JOURNAL OF CHEMICAL INFORMATION AND MODELING

Article

Cite This: J. Chem. Inf. Model. 2019, 59, 282−293

pubs.acs.org/jcim

# Practical Model Selection for Prospective Virtual Screening

Shengchao Liu,[†,‡,∇] Moayad Alnammi,[†,‡,∇] Spencer S. Ericksen,[§] Andrew F. Voter,[‖] Gene E. Ananiev,[§] James L. Keck,[‖] F. Michael Hoffmann,[§,⊥] Scott A. Wildman,[§] and Anthony Gitter*[,#,†,‡]

[†]Department of Computer Sciences, University of Wisconsin-Madison, Madison, Wisconsin 53706, United States

[‡]Morgridge Institute for Research, Madison, Wisconsin 53715, United States

[§]Small Molecule Screening Facility, University of Wisconsin Carbone Cancer Center, Madison, Wisconsin 53792, United States

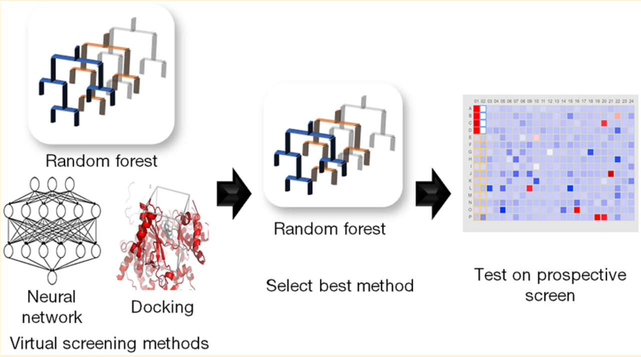
[‖]Department of Biomolecular Chemistry, University of Wisconsin School of Medicine and Public Health, Madison, Wisconsin 53706, United States

[⊥]McArdle Laboratory for Cancer Research, University of Wisconsin-Madison, Madison, Wisconsin 53705, United States

[#]Department of Biostatistics and Medical Informatics, University of Wisconsin-Madison, Madison, Wisconsin 53792, United States

*Supporting Information*

**ABSTRACT:** Virtual (computational) high-throughput screening provides a strategy for prioritizing compounds for experimental screens, but the choice of virtual screening algorithm depends on the data set and evaluation strategy. We consider a wide range of ligand-based machine learning and docking-based approaches for virtual screening on two protein−protein interactions, PriA-SSB and RMI-FANCM, and present a strategy for choosing which algorithm is best for prospective compound prioritization. Our workflow identifies a random forest as the best algorithm for these targets over more sophisticated neural network-based models. The top 250 predictions from our selected random forest recover 37 of the 54 active compounds from a library of 22,434 new molecules assayed on PriA-SSB. We show that virtual screening methods that perform well on public data sets and synthetic benchmarks, like multi-task neural networks, may not always translate to prospective screening performance on a specific assay of interest.



## 1. INTRODUCTION

Drug discovery is time consuming and expensive. After a specific protein or mechanistic pathway is identified to play an essential role in a disease process, the search begins for a chemical or biological ligand that can perturb the action or abundance of the disease target in order to mitigate the disease phenotype. A standard approach to discover a chemical ligand is to screen thousands to millions of candidate compounds against the target in biochemical- or cell-based assays via a process called high-throughput screening (HTS), which produces vast sets of valuable pharmacological data. Even though HTS assays are highly automated, screens of thousands of compounds sample only a small fraction of the millions of commercially available drug-like compounds. Cost and time preclude academic laboratories and even pharmaceutical companies from blindly testing the full set of drug-like compounds in HTS assays. Thus, there is a crucial need for an effective virtual screening (VS) process as a preliminary step in prioritizing compounds for HTS assays.

Virtual screening comprises two categories: structure-based[1,2] and ligand-based methods.[3,4] Structure-based methods require that the target protein's molecular structure is known so that the 3D interactions between the target and each chemical compound (binding poses) may be predicted *in silico*. These interactions are given numeric scores, which are then used to rank compounds for potential binding to the target. These methods do not require or typically make use of historical screening data in compound scoring. In contrast, ligand-based methods require no structural information about the target. They use data generated from testing molecules in biochemical or functional assays of the target to fit empirical models that relate compound attributes to assay outcomes.

For targets with abundant assay data or where a druggable binding site is not well defined, such as the targets considered here, ligand-based methods are generally superior to structure-based methods.[5−7] Confronted with the variety of ligand-based model building methods (e.g., regression models, random forests, support vector machines, etc.),[8] compound input representations, and performance metrics, how should one proceed with VS on a new target? The Merck Molecular Activity Challenge[9] incited the development of many ligand-

**Received:** June 8, 2018
**Published:** November 30, 2018

based deep learning VS methods,[10−14] as recently reviewed.[15,16] These methods are often assessed with cross-validation on existing HTS data, but there is presently little experimental evidence on the best option for prioritizing new compounds given a fixed screening budget.

We critically evaluated a collection of VS algorithms that include both structure-based and ligand-based methods, with a focus on the subset of quantitative structure−activity relationship ligand-based methods that use machine learning to predict active compounds for a target based on initial screening data. We present a VS workflow that first uses available HTS training data to systematically prune the specific versions of the algorithms and calculate their cross-validation performance on a variety of evaluation metrics. Based on the cross-validation results and analysis of the various evaluation metrics, we selected a single virtual screening algorithm. The selected method, a random forest model, was the best option for prioritizing a small number of compounds from a new library, as verified by experimental screening. These model selection and evaluation strategies can guide VS practitioners to select the best model for their target even as the landscape of available VS algorithms continues to evolve.

## 2. METHODS

### 2.1. Data Sets.

Our case studies were on new and recently generated data sets[17,18] for the targets PriA-SSB and RMI-FANCM. The PriA-SSB interaction is important in bacterial DNA replication and is a potential target for antibiotics.[19] The RMI-FANCM interaction is involved in DNA repair that is induced in human cancer cells to confer chemoresistance to cytotoxic DNA-cross-linking agents, making it an attractive drug target.[20] We previously screened these targets with a library of compounds obtained from Life Chemicals, Inc. (LC) in different assay formats. In addition, we screened new LC compounds on the PriA-SSB target to evaluate our VS models. The four data sets derived from these screens are described below and summarized in Table 1.

**Table 1. Summary Statistics for the Four Binary Data Sets**

| Stage | Data set | % inhibition threshold | # actives | # inactives |
|---|---|---|---|---|
| Cross-validation | PriA-SSB AS | ≥35% | 79 | 72,344 |
| | PriA-SSB FP | ≥30% | 24 | 72,399 |
| | RMI-FANCM FP | ≥ mean + 2 SD | 230 | 49,566 |
| Prospective | PriA-SSB prospective | ≥35% | 54 | 22,380 |

*2.1.1. PriA-SSB AlphaScreen.* PriA-SSB was initially screened using an AlphaScreen (AS) assay in a 1536-well format[18] on 72,423 LC compounds at a single concentration (33.3 $\mu$M), with data reported as % inhibition compared to controls. We refer to these continuous values as a "PriA-SSB % inhibition". Those compounds that tested above an activity threshold (≥35% inhibition) and passed PAINS chemical structural filters[21,22] were retested in the same AS assay. PAINS filters are not a technical necessity of any VS method, and some analyses have shown they are imperfect filters of nonspecific pan assay interference.[23] Nevertheless, they are a common requirement for publication of HTS and medicinal chemistry projects. We did not remove compounds detected by PAINS filters from the data set but rather flagged them and labeled them as inactive. Compounds that were confirmed in the AS retest screen (≥35% inhibition) were marked as actives, creating the binary data set PriA-SSB AS.

*2.1.2. PriA-SSB Fluorescence Polarization.* Compounds that had PriA-SSB % inhibition ≥ 35% and passed the PAINS filters were also tested in a fluorescence polarization (FP) assay as a secondary screen. Those compounds with FP inhibition ≥ 30% were labeled as actives, creating the binary data set PriA-SSB FP, with all other compounds in the screening set labeled inactive.

*2.1.3. RMI-FANCM Fluorescence Polarization.* The RMI-FANCM interaction was initially screened with a subset of 49,796 compounds from the same LC library as PriA-SSB.[17] This FP assay was run at a single compound concentration (32 $\mu$M). We refer to these continuous values as "RMI-FANCM % inhibition". Those compounds that demonstrated activity ≥ 2 standard deviations (SD) above the assay mean and passed PAINS filters were marked as actives in the binary data set RMI-FANCM FP.

*2.1.4. PriA-SSB Prospective.* For prospective testing, we experimentally screened an additional 22,434 compounds after the VS methods predicted their activity. We removed compounds that were already included in the 72,423 LC compounds in the PriA-SSB AS data set to ensure there was no overlap between the prospective screen compounds and those used to train VS models. As with the initial library, the PriA-SSB AS assay was used in the same 1536-well format at a single concentration (33.3 $\mu$M) to test the additional 22,434 LC compounds. Actives were defined with the same criteria used for the binary data set PriA-SSB AS. Compounds with at least 35% inhibition that passed the PAINS filters were retested with the AS assay. Those with at least 35% inhibition in the AS retest were labeled as actives, creating the binary data set PriA-SSB prospective.

Because secondary screens and structural filters were used to define the active compounds, there was no single primary screen % inhibition threshold that separated the actives from the inactives. Some compounds exhibiting high % inhibition values were labeled as inactive because they did not satisfy the structural requirements or were not active in the secondary screen.

*2.1.5. PubChem BioAssay.* To help learn a better chemical representation with multi-task neural networks, we considered other screening contexts from which to transfer useful knowledge. We used a subset of 128 assays (AIDs) from the PubChem BioAssay (PCBA)[24] repository. This data set was used in previous work on multi-task neural networks.[14] This subset contained assays for which the assays were developed to probe a specific protein target and dose−response measurements were obtained for each compound (see Part A in the Supporting Information for other assay query filters). Potency and curve quality are factored into a PubChem Activity Score. Regardless of the assay, compounds with a PubChem Activity Score of 40 or greater (range 0−100) were assigned a PubChem Bioactivity outcome (label) of "Active". Compounds with PubChem Activity Scores of 1−39 were labeled "Inconclusive", and those with 0 were labeled "Inactive" (Parts B and C, Supporting Information).

### 2.2. Compound Features.

Ligand-based virtual screening methods require each chemical compound to be represented in a particular format as input to the model. We adopted two common representations. All of the ligand-based algorithms except the Long Short-Term Memory (LSTM) neural network used 1024-bit Morgan fingerprints[25] with radius 2 generated with RDKit version 2016.03.4.[26] These circular fingerprints are similar to ECFP4 fingerprints,[27] though with a slightly different implementation. For LSTM networks, we used the Simplified Molecular Input Line Entry System (SMILES) representation,[28] where the characters were treated as sequential features.

**2.3. Virtual Screening Models.** We selected a variety of existing virtual screening approaches for our benchmarks and prospective predictions. These included ligand-based supervised machine learning approaches, structure-based docking, and a chemical similarity baseline. Table 2 summarizes the types of training data used by each algorithm.

**Table 2. Summary of Virtual Screening Methods and Which Labels Each Model Used during Training**[a]

| Model | Continuous % inhibition | Binary label | PCBA binary labels |
|---|---|---|---|
| Dock | | | |
| CD | | | |
| STNN-C | | √ | |
| STNN-R | √ | | |
| MTNN-C | | √ | √ |
| LSTM | | √ | |
| IRV | | √ | |
| RF | | √ | |
| Similarity baseline | | √ | |

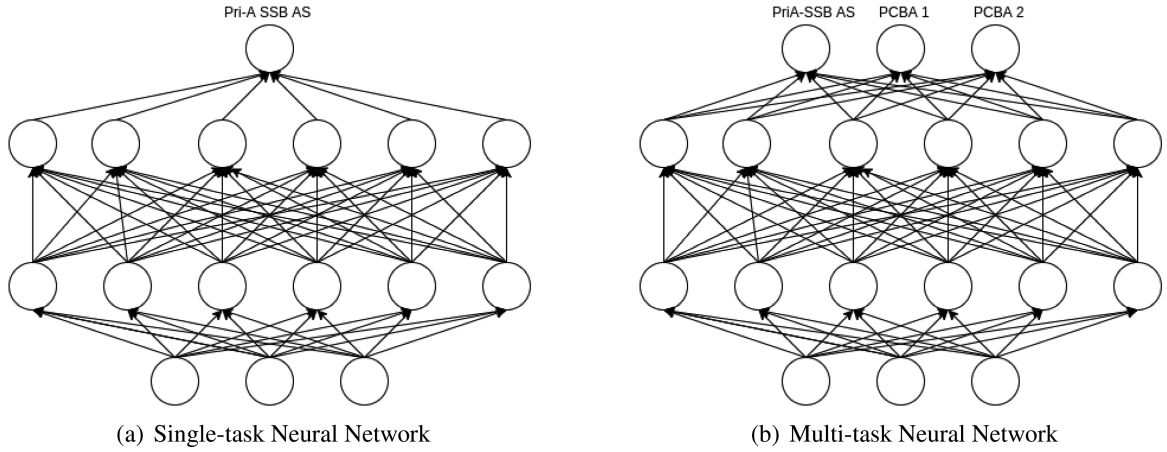
[a] The docking and consensus docking models do not train on the PriA-SSB or RMI-FANCM data sets.

*2.3.1. Ligand-Based Neural Networks.* Deep learning is a machine learning approach that encompasses neural network models with multiple hidden layer architectures and the techniques for training these models. It represents the state of the art for many predictive tasks, which has generated extensive interest in deep learning for biomedical research, including virtual screening.[15,16] We evaluated multiple types of established neural network architectures for virtual screening.

*2.3.1.1. Single-Task Neural Network (STNN).* A single-task neural network (Figure 1(a)) makes a single prediction for a single target (also referred to as a task). We trained a separate model for each of the PriA-SSB AS, PriA-SSB FP, and RMI-FANCM FP data sets, taking each compound's Morgan fingerprint as the input features. We trained the neural networks using Keras[29] with the Theano backend.[30] The single-task neural networks were trained on each task to predict either the binary activity label in the classification setting (STNN-C) or the continuous % inhibition in the regression setting (STNN-R). Because the STNN-R models were trained directly on the % inhibition, they do not depend on the PAINS filters. These neural networks used two hidden layers with 2000 hidden units each, Adam optimization,[31] 0.25 dropout rate, and other hyperparameters described in Tables S2 and S3.

*2.3.1.2. Multi-Task Neural Network (MTNN).* Multi-task neural networks make different predictions for multiple targets or tasks but share knowledge by training the first few hidden layers together. Each of our multi-task neural networks included one target task (PriA-SSB AS, PriA-SSB FP, or RMI-FANCM FP) and 128 tasks from PCBA. We only trained multi-task neural networks in the classification setting (MTNN-C). The MTNN-C models used two hidden layers with 2000 hidden units each, Adam optimization, 0.25 dropout rate, and other hyperparameters described in Table S2.

*2.3.1.3. Single-Task Atom-Level LSTM (LSTM).* The LSTM is one of most prevalent recurrent neural network models,[32] which has been applied previously in virtual screening.[33] An LSTM assumes there exists a sequential pattern in the input string. We used a one hot encoding of the SMILES strings as input for the LSTM model. In a one hot encoding, each character in a SMILES string is replaced by a binary vector. The binary vector has one bit for each possible unique character in all SMILES strings. At each position in a SMILES string, the bit corresponding to the character at that position is set to 1, and all other bits are set to 0. We trained the LSTM model to predict the binary activity labels. The LSTM models used one or two hidden layers with 10 to 100 hidden units each, Adam optimization, 0.2 or 0.5 dropout rate, and other hyperparameters described in Table S4. The compounds in the cross-validation stage used SMILES generated by OpenEye Babel version 3.3. The compounds in the prospective screen were processed separately and used SMILES from RDKit version 2016.03.4.[26]



(a) Single-task Neural Network

(b) Multi-task Neural Network

**Figure 1.** Neural network structures. The neural networks map the input features (e.g., fingerprints) in the input (bottom) layer to intermediate chemical representations in the hidden (middle) layers and finally to the output (top) layer, which makes either continuous or binary predictions. Panel (a) has only one unit in the output layer. Panel (b) has multiple units in the output layer representing different targets, one for our new target of interest and the others for PCBA targets.

DOI: 10.1021/acs.jcim.8b00363
J. Chem. Inf. Model. 2019, 59, 282−293

*2.3.1.4. Influence Relevance Voter (IRV).* IRV[34,35] is a hybrid between *k*-nearest neighbors and neural networks. Each compound's predicted value is a nonlinear combination of the similarity scores from its most closely related compounds in the training data set. We used Morgan fingerprints as the input and trained separate IRV models for each data set. The IRV models used 5 to 80 neighbors and other hyperparameters described in Table S5.

*2.3.2. Ligand-Based Random Forest (RF).* Random forests[36] are ensembles of decision trees that are often used as a baseline in virtual screening benchmarks.[37,38] We used scikit-learn[39] to train a random forest classifier for each binary label with Morgan fingerprints as features. The RF models used 4000 to 16,000 estimators, 1 to 1000 minimum samples at a leaf node, a bound on the maximum number of features, and other hyperparameters described in Table S6.

*2.3.3. Protein–Ligand Docking. 2.3.3.1. Target Preparation.* Our structure-based VS approach involved the docking-based ranking of the LC library to the holo-form of PriA using the crystal structure (PDB: 4NL8),[40] in which it is bound to a C-terminal segment of an SSB protein. A missing loop in this structure was added from the apo-form (PBD: 4NL4), though this is not near the SSB binding site. The docking search space was limited to 8 Å from the coordinates of the cocrystallized SSB C-terminal tripeptide.

For RMI-FANCM, the RMI protein was built from both the A and B chains from the structure (PDB: 4DAY).[41] The docking search space was defined by the central five residues of the MM2 peptide (PDB: 4DAY chain C), Val-Thr-Phe-Asp-Leu, also with an 8 Å bounding box.

*2.3.3.2. Compound Preparation.* LC library compounds were assigned 3D coordinates and Merck Molecular Force Field partial charges using OpenEye OMEGA and Molcharge.[42] Compounds in the LC library with ambiguous stereochemistry were enumerated in all possibilities, and the best resulting docking score was retained for each.

*2.3.3.3. Docking (Dock) and Consensus Docking (CD).* We ran eight different docking programs and generated nine docking scores as a broad comparison to the ligand-based methods under consideration. The docking programs and names we use for their scores are AutoDock version 4.2.6[43] (Dock_ad4), Dock version 6.7[44] (Dock_dock6), FRED version 3.0.1[45] (Dock_fred), HYBRID version 3.0.1[45] (Dock_hybrid), PLANTS version 1.2[46] (Dock_plants), rDock version 2013.1[47] (Dock_rdocktot and Dock_rdockint), Smina version 1.1.2[48] (Dock_smina), and Surflex-Dock version 3.040[49] (Dock_surflex). In addition, we calculated consensus docking scores using three traditional approaches (CD_mean, CD_median, and CD_max) and two versions of the Boosting Consensus Score (CD_efr1_opt and CD_rocauc_opt).[50] The consensus docking methods were developed without any knowledge of the PriA-SSB or RMI-FANCM assay data. Compounds with missing scores due to preparation or docking failures were not considered during evaluation.

*2.3.4. Chemical Similarity Baseline.* We introduced a compound ranking method based on chemical structure similarity to serve as a baseline for the ligand-based VS methods. The active compounds in the training set were used as seeds for similarity searching through all test set compounds. The test set compounds were ranked by their maximum Tanimoto similarity to any of the training set actives with MayaChemTools[51] using Morgan fingerprints from RDKit version 2013.09.1. Unlike the ligand-based machine learning algorithms, the similarity baseline does not consider inactive compounds in the training set.

In addition, all compounds were clustered by two separate approaches to describe chemical series. Chemical similarity-based hierarchical clusters on Morgan fingerprints using Ward's clustering are described as SIM. Maximum common substructure clusters, used to group molecules with similar scaffolds, are described as MCS. JKlustor was used for both types of clustering (JChem version 17.26.0, ChemAxon).

**2.4. Evaluation Metrics.** Given our goal of developing VS methods that enable very small, cost-effective, productive screens, we considered how evaluation metrics weight early active retrieval. All of the VS algorithms produce a ranked list of compounds, where compounds are ordered by the probability of being active, the continuous predicted % inhibition, the docking score, or a comparable output value. For a ranked list of compounds, we can threshold the ranked list and consider all compounds above the threshold as positive (active) predictions and those below the threshold as negative (inactive). Classification models output class probabilities. Regression models, docking, and the similarity baseline output different types of continuous scores. Thresholding on the compound rank is equivalent to thresholding on the class probability or continuous score because for each rank there is a corresponding probability or score. By comparing those predictions to the experimentally observed activity, we can compute true positive (TP), false positive (FP), true negative (TN), and false negative (FN) predictions for the ranked list at that threshold. We explored several options for summarizing how well each algorithm ranks the known active compounds. Because most of the compounds have only single-replicate measurements of % inhibition, we focus on evaluating active versus inactive compounds instead of correlation with the % inhibition.

The area under the receiver operating characteristic curve (AUC[ROC]) has been recommended for virtual screening because it is robust, interpretable, and does not depend on user-defined parameters.[52] The ROC curve plots the relationship between true positive rate (TPR, also known as sensitivity or recall) and false positive rate (FPR, equivalent to 1 − specificity), which are defined in eq 1. As the FPR goes to 100%, all ROC curves converge, whereas early active retrieval (a more meaningful characteristic of VS performance) can be assessed in the low FPR region of the ROC curve, which exhibits greater variability across VS methods. Thus, we also considered the Boltzmann-enhanced discrimination of receiver operating characteristic (BEDROC).[53] It emphasizes the early part of the ROC curve through a scaling function $\alpha$, which we set to 10 for our purposes of early enrichment up to 20%. We used the BEDROC implementation from the CROC Python package.[54]

$$\mathrm{TPR} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \quad \mathrm{FPR} = \frac{\mathrm{FP}}{\mathrm{FP} + \mathrm{TN}} \tag{1}$$

$$\mathrm{Recall} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FN}}, \quad \mathrm{Precision} = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}} \tag{2}$$

Area under the precision-recall curve (AUC[PR]) is another common metric (eq 2). AUC[PR] has an advantage over AUC[ROC] for summarizing classifier performance when the class labels are highly skewed, as in virtual screening where there are few active compounds in a typical library. AUC[PR] evaluates a classifier's ability to retrieve actives (recall) and

DOI: 10.1021/acs.jcim.8b00363

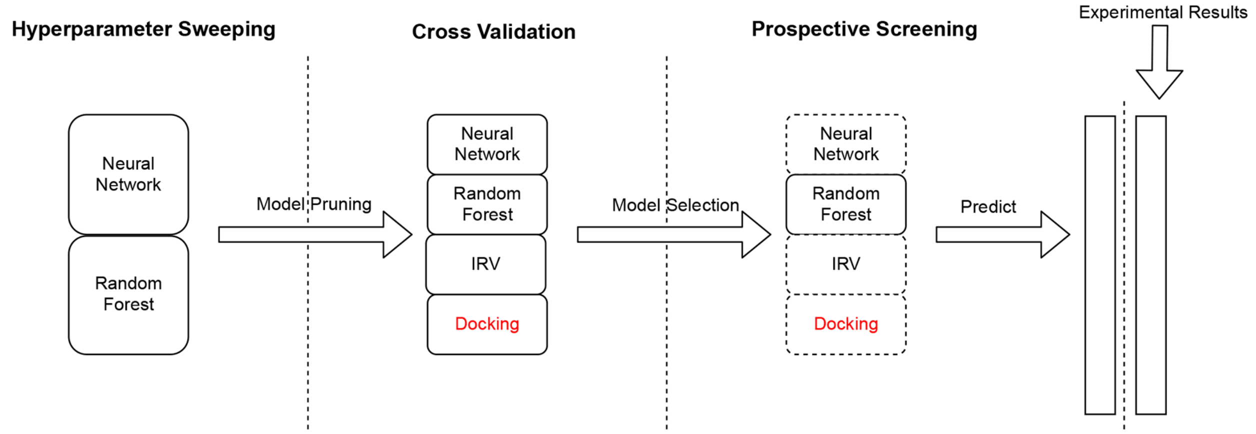


**Figure 2.** Initially, 258 neural network and random forest models were evaluated to eliminate poorly performing hyperparameter combinations. The models with the best hyperparameters advanced to cross-validation along with IRV and docking-based methods for a total of 35 models. Cross-validation identified a random forest as the best overall model. The VS methods and similarity baseline then predicted active compounds in the PriA-SSB prospective data set. After the predictions were finalized, we experimentally screened the compounds to evaluate the predictions. Black text denotes ligand-based machine learning models. Red text denotes docking-based models, which did not train on the target-specific HTS data.

which of the predicted actives are correctly classified (precision) as the prediction threshold varies. We used the PRROC R package's "auc.integral"[55] to compute AUC[PR].

Another VS metric is enrichment factor (EF), which is the ratio between the number of actives found in a prioritized subset of compounds versus the expected number of actives in a random subset of same size. In other words, it assesses how much better the VS method performs over random compound selection. Let $R \in [0\%,100\%]$ be a predefined fraction of the compounds from the total library of compounds screened.

$$\mathrm{EF}_R = \frac{\#\,\text{active in top } R \text{ ranked compounds}}{\#\,\text{actives in entire library} \times R} \tag{3}$$

$$\mathrm{EF}_{\max,R} = \frac{\min\{\#\,\text{actives, total}\,\#\,\text{compounds} \ \times R\}}{\#\,\text{actives in entire library} \times R} \tag{4}$$

$\mathrm{EF}_{\max,R}$ represents the maximum enrichment factor possible at $R$. Difficulty arises when interpreting EF scores because they vary with the data set and threshold $R$. We defined the normalized enrichment factor (NEF) as

$$\mathrm{NEF}_R = \frac{\mathrm{EF}_R}{\mathrm{EF}_{\max,R}} \tag{5}$$

Because $\mathrm{NEF}_R \in [0,1]$, it is easier to compare performance across data sets and thresholds. Here, 1.0 is the perfect NEF. Furthermore, we can create an NEF curve as $\mathrm{NEF}_R$ versus $R \in [0\%,100\%]$ and compute the area under that curve to obtain AUC[NEF] $\in [0,1]$. However, most models tend to exhibit similar late enrichment behavior. We are typically interested in early enrichment behavior so we computed AUC[NEF] using $R \in [0\%,20\%]$.

Finally, we considered the metric $n_{\text{hits}}$, which is simply the number of actives found in a selected number of tested compounds (e.g., how many hits or actives were found in 250 tested compounds). This metric represents the typical desired utility of a screening process: retrieve as many actives as possible in the selected number of tested compounds (denoted as $n_{\text{tests}}$). We compared $n_{\text{hits}}$ at various $n_{\text{tests}}$ to the different evaluation metrics to identify which metrics best mimic the $n_{\text{hits}}$ utility.

## 2.5. Pipeline.

Our virtual screening workflow contains three stages: (1) Tune hyperparameters in order to prune the model search space. (2) Train, evaluate, and compare models with cross-validation to select the best models. (3) Assess the best models' ability to prospectively identify active compounds in a new set.

In contrast to most other virtual screening studies, the experimental screen was not conducted until after all models were trained and evaluated in the cross-validation stage (Figure 2). For the first two stages, we first split the PriA-SSB AS, PriA-SSB FP, and RMI-FANCM FP data sets into five stratified folds as described in Part B in the Supporting Information.

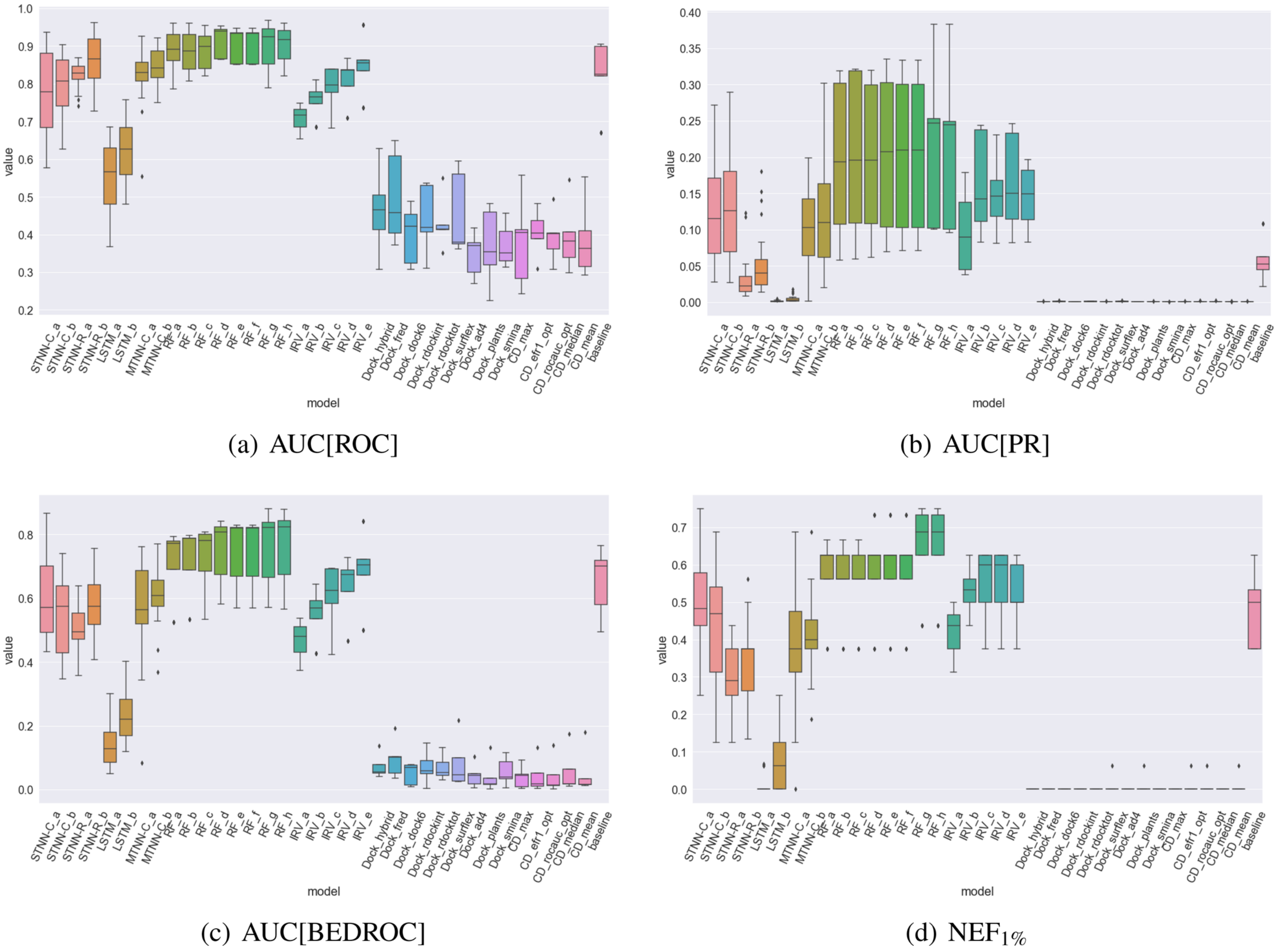
### 2.5.1. Hyperparameter Sweeping Stage.

Hyperparameters are model configurations or settings that are set by an expert as opposed to the weights or parameters that are learned or fit during model training. For most of the ligand-based machine learning models, the hyperparameter space was too large for exhaustive searches using the full data set. Therefore, we applied a grid search on a predefined set of hyperparameters in a smaller data set and pruned those that performed poorly. We performed a single iteration of training on the first four folds of PriA-SSB AS to avoid overfitting. The hyperparameters considered are listed in Part D in the Supporting Information.

### 2.5.2. Cross-Validation Stage.

To identify which VS algorithms are likely to have the best performance in a prospective screen, we applied a traditional cross-validation training strategy on data sets PriA-SSB AS, PriA-SSB FP, and RMI-FANCM FP after reducing the hyperparameter combinations to consider. Selecting the best model is nontrivial. Ideally, the best model would have dominant performance on all evaluation metrics, but this is rarely observed with existing models. Each evaluation metric prioritizes different performance characteristics. Our cross-validation results illustrate which models consistently perform well over different metrics, the correspondence of metrics relative to a desired utility ($n_{\text{hits}}$), and how to choose models and evaluation metrics in order to successfully identify active compounds in a prospective screen.

Cross-validation is commonly used to avoid overfitting when there are few training samples. We split the training data into five folds: four folds for training and one for testing. Models like RF and IRV that do not require a hold-out data set for



(a) AUC[ROC]

(b) AUC[PR]

(c) AUC[BEDROC]

(d) $NEF_{1\%}$

**Figure 3.** Evaluation metric distributions on PriA-SSB AS over the cross-validation folds. The metrics are (a) AUC[ROC], (b) AUC[PR], (c) AUC[BEDROC], and (d) $NEF_{1\%}$ as described in Section 2.4. Unlike the ligand-based models, the docking methods do not train on the PriA-SSB AS training folds and are applied directly to the test fold during cross-validation (see Section 4).

early stopping used four folds for training. The neural networks perform early stopping based on a hold-out set, so we iteratively selected one of the four training data folds for this purpose. This led to a nested cross-validation with 5 × 4 = 20 trained neural networks.

*2.5.3. Prospective Screening Stage.* Our prospective screen used a library of 22,434 new LC compounds that were not present in the PriA-SSB AS training set. We used each VS model to prioritize 250 of these compounds that are most likely to be active. This emulates virtual screening on much larger compound libraries, in which only a small fraction of all computationally scored compounds can be tested experimentally. When models assigned the same score to multiple compounds, we broke ties arbitrarily to obtain exactly 250 compounds.

After finalizing the models' predictions, we screened all 22,434 compounds in the wet lab and assigned actives based on a 35% inhibition threshold and structural filters (the PriA-SSB prospective data set). Finally, we evaluated how many of the experimental actives each VS method identified in its top 250 predictions, the number of distinct chemical clusters recovered, and the number of active compounds that were not in the top 250 predictions from any of the VS algorithms. The prospective screen allowed us to assess how well the cross-validation results generalized to new compounds and further verified our conclusions from the retrospective cross-validation tests.

**2.6. Data and Software Availability.** Code implementing our ligand-based virtual screening algorithms is available at https://github.com/gitter-lab/pria_lifechem and archived on Zenodo (DOI: 10.5281/zenodo.1257673). This GitHub repository also contains additional Jupyter notebooks to reproduce the visualizations and analyses. Our new PriA-SSB HTS data are available on PubChem (PubChem AID:1272365) along with the existing RMI-FANCM HTS data (PubChem AID:1159607). A formatted version of this data set for training virtual screening algorithms is available on Zenodo (DOI: 10.5281/zenodo.1257462).

## 3. RESULTS

**3.1. Cross-Validation Results.** In the cross-validation stage, we assessed 35 models: eight neural networks (STNN-C (Table S7), STNN-R (Table S8), MTNN-C (Table S9), and LSTM (Table S10)), five IRV (Table S11), eight RF (Table S12), and 14 from docking (Dock) or consensus docking (CD). When there are multiple versions of a model that use different hyperparameters, we distinguish them with alphabetic suffixes such as "_a" and "_b". Tables S7−S12 describe the

Table 3. Top-Ranked Models by Means versus DTK+Mean on the Three Tasks. Evaluation metric means were computed over all cross-validation folds[a]

| | Best by Mean Model | | | Best by DTK+Mean Model | | |
|---|---|---|---|---|---|---|
| Metric | PriA-SSB AS | PriA-SSB FP | RMI-FANCM FP | PriA-SSB AS | PriA-SSB FP | RMI-FANCM FP |
| AUC[ROC] | RF_d | STNN-R_a | RF_h | RF_d | STNN-R_a | RF_h |
| AUC[BEDROC] | RF_h | STNN-R_b | RF_h | RF_h | STNN-R_b | RF_h |
| AUC[PR] | RF_g | STNN-R_a | RF_h | STNN-C_b | STNN-R_b | STNN-C_b |
| AUC[NEF] | RF_h | STNN-R_b | RF_h | RF_h | STNN-R_b | RF_h |
| $NEF_{1\%}$ | RF_h | STNN-R_b | RF_h | RF_h | STNN-R_b | RF_h |

[a] The prospective screening was only performed on PriA-SSB. Model names are mapped to their hyperparameter values in Part E of the Supporting Information.

hyperparameters associated with these suffixes. We highlight the PriA-SSB AS data set as a representative example, but the VS workflow is applicable for all tasks.

#### 3.1.1. Comparing Virtual Screening Algorithms.

We tested all 35 models on three data sets, and the results for four evaluation metrics on the PriA-SSB AS data set are shown in Figure 3. Part F of the Supporting Information contains the results for PriA-SSB FP and RMI-FANCM FP. The PriA-SSB AS performance using AUC[ROC] was comparable for many models. All models except LSTM, some IRV models, and docking were above 0.8 AUC[ROC]. Some of the other evaluation metrics better stratify the ligand-based VS methods. Random forest was the best model, especially for the most-relevant metrics that prioritize early enrichment. We also ran the chemical similarity-based method for PriA-SSB AS and confirmed that random forest outperformed this simple baseline.

Random forest was again the best overall method for the RMI-FANCM FP data set (Part F, Supporting Information). On the PriA-SSB FP data set, STNN-R achieved the highest scores over the majority of the metrics (Part F, Supporting Information). The other types of VS models were effectively tied for most metrics (Part G, Supporting Information).

#### 3.1.2. Evaluation Metrics.

Given a fixed evaluation metric, we could compare two models with a $t$ test to assess if one statistically outperforms the other. However, we needed to make such comparisons repeatedly between each pair of models and required a statistical test that accounts for multiple hypothesis testing. Due to unequal variances and sample sizes (Figure 3), we used Dunnett's modified Tukey–Kramer test (DTK)[56,57] for pairwise comparison to assess whether the mean metric scores of two models were significantly different. Using DTK results for *each metric*, we scored each model based on how many times it attained a statistically significantly better result than other models (Part G, Supporting Information). For most metric-target pairs, many models have the same rank because DTK does not report a significant difference.

In a prospective screen, our goal is to maximize the number of active compounds identified by a VS algorithm given a fixed budget (number of predictions). We wanted to determine which of the VS evaluation metrics best aligns with $n_{\text{hits}}$. Thus, we compared the model ranking induced by each metric with the model ranking induced by $n_{\text{hits}}$ for a varying number of tests.

To score the evaluation metrics, we used Spearman's rank correlation coefficient based on the model rankings induced by the metric of concern versus $n_{\text{hits}}$ at a specific $n_{\text{tests}}$. We then ranked the metrics based on their correlation with $n_{\text{hits}}$ (Part H, Supporting Information). The metric ranking varies depending on $n_{\text{tests}}$ and the target. Some metrics overtake one another as we increase $n_{\text{tests}}$. For PriA-SSB AS, $NEF_R$ consistently placed in the top ranking correlations when $R$ coincided with $n_{\text{tests}}$. This is evident when we focus on a single metric and see the top ranking metrics for $n_{\text{tests}} \in [100,250,500,1000,2500]$. Only for a large enough $n_{\text{tests}}$ do metrics like AUC[ROC] that evaluate the complete ranked list become comparable. This suggests that if we know *a priori* how many new compounds we can afford to screen, then $NEF_R$ at a suitable $R$ is a viable metric for choosing a VS algorithm during cross-validation in the hopes of maximizing $n_{\text{hits}}$.

#### 3.1.3. Selecting the Best Model.

Based on these results, we selected the VS screening models that are most likely to generalize to new compounds and identify actives in our experimental screen of 22,434 new compounds. We focused on PriA-SSB for the prospective screen using models trained on PriA-SSB AS because the assay was more readily available for us to generate data for the new compounds.

Table 3 compares model selection based on evaluation metric means alone versus the DTK+Mean approach for multiple evaluation metrics on the three tasks. The complete model rankings for means only and DTK+Mean can be found in Part I in the Supporting Information. DTK+Mean ranks models by statistical significance and uses the mean value only for tie-breaking. Both strategies selected the same models for a fixed evaluation metric, except for AUC[PR] on all three tasks (Table 3). This is mainly due to DTK not detecting statistically significant differences among the models' evaluation scores, so tie-breaking by means selected the same models as ranking by means. Recall that PriA-SSB FP has fewer actives than PriA-SSB AS and RMI-FANCM FP (Table 1). Similar RF and STNN-C models were selected for PriA-SSB AS and RMI-FANCM FP. However, PriA-SSB FP prioritized STNN-R models exclusively.

In our prospective screen, each model prioritizes 250 top-ranked compounds, approximately 1% of the new LC library. In this setting where each model has a fixed budget for the predicted compounds, $NEF_R$ is a suitable metric. Therefore, we used $NEF_{1\%}$ with DTK+Means to choose the best models from each class. The best-in-class models were RandomForest_h, SingleClassification_a, SingleRegression_b, MultiClassification_b, LSTM_b, IRV_d, and ConsensusDocking_efr1_opt, with RandomForest_h being the strongest model overall (Part I, Supporting Information).

### 3.2. Prospective Screening Results.

After selecting the best model from each class based on cross-validation and the $NEF_{1\%}$ metric, we retrained the models on all 72,423 LC compounds to predict PriA-SSB inhibition using the same types of data shown in Table 2. This provided a single version of each model instead of one for each cross-validation fold. All models then ranked 22,434 new LC compounds that were provided without activity labels. We selected the top 250 ranked new compounds from each model. Then, we experimentally screened all 22,434 new compounds to assess PriA-SSB % inhibition and defined actives based on a 35% inhibition threshold and PAINS filters. The new binary data set PriA-SSB prospective contained 54 actives.

Table 4 presents how many of the 54 actives were identified by each best-in-class virtual screening method and the chemical structure similarity baseline. For context, randomly selecting 250 compounds from the PriA-SSB prospective data set is expected to identify less than one active based on the overall hit rate. Parts J and K of the Supporting Information show the VS models' PriA-SSB prospective performance for the other evaluation metrics.

**Table 4. Number of Active Compounds in the Top 250 Predictions from the Seven Selected Models and the Chemical Similarity Baseline Compared to the Number of Experimentally Identified Actives**[a]

| Model | Actives | Actives not in baseline | SIM clusters | MCS clusters |
|---|---|---|---|---|
| Experimental | 54 | – | 27 | 35 |
| Similarity baseline | 31 | – | 14 | 17 |
| CD_efr1_opt | 0 | 0 | 0 | 0 |
| STNN-C_a | 21 | 2 | 11 | 13 |
| STNN-R_b | 28 | 8 | 14 | 18 |
| LSTM_b | 1 | 1 | 1 | 1 |
| MTNN-C_b | 27 | 3 | 13 | 17 |
| RF_h | 37 | 7 | 17 | 22 |
| IRV_d | 29 | 4 | 15 | 18 |

[a] These selected models are the best in each algorithm category from cross-validation. The last two columns correspond to the number of distinct chemical clusters from similarity or maximum common substructure clustering that are represented among the 54 actives. The consensus docking model CD_efr1_opt ranks the PriA-SSB prospective compounds without using information from the PriA-SSB AS training data. Prospective performance for all VS models is in Part L of the Supporting Information.

Table 4 also lists the number of distinct chemical clusters identified by each method, with the goal of identifying as many diverse active compounds as possible. The 22,434 compounds form 124 SIM and 714 MCS clusters or chemical series. Of these, the 54 experimental actives represent 27 SIM and 35 MCS clusters. Commonly, virtual screening is followed by a medicinal chemistry effort that would be expected to identify other members of these clusters.

In general, the number of distinct chemical clusters captured in the top 250 predictions is correlated with the number of actives (Table 4), meaning that the methods selected structurally diverse hits. The similarity baseline identified compounds from roughly half of the SIM or MCS clusters. With the exception of docking, each of the methods in Table 4 found at least one cluster not present in the baseline. The machine learning techniques are not limited to finding only the chemotypes that are present in the training set (Part M, Supporting Information).

The ligand-based VS methods recovered many of the same actives as the chemical similarity baseline, but they also found actives that were missed by the baseline (Figure 4). There was a group of 11 active compounds that were identified by most ligand-based methods, including the baseline model. The compounds identified were not the most potent, either within their cluster or overall, nor did any of the methods exhibit any correspondence between the number of compounds identified from a cluster and their potency.

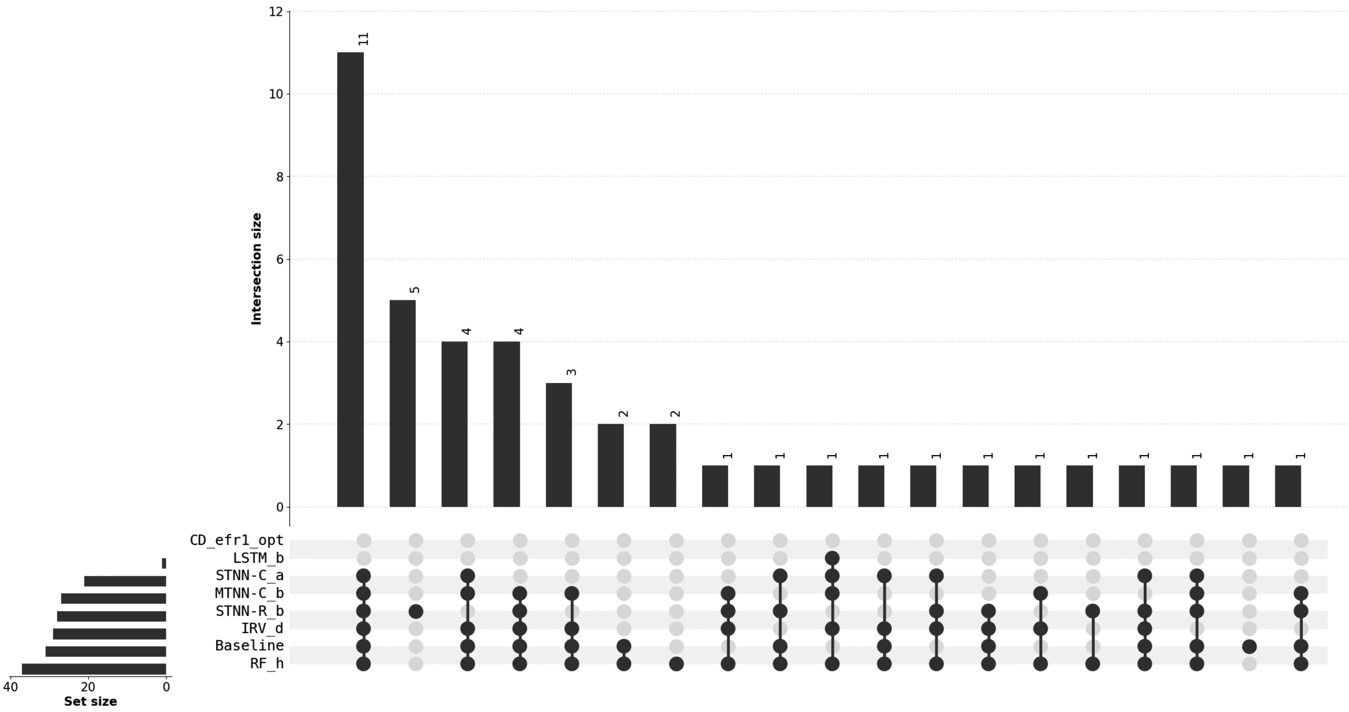


**Figure 4.** UpSet plot showing the overlap between the top 250 predictions from the selected VS models and the chemical similarity baseline on PriA-SSB prospective. The plot generalizes a Venn diagram by indicating the overlapping sets with dots on the bottom and the size of the overlaps with the bar graph.[58] Altogether, the combined predictions from the best-in-class VS methods and the baseline found 43 of the 54 actives.

The similarity baseline included one active compound that was excluded from the top 250 compounds from RF (Figure 4), but RF recovered a different member from this active compound's SIM cluster (Part M, Supporting Information). Only the RF model recovered more active compounds in its top 250 predictions than the chemical similarity baseline, including two unique actives not identified by any other model. Therefore, cross-validation with $NEF_{1\%}$ as the metric successfully identified the best PriA-SSB model before the prospective screen.

*3.2.1. Trained Models are Target Specific.* As a control, we retrained the best RF model on randomized data to confirm that its strong prospective performance was due to meaningful detected patterns among the active compounds instead of biases in the data set. Similar to y-scrambling or y-randomization,[59] we randomly permuted the binary activity labels in the PriA-SSB AS data set, retrained the RF_h model on the randomized data, and evaluated the classifier on the PriA-SSB prospective data set. This procedure was repeated 100 times with different y-scrambling performed each time. The number of active compounds in the top 250 predictions for these 100 runs is summarized in Figure S30. The mean number of actives was 0.83, and 55 of the runs found zero actives. The best y-scrambled run found only 10 actives, far less than the 37 actives when RF_h was trained on the real data.

In addition, we assessed the performance of all models trained on RMI-FANCM FP instead of PriA-SSB AS for making PriA-SSB prospective predictions on the new 22,434 compounds. As expected, the RMI-FANCM FP models perform poorly on PriA-SSB prospective (Table S29), indicating that the best PriA-SSB AS models have learned compound properties that are specific to PriA-SSB.

## 4. DISCUSSION

We followed a VS pipeline with the goal of maximizing the number of active compounds identified in a prospective screen with a limited number of predictions. From an initial pool of structure-based and ligand-based models, we pruned models in a hyperparameter search stage and conducted cross-validation with multiple evaluation metrics. We used DTK+Means with the $NEF_{1\%}$ metric to select the best models based on the cross-validation results and experimentally evaluated their top 250 prospective predictions from a new library of 22,434 compounds. The single best model from our pool, which was RandomForest_h for PriA-SSB AS, was also the top performing model on PriA-SSB prospective. Therefore, our overall pipeline successfully identified the best prospective model.

Metrics like AUC[ROC] can compare models in general, regardless of cost or other additional constraints.[52] However, for virtual screening in practice, one typically only experimentally tests a small fraction of all available compounds. In this setting, metrics like EF that capture early enrichment are preferable. In our prospective screen, STNN-R_a had higher AUC[ROC] than RF_h (Part J, Supporting Information), but the random forest found eight more active compounds in its top 250 predictions (Part L, Supporting Information). Our study suggests that $EF_R$, or its normalized version $NEF_R$, are the preferred metrics for identifying the best target-specific virtual screening method that maximizes $n_{\text{hits}}$ when there is a budget for experimental testing. Other metrics like AUC[ROC] or AUC[PR], which is more appropriate for problems where the inactive compounds far outnumber the actives,[60] may still be reasonable for benchmarking virtual screening methods on large existing data sets where the entire ranked list of compounds is evaluated.[38]

Some recent studies[3,37,61] reported that deep learning models substantially outperform traditional supervised learning approaches, including random forests. Our finding that a random forest model was the most accurate in both cross-validations, and our prospective screen does not refute those results. Rather, it reinforces that the ideal virtual screening method can depend on the training data available, target attributes, and other factors. Therefore, careful target-specific cross-validation is important to optimize prospective performance. One cannot assume that deep learning models will be dominant for all targets and all virtual screening scenarios. We also recommend hyperparameter exploration for all models, including traditional supervised learning methods. For example, our best random forest model contained 8000 estimators, whereas a previous benchmark considered at most 50 estimators.[3]

Ramsundar et al.[14] showed that performance improved in multi-task neural networks as they added more training compounds and tasks. Furthermore, the degree of improvement varied across the data sets and was moderately correlated with the number of shared active compounds among the targets within a single data set. Task-relatedness also affects the success of multi-task learning but is difficult to quantify.[62,63] We observed that PriA-SSB AS, PriA-SSB FP, and RMI-FANCM FP have no shared actives with any of the PCBA tasks, and multi-task neural networks were not substantially better than single-task neural networks in PriA-SSB AS cross-validation (Figure 3). The MTNN-C model outperformed the STNN-C model in the prospective evaluation (Table 4), possibly because multi-task learning can help prevent overfitting,[64] but was still considerably worse than the random forest. Multi-task random forests can also be constructed by using multi-task decision trees as the base learner.[65] However, these methods have not been used widely in the context of virtual screening.

We focused on well-established machine learning models instead of more recent deep learning models, such as graph-based neural networks.[38,66−69] This is because our main goal was to investigate the virtual screening principles for choosing the best model for a specific task (PriA-SSB AS) in a practical setting instead of broadly benchmarking virtual screening algorithms. In addition, a recent benchmark showed that conventional methods outperformed graph-based methods on most biophysics data sets.[38]

Consensus docking[50] failed to recover any actives in the PriA-SSB prospective data set, even though some of the individual docking programs did. Specifically for the PriA-SSB protein−protein interaction, docking is limited by the large, flat nature of the binding site. Many compounds that are inactive in the experimental screen have good scores and reasonable binding poses (per visual inspection) but fail to interrupt necessary specific interactions in the protein−protein interface. This will the limit overall performance by pushing true actives down the ranked list.

Our results are not intended to make general conclusions about the performance of ligand-based versus structure-based models. We use docking only for comparison to traditional structure-based VS methods and do not evaluate more sophisticated structure-based scoring functions. In addition, the individual docking and consensus docking methods do not train and optimize hyperparameters on the target-specific HTS screening data, whereas the ligand-based machine learning methods do. A more direct comparison would be to retrain a custom structure-based model or consensus scoring function to include the initial HTS data, though this effort is out of scope for this study. In addition, there are computational trade-offs between docking and ligand-based machine learning approaches. The machine learning models require substantial training time to select hyperparameters and fit models, but the trained models make predictions on new compounds very quickly. The docking programs take more time to score each new compound but have the advantage of not requiring training compounds.

The random forest model performed the best overall, but there were six active compounds identified by the other methods that the random forest missed (Figure 4). The single-task regression neural network recovered five of those six as well as unique active compound clusters (Part M, Supporting Information). In addition, this regression model performed the best on PriA-SSB FP during cross-validation (Table 3), possibly because there are fewer binary actives in this data set. In future work, we will explore whether ensembling classification and regression models, potentially in combination with structure-based VS algorithms, can further improve accuracy.

We emphasize our prospective performance on the new LC library, which minimizes the biases that make evaluation with retrospective benchmarks challenging.[70] There are many sources of experimental error in HTS, and the active compounds in the prospective evaluation must still be interpreted conservatively. However, a VS algorithm that can prioritize compounds with high % inhibition in primary and retest screens is valuable for further compound optimization even if not all of the actives confirm experimentally. Our study provides guidelines for selecting a target-specific VS model and complements other practical recommendations for VS pertaining to hit identification, validation, and filtering,[71] as well as avoiding common pitfalls.[72] Having established that our best virtual screening model successfully prioritized new active compounds in the LC library, another future direction will be to test prospective performance on much larger, more diverse chemical libraries.

## ■ ASSOCIATED CONTENT

### Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jcim.8b00363.

Supporting text, Figures S1−S30, and Tables S1−S30. (PDF)

## ■ AUTHOR INFORMATION

### Corresponding Author

*E-mail: gitter@biostat.wisc.edu.

### ORCID

Scott A. Wildman: 0000-0002-8598-0751
Anthony Gitter: 0000-0002-5324-9833

### Author Contributions

$^{\nabla}$S. Liu and M. Alnammi are co-first authors.

### Notes

The authors declare no competing financial interest.

## ■ ACKNOWLEDGMENTS

We acknowledge GPU hardware from NVIDIA, computing resources from the University of Wisconsin-Madison Center for High Throughput Computing, and funding from the Center for Predictive Computational Phenotyping NIH U54 AI117924, the University of Wisconsin Carbone Cancer Center Support Grant NIH P30 CA014520, and the Morgridge Institute for Research. Additional support for this research was provided by the University of Wisconsin-Madison Office of the Vice Chancellor for Research and Graduate Education with funding from the Wisconsin Alumni Research Foundation. We are grateful for the assistance and feedback from Chengpeng Wang, Haozhen Wu, and many members of the Center for High Throughput Computing and the Gitter lab. We thank Julio Lopes for alerting us about duplicate compounds in a preliminary version of the PriA-SSB prospective data set.

## ■ REFERENCES

(1) Cross, J. B.; Thompson, D. C.; Rai, B. K.; Baber, J. C.; Fan, K. Y.; Hu, Y.; Humblet, C. Comparison of Several Molecular Docking Programs: Pose Prediction and Virtual Screening Accuracy. *J. Chem. Inf. Model.* **2009**, *49*, 1455−1474.

(2) Lionta, E.; Spyrou, G.; Vassilatis, D.; Cournia, Z. Structure-based Virtual Screening for Drug Discovery: Principles, Applications and Recent Advances. *Curr. Top. Med. Chem.* **2014**, *14*, 1923−1938.

(3) Korotcov, A.; Tkachenko, V.; Russo, D. P.; Ekins, S. Comparison of Deep Learning With Multiple Machine Learning Methods and Metrics Using Diverse Drug Discovery Data Sets. *Mol. Pharmaceutics* **2017**, *14*, 4462−4475.

(4) Tseng, Y. J.; Hopfinger, A. J.; Esposito, E. X. The Great Descriptor Melting Pot: Mixing Descriptors for the Common Good of QSAR Models. *J. Comput.-Aided Mol. Des.* **2012**, *26*, 39−43.

(5) Hawkins, P. C.; Skillman, A. G.; Nicholls, A. Comparison of Shape-matching and Docking As Virtual Screening Tools. *J. Med. Chem.* **2007**, *50*, 74−82.

(6) Krüger, D. M.; Evers, A. Comparison of Structure-and Ligand-Based Virtual Screening Protocols Considering Hit List Complementarity and Enrichment Factors. *ChemMedChem* **2010**, *5*, 148−158.

(7) Venkatraman, V.; Pérez-Nueno, V. I.; Mavridis, L.; Ritchie, D. W. Comprehensive Comparison of Ligand-based Virtual Screening Tools against the DUD Data Set Reveals Limitations of Current 3d Methods. *J. Chem. Inf. Model.* **2010**, *50*, 2079−2093.

(8) Mitchell, J. B. O. Machine Learning Methods in Chemoinformatics. *Wiley Interdisciplinary Reviews. Computational Molecular Science* **2014**, *4*, 468−481.

(9) Merck. Merck Molecular Activity Challenge. https://www.kaggle.com/c/MerckActivity (accessed 2017−10−01).

(10) Dahl, G. E.; Jaitly, N.; Salakhutdinov, R. Multi-task Neural Networks for QSAR Predictions. *arXiv preprint arXiv:1406.1231*, 2014.

(11) Ma, J.; Sheridan, R. P.; Liaw, A.; Dahl, G. E.; Svetnik, V. Deep Neural Nets As a Method for Quantitative Structure−activity Relationships. *J. Chem. Inf. Model.* **2015**, *55*, 263−274.

(12) Mayr, A.; Klambauer, G.; Unterthiner, T.; Hochreiter, S. DeepTox: Toxicity Prediction Using Deep Learning. *Front. Environ. Sci.* **2016**, *3*, 80.

(13) Unterthiner, T.; Mayr, A.; Klambauer, G.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Hochreiter, S. Deep Learning as an Opportunity in Virtual Screening. *Deep Learning and Representation Learning Workshop: Neural Information Processing Systems* **2014**, *2014*, 27.

(14) Ramsundar, B.; Kearnes, S.; Riley, P.; Webster, D.; Konerding, D.; Pande, V. Massively Multitask Networks for Drug Discovery. *arXiv preprint arXiv:1502.02072*, 2015.

(15) Ching, T.; Himmelstein, D. S.; Beaulieu-Jones, B. K.; Kalinin, A. A.; Do, B. T.; Way, G. P.; Ferrero, E.; Agapow, P.-M.; Zietz, M.; Hoffman, M. M.; Xie, W.; Rosen, G. L.; Lengerich, B. J.; Israeli, J.; Lanchantin, J.; Woloszynek, S.; Carpenter, A. E.; Shrikumar, A.; Xu, J.; Cofer, E. M.; Lavender, C. A.; Turaga, S. C.; Alexandari, A. M.; Lu, Z.; Harris, D. J.; DeCaprio, D.; Qi, Y.; Kundaje, A.; Peng, Y.; Wiley, L. K.; Segler, M. H. S.; Boca, S. M.; Swamidass, S. J.; Huang, A.; Gitter, A.; Greene, C. S. Opportunities and Obstacles for Deep Learning in Biology and Medicine. *J. R. Soc., Interface* **2018**, *15*, 20170387.

(16) Chen, H.; Engkvist, O.; Wang, Y.; Olivecrona, M.; Blaschke, T. The Rise of Deep Learning in Drug Discovery. *Drug Discovery Today* **2018**, *23*, 1241−1250.

(17) Voter, A. F.; Manthei, K. A.; Keck, J. L. A High-throughput Screening Strategy to Identify Protein-protein Interaction Inhibitors That Block the Fanconi Anemia DNA Repair Pathway. *J. Biomol. Screening* **2016**, *21*, 626−633.

(18) Voter, A. F.; Killoran, M. P.; Ananiev, G. E.; Wildman, S. A.; Hoffmann, F. M.; Keck, J. L. A High-Throughput Screening Strategy to Identify Inhibitors of SSB Protein−Protein Interactions in an Academic Screening Facility. *SLAS DISCOVERY: Advancing Life Sciences R&D* **2018**, *23*, 94−101.

(19) Nordmann, P.; Cuzon, G.; Naas, T. The Real Threat of Klebsiella Pneumoniae Carbapenemase-producing Bacteria. *Lancet Infect. Dis.* **2009**, *9*, 228−236.

(20) Manthei, K. A.; Keck, J. L. The BLM Dissolvasome in DNA Replication and Repair. *Cell. Mol. Life Sci.* **2013**, *70*, 4067−4084.

(21) Baell, J. B.; Holloway, G. A. New Substructure Filters for Removal of Pan Assay Interference Compounds (PAINS) from Screening Libraries and for Their Exclusion in Bioassays. *J. Med. Chem.* **2010**, *53*, 2719−2740.

(22) Lagorce, D.; Sperandio, O.; Baell, J. B.; Miteva, M. A.; Villoutreix, B. O. FAF-Drugs3: a Web Server for Compound Property Calculation and Chemical Library Design. *Nucleic Acids Res.* **2015**, *43*, W200−W207.

(23) Capuzzi, S. J.; Muratov, E. N.; Tropsha, A. Phantom PAINS: Problems with the Utility of Alerts for Pan-Assay INterference CompoundS. *J. Chem. Inf. Model.* **2017**, *57*, 417−427.

(24) Wang, Y.; Bryant, S. H.; Cheng, T.; Wang, J.; Gindulyte, A.; Shoemaker, B. A.; Thiessen, P. A.; He, S.; Zhang, J. PubChem BioAssay: 2017 Update. *Nucleic Acids Res.* **2017**, *45*, D955−D963.

(25) Morgan, H. L. The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service. *J. Chem. Doc.* **1965**, *5*, 107−113.

(26) RDKit: Open-source Cheminformatics. http://www.rdkit.org (accessed 03/04/2016).

(27) Rogers, D.; Hahn, M. Extended-connectivity Fingerprints. *J. Chem. Inf. Model.* **2010**, *50*, 742−754.

(28) Weininger, D.; Weininger, A.; Weininger, J. L. SMILES. 2. Algorithm for Generation of Unique SMILES Notation. *J. Chem. Inf. Model.* **1989**, *29*, 97−101.

(29) Chollet, F. Keras. https://github.com/fchollet/keras (accessed 12/20/2016).

(30) The Theano Development Team. Al-Rfou, R.; Alain, G.; Almahairi, A.; Angermueller, C.; Bahdanau, D.; Ballas, N.; Bastien, F.; Bayer, J.; Belikov, A.; Belopolsky, A.; Bengio, Y.; Bergeron, A.; Bergstra, J.; Bisson, V.; Snyder, J. B.; Bouchard, N.; Boulanger-Lewandowski, N.; Bouthillier, X.; de Brébisson, A.; Breuleux, O.; Carrier, P.-L.; Cho, K.; Chorowski, J.; Christiano, P.; Cooijmans, T.; Côté, M.-A.; Côté, M.; Courville, A.; Dauphin, Y. N.; Delalleau, O.; Demouth, J.; Desjardins, G.; Dieleman, S.; Dinh, L.; Ducoffe, M.; Dumoulin, V.; Kahou, S. E.; Erhan, D.; Fan, Z.; Firat, O.; Germain, M.; Glorot, X.; Goodfellow, I.; Graham, M.; Gulcehre, C.; Hamel, P.; Harlouchet, I.; Heng, J.-P.; Hidasi, B.; Honari, S.; Jain, A.; Jean, S.; Jia, K.; Korobov, M.; Kulkarni, V.; Lamb, A.; Lamblin, P.; Larsen, E.; Laurent, C.; Lee, S.; Lefrancois, S.; Lemieux, S.; Léonard, N.; Lin, Z.; Livezey, J. A.; Lorenz, C.; Lowin, J.; Ma, Q.; Manzagol, P.-A.; Mastropietro, O.; McGibbon, R. T.; Memisevic, R.; van Merriënboer, B.; Michalski, V.; Mirza, M.; Orlandi, A.; Pal, C.; Pascanu, R.; Pezeshki, M.; Raffel, C.; Renshaw, D.; Rocklin, M.; Romero, A.; Roth, M.; Sadowski, P.; Salvatier, J.; Savard, F.; Schlüter, J.; Schulman, J.; Schwartz, G.; Serban, I. V.; Serdyuk, D.; Shabanian, S.; Simon, E.; Spieckermann, S.; Subramanyam, S. R.; Sygnowski, J.; Tanguay, J.; van Tulder, G.; Turian, J.; Urban, S.; Vincent, P.; Visin, F.; de Vries, H.; Warde-Farley, D.; Webb, D. J.; Willson, M.; Xu, K.; Xue, L.; Yao, L.; Zhang, S.; Zhang, Y. Theano: A Python Framework for Fast Computation of Mathematical Expressions. *arXiv preprint arXiv:1605.02688*, 2016.

(31) Kingma, D. P.; Ba, J. Adam: A Method for Stochastic Optimization. *arXiv preprint arXiv:1412.6980*, 2014.

(32) Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **1997**, *9*, 1735−1780.

(33) Jastrzębski, S.; Leśniak, D.; Czarnecki, W. M. Learning to SMILE(S). *arXiv preprint arXiv:1602.06289*, 2016.

(34) Swamidass, S. J.; Azencott, C.-A.; Lin, T.-W.; Gramajo, H.; Tsai, S.-C.; Baldi, P. Influence Relevance Voting: an Accurate and Interpretable Virtual High Throughput Screening Method. *J. Chem. Inf. Model.* **2009**, *49*, 756−766.

(35) Lusci, A.; Fooshee, D.; Browning, M.; Swamidass, J.; Baldi, P. Accurate and Efficient Target Prediction Using a Potency-sensitive Influence-relevance Voter. *J. Cheminf.* **2015**, *7*, 63.

(36) Breiman, L. Random Forests. *Machine Learning* **2001**, *45*, 5−32.

(37) Ramsundar, B.; Liu, B.; Wu, Z.; Verras, A.; Tudor, M.; Sheridan, R. P.; Pande, V. Is Multitask Deep Learning Practical for Pharma? *J. Chem. Inf. Model.* **2017**, *57*, 2068−2076.

(38) Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a Benchmark for Molecular Machine Learning. *Chemical Science* **2018**, *9*, 513.

(39) Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; Duchesnay, E. Scikit-learn: Machine Learning in Python. *J. Machine Learning Res.* **2011**, *12*, 2825−2830.

(40) Bhattacharyya, B.; George, N. P.; Thurmes, T. M.; Zhou, R.; Jani, N.; Wessel, S. R.; Sandler, S. J.; Ha, T.; Keck, J. L. Structural Mechanisms of PriA-mediated DNA Replication Restart. *Proc. Natl. Acad. Sci. U. S. A.* **2014**, *111*, 1373−1378.

(41) Hoadley, K. A.; Xue, Y.; Ling, C.; Takata, M.; Wang, W.; Keck, J. L. Defining the Molecular Interface That Connects the Fanconi Anemia Protein FANCM to the Bloom Syndrome Dissolvasome. *Proc. Natl. Acad. Sci. U. S. A.* **2012**, *109*, 4437−4442.

(42) Hawkins, P. C. D.; Skillman, A. G.; Warren, G. L.; Ellingson, B. A.; Stahl, M. T. Conformer Generation with OMEGA: Algorithm and Validation Using High Quality Structures from the Protein Databank and Cambridge Structural Database. *J. Chem. Inf. Model.* **2010**, *50*, 572−584.

(43) Morris, G. M.; Huey, R.; Lindstrom, W.; Sanner, M. F.; Belew, R. K.; Goodsell, D. S.; Olson, A. J. AutoDock4 and AutoDockTools4: Automated Docking with Selective Receptor Flexibility. *J. Comput. Chem.* **2009**, *30*, 2785−2791.

(44) Allen, W. J.; Balius, T. E.; Mukherjee, S.; Brozell, S. R.; Moustakas, D. T.; Lang, P. T.; Case, D. A.; Kuntz, I. D.; Rizzo, R. C. DOCK 6: Impact of New Features and Current Docking Performance. *J. Comput. Chem.* **2015**, *36*, 1132−1156.

(45) McGann, M. FRED and HYBRID Docking Performance on Standardized Datasets. *J. Comput.-Aided Mol. Des.* **2012**, *26*, 897−906.

(46) Korb, O.; Stützle, T.; Exner, T. E. Empirical Scoring Functions for Advanced Protein-Ligand Docking with PLANTS. *J. Chem. Inf. Model.* **2009**, *49*, 84−96.

(47) Ruiz-Carmona, S.; Alvarez-Garcia, D.; Foloppe, N.; Garmendia-Doval, A. B.; Juhos, S.; Schmidtke, P.; Barril, X.; Hubbard, R. E.; Morley, S. D. rDock: A Fast, Versatile and Open Source Program for Docking Ligands to Proteins and Nucleic Acids. *PLoS Comput. Biol.* **2014**, *10*, No. e1003571.

(48) Koes, D. R.; Baumgartner, M. P.; Camacho, C. J. Lessons Learned in Empirical Scoring with Smina from the CSAR 2011 Benchmarking Exercise. *J. Chem. Inf. Model.* **2013**, *53*, 1893−1904.

(49) Cleves, A. E.; Jain, A. N. Knowledge-guided Docking: Accurate Prospective Prediction of Bound Configurations of Novel Ligands Using Surflex-Dock. *J. Comput.-Aided Mol. Des.* **2015**, *29*, 485−509.

(50) Ericksen, S. S.; Wu, H.; Zhang, H.; Michael, L. A.; Newton, M. A.; Hoffmann, F. M.; Wildman, S. A. Machine Learning Consensus Scoring Improves Performance across Targets in Structure-based Virtual Screening. *J. Chem. Inf. Model.* **2017**, *57*, 1579−1590.

(51) Sud, M. MayaChemTools: An Open Source Package for Computational Drug Discovery. *J. Chem. Inf. Model.* **2016**, *56*, 2292−2297.

(52) Nicholls, A. What Do We Know and When Do We Know It? *J. Comput.-Aided Mol. Des.* **2008**, *22*, 239−255.

(53) Truchon, J.-F.; Bayly, C. I. Evaluating Virtual Screening Methods: Good and Bad Metrics for the "Early Recognition" Problem. *J. Chem. Inf. Model.* **2007**, *47*, 488−508.

(54) Swamidass, S. J.; Azencott, C.-A.; Daily, K.; Baldi, P. A CROC Stronger Than ROC: Measuring, Visualizing and Optimizing Early Retrieval. *Bioinformatics* **2010**, *26*, 1348−1356.

(55) Grau, J.; Grosse, I.; Keilwagen, J. Grosse I PRROC: Computing and Visualizing Precision-recall and Receiver Operating Characteristic Curves in R. *Bioinformatics* **2015**, *31*, 2595−2597.

(56) Lau, M. DTK: Dunnett-Tukey-Kramer Pairwise Multiple Comparison Test Adjusted for Unequal Variances and Unequal Sample Sizes. *R package*, 2013.

(57) Dunnett, C. W. Pairwise Multiple Comparisons in the Unequal Variance Case. *J. Am. Stat. Assoc.* **1980**, *75*, 796−800.

(58) Lex, A.; Gehlenborg, N.; Strobelt, H.; Vuillemot, R.; Pfister, H. UpSet: Visualization of Intersecting Sets. *IEEE Transactions on Visualization and Computer Graphics* **2014**, *20*, 1983−1992.

(59) Rücker, C.; Rücker, G.; Meringer, M. y-Randomization and Its Variants in QSPR/QSAR. *J. Chem. Inf. Model.* **2007**, *47*, 2345−2357.

(60) Davis, J.; Goadrich, M. The Relationship between Precision-Recall and ROC Curves. *Proceedings of the 23rd International Conference on Machine Learning.* **2006**, 233−240.

(61) Lenselink, E. B.; ten Dijke, N.; Bongers, B.; Papadatos, G.; van Vlijmen, H. W. T.; Kowalczyk, W.; IJzerman, A. P.; van Westen, G. J. P. Beyond the Hype: Deep Neural Networks Outperform Established Methods Using a ChEMBL Bioactivity Benchmark Set. *J. Cheminf.* **2017**, *9*, 45.

(62) Kearnes, S.; Goldman, B.; Pande, V. Modeling Industrial ADMET Data with Multitask Networks. *arXiv preprint arXiv:1606.08793*, 2016.

(63) Altae-Tran, H.; Ramsundar, B.; Pappu, A. S.; Pande, V. Low Data Drug Discovery with One-Shot Learning. *ACS Cent. Sci.* **2017**, *3*, 283−293.

(64) Ruder, S. An Overview of Multi-Task Learning in Deep Neural Networks. *arXiv preprint arXiv:1706.05098*, 2017.

(65) Caruana, R. Multitask Learning: A Knowledge-Based Source of Inductive Bias. Proceedings of the Tenth International Conference on Machine Learning. 1993; pp 41−48.

(66) Duvenaud, D. K.; Maclaurin, D.; Iparraguirre, J.; Bombarell, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional Networks on Graphs for Learning Molecular Fingerprints. *Advances in Neural Information Processing Systems* **2015**, 2224−2232.

(67) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular Graph Convolutions: Moving beyond Fingerprints. *J. Comput.-Aided Mol. Des.* **2016**, *30*, 595−608.

(68) Coley, C. W.; Barzilay, R.; Green, W. H.; Jaakkola, T. S.; Jensen, K. F. Convolutional Embedding of Attributed Molecular Graphs for Physical Property Prediction. *J. Chem. Inf. Model.* **2017**, *57*, 1757−1772.

(69) Matlock, M. K.; Dang, N. L.; Swamidass, S. J. Learning a Local-Variable Model of Aromatic and Conjugated Systems. *ACS Cent. Sci.* **2018**, *4*, 52−62.

(70) Wallach, I.; Heifets, A. Most Ligand-Based Classification Benchmarks Reward Memorization Rather Than Generalization. *J. Chem. Inf. Model.* **2018**, *58*, 916−932.

(71) Zhu, T.; Cao, S.; Su, P.-C.; Patel, R.; Shah, D.; Chokshi, H. B.; Szukala, R.; Johnson, M. E.; Hevener, K. E. Hit Identification and Optimization in Virtual Screening: Practical Recommendations Based on a Critical Literature Analysis. *J. Med. Chem.* **2013**, *56*, 6560−6572.

(72) Scior, T.; Bender, A.; Tresadern, G.; Medina-Franco, J. L.; Martínez-Mayorga, K.; Langer, T.; Cuanalo-Contreras, K.; Agrafiotis, D. K. Recognizing Pitfalls in Virtual Screening: A Critical Review. *J. Chem. Inf. Model.* **2012**, *52*, 867−881.