Large-scale Pretraining Improves Sample Efficiency of Active Learning based Molecule Virtual Screening

Zhonglin Cao, Simone Sciabola, Ye Wang
Medicinal Chemistry
Biogen
Cambridge, MA 02142
{zhonglin.cao, simone.sciabola, ye.wang}@biogen.com

Abstract

Virtual screening of large compound libraries to identify potential hit candidates is one of the earliest steps in drug discovery. As the size of commercially available compound collections grows exponentially to the scale of billions, brute-force virtual screening using traditional tools such as docking becomes infeasible in terms of time and computational resources. Active learning and Bayesian optimization has recently been proven as effective methods of narrowing down the search space. An essential component in those methods is a surrogate machine learning model that is trained with a small subset of the library to predict the desired properties of compounds. Accurate model can achieve high sample efficiency by finding the most promising compounds with only a fraction of the whole library being virtually screened. In this study, we examined the performance of pretrained transformer-based language model and graph neural network in Bayesian optimization active learning framework. The best pretrained models identifies 58.97% of the top-50000 by docking score after screening only 0.6% of an ultra-large library containing 99.5 million compounds, improving 8% over previous state-of-the-art baseline. Through extensive benchmarks, we show that the superior performance of pretrained models persists in both structure-based and ligand-based drug discovery. Such model can serve as a boost to the accuracy and sample efficiency of active learning based molecule virtual screening.

1 Introduction

As an important step in early-stage drug discovery, structure-based virtual screening methods, such as molecular docking[1–3] and molecular dynamics simulation[4, 5], predict the conformation and pose of a ligand in target protein binding pockets and formulate a quantitative measurement of binding affinity[1]. Among all structure-based virtual screening methods, molecular docking estimate the protein-ligand binding affinity by optimizing a parameterized scoring function[6–8]. Molecular docking is one of the most popular choices thanks to its relatively lower computational cost[2, 9, 10] (usually a few CPU seconds for each ligand[11]) and acceptable accuracy. Many successes are witnessed in identifying potential drug candidates[12–15] using molecular docking.

In the past decade, the size of synthesizable on-demand compound libraries has grown exponentially. For example, the size of ZINC database has increased from 120 million molecules in 2015[16] to more than 1 billion molecules in 2020[17]. Commercially available Enamine *REAL* database[18] now includes 6 billion synthetically feasible compound selected from a larger Enamine *REAL* Space containing 36 billion compounds[19]. As the the number of organic molecules with 30 heavy atoms is estimated to be 10^{60}[20], the ultralarge compound libraries can be expected to expand in the future.

The NeurIPS 2023 Workshop on New Frontiers of AI for Drug Discovery and Development (AI4D3 2023), New Orleans, LA, USA, 2023.

Although the sheer size of those libraries allows the possibility of identifying new and non-proprietary drug candidates, it brings challenges to the virtual screening campaign. Exhaustive virtual screening using molecular docking on the ultralarge libraries can be a Herculean task or even infeasible due to required computational resources and time[21–24]. Therefore, docking strategies that can accelerate screening of ultralarge libraries or reduce the screening space without compromising hit recovery rate will be beneficial to the drug discovery industry.

The rapid advances of machine learning provide researchers a powerful tool to accelerate docking-based virtual screening. Machine learning models are trained as a faster surrogate of molecular docking to either predict the docking score[25] or classify if a compound is a hit[26, 27]. In the application of machine learning to molecule virtual screening, the accuracy of surrogate model can directly influence the hit recovery rate. The data used for the training of machine learning model is obtained through docking a subset of the library. Larger training set generally leads to higher prediction accuracy but docking more compounds increases the computation burden. Optimizing the trade-off between computation cost and model accuracy is a problem within the scope of active learning. Active learning is a subfield of machine learning that aims to train accurate model using minimum amount of data by allowing the model to actively sample unlabelled data to be added into the training set[28, 29]. Many previous works have shown the applicability of active learning in sample efficient molecule virtual screening. Yang et al.[11] demonstrate that active learning with graph convolutional neural network can recover $> 90\%$ by docking only 5% of the library. Deep Docking proposed by Gentile et al.[26, 27] enables the virtual screening on 1.36 billion compounds by 100-fold data reduction. Graff et al.[30] develop a framework (MolPAL) based on batched Bayesian optimization[31, 32], which formulate a strong synergy with pool-based active learning[33, 29], to successfully recover 94.8% of the top-50000 compounds from a 99.5 million sized library. Such a framework is shown to be effective in noisy environment[34] (a common problem with docking data), and a pruning algorithm is developed to further improve efficiency by reducing the screening space[35].

The choice of surrogate machine learning model in the active learning framework is one of the dominant factors that affect the hit recovery rate. A more accurate surrogate model can achieve high sample efficiency by allowing the active learning virtual screening to identify more hit candidates with less docking necessary. Self-supervised pretraining is a well-established technique to improve the models' performance in downstream tasks like regression. The goal of self-supervised pretraining to enable model to learn a better representation of data using the large quantity of unlabeled data. With the growth of molecular data, molecular representation learning through self-supervision has attracted academic attention[36]. Language models such as transformer pretrained using mask-language-modeling on large scale SMILES data[37–39] show promising accuracy in molecular property prediction. Graph neural network (GNN) can also be pretrained in a contrastive learning manner[40] using molecular graph augmentation[41, 42]. Moreover, joint pretraining across graph/text or 2D/3D data has been proven to be beneficial to models in molecular or material property prediction[43–45]. The success of pretrained models in molecular representation learning inspires us to leverage them for more sample-efficient virtual screening. In this work, we benchmark two pretrained models including molecular language transformer[39] (MoLFormer) and molecular contrastive learning pretrained graph isomorphism network[41, 46] (MolCLR) in the MolPAL framework for molecule virtual screening. We demonstrate that pretrained models can achieve consistently higher hit recovery rate and enrichment factor on 50 thousand to 99.5 million compound datasets compared with a strong baseline, directed message passing neural network[47] (D-MPNN). We also compare the models' performance with different acquisition functions, including greedy and upper confidence boundary (UCB), and study the effect of uncertainty level on hit recovery rate and chemical diversity of identified compounds. At last, we extend the applicability of MolPAL with pretrained surrogate models to ligand-based drug discovery by virtual screening large library based on the 3D similarity to known active compounds. Pretrained deep learning models are shown to be superior surrogate model choices in both structure-based and ligand-based active learning virtual screening.

2 Results and discussion

The active learning is conducted using the batched Bayesian optimization framework, MolPAL, implemented by Graff et al[30] (Figure 1a, details in the Appendix). MolPAL consists of three important components including a surrogate model, acquisition function, and an objective function.

The objective function can be a docking protocol in structure-based drug discovery. During the virtual screening of a large pool of molecules for potential hit to a protein target, a small batch of molecules are randomly selected and docked. The docking scores are used to train the surrogate model in a supervised manner such that the surrogate model can be used to predict the docking scores of all other molecules in the pool. The acquisition function evaluates the molecule pool based on the prediction from surrogate model and further selects another batch of molecules to augment the surrogate model training dataset. The above iterative process repeats until a user-defined stopping criterion, which is a fixed number of iterations in our case, is satisfied. We conduct retrospective studies on all datasets, meaning that the docking score of all molecules are pre-calculated and an oracle lookup function is used to retrieve the docking score during the active learning process instead of performing docking on-the-fly. Two acquisition functions tested in this work are the Greedy (Eq.1) and upper confidence boundary (UCB, Eq. 2). Details of all surrogate models and the training process are included in the Appendix. Pretraining strategies of MoLFormer and MolCLR are visualized in Figure 1b and 1c. The metrics we use to evaluate the performance of the Bayesian optimization active learning is the percentage of top-k molecules retrieved (identified through SMILES) or top-k retrieval rate and the enrichment factor (EF). The EF is defined as the percentage of top-k molecules retrieved by active learning over the percentage of top-k molecules retrieved by random selection.

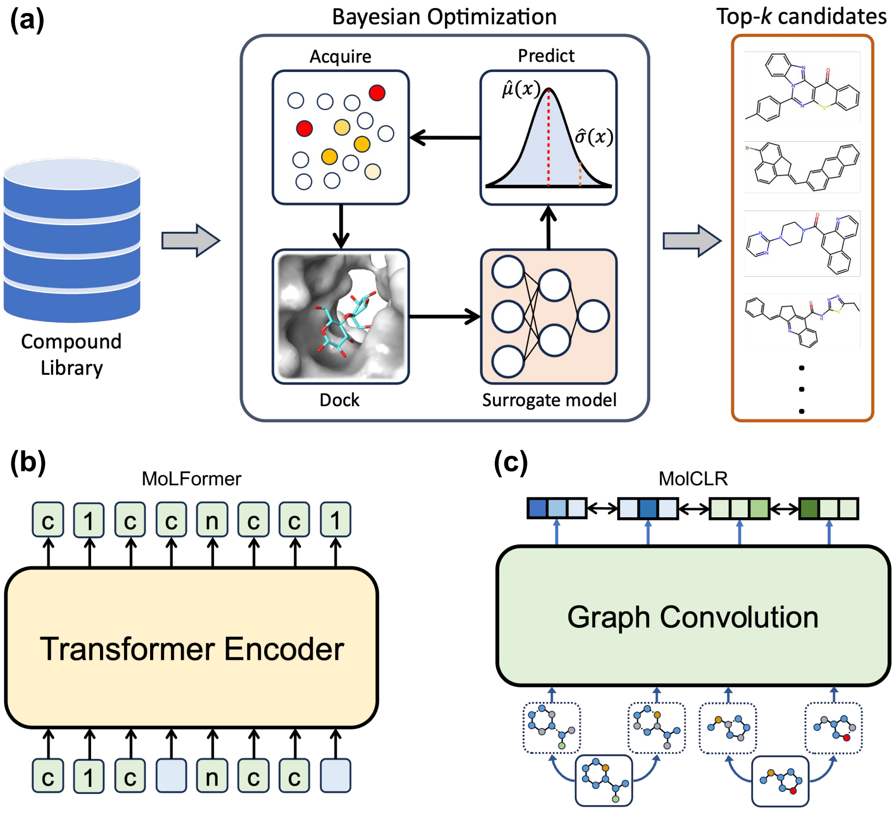

Figure 1: (a) The Bayesian optimization based active learning framework. For ligand-based virtual screening, the docking can be substitute by ligand similarity calculation such as Rapid Overlay of Chemical Structures (ROCS). (b) Schematic of mask-language-modeling pretraining process of the MoLFormer model. Green pads are tokenized SMILES, and blue pads represent masked tokens. MoLFormer is trained to predict masked tokens. (c) Schematic of the molecular contrastive learning pretraining process of the MolCLR model. Solid boxes contain original molecular graphs and dashed boxes contains augmented graphs. Grey circles represent masked nodes and dashed edges represent deleted bond during the augmentation process. Contrastive loss is applied on representations of augmented molecular graphs.

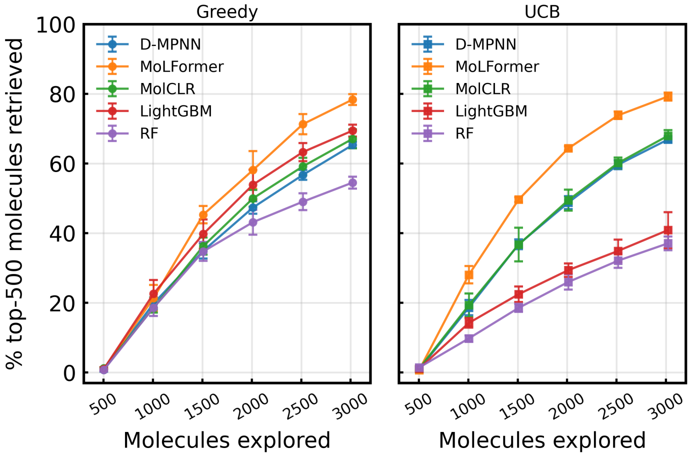

Figure 2: Percentage of the top-500 (top-1%) molecules in the Enamine 50K dataset retrieved after 5 iterations of Bayesian optimization using different surrogate models. The initial training set of the surrogate model contains 1% of randomly selected molecules. Each following acquisition selects extra 1% molecules according to the Greedy (left) or the UCB (right) strategy. Each data point is an average of 5 runs with different initial seeds and error bars represent one standard deviation.

2.1 Enamine libraries

Two Enamine compound libraries are used for benchmark in this work. The smaller one is the Enamine Discovery Diversity Set which contains 50,240 compounds (Enamine 50k). The larger one is the Enamine HTS collection (Enamine HTS) containing 2,141,514 molecules. The Enamine 50K and HTS datasets are publicly available on the code repository of MolPAL[30]. Molecules in both of the Enamine datasets are docked using AutoDock Vina[7] against thymidylate kinase (PDB ID: 4UNN)[48]. The docking procedure is detailed in ref[30]. When the models are evaluated on the Enamine 50k, 1% of the dataset is randomly selected for the initial model training. Following the initialization, 5 iterations of 1% batch acquisition are done resulting in 6% of the dataset being explored. MoLFormer retrieves 78.36% of the top-500 molecules after 5 iterations of acquisition, followed by LightGBM (69.44%), MolCLR (67.08%), D-MPNN (65.32%), and RF (54.52%), using the Greedy strategy (Figure 2 left). UCB strategy slightly improves the top-500 retrieval rate of deep learning models, MoLFormer, MolCLR, and D-MPNN to 79.24%, 67.96%, and 66.88%, respectively. However, the UCB negatively impacts LightGBM and RF and causes their top-500 retrieval rate drop to 40.88% and 37.08%, respectively (Figure 2 right). Calculated based on the higher retrieval rate of either Greedy or UCB acquisition strategy, the EF of MoLFormer is 13.2, exceeding MolCLR (11.33), D-MPNN (11.15), LightGBM (11.57), and RF (9.09). The difference in the performance of GNN models, MolCLR and D-MPNN, is very small on the Enamine 50k dataset. LightGBM when used with the Greedy strategy is the second only to the MoLFormer. Considering its low computational cost, LightGBM is a promising surrogate model for screening small libraries.

The same five surrogate models are further benchmarked on the larger Enamine HTS set (Figure 3) after their high EF is validated on the smaller Enamine 50k set. Given that the ultimate goal of active learning is to retrieve top performing compounds by minimum amount of evaluation, reduced batch size (0.2% or 0.1%) is used for the initial selection and iterative acquisition for the Enamine HTS set. Using the Greedy strategy and 0.2% of acquisition batch size, MoLFormer retrieves 92.24% of the top-1000 molecules by exploring only 1.2% of the whole dataset, exceeding all other models. With the increase size of dataset, D-MPNN (89.88%) and MolCLR (86.98%) consistently outperform LightGBM (82.5%) and RF (68.18%). Deep learning models are generally better surrogate model choices when virtual screening large libraries. Smaller acquisition batch size improves the top-k retrieval rate when the total number of explored compound remains the same. For example, after exploring 0.6% of the dataset (12,855 molecules) using the Greedy strategy, MoLFormer retrieves 81.58% of the top-1000 compounds with the 0.1% batch size, higher than 75.9% with the 0.2% batch size. Smaller acquisition batch size also minimizes the performance gap between the D-MPNN and MolCLR. With Greedy strategy and 0.1% batch size, MolCLR retrieves 72.78% of the top-1000

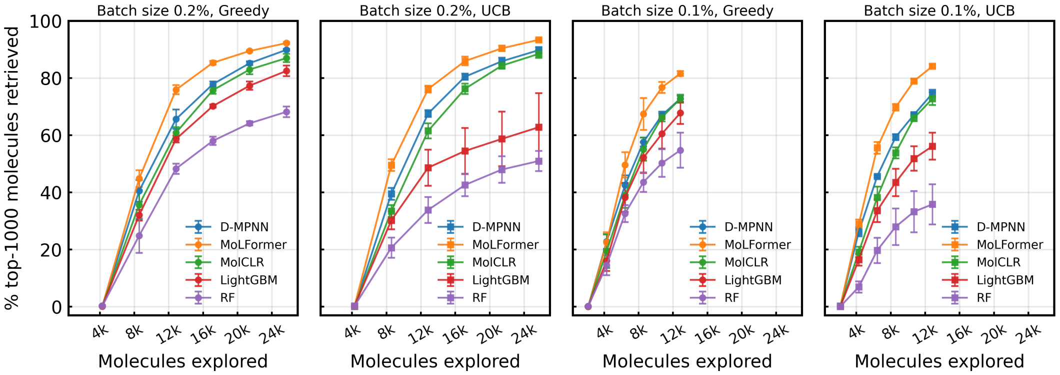

Figure 3: Percentage of the top-1000 (top-0.05%) molecules in the Enamine 50K dataset retrieved after 5 iterations of Bayesian optimization using different surrogate models. The initial training set of the surrogate model has the same size of the molecule batch acquired in each following iteration. The results of 0.2% batch size (left two panels) and 0.1% batch size (right two panels) are shown. Each data point is an average of 5 runs with different initial seeds and error bars represent one standard deviation.

compounds after exploring 0.6% of the dataset, which is slightly higher than 72.76% of the D-MPNN. When the batch size is 0.2%, the top-1000 retrieval rate of D-MPNN (65.62%) is noticeably higher than that of MolCLR (60.86%) after exploring 0.6% of the dataset. UCB strategy consistently improve the top-1000 retrieval rate of MoLFormer by 1-2.5% regardless of the batch size on the Enamine HTS set. Similar to the results on the Enamine 50k set, top-1000 retrieval rates of LightGBM and RF drastically decrease by 11.6-19.7% using the UCB instead of the Greedy strategy. However, neither Greedy nor UCB strategy is conclusively better than the other one based on the top-1000 retrieval rate comparison between D-MPNN and MolCLR. Calculated based on the higher retrieval rate of either Greedy or UCB acquisition strategy with 0.1% batch size, the EF of MoLFormer is 140.2, higher than D-MPNN (124.37), MolCLR (121.3), LightGBM (112.97), and RF (91.23).

2.2 Effect of uncertainty weight in UCB acquisition

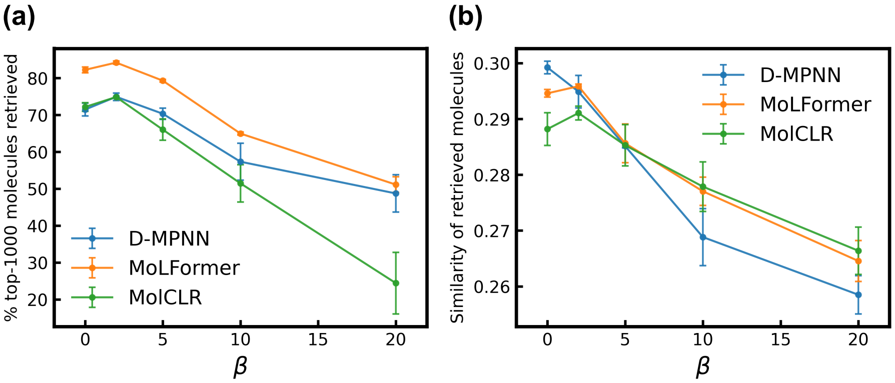

Figure 4: The effect of increasing uncertainty weight (β) in the UCB acquisition function. (a) Top-1000 molecule retrieval rate of different surrogate models with increasing β (b) Average Dice similarity of molecules retrieved by different surrogate models with increasing β. The results shown are from exploring 0.6% of the EnamineHTS set with 0.1% initialization and acquisition batch size. Each point is averaged value over three runs with different initial seed and error bar represents one standard deviation.

In the UCB acquisition function (Eq. 2), the weight of uncertainty in acquiring new samples is regulated by a hyperparameter β. To further understand the effect of β on the top-*k* molecule retrieval rate and the diversity of retrieved molecules, we run the BO active learning with β value enumerating from $0, 2, 5, 10, 20$ on the Enamine HTS set. The results (Figure. 4) are obtained using the 0.1%

batch size for initialization and iterative acquisition (0.6% of total dataset explored at the end). When $\beta = 0$, Greedy acquisition function is used. It is noticeable that the top-1000 retrieval rate peaks (Figure. 4a) for all models when $\beta = 2$, which is the default value of β, indicating that adding a term of uncertainty with small weight during acquisition can benefit the performance of the active learning. However, the top-1000 retrieval rate drops with greater β value. For MoLFormer and D-MPNN, the retrieval rate converges to around 50%, while the retrieval rate of MolCLR has negatively linear relationship with β and drops to 24.4% when $\beta = 20$ without converging. To quantify the diversity of retrieved molecules, we compute the pairwise Dice similarity of molecules based on their Morgan Fingerprint with radius of 3. The average similarity between retrieved molecules drops with greater β value, reflecting that the models tend to acquire more diverse molecules when uncertainty weight is higher in the acquisition function. If exploring a more diverse set of compound is desired during the active learning based virtual screening, tuning up the β value is an effective option.

2.3 Hundred-million scale libraries

Industrial level virtual screening is usually conducted on ultra-large libraries containing hundred-millions or billions of compounds. To evaluate the performance of MolPAL with pretrained models on ultra-large library, we run the previous benchmarks on the dataset curated by Lyu et al.[23]. The dataset contains 99.5 million compounds docked against the AmpC β-lactamase (the AmpC dataset in short, PDB ID: 1L2S) using Dock3.7[8]. Since the deep learning surrogate models including MoLFormer, D-MPNN, and MolCLR significantly outperforms LightGBM and RF on large dataset like the Enamine HTS, only them are selected to be benchmarked on the AmpC dataset. The initialization and acquisition batch size is set to be 0.1% of the whole dataset as we want to evaluate the models' performance given minimum amount of docking calculation. The ranking of top-50000 retrieval rate using the Greedy strategy is MolCLR (58.965%) > MoLFormer (55.497%) > D-MPNN (50.021%). The UCB strategy reduces the top-50000 retrieval rate of MolCLR and MoLFormer to 55.659% and 54.633%, respectively, but improves that of the D-MPNN to 55.03%. Among the three models (Table S4), MolCLR has the highest EF of 98.28, which is 6.2% higher than MoLFormer (92.49) and 7.2% higher than D-MPNN (91.72). By benchmarking on three datasets with various sizes, we demonstrate that pretrained deep learning surrogate models are more sample-efficient choices to be included in the Bayesian optimization active learning framework.

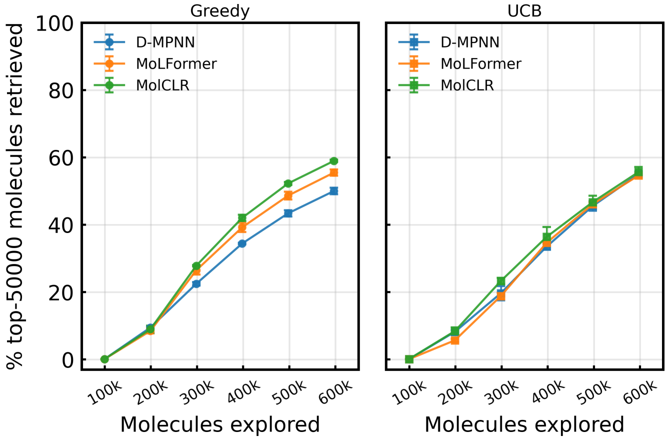

Figure 5: Percentage of the top-50000 (top-0.05%) molecules in the AmpC dataset retrieved after 5 iterations of Bayesian optimization using different surrogate models. The initial training set of the surrogate model contains 0.1% of randomly selected molecules. Each following acquisition selects extra 0.1% molecules according to the Greedy (left) or the UCB (right) strategy. Each data point is an average of 3 runs with different initial seeds and error bars represent one standard deviation.

2.4 Extending the application to ligand-based virtual screening

Ligand-based drug design is another important domain in the drug discovery industry. The fundamental philosophy of ligand-based drug design is that compounds similar to a known, active compound are more likely to be active. Rapid Overlay of Chemical Structures (ROCS) by OpenEye Inc. is a established tool to calculate the 3D shape similarity of compounds. To extend the applicability of the MolPAL to ligand-based virtual screening, we compute the ROCS similarity score of all compounds in the Enamine HTS set to the original ligand in the thymidylate kinase complex[48]. MolPAL with different surrogate models is then benchmarked on the EnamineHTS ROCS dataset for the top-1000 molecules retrieval rate. The top-1000 molecules of the EnamineHTS ROCS dataset have ROCS score of at least 1.1771. 0.1% batch size is used for the initialization and iterative acquisition. With the Greedy strategy (Figure 6 left), MoLFormer has the highest top-1000 retrieval rate (70.54%), followed by MolCLR (66.30%). The pretrained models have significantly higher top-1000 retrieval rate than LightGBM (51.88%), D-MPNN (47.40%), and RF (17.68%). On the EnamineHTS ROCS dataset, the standard deviation of top-1000 retrieval rate across 5 runs is very large for all models except MoLFormer when Greedy strategy is used. Specifically, the retrieval rate standard deviation at the 5th iteration is higher than 8% for both MolCLR and D-MPNN, indicating unstable performance. The UCB strategy effectively alleviates this problem for deep learning models (Figure 6 right). With UCB strategy, the average top-1000 retrieval rate of MoLFormer, MolCLR, and D-MPNN is improved to 75.78%, 70.52%, and 64.5%, respectively. The retrieval rate standard deviation after the last iteration is also reduced to 0.66% (MoLFormer), 2.04% (MolCLR), and 4.24% (D-MPNN) when following the UCB strategy. Given that UCB strategy can lead to higher and more consistent top-1000 retrieval rate for the deep learning models, it is the better choice when MolPAL is applied on ligand-based virtual screening with ROCS score.

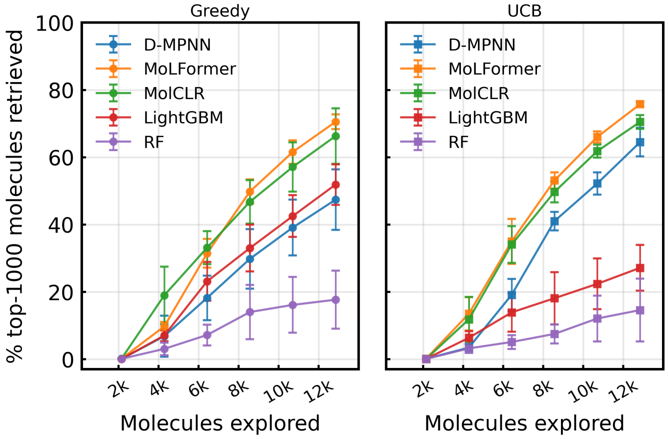

Figure 6: Percentage of the top-1000 (top-0.05%) molecules in the EnamineHTS ROCS dataset retrieved after 5 iterations of Bayesian optimization using different surrogate models. The initial training set of the surrogate model contains 0.1% of randomly selected molecules. Each following acquisition selects extra 0.1% molecules according to the Greedy (left) or the UCB (right) strategy. Each data point is an average of 5 runs with different initial seeds and error bars represent one standard deviation.

3 Conclusion

In this work, we demonstrate that Transformer-based language models (MoLFormer) and graph neural network (MolCLR) after large-scale pretraining can serve as more sample-efficient surrogate model in the Bayesian optimization active learning framework for molecule virtual screening. On Enamine libraries with size ranging from 50 thousand to 2.1 million compounds, MoLFormer has consistently higher top-k docking score molecules retrieval rate than the previous state-of-the-art D-MPNN model. On the AmpC dataset containing 99.5 million compounds docked against AmpC β-lactamase, MoLFormer and MolCLR retrieves 5.5% and 8.9% more top-50000 compounds than D-MPNN, respectively, after exploring the same 0.6% of the whole dataset using Greedy strategy. We demonstrate that smaller acquisition batch size can improve the top-k molecules retrieval rate. Greedy acquisition strategy is effective in most cases, rendering it a suitable default option. The effect of UCB strategy varies on different surrogate models and datasets. The larger uncertainty weight in the UCB acquisition function is shown to increase the diversity of the acquired molecules at the cost of reduced top-k retrieval rate. At last, the application of MolPAL is extended to ligand-based virtual screening. We curate a dataset containing the compounds in the EnamineHTS dataset and their ROCS score to the ligand in the 4UNN complex. With only 0.6% of the EnamineHTS ROCS dataset explored, MolFormer and MolCLR again outperforms D-MPNN by large margins in retrieving the top-1000 similar molecules. UCB strategy is shown to be a better option on the ligand-based virtual screening because it delivers higher and more consistent top-k retrieval rate. In conclusion, deep learning models after pretraining on large-scale chemical space, such as MoLFormer, should be used as surrogate models in active learning molecule virtual screening because of their higher sample-efficiency. The application of pretrained models can benefit both the structure-based and ligand-based domains in the field of drug discovery.

References

[1] Jin Li, Ailing Fu, and Le Zhang. An overview of scoring functions used for protein–ligand interactions in molecular docking. *Interdisciplinary Sciences: Computational Life Sciences*, 11:320–328, 2019.

[2] John J Irwin and Brian K Shoichet. Docking screens for novel ligands conferring new biology: Miniperspective. *Journal of medicinal chemistry*, 59(9):4103–4120, 2016.

[3] Elizabeth Yuriev, Jessica Holien, and Paul A Ramsland. Improvements, trends, and new ideas in molecular docking: 2012–2013 in review. *Journal of Molecular Recognition*, 28(10):581–604, 2015.

[4] Ron O Dror, Albert C Pan, Daniel H Arlow, David W Borhani, Paul Maragakis, Yibing Shan, Huafeng Xu, and David E Shaw. Pathway and mechanism of drug binding to g-protein-coupled receptors. *Proceedings of the National Academy of Sciences*, 108(32):13118–13123, 2011.

[5] Ignasi Buch, Toni Giorgino, and Gianni De Fabritiis. Complete reconstruction of an enzyme-inhibitor binding process by molecular dynamics simulations. *Proceedings of the National Academy of Sciences*, 108(25):10184–10189, 2011.

[6] Richard A Friesner, Jay L Banks, Robert B Murphy, Thomas A Halgren, Jasna J Klicic, Daniel T Mainz, Matthew P Repasky, Eric H Knoll, Mee Shelley, Jason K Perry, et al. Glide: a new approach for rapid, accurate docking and scoring. 1. method and assessment of docking accuracy. *Journal of medicinal chemistry*, 47(7):1739–1749, 2004.

[7] 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.

[8] Todd JA Ewing, Shingo Makino, A Geoffrey Skillman, and Irwin D Kuntz. Dock 4.0: search strategies for automated molecular docking of flexible molecule databases. *Journal of computer-aided molecular design*, 15:411–428, 2001.

[9] David E. Graff and Connor W. Coley. pyscreener: A python wrapper for computational docking software. *Journal of Open Source Software*, 7(71):3950, 2022.

[10] Christoph Gorgulla, Andras Boeszoermenyi, Zi-Fu Wang, Patrick D Fischer, Paul W Coote, Krishna M Padmanabha Das, Yehor S Malets, Dmytro S Radchenko, Yurii S Moroz, David A Scott, et al. An open-source drug discovery platform enables ultra-large virtual screens. *Nature*, 580(7805):663–668, 2020.

[11] Ying Yang, Kun Yao, Matthew P Repasky, Karl Leswing, Robert Abel, Brian K Shoichet, and Steven V Jerome. Efficient exploration of chemical space with docking and deep learning. *Journal of Chemical Theory and Computation*, 17(11):7106–7119, 2021.

[12] Lars Richter, Chris De Graaf, Werner Sieghart, Zdravko Varagic, Martina Mörzinger, Iwan JP De Esch, Gerhard F Ecker, and Margot Ernst. Diazepam-bound gabaa receptor models identify new benzodiazepine binding-site ligands. *Nature chemical biology*, 8(5):455–464, 2012.

[13] Jun Min, Da Lin, Qingye Zhang, Jibing Zhang, and Ziniu Yu. Structure-based virtual screening of novel inhibitors of the uridyltransferase activity of xanthomonas oryzae pv. oryzae glmu. *European journal of medicinal chemistry*, 53:150–158, 2012.

[14] Yu Chen and Brian K Shoichet. Molecular docking and ligand specificity in fragment-based inhibitor discovery. *Nature chemical biology*, 5(5):358–364, 2009.

[15] Denise G Teotico, Kerim Babaoglu, Gabriel J Rocklin, Rafaela S Ferreira, Anthony M Giannetti, and Brian K Shoichet. Docking for fragment inhibitors of ampc β-lactamase. *Proceedings of the National Academy of Sciences*, 106(18):7455–7460, 2009.

[16] Teague Sterling and John J Irwin. Zinc 15–ligand discovery for everyone. *Journal of chemical information and modeling*, 55(11):2324–2337, 2015.

[17] John J Irwin, Khanh G Tang, Jennifer Young, Chinzorig Dandarchuluun, Benjamin R Wong, Munkhzul Khurelbaatar, Yurii S Moroz, John Mayfield, and Roger A Sayle. Zinc20—a free ultralarge-scale chemical database for ligand discovery. *Journal of chemical information and modeling*, 60(12):6065–6073, 2020.

[18] Enamine real. `https://enamine.net/compound-collections/real-compounds/real-database`. Accessed: 2023-08-29.

[19] Louis Bellmann, Patrick Penner, Marcus Gastreich, and Matthias Rarey. Comparison of combinatorial fragment spaces and its application to ultralarge make-on-demand compound catalogs. *Journal of Chemical Information and Modeling*, 62(3):553–566, 2022.

[20] Regine S Bohacek, Colin McMartin, and Wayne C Guida. The art and practice of structure-based drug design: a molecular modeling perspective. *Medicinal research reviews*, 16(1):3–50, 1996.

[21] Christoph Grebner, Erik Malmerberg, Andrew Shewmaker, Jose Batista, Anthony Nicholls, and Jens Sadowski. Virtual screening in the cloud: how big is big enough? *Journal of Chemical Information and Modeling*, 60(9):4274–4282, 2019.

[22] Christoph Gorgulla, Krishna M Padmanabha Das, Kendra E Leigh, Marco Cespugli, Patrick D Fischer, Zi-Fu Wang, Guilhem Tesseyre, Shreya Pandita, Alec Shnapir, Anthony Calderaio, et al. A multi-pronged approach targeting sars-cov-2 proteins using ultra-large virtual screening. *Iscience*, 24(2), 2021.

[23] Jiankun Lyu, Sheng Wang, Trent E Balius, Isha Singh, Anat Levit, Yurii S Moroz, Matthew J O'Meara, Tao Che, Enkhjargal Algaa, Kateryna Tolmachova, et al. Ultra-large library docking for discovering new chemotypes. *Nature*, 566(7743):224–229, 2019.

[24] Arman A Sadybekov, Anastasiia V Sadybekov, Yongfeng Liu, Christos Iliopoulos-Tsoutsouvas, Xi-Ping Huang, Julie Pickett, Blake Houser, Nilkanth Patel, Ngan K Tran, Fei Tong, et al. Synthon-based ligand discovery in virtual libraries of over 11 billion compounds. *Nature*, 601(7893):452–459, 2022.

[25] Francois Berenger, Ashutosh Kumar, Kam YJ Zhang, and Yoshihiro Yamanishi. Lean-docking: exploiting ligands' predicted docking scores to accelerate molecular docking. *Journal of Chemical Information and Modeling*, 61(5):2341–2352, 2021.

[26] Francesco Gentile, Vibudh Agrawal, Michael Hsing, Anh-Tien Ton, Fuqiang Ban, Ulf Norinder, Martin E Gleave, and Artem Cherkasov. Deep docking: a deep learning platform for augmentation of structure based drug discovery. *ACS central science*, 6(6):939–949, 2020.

[27] Francesco Gentile, Jean Charle Yaacoub, James Gleave, Michael Fernandez, Anh-Tien Ton, Fuqiang Ban, Abraham Stern, and Artem Cherkasov. Artificial intelligence–enabled virtual screening of ultra-large chemical libraries with deep docking. *Nature Protocols*, 17(3):672–697, 2022.

[28] Burr Settles. Active learning literature survey. Computer Sciences Technical Report 1648, University of Wisconsin–Madison, 2009.

[29] Francesco Di Fiore, Michela Nardelli, and Laura Mainini. Active learning and bayesian optimization: a unified perspective to learn with a goal. *arXiv preprint arXiv:2303.01560*, 2023.

[30] David E Graff, Eugene I Shakhnovich, and Connor W Coley. Accelerating high-throughput virtual screening through molecular pool-based active learning. *Chemical science*, 12(22):7866–7881, 2021.

[31] Bobak Shahriari, Kevin Swersky, Ziyu Wang, Ryan P Adams, and Nando De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.

[32] Edward O Pyzer-Knapp. Bayesian optimization for accelerated drug discovery. *IBM Journal of Research and Development*, 62(6):2–1, 2018.

[33] Xueying Zhan, Huan Liu, Qing Li, and Antoni B Chan. A comparative survey: Benchmarking for pool-based active learning. In *IJCAI*, pages 4679–4686, 2021.

[34] Hugo Bellamy, Abbi Abdel Rehim, Oghenejokpeme I Orhobor, and Ross King. Batched bayesian optimization for drug design in noisy environments. *Journal of Chemical Information and Modeling*, 62(17):3970–3981, 2022.

[35] David E Graff, Matteo Aldeghi, Joseph A Morrone, Kirk E Jordan, Edward O Pyzer-Knapp, and Connor W Coley. Self-focusing virtual screening with active design space pruning. *Journal of Chemical Information and Modeling*, 62(16):3854–3862, 2022.

[36] Rayan Krishnan, Pranav Rajpurkar, and Eric J Topol. Self-supervised learning in medicine and healthcare. *Nature Biomedical Engineering*, 6(12):1346–1352, 2022.

[37] Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. Chemberta: large-scale self-supervised pretraining for molecular property prediction. *arXiv preprint arXiv:2010.09885*, 2020.

[38] Walid Ahmad, Elana Simon, Seyone Chithrananda, Gabriel Grand, and Bharath Ramsundar. Chemberta-2: Towards chemical foundation models. *arXiv preprint arXiv:2209.01712*, 2022.

[39] Jerret Ross, Brian Belgodere, Vijil Chenthamarakshan, Inkit Padhi, Youssef Mroueh, and Payel Das. Large-scale chemical language representations capture molecular structure and properties. *Nature Machine Intelligence*, 4(12):1256–1264, 2022.

[40] Yuning You, Tianlong Chen, Yongduo Sui, Ting Chen, Zhangyang Wang, and Yang Shen. Graph contrastive learning with augmentations. *Advances in neural information processing systems*, 33:5812–5823, 2020.

[41] Yuyang Wang, Jianren Wang, Zhonglin Cao, and Amir Barati Farimani. Molecular contrastive learning of representations via graph neural networks. *Nature Machine Intelligence*, 4(3):279–287, 2022.

[42] Yuyang Wang, Rishikesh Magar, Chen Liang, and Amir Barati Farimani. Improving molecular contrastive learning via faulty negative mitigation and decomposed fragment contrast. *Journal of Chemical Information and Modeling*, 62(11):2713–2725, 2022.

[43] Jinhua Zhu, Yingce Xia, Tao Qin, Wengang Zhou, Houqiang Li, and Tie-Yan Liu. Dual-view molecule pre-training. *arXiv preprint arXiv:2106.10234*, 2021.

[44] Hannes Stärk, Dominique Beaini, Gabriele Corso, Prudencio Tossou, Christian Dallago, Stephan Günnemann, and Pietro Liò. 3d infomax improves gnns for molecular property prediction. In *International Conference on Machine Learning*, pages 20479–20502. PMLR, 2022.

[45] Zhonglin Cao, Rishikesh Magar, Yuyang Wang, and Amir Barati Farimani. Moformer: self-supervised transformer model for metal–organic framework property prediction. *Journal of the American Chemical Society*, 145(5):2958–2967, 2023.

[46] Keyulu Xu, Weihua Hu, Jure Leskovec, and Stefanie Jegelka. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.

[47] Kevin Yang, Kyle Swanson, Wengong Jin, Connor Coley, Philipp Eiden, Hua Gao, Angel Guzman-Perez, Timothy Hopper, Brian Kelley, Miriam Mathea, et al. Analyzing learned molecular representations for property prediction. *Journal of chemical information and modeling*, 59(8):3370–3388, 2019.

[48] Maruti Naik, Anandkumar Raichurkar, Balachandra S Bandodkar, Begur V Varun, Shantika Bhat, Rajesh Kalkhambkar, Kannan Murugan, Rani Menon, Jyothi Bhat, Beena Paul, et al. Structure guided lead generation for m. tuberculosis thymidylate kinase (mtb tmk): discovery of 3-cyanopyridone and 1, 6-naphthyridin-2-one as potent inhibitors. *Journal of medicinal chemistry*, 58(2):753–766, 2015.

[49] Peter Auer. Using confidence bounds for exploitation-exploration trade-offs. *Journal of Machine Learning Research*, 3(Nov):397–422, 2002.

[50] Niranjan Srinivas, Andreas Krause, Sham M Kakade, and Matthias Seeger. Gaussian process optimization in the bandit setting: No regret and experimental design. *arXiv preprint arXiv:0912.3995*, 2009.

[51] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. Lightgbm: A highly efficient gradient boosting decision tree. *Advances in neural information processing systems*, 30, 2017.

[52] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[53] Raymond E Carhart, Dennis H Smith, and R Venkataraghavan. Atom pairs as molecular features in structure-activity studies: definition and applications. *Journal of Chemical Information and Computer Sciences*, 25(2):64–73, 1985.

[54] David A Nix and Andreas S Weigend. Estimating the mean and variance of the target probability distribution. In *Proceedings of 1994 ieee international conference on neural networks (ICNN'94)*, volume 1, pages 55–60. IEEE, 1994.

[55] Lior Hirschfeld, Kyle Swanson, Kevin Yang, Regina Barzilay, and Connor W Coley. Uncertainty quantification using neural networks for molecular property prediction. *Journal of Chemical Information and Modeling*, 60(8):3770–3780, 2020.

[56] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

[57] Philippe Schwaller, Teodoro Laino, Théophile Gaudin, Peter Bolgar, Christopher A Hunter, Costas Bekas, and Alpha A Lee. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS central science*, 5(9):1572–1583, 2019.

[58] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *International conference on machine learning*, pages 5156–5165. PMLR, 2020.

[59] Jianlin Su, Yu Lu, Shengfeng Pan, Ahmed Murtadha, Bo Wen, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *arXiv preprint arXiv:2104.09864*, 2021.

[60] John J Irwin and Brian K Shoichet. Zinc- a free database of commercially available compounds for virtual screening. *Journal of chemical information and modeling*, 45(1):177–182, 2005.

[61] Sunghwan Kim, Jie Chen, Tiejun Cheng, Asta Gindulyte, Jia He, Siqian He, Qingliang Li, Benjamin A Shoemaker, Paul A Thiessen, Bo Yu, et al. Pubchem 2019 update: improved access to chemical data. *Nucleic acids research*, 47(D1):D1102–D1109, 2019.

[62] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[63] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

[64] Dan Hendrycks and Kevin Gimpel. Gaussian error linear units (gelus). *arXiv preprint arXiv:1606.08415*, 2016.

[65] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[66] Yang You, Jing Li, Sashank Reddi, Jonathan Hseu, Sanjiv Kumar, Srinadh Bhojanapalli, Xiaodan Song, James Demmel, Kurt Keutzer, and Cho-Jui Hsieh. Large batch optimization for deep learning: Training bert in 76 minutes. *arXiv preprint arXiv:1904.00962*, 2019.

[67] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *IEEE transactions on neural networks*, 20(1):61–80, 2008.

[68] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[69] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[70] Greg Landrum et al. Rdkit: A software suite for cheminformatics, computational chemistry, and predictive modeling. *Greg Landrum*, 8:31, 2013.

[71] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

Appendix

3.1 Bayesian optimization

The active learning is conducted using the bacthed Bayesian optimization framework implemented by Graff et al.[30] called MolPAL. MolPAL consists of three important components including a surrogate model, acquisition function, and an objective function (e.g. docking or ROCS). When applied on virtual screening of a large library of molecules (denoted as a set $\mathcal{X}$), the goal of Bayesian optimization is to select a subset $\mathbf{x}$ that maximizes the objective $f(\mathbf{x})$, which can be the docking score or ROCS score. Due to the relatively high computational cost of objective function, a machine learning model parameterized by θ, $f_\theta(\cdot)$, is used as a faster surrogate to approximate the original objective function.

At the beginning of active learning, a batch of molecules $\mathbf{x}_0$ are randomly selected and their corresponding objective values $f(\mathbf{x}_0)$ are calculated. The surrogate model is trained using the initial labeled training set $\mathcal{D}_0 = \{\mathbf{x}_0, f(\mathbf{x}_0)\}$. Then, for n iterations, new batches of molecules, $\mathbf{x}_i$, will be selected by $\mathbf{x}_i = \underset{\mathbf{x}\in\mathcal{X}}{\operatorname{argmax}}\ \alpha(\mathbf{x}, f_\theta(\mathbf{x}))$, where $0 < i \leq n$ is the i-th iteration and $\alpha(\cdot)$ is the acquisition function. After each iteration of selection, the training set will be augmented as $\mathcal{D}_i = \{\mathcal{D}_{i-1}, (\mathbf{x}_i, f(\mathbf{x}_i))\}$ and the surrogate model will be retrained using $\mathcal{D}_i$. In this work, $n = 5$ is set for all benchmarks. The top-k molecules retrieval rate is calculated as the percentage of real top-k molecules being included in the $\mathcal{D}_n$. Retrospective studies are conducted on all datasets, meaning that docking score of all molecules are pre-calculated and $f(\mathrm{x})$ is a oracle lookup function that retrieves the docking or ROCS score during the active learning process.

3.2 Acquisition functions

Based on the benchmarks in the work of Graff et al.[30], we test two of the best performing acquisition functions including the greedy strategy:

$$\alpha_{\text{greedy}}(x) = \hat{\mu}(x) \tag{1}$$

and the upper confidence boundary (UCB) strategy[49, 50]:

$$\alpha_{\text{UCB}}(x) = \hat{\mu}(x) + \beta\hat{\sigma}(x) \tag{2}$$

where $\hat{\mu}(x)$ and $\hat{\sigma}(x)$ are the predicted docking or ROCS score and predicted uncertainty of a molecule x, respectively. Since new batches are selected by maximizing the acquisition function, docking scores are multiplied by -1. β, set toa default value of 2 in our work, is a hyperparameter that determines the weight of uncertainty when augmenting the surrogate model training data.

3.3 Surrogate models

Five surrogate models benchmarked in this work are Molecular Language Transformer[39] (MoLFormer), directed message passing neural network[47] (D-MPNN), graph isomorphism network[46] pretrained using molecular contrastive learning[41] (MolCLR), LightGBM[51], and Random Forest[52] (RF). Deep learning models, including MoLFormer, D-MPNN, and MolCLR, consist of two components, a feature extractor and a prediction head. The feature extractor (Transformer encoder in MoLFormer and graph neural network in D-MPNN and MolCLR) learns a molecular representation from the input SMILES. The prediction head is a two-layer fully connected neural network that predicts the docking score and uncertainty using the learned molecular representation. For LightGBM and RF, each molecule is represented by a 2048-bit atom-pair fingerprint[53] with minimum radius of 1 and maximum radius of 3. When greedy acquisition function is used, the all surrogate models predict only the docking score of each molecule and are trained with the mean squared error (MSE) loss. When the UCB acquisition function is used, neural network based surrogate models predict both the docking score and the uncertainty (variance), and is trained on the gaussian negative log-likelihood loss[54, 55]:

$$\mathcal{L}(y, \hat{y}, \hat{\sigma}^2) = \frac{1}{2}(\log\hat{\sigma}^2 + \frac{(\hat{y} - y)^2}{\hat{\sigma}^2}) \tag{3}$$

where the y and $\hat{y}$ are the ground truth and predicted docking or ROCS score, respectively. $\hat{\sigma}^2$ is the predicted variance which is clamped to 10^{-5} for training stability. For random forest based models

(RF and LightGBM), the uncertainty is calculated as the variance of predictions of all decision trees in the ensemble and they are always trained using the MSE loss function.

MoLFormer[39] is a transformer-based[56] model. It receives tokenized[57] SMILES string as input and generates molecular representation by learning the intrinsic spatial relationships between atoms in a molecule. Linear attention mechanism[58] and rotary position embedding[59] are adopted over the regular quadratic attention to improve the computation efficiency of the model. MoLFormer has been pretrained on a ultra-large 1.1 billion small molecule dataset combining ZINC[60] and PubChem[61] using the mask-language-modeling technique[62, 63] (Figure 1b). In this work, we use the MoLFormer-XL variant proposed in ref[39] which consists of 12 linear attention layer with 12 heads in each layer and a hidden dimension of 768. Molecular representation learned by MoLFormer is the average of the embeddings of all input tokens. The prediction head of MoLFormer has hidden dimension as 768 with gaussian error linear unit[64] activation and 0.1 rate of dropout[65]. LAMB[66] optimizer is used to enable large batch training. The training batch size is 600 and learning rate is fixed at 1.6×10^{-4}.

D-MPNN and MolCLR pretrained graph isomorphism network[46] are variants of the graph neural network[67] (GNN). They directly take the molecular graph, in which each atoms are represented as nodes and bonds as edges, as input thanks to their GNN architecture. Both model learn molecular representations through the message passing process. In message passing, the feature vector of each node in the molecular graph is updated by aggregating message from neighboring nodes. Through multiple aggregation update operations, the graph-level feature or molecular representation is calculated by a mean-pooling readout function over all node features in the graph. D-MPNN and MolCLR differs from each other by their unique aggregation update operation, which are detailed in ref[47] and ref[46, 41], respectively. MolCLR is pretrained on approximately 10 million molecules from PubChem[61] using contrastive learning strategy[68] (Figure 1c). For fair comparison, we use the default architecture setting of both the D-MPNN (3 graph layers and hidden size as 300) and MolCLR (5 graph layers, embedding size as 300, and feature size as 512). The prediction head of D-MPNN has hidden size of 300 and ReLU activation function. Following ref[30], D-MPNN is trained 50 epochs in batches of 50 using Adam[69] optimizer with Noam learning rate scheduler[56] (10^{-4}, 10^{-3}, and 10^{-4} as the initial, maximum, and final learning rate, respectively). The prediction head of MolCLR has hidden size of 256 and 128 for the first and second layer, respectively, and ReLU activation function. MolCLR is trained 50 epoches in batches of 32 using Adam[69] optimizer. The learning rate is 0.0002 and 0.0005 for the pretrained graph neural network and prediction head, respectively. RDKit[70] is used to convert SMILES to molecular graph. Molecules acquired at each iteration are split by 80%-20% to the training and validation set. Early stopping strategy with patient value of 10 on validation loss is adopted during the training of all neural network based models.

RF and LightGBM are decision tree based ensemble methods. RF is implemented using the scikit-learn[71] package. The `n_estimator` value is set to 100, `max_depth` set to 8, and `min_samples_leaf` set to 1. LightGBM is implemented using its dedicated python package[51]. The `n_estimator` value for LightGBM is set to 100, while the `max_depth` is unlimited.