Progressive Docking - Deep Learning Based Approach for Accelerated Virtual Screening

Vibudh Agrawal^{1,2} , Francesco Gentile¹ , Michael Hsing¹ ,
Fuqiang Ban¹ , and Artem Cherkasov¹

¹ Vancouver Prostate Centre, Department of Urologic Sciences,
Faculty of Medicine, University of British Columbia, 2660 Oak Street,
Vancouver, BC V6H 3Z6, Canada
{vagrawal, fgentile, mhsing, fban,
acherkasov}@prostatecentre.com

² The Bioinformatics Graduate Program, University of British Columbia,
Vancouver, Canada

Abstract. We have developed a novel, hybrid QSAR-docking approach (called ‘progressive docking’) that can speed up the process of virtual screening by enhancing it with Deep Learning models trained on-the-go on produced docking scores. The developed method can, therefore, predict docking outcome for yet unprocessed molecular entries and hence to progressively remove unfavorable chemical structures from the remaining docking base. This approach provides 50–100X speed increase for the standard docking procedures while retaining >90% of qualified molecules. We demonstrate that the use of PD allows processing of about 360 million molecules just in 2 weeks using a standard 200 CPU setup.

Keywords: Computer aided drug discovery · QSAR · Progressive docking · Deep Learning · Artificial intelligence · Cheminformatics

Drug screening is an extensive and expensive process [1], where computational (docking) models can significantly increase both the speed and hit rate of the discovery [2]. On the other hand, even the most advanced and elaborate docking algorithms [3–5], coupled with significant computational resources, are not fast enough to fully scan currently available molecular database (such as ZINC [6]), which has reached billions of entries in size.

The current solution to the problem is to apply filters based on simplistic physicochemical properties such as molecular weight and number of H-bond donors/acceptors [7], and various drug-likeness filters to reduce the size of the docking base to typically 10 s of millions of entries. With such a reduction, the vast majority of available chemical entities in sources like ZINC remain unprocessed.

In this work, we propose a new pipeline (Fig. 1) integrated with the machine learning framework which can speed the process of virtual screening (molecular docking) of the entire ZINC database by above 50X while retaining >90% of qualified molecules and using a fairly modest 200 CPUs setup (Fig. 2). The different statistics (AUC, enrichment, precision, recall) for every model at each iteration is shown in Table 2. Our method is based on the idea of ‘progressive docking’ pipeline which

Fig. 1. Flowchart of progressive docking. Features are extracted for all the molecules from the small molecule database and are clustered. About 1 million molecules are then sampled from this clustered dataset. These 1 million molecules are docked and a cut-off value is chosen to convert this problem of regression to classification. A QSAR model (DNN) is built and is used to predict the classes for all the molecules in the clustered database. All the molecules classified as ‘bad’ are rejected. This process is repeated multiple times to bring down the total molecules in the database to a manageable size.

utilizes docking scores generated by the slower but more accurate docking method (such as GLIDE [3]) on a small subset of molecules, to quickly predict the docking score of the remaining undocked molecules in the database [8]. For our machine learning model, we use DNN (Deep Neural Networks) with the number of hidden layers ranging from 1–4 (hyperparameter) and same number of neurons (hyperparameter) in each of the layers. Since the dataset is very unbalanced, we also use oversampling along with class weights (both oversampling ratios, as well as weights, are hyperparameters). Each model is trained on a single Tesla P100 NVIDIA GPU. We used Morgan fingerprints with radius 2 and 1024 bits for all the molecules.

Docking and model training are two major time-consuming steps in our method. Out of the two, docking is the rate determining step for our case (docking one million takes ~4 days while hyperparameter search takes <1 day). Hence, we can conclude that a more accurate model will also result in a faster progressive docking implementation (since fewer molecules will be left in the end for docking). We further compared the performance of Deep Neural Networks (DNN) model with other machine learning approaches like Random Forest and Logistic Regression to demonstrate that the use of DNN provides the most accurate results (Table 1) and hence a faster PD implementation. All generated Enrichment Factor (EF) and AUC_ROC values are significant and consistent.

Fig. 2. Flowchart of progressive docking for ERG protein. The time required to dock 360 M molecules without PD is about ~2years based on available resources. After each iteration, the size of the remaining dataset decreases with only losing 10% good molecules (not compounded after each iteration). The definition of a good molecule gets better after each iteration (cutoff). Enrichment of good molecules compared to random also increases exponentially. The overall time for PD to dock 360 M million molecules is about 2–3 weeks

Table 1. Performance of different QSAR models on the same dataset. A decrease in precision value, as well as AUC_ROC, is observed from DNN to logistic regression. DNN is trained on 1 GPU while random forest and logistic regression are trained on 1 CPU. Training time is very quick for logistic regression while almost similar for DNN and random forest. Predicting time increases from logistic regression to random forest.

Model	Enrichment factor	AUC_ROC	CPU/GPU	Training time	Predicting time
DNN	2.5	0.88	GPU	30 min	0.5 min
Random forest	2	0.85	CPU	40 min	4 min
Logistic regression	1.88	0.82	CPU	1 min	1 s

Table 2. Performance statistics of different QSAR models (DNN) at each iteration steps. The model is selected to have a recall of 90% (capturing 90% of good molecules). The enrichment factor, as well as AUC, gets better after each iteration.

Iteration	Enrichment factor	AUC_ROC	Docking score cut-off
1	6	0.81	−6.0
2	18	0.90	−6.5
3	90	0.91	−7.0

We anticipate that the progressive docking framework can be integrated with any choice of docking software that the user trusts and can be used with any machine learning algorithm (we recommend random forest if no GPUs are available or else DNN).

References

1. DiMasi, J.A., Grabowski, H.G., Hansen, R.W.: Innovation in the pharmaceutical industry: new estimates of R&D costs. *J Health Econ.* **47**, 20–33 (2016). <https://doi.org/10.1016/j.jhealeco.2016.01.012>
2. Ban, F., Dalal, K., Li, H., LeBlanc, E., Rennie, P.S., Cherkasov, A.: Best practices of computer-aided drug discovery: lessons learned from the development of a preclinical candidate for prostate cancer with a new mechanism of action. *J. Chem. Inf. Model.* **57**(5), 1018–1028 (2017). <https://doi.org/10.1021/acs.jcim.7b00137>
3. Friesner, R.A., Banks, J.L., Murphy, R.B., et al.: Glide: a new approach for rapid, accurate docking and scoring. 1. method and assessment of docking accuracy. *J. Med. Chem.* **47**(7), 1739–1749 (2004). <https://doi.org/10.1021/jm0306430>
4. McGann, M.: FRED and HYBRID docking performance on standardized datasets. *J. Comput. Aided Mol. Des.* **26**(8), 897–906 (2012). <https://doi.org/10.1007/s10822-012-9584-8>
5. Abagyan, R., Totrov, M., Kuznetsov, D.: ICM—a new method for protein modeling and design: Applications to docking and structure prediction from the distorted native conformation. *J. Comput. Chem.* **15**(5), 488–506 (1994). <https://doi.org/10.1002/jcc.540150503>
6. Sterling, T., Irwin, J.J.: ZINC 15—ligand discovery for everyone. *J. Chem. Inf. Model.* **55**(11), 2324–2337 (2015). <https://doi.org/10.1021/acs.jcim.5b00559>
7. Lipinski, C.A.: Lead- and drug-like compounds: the rule-of-five revolution. *Drug Discov. Today Technol.* **1**(4), 337–341 (2004). <https://doi.org/10.1016/j.ddtec.2004.11.007>
8. Cherkasov, A., Ban, F., Li, Y., Fallahi, M., Hammond, G.L.: Progressive docking: a hybrid QSAR/docking approach for accelerating in silico high throughput screening. *J. Med. Chem.* **49**(25), 7466–7478 (2006). <https://doi.org/10.1021/jm060961+>

Open Access This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.