 \\ 0, & d_{\text{diff}}(a_1, a_2) \geq 0 \\ \frac{d_{\text{diff}}(a_1, a_2)}{-0.7}, & \text{otherwise} \end{cases} \\
\text{num\_tors\_div} &= \ldots
\end{aligned}$$

B Model details

We include hyperparameters and training settings used in our models. Our code is available at https://github.com/cieplinski-tobiasz/smina-docking-benchmark.

MLP is used to predict docking score from CVAE or GVAE latent space representation of molecule. It is a simple feed forward neural network with one hidden layer. Hyperparameters of this model are listed in 9.

	Parameter
Training epochs	50
Layers number	1
Hidden layer dim	1000
Loss function	Mean Squared Error
Optimizer	Adam
Learning rate	0.001

Table 9: MLP hyperparameters

Both Chemical VAE and Grammar VAE are based on variational autoencoder model with stacked convolution layers in its encoder part and stacked GRU layers in decoder part. What differs them is the way that SMILES is encoded to one hot vector. Chemical VAE encodes each character of SMILES to separate one-hot vector, while Grammar VAE forms a parse tree from SMILES and encodes the parse rules. Details for CVAE are listed in 10 and for GVAE in 11.

	Parameter
MLP learning rate	0.05
MLP descent iterations	50
Fine-tuning batch size	256
Fine-tuning epochs	5
Latent space dim	196
Encoder convolution layers number	4
Decoder GRU layers number	4

Table 10: Chemical VAE hyperparameters

	Parameter
MLP learning rate	0.01
MLP descent iterations	50
Fine-tuning batch size	256
Fine-tuning epochs	5
Latent space dim	56
Encoder convolution layers number	3
Decoder GRU layers number	3

Table 11: Grammar VAE hyperparameters