Docking-based Virtual Screening with Multi-Task Learning

Zijing Liu
Baidu Inc.
Shenzhen, China
liuzijing01@baidu.com

Xianbin Ye
Jinan University
Guangzhou, China
yexianbin@stu2019.jnu.edu.cn

Xiaomin Fang
Baidu Inc.
Shenzhen, China
fangxiaomin01@baidu.com

Fan Wang
Baidu Inc.
Shenzhen, China
wangfan04@baidu.com

Hua Wu
Baidu Inc.
Beijing, China
wu_hua@baidu.com

Haifeng Wang
Baidu Inc.
Beijing, China
wanghaifeng@baidu.com

***Abstract*—Machine learning shows great potential in virtual screening for drug discovery. Current efforts on accelerating docking-based virtual screening do not consider using existing data of other previously developed targets. To make use of the knowledge of the other targets and take advantage of the existing data, in this work, we apply multi-task learning to the problem of docking-based virtual screening. With two large docking datasets, the results of extensive experiments show that multi-task learning can achieve better performances on docking score prediction. By learning knowledge across multiple targets, the model trained by multi-task learning shows a better ability to adapt to a new target. Additional empirical study shows that other problems in drug discovery, such as the experimental drug-target affinity prediction, may also benefit from multi-task learning. Our results demonstrate that multi-task learning is a promising machine learning approach for docking-based virtual screening and accelerating the process of drug discovery.**

***Index Terms*—Multi-Task Learning, Docking, Virtual Screening, Drug Discovery**

I. Introduction

Bringing a new drug to market is time-consuming and expensive. The pipeline of drug discovery and development is long and complex, with a high failure rate. The cost of a new drug was estimated to be $2.8 billion, and it took more than ten years [1]. Enormous efforts have been made to accelerate this process and lower the cost-to-market. Computer-aided drug discovery (CADD) has long been one of the most promising directions. One initial step of drug discovery is to identify new drug-like chemicals which has a strong affinity to a predefined protein target (Figure 1 (a)). However, due to the high cost and time limit, it is impossible to perform chemical experiments to screen the massive chemical space. In CADD, computers are used to perform a preliminary virtual screening of a large number of compounds and select the most promising compounds for further chemical experiments [2]. A common approach for virtual screening is to use molecular docking to predict the binding affinity and bound conformation. It has been shown that structure-based docking can achieve high hit-rates on ultra-large chemical libraries of more than 100 million compounds [3], [4].

However, the commercially available compound libraries have been growing, and their sizes have become so large that the computational time of virtual screening emerges to be an issue. For example, ZINC, a widely used compound library, now has more than 1.3 billion purchasable compounds containing 736 million lead-like molecules [5]. With a platform for ultra-large virtual screening [6], docking one billion compounds needs more than 4 million CPU-hours or 173 days with 1000 CPUs (15 seconds/ligand). The high requirement of such computational resources brings difficulties for many research and drug discovery organizations.

In order to mitigate the computational cost of growing virtual libraries, many new computational methods have been developed based on machine learning. A common approach is that a machine learning model is trained as a surrogate model to predict the output of molecular docking using the docking data from a small set of the compound library (Figure 1 (b)). The machine learning model is then used as a preliminary filter, and the top compounds predicted by the model are chosen for further validation.

To further improve the accuracy of the machine learning model, various machine learning techniques have been applied to the problem of docking-based virtual screening. One approach that attracts a lot of interest in drug discovery [7] is active learning, with applications in docking [8]–[10] and free energy estimation [11], [12]. Although active learning has been shown able to reduce the computational cost to screen a large library, it suffers from the bias introduced by the selection process [13]. More importantly, current machine learning methods for docking-based virtual screening, including recent works on active learning [8]–[10], try to train the model with only the data of the new target, neglecting any existing data of other targets (Figure 1 (b)). In practice, a research organization or a pharmaceutical company, which is going to perform virtual screening on a new target, usually has a lot of related data of other targets accumulated from its past drug development projects. In such a scenario, it is essential to take full advantage of the available data and avoid wasting

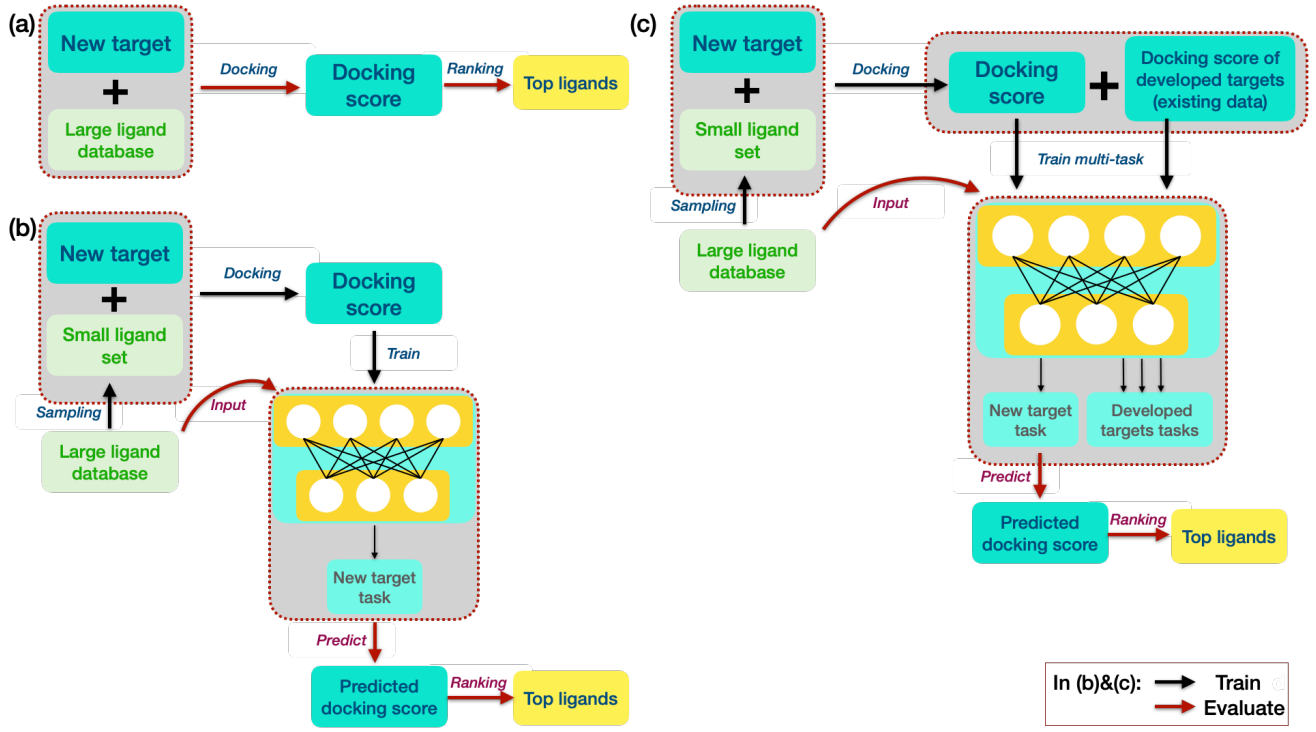


Fig. 1: The processes of docking-based virtual screening of a new target using (a) direct exhaustive docking, (b) machine learning accelerated screening, which samples a small subset of the ligands and use docking to obtain the data, and then train a machine learning model to predict the docking score of all the compounds in the large ligand database, (c) multi-task learning for screening, where the existing data of the other targets are combined with the new target docking scores to jointly train the model. Note that for active learning, the sampling and training processes in (b) may iterate for multiple rounds, but it does not use any other existing data.

data resources. As more and more data will be produced over time, these related data will play an increasingly important role.

The structure and property of a compound are closely related, which can be shown by the various applications of the quantitative structure-activity relationship (QSAR) or quantitative structure-property relationships (QSPR) models in drug discovery [14]. Docking is essentially a structure-based CADD approach. The docking score prediction models learned from other targets may contain useful knowledge about the new target. From this insight, it is reasonable to jointly model the docking data of the new and previous targets, which leads us to use Multi-Task Learning (MTL) [15]. MTL aims to learn multiple related tasks jointly such that a task can use the knowledge contained in other tasks to improve its performance, such as predictive accuracy and generalization ability. In previous studies, MTL has been mostly used for QSAR studies [16]–[19]. In this paper, we investigate MTL in docking score prediction for the purpose of virtual screening (Figure 1 (c)). We show that MTL improves the accuracy of docking score prediction. By learning the docking scores of the new and other (developed) targets together, the trained model has a large boost and outperforms both single-task learning and active learning. Our study also reports that the model trained with MTL learns a better representation of the compounds by sharing knowledge across multiple tasks, and thus can be used to adapt to a new task and achieve better performances. Besides docking score prediction, we also apply MTL to drug-target affinity prediction, showing that MTL can be employed in other problems of drug discovery.

The contributions of our paper can be concluded as follows:

- By integrating the existing docking data, we use multi-task learning in the process of docking-based virtual screening.
- Empirical studies show that multi-task learning can have better performances than single-task machine learning and active learning with the same number of docking compounds.
- With multi-task learning, the model can learn the common knowledge from related tasks and adapt to a new task with better performances.
- Multi-task learning is also applicable to other low data tasks in drug discovery, such as drug-target affinity prediction.

II. Methods

In this section, we introduce the MTL for virtual screening, where the targets with existing docking data are used to jointly train the model. The compound representation and neural network model are also described.

A. Multi-task learning

Given m tasks $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_m\}$, the aim of MTL is to learn all the tasks together to improve the performance for each task via learning the knowledge in other tasks. In the case of docking score prediction, one significant difference from traditional MTL is that we aim to improve the accuracy of predicting a new protein target, which we denote by $\mathcal{T}_0$, using the information learned from data of m existing targets. Those tasks from existing targets are denoted as $\{\mathcal{T}_1, \mathcal{T}_2, \ldots, \mathcal{T}_m\}$.

Nowadays, most of the models are based on deep learning due to their representation power and predictive ability. With deep neural networks, sharing the knowledge from other tasks can be easily achieved by parameter sharing. In a neural network model, some layers have shared weights across the tasks to jointly learn the knowledge from all the tasks, and there are also layers with the task-specific weights to capture the task-dependent features. The neural network model (Figure 2) can be represented by continuous and differentiable parametric functions with input x:

$$z = G_\phi(x), \tag{1}$$

$$\tilde{y}_i = F_{\theta_i}(z)\,,\ i = 0, 1, \ldots, m, \tag{2}$$

where G_ϕ is the layers with shared parameters ϕ, F_{θ_i} is the layers with task-specific weights θ_i, and $\tilde{y}_i$ is the predicted output for task i. The parameters ϕ and θ_i can be learned by optimizing a loss function $L(\tilde{y}_i, y_i)$ where y_i is the true label of task i. In virtual screening or docking score prediction, it is a regression problem, and the mean-squared error (MSE) loss is used. Note that a compound may have no label for some tasks. Let $\mathcal{X}_i$ denote the set of compounds that have the label of task i. The total loss can be written as

$$\sum_{i=0}^{m} \sum_{x \in \mathcal{X}_i} L(\tilde{y}_i, y_i), \tag{3}$$

which is minimized to learn the parameters of the neural networks.

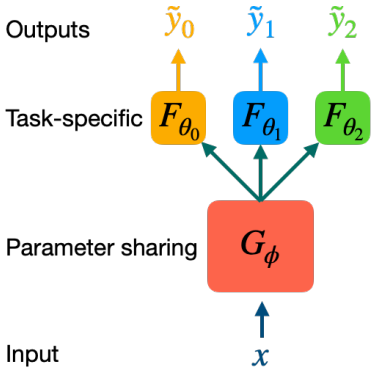


Fig. 2: MTL with shared-parameter and task-specific-parameter neural network layers. ϕ and θ_i are learnable parameters.

B. MTL model for docking score prediction

We now introduce the MTL model for docking score prediction in detail (Figure 3).

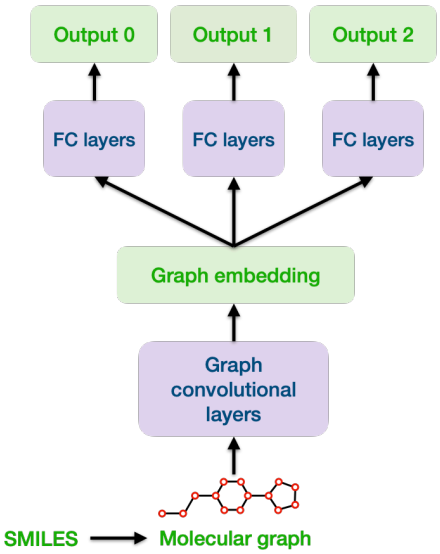


Fig. 3: The MTL model for virtual screening with graph neural networks. The weights of the graph convolutional layers are shared across all the tasks. For each task, the output embedding of the graph convolutional layers goes into task-specific fully connected layers to get the final prediction.

1) Input molecular features: In a chemical library, the compounds are usually represented by SMILES (Simplified Molecular Input Line Entry System) [20]. Recently, modeling the molecules as graphs has shown a great success in many problems related to drug discovery, including molecular property prediction, drug-target interaction prediction, and molecule design [21]–[24]. Thus we convert the SMILES codes to molecular graphs with the package RDKit [25], such that a compound is a graph with the nodes being the atoms and the edges being the chemical bonds. RDKit is also used to generate features for the molecular graph, including seven atom features and two bond features. All the features are discrete values and encoded as one-hot vectors with different lengths, as shown in Table I.

TABLE I: Input features for the molecular graph

Atom feature	atomic type	119 bits one hot
	formal charge	16 bits one hot
	degree	11 bits one hot
	chirality tag	4 bits one hot
	number of hydrogens	9 bits one hot
	aromaticity	2 bits one hot
	hybridization	5 bits one hot
Bond feature	bond direction,	7 bits one hot
	bond type	4 bits one hot
	is in ring	2 bits one hot

2) Parameter sharing module: With the input of molecular graph data, the parameter-sharing part of our MTL model G_ϕ

is based on the Graph Isomorphism Network (GIN), which has been shown to achieve the state-of-the-art performance in drug-target affinity prediction [24], [26]. As shown in Table I, the features generated by RDKit are all one-hot vectors, which are denoted as $f_{v,i}$ $(i = 1, \ldots, 7)$ and $f_{e,j}$ $(j = 1, 2, 3)$ for node v and edge e. In order to input the features to GNNs, we first use embedding operations to map the one-hot vectors into d-dimensional real vectors:

$$h_v^0 = \sum_{i=1}^{7} \text{NodeEmbedding}_i(f_{v,i}),$$

$$h_e^k = \sum_{j=1}^{3} \text{EdgeEmbedding}_j^k(f_{e,j}) \quad k = 0, 1, \ldots, K-1,$$

where K is the total number of GNN layers. In the k-th layer, GIN updates the node representations by

$$h_v^k = \sigma\left(g^k\left(\sum_{u \in \mathcal{N}(v) \cup \{v\}} h_u^{k-1} + \sum_{e=(v,u): u \in \mathcal{N}(v) \cup \{v\}} h_e^{k-1}\right)\right),$$

where $\mathcal{N}(v)$ is the set of nodes adjacent to node v, $\sigma(\cdot)$ is the ReLU activation function, and $g^k(\cdot)$ is a two-layer perceptron with $2d$ hidden neurons followed by batch normalization [27]. Since the docking score prediction is a graph-level regression task, the graph representation is obtained by averaging the node embeddings of the last GIN layer:

$$z = \text{MEAN}_v(h_v^K).$$

3) Task-specific module: The output of the GIN layers gives a vector representation of the compounds, which contains shared knowledge across all the tasks. These vector representations are then fed into the task-specific neural networks to make predictions for each task. Here, we apply a two-layer fully connected (FC) neural network on top of the graph representation z to obtain the final prediction for each task.

C. Other methods for comparison

The first baseline for comparison is the traditional single-task machine learning method for virtual screening. In addition, several recent works for docking-based virtual screening have focused on the technique of active learning [9], [10]. We therefore also compare MTL with the (single-task) active learning approach. For both cases, the same neural network architecture is used. The only difference is that the number of tasks is set to be one. For active learning, we adopt the same data acquisition strategy as described in [10], where an ensemble of five models with the same architecture is used. To ensure a fair comparison, the training samples of the new target task always have the same size for all the approaches.

III. Results

The performances of different machine learning approaches for virtual screening are evaluated with two large docking datasets. In addition, we also collect a drug-target affinity dataset to illustrate that MTL is applicable to other problems for drug discovery.

A. Experiment details

1) Dataset: The first docking dataset was downloaded from [8], which contained 12 targets (PDB id: 1ERR, 1T7R, 2ZV2, 4AG8, 4F8H, 4R06, 4YAY, 5EK0, 5L2S, 5MZJ, 6D6T, 6IIU), and each target was docked with 3 million compounds randomly sampled from ZINC15 by the program FRED [28]. The second dataset was obtained from [3], in which 99 million lead-like compounds were docked against the target AmpC (AmpC β-lactamase) using DOCK3.7.2 [29]. These two datasets allow us to test the performance of MTL for virtual screening in a large scale and across different docking softwares. To further test the ability of MTL in drug discovery, we also collect an experimental drug-target affinity dataset (the half-maximal inhibitory concentrations of the ligands on their protein targets, i.e., IC_{50}) with eight targets from the ChEMBL database [30]. ChEMBL uses the pChEMBL value as the measurement of the ligand-target affinity, which is defined as:

$$\text{pChEMBL} = -\log(\text{IC}_{50}). \tag{4}$$

For example, an IC_{50} measurement of 10 uM would have a pChEMBL value of 5. We use this pChEMBL value as the ground truth in our experiment. When a ligand-target pair has multiple pChEMBL values, we take the average of all the pChEMBL values. The details of the drug-target affinity dataset are summarized in TABLE II.

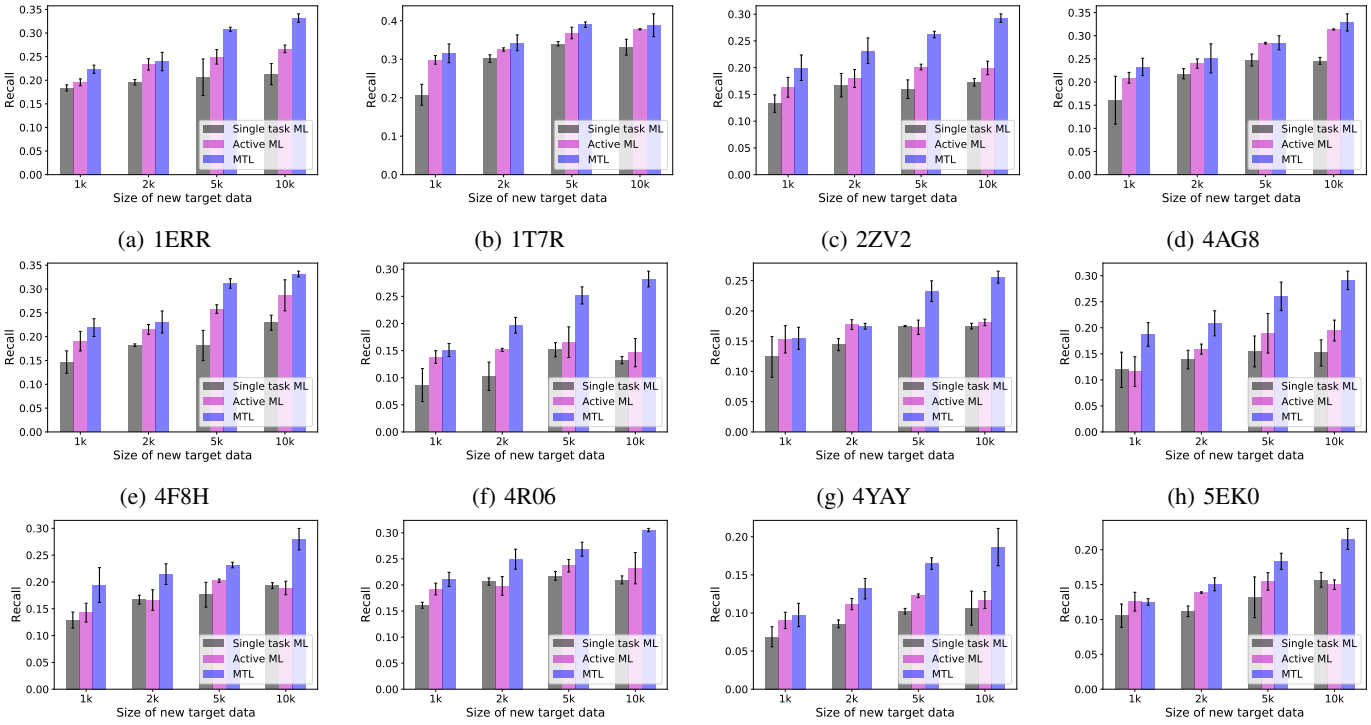
TABLE II: Details of the drug-target affinity dataset collected from ChEMBL

ChEMBL ID	Target name	Data points
203	Epidermal growth factor receptor erbB1	6414
279	Fetal liver kinase 1	7777
267	Tyrosine-protein kinase SRC	2775
325	Histone deacetylase 1	4473
333	Matrix metalloproteinase-2	2673
2842	Serine/threonine-protein kinase mTOR	3662
2971	Tyrosine-protein kinase JAK2	4855
4005	PI3-kinase p110-alpha subunit	4459

2) Evaluation metrics: The performance of the machine learning approaches for docking score prediction is evaluated by the virtual hit ratio. We define the top-k compounds as the virtual hits. For the first dataset of 3 million compounds, k is set to be 3,000 or 30,000. For the larger dataset of 99 million compounds, k is 10,000 or 100,000. The predicted virtual hits are taken as the top 2%, 3% or 5% compounds of the predictions. The hit ratio is calculated as the recall:

$$\text{recall} = \frac{TP}{TP + FN},$$

where TP (true positives) are the correctly predicted virtual hits, and FN (false negatives) are the virtual hits discarded by the prediction incorrectly.



(a) 1ERR (b) 1T7R (c) 2ZV2 (d) 4AG8

(e) 4F8H (f) 4R06 (g) 4YAY (h) 5EK0

(i) 5L2S (j) 5MZJ (k) 6D6T (l) 6IIU

Fig. 4: Recall of top 3k virtual hit of three approaches with different 'new target' data sizes for all 12 targets at top 2% predicted values. For MTL, the 'new target' data is trained together with 2 million samples from the 11 'developed targets'. The error bars are obtained from three independent runs.

The performance of the drug-target affinity prediction is measured by the mean-square error (MSE) and concordance index (CI). The CI is defined as

$$\mathrm{CI} = \frac{\sum_{k,l} \mathbf{1}_{\tilde{y}_k > \tilde{y}_l} \cdot \mathbf{1}_{y_k > y_l}}{\sum_{k,l} \mathbf{1}_{y_k > y_l}},$$

where $\tilde{y}_k$ and y_k are the predicted and true value of the k-th sample respectively, and $\mathbf{1}_{y_k > y_l} = 1$ if $y_k > y_l$, else 0.

3) Model training details: The embedding dimension for the features d is 256. The number of GIN layers is eight. The number of hidden neurons in the task-specific FC layers is also 256. The model is trained with the ADAM optimizer with a learning rate 0.001 and batch size 128 [31]. During training, we apply dropout after the ReLU for all GNN layers and the task-specific FC layers [32]. The dropout rate is set to be 0.2. 20% of the training data are used for validation to avoid over-fitting. The model is trained for at least 100 epochs until the validation loss is larger than the training loss for more than 50 consecutive epochs. The epoch with the smallest validation loss is chosen as the final model. For active learning, the final prediction is the average of the five models in the ensemble.

B. MTL for docking-based virtual screening

For the first dataset, one of the 12 targets is taken as the 'new target' and the other 11 targets are treated as the 'developed targets' with existing data. For each new target, we randomly choose 1k, 2k, 5k or 10k samples as the training data. For active learning, the total number of samples for training is kept to be the same to ensure a fair comparison. For MTL, we choose 100k, 500k, 1m (million), and 2m samples randomly from the 11 developed targets to jointly train the model with the samples from the new target. Figure 4 shows the bar plots comparing the recalls of the top 3000 virtual hits of the top 2% of the compounds predicted by three machine learning approaches across different new target data sizes. As all three approaches have better performances when the training size of the new target samples increases, MTL achieves higher recall values than single-task and active learning in all the targets across four different new target data sizes. In several targets (such as 2ZV2 and 5EK0), MTL with 1k new target data can obtain a comparable recall value to the single-task machine learning and active learning with 10k new target data, which means that MTL requires less docking data of the new target to achieve the same performance. Thus in practice, MTL can further accelerate docking-based virtual screening without sacrificing the performance. As a summary of the prediction performance, TABLE III gives the Pearson correlation between the real and predicted docking scores. The complete results of all the targets across different new target sizes and other target sizes for MTL can be found in the Supplementary Table [1].

[1]https://github.com/PaddlePaddle/PaddleHelix/tree/dev/apps/drug_target_interaction/MTL_docking/BIBM_SupplementaryMaterials

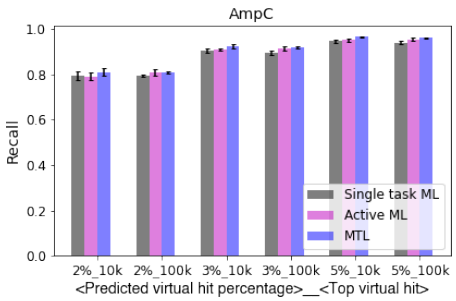


Fig. 5: Recall of the top 10k and 100k virtual hits of different percentages of the predicted values of the 99 million AmpC docking data. Although single-task and active learning already achieve very high recall values, MTL performs better.

To further examine the ability of MTL for large-scale virtual screening, we use the second dataset, which contains the docking scores of 99 million compounds against the target AmpC. Out of the 99 million docking results, 100k samples of AmpC are chosen as the new target data. 2m random samples from the 12 targets in the first dataset are used in MTL as the data of other tasks. As shown in Figure 5, MTL performs the best when evaluated across different numbers of virtual hits or predicted virtual hits. Note that the AmpC docking data is produced by a different docking software. So our result also indicates that MTL allows knowledge-sharing across different docking programs.

C. Results on drug-target affinity prediction

Next, we use the collected drug-target affinity dataset to further test the ability of MTL in other problems of drug discovery. As in the first docking dataset, we take one of the targets as the new target, of which 60% samples are used as the testing set for evaluation. The other 40% samples are used for single-task machine learning, or combined with the data of the other targets for MTL training. TABLE IV shows that compared to single-task machine learning, MTL has a substantial improvement in terms of both MSE and CI, indicating that MTL helps predict experimental drug-target affinity values and is applicable to not only docking score prediction but also other tasks in drug discovery.

To obtain an intuitive understanding of the difference between the single and multi-task learning models, we visualize the compound embedding learned by the GIN layers. Specifically, we randomly sample 1000 compounds for each of the 10 most common scaffolds from the ZINC database [33]. UMAP is then used to visualize the embedding of these 10000 compounds from the last GIN layer [34]. As shown in Figure 6, the model trained by MTL gives more distinctive clusters with correspondence to the 10 scaffolds than the single-task model. Quantitatively, the embedding of the MTL model has a smaller DB index and higher Silhouette index [35], [36]. As the scaffold represents the core structure of a compound, our results indicate that the model trained with MTL captures more inherent characteristics via sharing knowledge across multiple tasks. By MTL, the learned model achieves a similar effect of representation learning by self-supervised learning [37]. Thus the high-quality representation learned by MTL leads to greater predictive power compared to single-task learning.

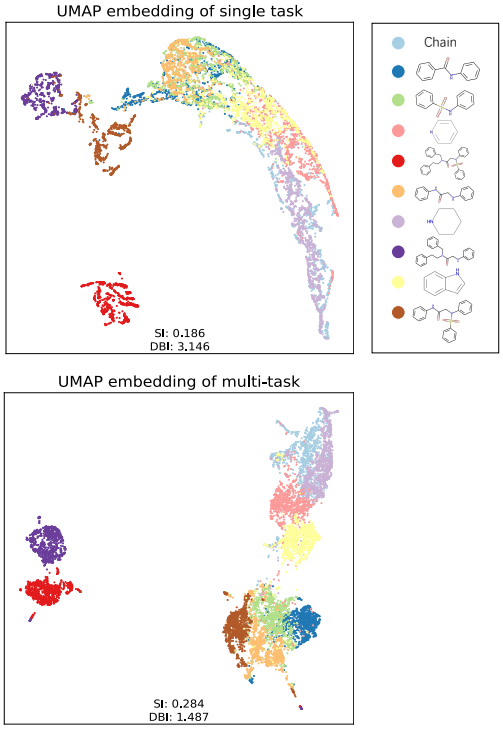


Fig. 6: Visualization of the compound embeddings extracted from the final layer of the GINs from the trained single-task and multi-task models. A lower DB index or higher Silhouette index indicates a more appropriate separation.

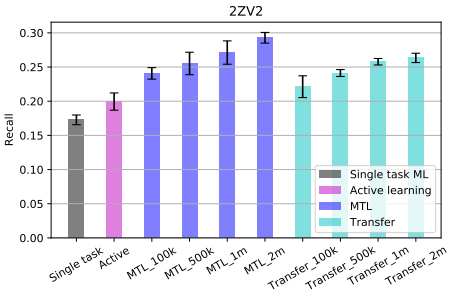


Fig. 7: Bar plots of the recall of top 3k virtual hit for the target 2ZVS at top 2% predicted values of different ML approaches. The training size for the new target is 10k. MTL and transfer learning include the results of 4 different training sizes for the other targets. The error bars are obtained from three independent runs.

D. MTL trained model contains common reusable knowledge

We have shown that MTL helps learn a good representation of the chemical compounds, which indicates that the model is able to capture common knowledge across all the tasks.

TABLE III: Pearson correlation of single-task machine learning (ML), active learning (AL) and MTL with different new target sizes. MTL is jointly trained with 2 million data from other targets.

	Target size=1k			Target size=2k			Target size=5k			Target size=10k		
	ML	AL	MTL	ML	AL	MTL	ML	AL	MTL	ML	AL	MTL
1ERR	0.562	0.566	**0.630**	0.605	0.610	**0.662**	0.617	0.635	**0.710**	0.627	0.645	**0.732**
1T7R	0.800	0.790	**0.844**	0.823	0.826	**0.859**	0.851	0.852	**0.882**	0.859	0.854	**0.879**
2ZV2	0.531	0.545	**0.618**	0.581	0.582	**0.646**	0.588	0.606	**0.680**	0.608	0.610	**0.696**
4AG8	0.671	0.685	**0.751**	0.718	0.737	**0.771**	0.758	0.769	**0.790**	0.770	0.777	**0.812**
4F8H	0.566	0.596	**0.658**	0.636	0.628	**0.689**	0.650	0.668	**0.721**	0.667	0.694	**0.747**
4R06	0.493	0.543	**0.597**	0.548	0.585	**0.639**	0.610	0.602	**0.680**	0.603	0.596	**0.697**
4YAY	0.528	0.553	**0.550**	0.568	0.580	**0.589**	0.585	0.590	**0.639**	0.595	0.601	**0.669**
5EK0	0.492	0.496	**0.564**	0.525	0.532	**0.587**	0.564	0.560	**0.650**	0.570	0.580	**0.680**
5L2S	0.474	0.496	**0.579**	0.519	0.516	**0.620**	0.562	0.563	**0.655**	0.586	0.527	**0.679**
5MZJ	0.663	0.677	**0.743**	0.704	0.704	**0.762**	0.725	0.744	**0.778**	0.733	0.744	**0.798**
6D6T	0.498	0.534	**0.616**	0.587	0.590	**0.655**	0.611	0.620	**0.694**	0.610	0.623	**0.715**
6IIU	0.585	0.609	**0.670**	0.639	0.648	**0.708**	0.669	0.679	**0.747**	0.694	0.667	**0.764**

TABLE IV: The MSE and CI of the drug-target affinity prediction. The standard deviation in the parentheses is computed from three independent runs.

MSE

ChEMBL ID	325	333	4005	2971	2842	267	203	279
Single-task ML	0.779(0.017)	1.173(0.017)	0.632(0.047)	0.739(0.004)	0.624(0.010)	0.907(0.017)	1.033(0.036)	0.875(0.026)
MTL	0.646(0.019)	0.960(0.035)	0.480(0.022)	0.586(0.017)	0.477(0.006)	0.775(0.010)	0.693(0.004)	0.606(0.014)
Improvement	17.09%	18.17%	24.12%	20.73%	23.64%	14.60%	32.92%	30.78%

CI

ChEMBL ID	325	333	4005	2971	2842	267	203	279
Single-task ML	0.721(0.011)	0.755(0.005)	0.756(0.007)	0.720(0.011)	0.775(0.005)	0.761(0.003)	0.729(0.005)	0.691(0.012)
MTL	0.753(0.005)	0.785(0.005)	0.793(0.003)	0.765(0.004)	0.810(0.001)	0.779(0.003)	0.789(0.001)	0.762(0.004)
Improvement	4.41%	3.90%	4.97%	6.33%	4.50%	2.31%	8.19%	10.16%

We next examine if the knowledge learned by MTL can be reused for a new task by transferring the MTL trained model to a new target and checking if it can improve the prediction performance. We test this idea using the first docking dataset with 12 targets. Similarly, one target is taken as the new target. An MTL model is first trained with the data of the other 11 targets. The weights of the MTL trained GIN layers are used as the initialization to train a single-task model of the new target. We first train the FC layers for 20 epochs and then train the whole neural network. From Figure 7, it is clear to see that transfer learning based on the MTL model achieves better recall values than single-task machine learning and active learning. Figure 7 also reports that when the data size of the other tasks increases from 100k to 2m, the performances of MTL and MTL initialized transfer learning become better.

IV. Conclusion

As the number of compounds in chemical libraries available to screening grows rapidly, machine learning approaches play an important role in docking-based virtual screening. Current works for docking barely consider using the docking data of other targets from previous screens. Our work is motivated by a practical question of how to utilize the existing data with machine learning during the development of a new drug target. With this motivation, we investigate MTL for the task of docking-based virtual screening in drug discovery. Combining the data from other tasks, we show that MTL achieves better performances than single-task and active machine learning by sharing knowledge across multiple tasks. Further experiments on a collected drug-target affinity dataset indicate that MTL learns better representations of the compounds and shows the potential in other problems for drug discovery. For example, it is straightforward to extend the MTL framework to applications of more computational costly jobs such as free energy calculation [38].

In this work, we focus on the application of MTL in docking-based virtual screening. In the aspect of deep learning model development, a lot of efforts have been made to design new neural network architectures for MTL. One direction of future works can be optimization of the MTL neural network architecture for virtual screening in drug discovery. Regarding how to make use of additional data, various other machine learning techniques can also be investigated in the future, such as meta-learning, self-supervised learning, and semi-supervised learning, which are all closely related to MTL [39]–[41].

References

[1] J. A. DiMasi, H. G. Grabowski, and R. W. Hansen, "Innovation in the pharmaceutical industry: new estimates of R&D costs," *Journal of health economics*, vol. 47, pp. 20–33, 2016.

[2] B. K. Shoichet, "Virtual screening of chemical libraries," *Nature*, vol. 432, no. 7019, pp. 862–865, 2004.

[3] J. Lyu, S. Wang, T. E. Balius, I. Singh, A. Levit, Y. S. Moroz, M. J. O'Meara, T. Che, E. Algaa, K. Tolmachova *et al.*, "Ultra-large library docking for discovering new chemotypes," *Nature*, vol. 566, no. 7743, pp. 224–229, 2019.

[4] R. M. Stein, H. J. Kang, J. D. McCorvy, G. C. Glatfelter, A. J. Jones, T. Che, S. Slocum, X.-P. Huang, O. Savych, Y. S. Moroz *et al.*, "Virtual discovery of melatonin receptor ligands to modulate circadian rhythms," *Nature*, vol. 579, no. 7800, pp. 609–614, 2020.

[5] J. J. Irwin, K. G. Tang, J. Young, C. Dandarchuluun, B. R. Wong, M. Khurelbaatar, Y. S. Moroz, J. Mayfield, and R. A. Sayle, "ZINC20—a free ultralarge-scale chemical database for ligand discovery," *Journal of chemical information and modeling*, vol. 60, no. 12, pp. 6065–6073, 2020.

[6] C. Gorgulla, A. Boeszoermenyi, Z.-F. Wang, P. D. Fischer, P. W. Coote, K. M. P. Das, Y. S. Malets, D. S. Radchenko, Y. S. Moroz, D. A. Scott *et al.*, "An open-source drug discovery platform enables ultra-large virtual screens," *Nature*, vol. 580, no. 7805, pp. 663–668, 2020.

[7] D. Reker, "Practical considerations for active machine learning in drug discovery," *Drug Discovery Today: Technologies*, vol. 32, pp. 73–79, 2019.

[8] F. Gentile, V. Agrawal, M. Hsing, A.-T. Ton, F. Ban, U. Norinder, M. E. Gleave, and A. Cherkasov, "Deep docking: a deep learning platform for augmentation of structure based drug discovery," *ACS central science*, vol. 6, no. 6, pp. 939–949, 2020.

[9] D. E. Graff, E. I. Shakhnovich, and C. W. Coley, "Accelerating high-throughput virtual screening through molecular pool-based active learning," *Chem. Sci.*, vol. 12, pp. 7866–7881, 2021. [Online]. Available: http://dx.doi.org/10.1039/D0SC06805E

[10] Y. Yang, K. Yao, M. P. Repasky, K. Leswing, R. Abel, B. Shoichet, and S. Jerome, "Efficient exploration of chemical space with docking and deep-learning," *ChemRxiv*, 2021.

[11] J. S. Smith, B. Nebgen, N. Lubbers, O. Isayev, and A. E. Roitberg, "Less is more: Sampling chemical space with active learning," *The Journal of chemical physics*, vol. 148, no. 24, p. 241733, 2018.

[12] K. D. Konze, P. H. Bos, M. K. Dahlgren, K. Leswing, I. Tubert-Brohman, A. Bortolato, B. Robbason, R. Abel, and S. Bhat, "Reaction-based enumeration, active learning, and free energy calculations to rapidly explore synthetically tractable chemical space and optimize potency of cyclin-dependent kinase 2 inhibitors," *Journal of chemical information and modeling*, vol. 59, no. 9, pp. 3782–3793, 2019.

[13] S. Farquhar, Y. Gal, and T. Rainforth, "On statistical bias in active learning: How and when to fix it," in *International Conference on Learning Representations*, 2021. [Online]. Available: https://openreview.net/forum?id=JiYq3eqTKY

[14] C. Nantasenamat, C. Isarankura-Na-Ayudhya, and V. Prachayasittikul, "Advances in computational methods to predict the biological activity of compounds," *Expert opinion on drug discovery*, vol. 5, no. 7, pp. 633–654, 2010.

[15] Y. Zhang and Q. Yang, "A survey on multi-task learning," *IEEE Transactions on Knowledge and Data Engineering*, 2021.

[16] Y. Xu, J. Ma, A. Liaw, R. P. Sheridan, and V. Svetnik, "Demystifying multitask deep neural networks for quantitative structure–activity relationships," *Journal of chemical information and modeling*, vol. 57, no. 10, pp. 2490–2504, 2017.

[17] B. Ramsundar, B. Liu, Z. Wu, A. Verras, M. Tudor, R. P. Sheridan, and V. Pande, "Is multitask deep learning practical for pharma?" *Journal of chemical information and modeling*, vol. 57, no. 8, pp. 2068–2076, 2017.

[18] K. Lee and D. Kim, "In-silico molecular binding prediction for human drug targets using deep neural multi-task learning," *Genes*, vol. 10, no. 11, p. 906, 2019.

[19] S. Sosnin, M. Vashurina, M. Withnall, P. Karpov, M. Fedorov, and I. V. Tetko, "A survey of multi-task learning methods in chemoinformatics," *Molecular informatics*, vol. 38, no. 4, p. 1800108, 2019.

[20] D. Weininger, "SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules," *Journal of chemical information and computer sciences*, vol. 28, no. 1, pp. 31–36, 1988.

[21] J. Gilmer, S. S. Schoenholz, P. F. Riley, O. Vinyals, and G. E. Dahl, "Neural message passing for quantum chemistry," in *Proceedings of the 34th International Conference on Machine Learning*, ser. Proceedings of Machine Learning Research, D. Precup and Y. W. Teh, Eds., vol. 70. PMLR, 06–11 Aug 2017, pp. 1263–1272. [Online]. Available: http://proceedings.mlr.press/v70/gilmer17a.html

[22] K. Liu, X. Sun, L. Jia, J. Ma, H. Xing, J. Wu, H. Gao, Y. Sun, F. Boulnois, and J. Fan, "Chemi-Net: a molecular graph convolutional network for accurate drug property prediction," *International journal of molecular sciences*, vol. 20, no. 14, p. 3389, 2019.

[23] M. Sun, S. Zhao, C. Gilvary, O. Elemento, J. Zhou, and F. Wang, "Graph convolutional networks for computational drug development and discovery," *Briefings in bioinformatics*, vol. 21, no. 3, pp. 919–935, 2020.

[24] T. Nguyen, H. Le, T. P. Quinn, T. Nguyen, T. D. Le, and S. Venkatesh, "GraphDTA: Predicting drug–target binding affinity with graph neural networks," *Bioinformatics*, vol. 37, no. 8, pp. 1140–1147, 2021.

[25] G. Landrum, "RDKit: Open-source cheminformatics," 2006.

[26] K. Xu, W. Hu, J. Leskovec, and S. Jegelka, "How powerful are graph neural networks?" in *International Conference on Learning Representations*, 2019. [Online]. Available: https://openreview.net/forum?id=ryGs6iA5Km

[27] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *International conference on machine learning*. PMLR, 2015, pp. 448–456.

[28] M. McGann, "FRED and HYBRID docking performance on standardized datasets," *Journal of computer-aided molecular design*, vol. 26, no. 8, pp. 897–906, 2012.

[29] R. G. Coleman, M. Carchia, T. Sterling, J. J. Irwin, and B. K. Shoichet, "Ligand pose and orientational sampling in molecular docking," *PloS one*, vol. 8, no. 10, p. e75992, 2013.

[30] D. Mendez, A. Gaulton, A. P. Bento, J. Chambers, M. De Veij, E. Félix, M. P. Magariños, J. F. Mosquera, P. Mutowo, M. Nowotka *et al.*, "ChEMBL: towards direct deposition of bioassay data," *Nucleic acids research*, vol. 47, no. D1, pp. D930–D940, 2019.

[31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[32] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[33] G. W. Bemis and M. A. Murcko, "The properties of known drugs. 1. molecular frameworks," *Journal of medicinal chemistry*, vol. 39, no. 15, pp. 2887–2893, 1996.

[34] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform manifold approximation and projection for dimension reduction," 2020.

[35] X. L. Xie and G. Beni, "A validity measure for fuzzy clustering," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 13, no. 8, pp. 841–847, 1991.

[36] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[37] P. Li, J. Wang, Y. Qiao, H. Chen, Y. Yu, X. Yao, P. Gao, G. Xie, and S. Song, "An effective self-supervised framework for learning expressive molecular global representations to drug discovery," *Briefings in Bioinformatics*, 05 2021, bbab109. [Online]. Available: https://doi.org/10.1093/bib/bbab109

[38] L. Wang, Y. Wu, Y. Deng, B. Kim, L. Pierce, G. Krilov, D. Lupyan, S. Robinson, M. K. Dahlgren, J. Greenwood *et al.*, "Accurate and reliable prediction of relative ligand binding potency in prospective drug discovery by way of a modern free-energy calculation protocol and force field," *Journal of the American Chemical Society*, vol. 137, no. 7, pp. 2695–2703, 2015.

[39] T. M. Hospedales, A. Antoniou, P. Micaelli, and A. J. Storkey, "Meta-learning in neural networks: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[40] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine Learning*, vol. 109, no. 2, pp. 373–440, 2020.

[41] X. Liu, F. Zhang, Z. Hou, L. Mian, Z. Wang, J. Zhang, and J. Tang, "Self-supervised learning: Generative or contrastive," *IEEE Transactions on Knowledge and Data Engineering*, 2021.