

Protein Family-Specific Models Using Deep Neural Networks and Transfer Learning Improve Virtual Screening and Highlight the Need for More Data

Fergus Imrie,[†] Anthony R. Bradley,^{‡,§} Mihaela van der Schaar,^{||,⊥} and Charlotte M. Deane^{*,†}

[†]Oxford Protein Informatics Group, Department of Statistics, University of Oxford, Oxford OX1 3LB, U.K.

[‡]Structural Genomics Consortium, University of Oxford, Oxford OX3 7DQ, U.K.

[§]Department of Chemistry, University of Oxford, Oxford OX1 3TA, U.K.

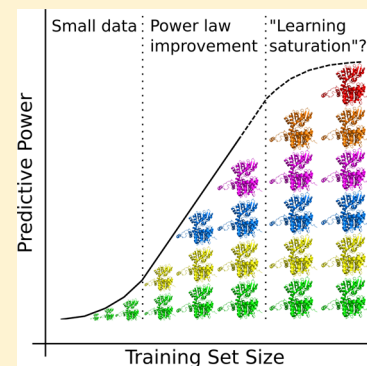
[§]Diamond Light Source Ltd., Didcot OX11 0DE, U.K.

^{||}Department of Engineering, University of Oxford, Oxford OX1 3PJ, U.K.

[⊥]Alan Turing Institute, London NW1 2DB, U.K.

Supporting Information

ABSTRACT: Machine learning has shown enormous potential for computer-aided drug discovery. Here we show how modern convolutional neural networks (CNNs) can be applied to structure-based virtual screening. We have coupled our densely connected CNN (DenseNet) with a transfer learning approach which we use to produce an ensemble of protein family-specific models. We conduct an in-depth empirical study and provide the first guidelines on the minimum requirements for adopting a protein family-specific model. Our method also highlights the need for additional data, even in data-rich protein families. Our approach outperforms recent benchmarks on the DUD-E data set and an independent test set constructed from the ChEMBL database. Using a clustered cross-validation on DUD-E, we achieve an average AUC ROC of 0.92 and a 0.5% ROC enrichment factor of 79. This represents an improvement in early enrichment of over 75% compared to a recent machine learning benchmark. Our results demonstrate that the continued improvements in machine learning architecture for computer vision apply to structure-based virtual screening.



INTRODUCTION

Drug discovery requires finding molecules that interact with targets with high affinity and specificity. While ultimately this is determined through experimental assays, computational techniques are frequently used to reduce the cost and improve the hit-rate of experimental verification. Successful applications of virtual screening in drug discovery processes are being increasingly reported;^{1–5} however, current methods still show relatively weak predictive power in many settings.^{6–8}

Traditional approaches have typically used experimental data to parametrize a physically inspired function.^{9–16} While interpretable, these techniques are inherently limited in their ability to capture complex interactions due to the use of rigid functional forms. Many machine learning-based scoring functions reuse the features of traditional approaches^{8,17} but exploit the greater flexibility in model structure to produce better representations of the same input data.¹⁸ However, this can lead to overfitting and often results in a loss of interpretability. In addition, the use of specific features, such as descriptors^{17,19} or fingerprints,²⁰ both biases the model to the choice of features and leads to an unnecessary loss of information through the elimination or approximation of the raw structural data. For these reasons, following the work of

Ragoza et al.,²¹ we have adopted an approach that minimizes initial featurization of input data.

Due to the importance of spatial configurations for physical interactions, determination of binding can be reframed as a computer vision problem. Ragoza et al. showed that a fully convolutional neural network (CNN), taking as input only spatial and atom type information, can outperform empirical and feature-based machine learning approaches at virtual screening,²¹ while Jiménez et al. exhibited state-of-the-art performance at binding affinity prediction using a similar approach.²²

Both of these methods were based on early CNN models used in the annual ImageNet Large Scale Visual Recognition Challenge (ILSVRC or ImageNet).²³ Since its introduction in 2010, this competition has been the source of many of the substantial advances not just in computer vision, but in machine learning more broadly. All entries in the first two years of the challenge had error rates of over 25% at the image classification task, while the winner of the 2017 edition, SENets,²⁴ achieved a 2.3% error rate. These advances have

Received: June 4, 2018

Published: October 1, 2018

been successfully applied to other areas outside of traditional computer vision tasks, such as medical imaging.^{25–27} However, limited use has been made in cheminformatics,²⁸ and none has been made that we are aware of for the study of protein–ligand interactions. We examine the applicability of modern CNNs to structure-based virtual screening by utilizing a densely connected convolutional neural network architecture (DenseNet).²⁹

A major challenge in virtual screening is the heterogeneity of binding between different targets arising from the structural diversity of proteins. Interactions of one target are not necessarily indicative of interactions of another target. However, proteins can be grouped into families, with proteins belonging to the same family having similar structures and physicochemical properties. As a result, it has been shown that in most cases a targeted scoring function will outperform a universal model.^{30,31} We investigate how we can improve predictions for a target using information from its protein family but not the target itself. This mimics the investigation of a novel target or one for which data is difficult to obtain.

Transfer learning is a general class of machine learning methods which improve performance on a new task by exploiting information that has already been learnt on a different, but often related, task. Finetuning is a transfer learning technique that allows a general model, trained on one data set, to be repurposed for a new task by retraining a part (or all) of the model on a data set specific to the new task. For example, Tajbakhsh et al. showed that finetuning a model for medical imaging that was originally trained on general image data outperformed training solely on the medical data, in particular when limited examples were available.³² We use this technique to create protein family-specific models and compare these to see if such a targeted scoring function outperforms a universal model.

In this work, we first show how recent advances in computer vision can be applied to structure-based virtual screening by adopting a CNN model based on the DenseNet architecture.²⁹ We then investigate the optimal number of poses to use as input. Finally, we detail how transfer learning can be used to construct models for specific protein families. We conduct an in-depth empirical study into the number of family members required to adopt family-specific models and, for the first time, present guidelines for the expected benefit from collecting additional data.

METHODS

To develop our virtual screening tool, we utilized a clustered cross-validation of our training set (DUD-E) to optimize our network and investigate other choices in the procedure. We then trained a final model on the full training set and evaluated our methods on two independent test sets (a subset of ChEMBL and MUV). These sets have been filtered to ensure that targets are sufficiently dissimilar from targets in the training set (see below).

Input Format. We generated ligand poses for all actives and decoys using AutoDock Vina,¹² specifically the smina¹³ implementation. Ligands were docked against a reference receptor within a box centered around a reference ligand with 8 Å of padding. We used smina's default arguments for exhaustiveness and sampling. We followed the approach described in Ragoza et al. and discretized the docked protein–ligand structures into a grid format to act as the input for the CNN. A schematic of the input featurization

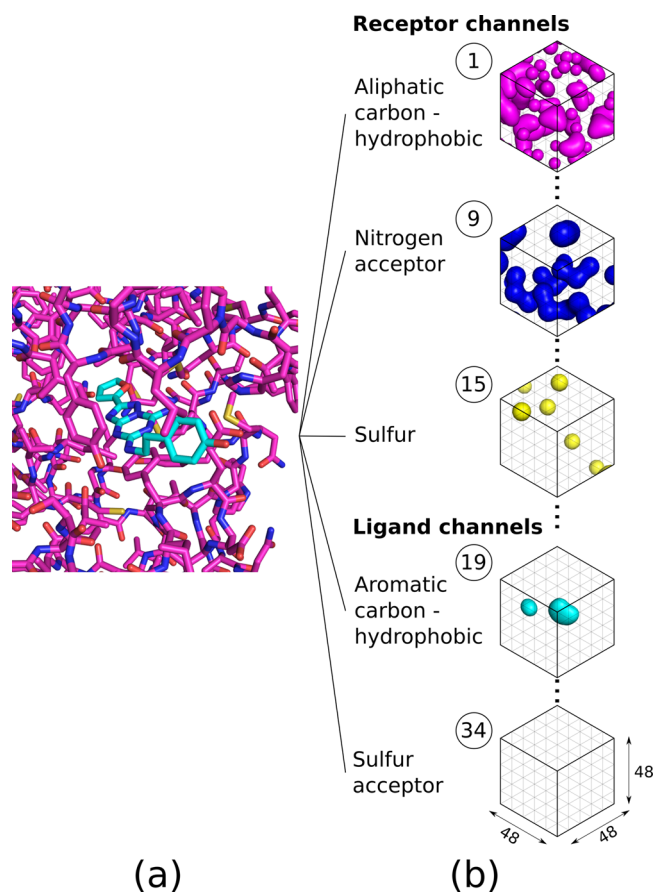


Figure 1. Input featurization for PDB ID 3EML (ligand ZMA). (a) The protein–ligand complex is cropped to a 24 \AA^3 box, centered on the ligand. The ligand is shown with carbons in cyan, the receptor with carbons in magenta, and the heteroatoms are colored with standard coloring. (b) The complex is decomposed into information channels, one for each atom type (Table 1), and divided into voxels with a resolution of 0.5 \AA . Atoms within each channel have a Gaussian representation. We show this for five of the 34 channels and visualize the final voxel grid as an isosurface. The resulting atom type channels are concatenated to produce the $(34, 48, 48, 48)$ input tensor for the CNNs.

process is shown in Figure 1. The grid used was a 24 \AA^3 cube, with a resolution of 0.5 \AA , centered around the binding site. Each point of the 3D grid has 34 information channels, corresponding to distinct heavy atoms on either the protein (16 channels) or ligand (18). A list of permitted atom types, based on smina atom types, is provided in Table 1 and is consistent with the work of Ragoza et al.²¹ Atoms have a Gaussian representation within the van der Waals radius that is quadratically smoothed to zero at $1.5\times$ the van der Waals radius from the input coordinates of a given atom. The input format provides no additional information beyond spatial coordinates and atom type and does not explicitly include bond order or hydrogens; the information provided in the input format is a comprehensive representation of the binding site of a single, static, docked protein–ligand complex, up to the chosen grid resolution and atom typing scheme.

Model Description. We based our model on the DenseNet architecture, introduced by Huang et al. in 2016. DenseNets have achieved state-of-art performance on several computer vision tasks, while exhibiting substantially improved parameter efficiency.²⁹ The key architectural difference in a

Table 1. Atom Typing Scheme^a

| receptor atom types | ligand atom types |
|--------------------------------|--------------------------------|
| AliphaticCarbonXSHydrophobe | AliphaticCarbonXSHydrophobe |
| AliphaticCarbonXSNonHydrophobe | AliphaticCarbonXSNonHydrophobe |
| AromaticCarbonXSHydrophobe | AromaticCarbonXSHydrophobe |
| AromaticCarbonXSNonHydrophobe | AromaticCarbonXSNonHydrophobe |
| Calcium | Bromine |
| | Chlorine |
| | Fluorine |
| | Iodine |
| Iron | Nitrogen |
| Magnesium | NitrogenXSAcceptor |
| Nitrogen | NitrogenXSDonor |
| NitrogenXSAcceptor | NitrogenXSDonorAcceptor |
| NitrogenXSDonor | Oxygen |
| NitrogenXSDonorAcceptor | OxygenXSAcceptor |
| OxygenXSAcceptor | OxygenXSDonorAcceptor |
| OxygenXSDonorAcceptor | Phosphorus |
| Phosphorus | Sulfur |
| Sulfur | SulfurAcceptor |
| Zinc | |

^aThe input format included 34 information channels, one for each atom type, 18 corresponding to ligand atoms and 16 to receptor atoms.

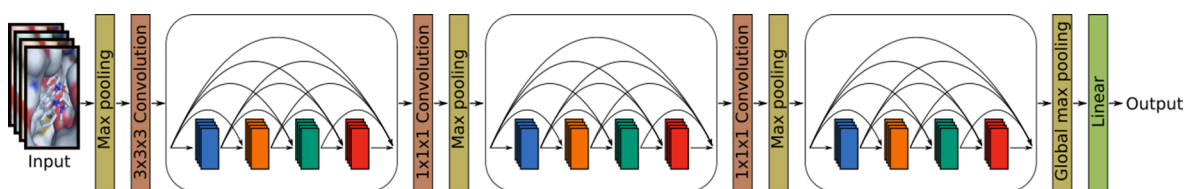


Figure 2. Schematic of the DenseNet architecture used in our model.

DenseNet compared with other convolutional networks is the fashion in which layers are connected (Figure S1). Instead of each layer receiving as input solely the output of the previous layer (standard connectivity), within each dense block a layer receives the output of all prior layers within that block (dense connectivity). The ability for the output of a layer to skip the next allows feature maps of different depths and hence complexities to form, and for new feature maps to be learned from a combination of existing maps of differing complexities. Furthermore, the dense connections improve gradient flow during backpropagation,³³ allowing deeper models to be trained effectively.²⁹

Our model contained three dense blocks, with four convolutional blocks within each (Figure 2). The convolutional blocks consisted of a batch normalization layer, a convolutional layer with a $3 \times 3 \times 3$ kernel, followed by a rectified linear unit. Between each dense block, we included a $1 \times 1 \times 1$ convolutional layer followed by a $2 \times 2 \times 2$ max pooling layer. The first convolutional layer contained 32 filters, after which each $3 \times 3 \times 3$ convolutional layer contained 16 filters, while the $1 \times 1 \times 1$ convolutional layers matched the number of input features. A schematic of our network architecture can be seen in Figure 2.

Training. Our CNN models were defined and trained using the Caffe deep learning framework,³⁴ with the MolGridData-Layer input format from the gnina fork.²¹ All networks were trained using stochastic gradient descent (SGD) to minimize

the multinomial logistic loss. We trained our model with a batch size of 16 for 25 000 iterations (corresponding to between 12 and 14 epochs of the active molecules during cross-validation). Due to the substantial class imbalance, we used oversampling and forced each batch to contain an equal number of positive and negative examples.³⁵ Following the work of Ragoza et al., we used a learning rate of 0.01, momentum of 0.9, an inverse learning rate decay with power = 1 and gamma = 0.001, and weight decay of 0.001.²¹ Input structures were augmented via random translations of up to 2 Å and random rotations (in multiples of 15°).

Test Time Scoring. We trained our CNN models only on the top-ranked AutoDock Vina pose for each complex. Many of these poses will be incorrect and restricting evaluation in the same way at test time would result in scoring inaccurate poses for many of the active molecules (Figure S2). At test time, any scoring system should ideally display a robustness to the poses selected, i.e. a similar score should be obtained if the poses assessed are similar. In particular, compounds, especially decoys, should not be ranked highly based on a single pose. Ragoza et al. demonstrated improvement over scoring only the top-ranked Vina pose by scoring all poses and using the maximum as the final score. We investigated averaging the scores of the top n ranked poses, where we determined n through cross-validation, and compared this to using the maximum.

Protein Family-Specific Models. It has been shown that in most cases a scoring function constructed for a specific protein family will outperform a universal model.^{30,31} As such, we constructed protein family-specific models using transfer learning by finetuning our universal model on data from targets belonging to the target's family.

Finetuning describes the technique where a model is pretrained on an initial, usually larger, training set, before the model is trained further on a second, targeted, set. This allows general features to be formed in the initial training stage, before the parameters become more specific during the second training stage. Deep learning models are prone to overfitting without sufficient training data. In many settings, data for a similar or related task is readily available but numerous high quality examples for the specific task are not. Thus, if training data is limited to only closely related targets, the examples would likely be more informative, but the risk of overfitting would be higher. In the other extreme, if all available data is included in the training set, then related, but not overly informative examples (such as those from a structurally dissimilar target), would make up the majority of the training set and thus nonspecific representations would be learned. Finetuning can be used to overcome these challenges.

Selecting the training set for finetuning is a nontrivial problem, especially given that we do not permit any overlap of targets between training and test sets, restricting training to off-target data. Proteins can be clustered into families where members typically have similar 3D structures and function. There are several widely used methods available to categorize proteins using combinations of sequence, structural, and functional information, such as SCOP³⁶ and CATH.³⁷ We have used the target protein classes provided in the works of Mysinger et al.,³⁸ Riniker and Landrum,³⁹ and Rohrer and Baumann⁴⁰ to cluster proteins and refer to these as the protein family. For all three data sets, these are consistent with the second level of ChEMBL's protein target classification. After pretraining on the entire training set, we investigated finetuning models for specific protein families using training sets constructed from targets belonging to the same family only. We examined the effect this had for different families and a varying number of family members.

We investigated the two extremes of finetuning by training only the classifier (final layer) and freezing the parameters in all other layers, or letting the parameters of all layers of the model train (Figure 3). We finetuned for five epochs of the active molecules. We reduced the learning rate by a factor of 10 to 0.001 when finetuning only the classifier and a factor of 20 to 0.0005 when finetuning all layers of the model. As is custom, we adopted a variable learning rate when finetuning all layers of the model, with the classifier's learning rate 10× higher than the convolutional layers. We compared these approaches to not finetuning the trained model, as well as training a new model from initialization (i.e., from scratch) only on data from a given protein family (Figure 3).

Model Evaluation. The choice of evaluation method should closely reflect the practical situation and desired outcome. We adopted the following approach, in line with the work of Ragoza et al.²¹ We first assessed the performance of our models by a three-fold cross-validation on the Directory of Useful Decoys: Enhanced (DUD-E)³⁸ and used the results to optimize our protocol. During cross-validation, we clustered proteins by sequence similarity using CD-HIT¹⁵ and ensured that targets with >80% sequence similarity were included in the

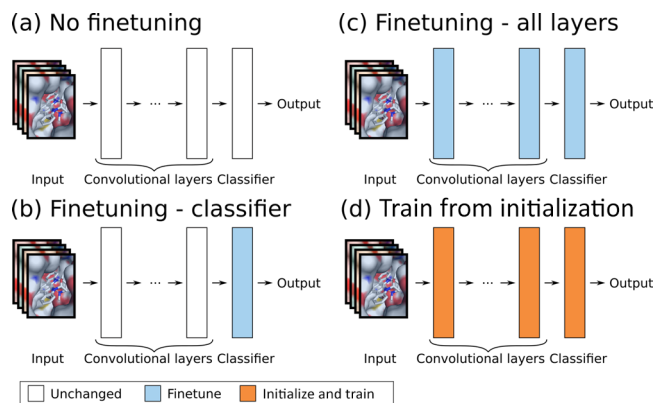


Figure 3. (a–d) Illustration of the different training regimes adopted to construct family-specific models. White corresponds to layers of the model that have been trained on all training data; blue, to layers that have been trained first on all training data and then finetuned on data from a specific protein family; and orange, to layers that have been trained only on data from a specific protein family.

same fold to avoid training and testing on overly similar targets. Wu et al. demonstrated almost perfect performance on DUD-E when performing a random training/test split across all targets,²⁰ allowing training and test sets to contain examples from the same targets. Wójcikowski et al. saw performance in cross-validation almost triple when splitting their data either randomly across all targets (“horizontal split”) or on a per-target basis, as opposed to keeping all examples for a given target in the same fold (“vertical split”).⁸ We did not permit any overlap of targets between training and validation sets, meaning that all testing is performed on unseen targets. The clustered approach we adopted represents a further refinement of the vertical split, and overall our cross-validation procedure more closely mimics the screening of a novel target.

The final protocol was then evaluated on two independently constructed test sets, a subset of the ChEMBL database⁴¹ curated by Riniker and Landrum,³⁹ following Heikamp and Bajorath,⁴² and the maximum unbiased validation (MUV)⁴⁰ data set, which is based on PubChem bioactivity data. Following the work of Ragoza et al., the independent sets have been further refined to avoid artificially enhancing performance using the following steps: (i) a global sequence alignment between all targets from the training and proposed test sets was performed, removing any test target with more than 80% sequence similarity with a training target, (ii) ProBiS⁴³ structural alignment on the binding sites of all pairs of targets from the training and proposed test sets was performed, removing any targets for which a significant alignment was found using the default ProBiS parameters. This resulted in a 13 target subset of the Riniker and Landrum ChEMBL set (from 50 initially) and a 9 target subset of the MUV set (from 17).

The models were evaluated with respect to two global metrics and four local ones. The global metrics used were area under curve (AUC) of the receiver operating characteristic (ROC) curve⁴⁴ and the precision recall curve (PRC). As noted by previous works,^{20,45} while AUC ROC and AUC PRC are highly correlated, substantial performance differences can emerge in the case of high class imbalance. We argue similarly to the work of Wu et al. that AUC PRC is a more informative metric given the extent of class imbalance inherent in virtual screening. Random performance has an expected AUC ROC of

0.5, whereas the expected AUC PRC of a random classifier is equal to the class imbalance (e.g., if there were 50 decoys for each active, the expected AUC PRC for a random classifier would be 0.02). We present AUC ROC due to its wide use, interpretability, and to allow direct comparison to other work. To measure early enrichment, we reported the ROC enrichment.^{46,47} ROC enrichment measures the ratio of true positive rate (TPR) to false positive rate (FPR) at a given FPR. We assessed the enrichment factor (EF) at 0.5%, 1%, 2%, and 5%. The maximum possible ROC enrichment depends on the FPR threshold (e.g., at an FPR of 2%, the highest possible ROC enrichment is 50), while random performance has an expected enrichment factor of 1 at any FPR threshold.

Visualization. We took a machine learning approach to visualization and replaced the final layer of the model (the classifier) with a $1 \times 1 \times 1$ convolutional layer with the same parameters as the classifier. This transformed the final layer from a global classifier into a regional classifier and allowed specific regions of the image to be assessed. A score is produced for each subsection, rather than a single score for the entire complex. A limitation of this technique is that our models have not been trained to assess subregions of the complex, but rather the complex as a whole. However, this technique provides a region-based assessment without modifying the protein–ligand complex at marginal additional computational cost.

Comparison to Previous Work. In structure-based virtual screening, traditional approaches have typically used experimental data to parametrize a physically inspired scoring function.^{9–14,16} Machine learning approaches have either reused the features of traditional approaches, or calculated new descriptors from docked protein–ligand complexes.^{8,17,19} Recently, several attempts to learn features relevant for binding in an end-to-end manner, rather than manually selecting them, have shown promise. Ragoza et al.²¹ and Wallach et al.⁴⁸ both used shallow CNN architectures for virtual screening. Shallow CNNs have also been used for binding affinity prediction.^{22,49}

Specifically, Ragoza et al.²¹ proposed a neural network for protein–ligand scoring consisting of three $3 \times 3 \times 3$ convolutional layers (with 32, 64, and 128 filters respectively), each preceded by a $2 \times 2 \times 2$ max pooling layer. They scored all docked poses using a single, universal model, and took the maximum as the final score. To our knowledge, their approach is the best performing CNN model, and has been shown to outperform both empirical scoring functions and feature-based machine learning approaches at virtual screening.²¹ We have therefore used this approach as the benchmark. Our method utilizes the same input format but differs in four key ways from all previous approaches. First, we adopted the densely connected neural network architecture described above. Second, at test time our model scored each protein–ligand complex by averaging over an ensemble of docked poses. Third, we used transfer learning to construct protein family-specific models for each of the four major protein classes in DUD-E. Finally, we employed an ensemble of models. The effect of each of these changes is detailed in a full ablation study (Table 3) and discussed below.

RESULTS AND DISCUSSION

Cross-Validation on DUD-E. We assessed performance using a clustered cross-validation on DUD-E. We present a final version of our method that incorporates the DenseNet architecture, average test time scoring, and combines an

ensemble of protein family-specific models (“DenseFS”). Our approach achieved state-of-the-art performance on the DUD-E benchmark, recording average per-target AUC ROC of 0.917, AUC PRC of 0.443, and 0.5% ROC enrichment factor of 79.3 (Table 2).

Table 2. Mean AUC ROC, AUC PRC, and ROC Enrichment Across Targets in the DUD-E Data Set for Our Method, DenseFS, Compared to Baseline CNN and the AutoDock Vina Scoring Function

| metric | Vina | Baseline CNN | DenseFS |
|---------|--------|--------------|---------|
| AUC ROC | 0.703 | 0.862 | 0.917 |
| AUC PRC | 0.093 | 0.263 | 0.443 |
| 0.5% EF | 15.017 | 44.521 | 79.321 |
| 1% EF | 10.383 | 30.652 | 47.986 |
| 2% EF | 7.135 | 19.724 | 28.408 |
| 5% EF | 4.726 | 10.595 | 13.744 |

We compared DenseFS to the CNN model described by Ragoza et al.²¹ (“Baseline CNN”). We also compared to the AutoDock Vina scoring function. Summary results from cross-validation on DUD-E are shown in Figures 4 and 5 and Table 2. For each of the CNN methods, we carried out three replicas with different random seeds. DenseFS exhibited around a 70% improvement in AUC PRC and 0.5% ROC enrichment over the Baseline CNN, with around a 400% gain over AutoDock Vina. Compared with the Baseline CNN, our method achieved a higher AUC ROC for 97 of the 102 targets (95%), and AUC PRC for 95 of the 102 targets (93%).

For both the Baseline CNN and DenseFS, substantial performance differences emerged between targets, in particular between different protein families (Figure 7 and Tables S1–S5). DenseFS exhibited the most predictive power for kinases followed by proteases, while the largest improvement in performance compared to the Baseline CNN was on the nuclear proteins family. Performance on GPCRs and the remaining targets (denoted as other in Figure 5) was lower compared to the families for which the data set contained more targets, albeit still much better than random. The ordering of performance by family precisely matched the overall number of targets in DUD-E, highlighting again the importance of the presence of data from the same protein family in the training set. However, while beneficial, this is not a requirement. DenseFS achieved an AUC ROC of 0.943 on DUD-E target DYR, an oxidoreductase, which shared no family members with the training set. More generally, DenseFS exhibited substantial predictive power for the “other” category, achieving an average AUC ROC of 0.865 (Table S5).

We performed an ablation study on the four key changes made between Baseline CNN and DenseFS (Table 3). All four changes had a material positive impact both on a standalone basis, but also when added to any pre-existing combination of changes. This suggests that each factor works somewhat independently. For the remainder of this section, we discuss the individual effect and implications of each of these advantages.

Advantage 1. Model Architecture. We found substantial benefit from changing the model topology to a DenseNet architecture and adopting a deeper model, shown in Table 3. Changing the model architecture alone was responsible for a 25% increase in AUC PRC compared to Baseline CNN (c. 37% of the overall improvement). This shows the suitability of

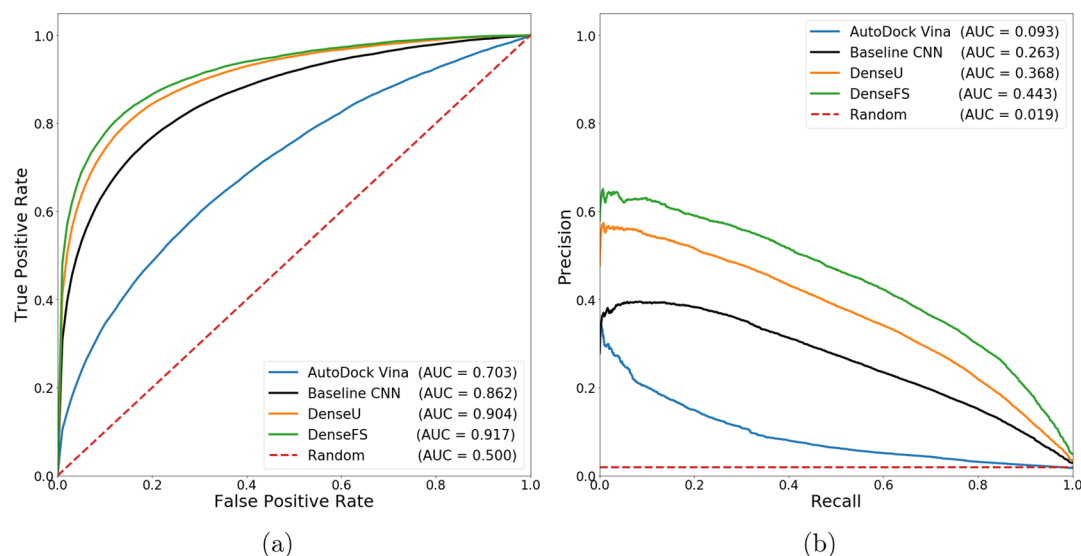


Figure 4. Average per-target ROC (a) and PRC (b) plots comparing our methods (DenseFS and DenseU) with Baseline CNN and the AutoDock Vina scoring function on the DUD-E data set during cross-validation. Our methods outperformed Baseline CNN by 6.4% and 4.9% with respect to AUC ROC and 68.4% and 39.9% with respect to AUC PRC.

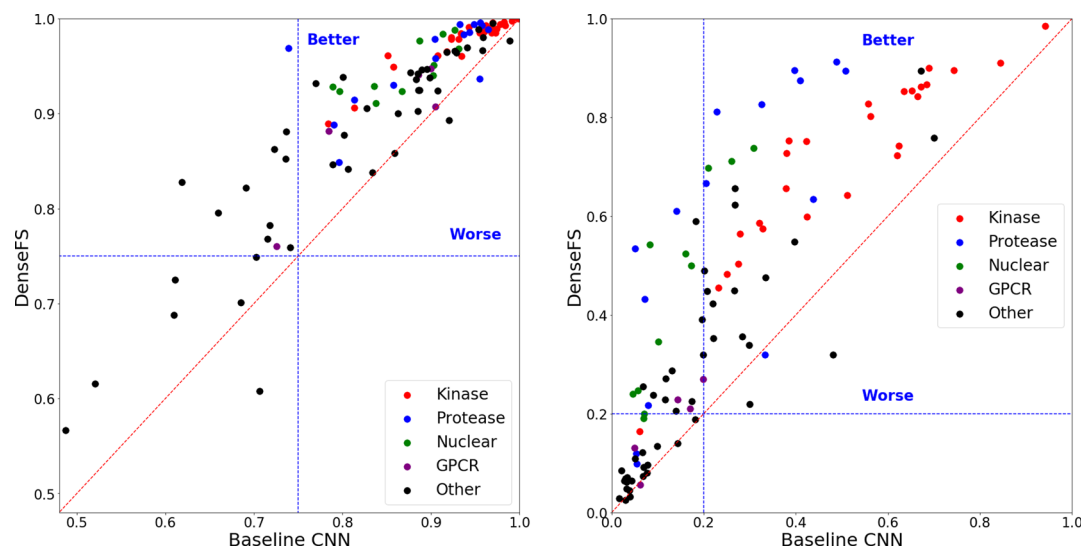


Figure 5. Performance of DenseFS compared to Baseline CNN during cross-validation on the DUD-E data set. We directly compare performance on each target in DUD-E with respect to AUC ROC (left) and AUC PRC (right). Points above the diagonal represent targets for which DenseFS outperforms Baseline CNN (and vice versa). Guidelines are set at 0.75 for AUC ROC and 0.2 for AUC PRC. This corresponds to approximately a 10X improvement over random for AUC PRC. DenseFS achieved a higher AUC ROC for 97 of the 102 targets (95%) and AUC PRC for 95 of the 102 targets (93%).

reformulating structure-based virtual screening as a computer vision problem and highlights how advances in computer vision tasks provide improvements in this setting.

Our final model contained more convolutional layers and more overall features than Baseline CNN. That a more expressive model led to improved performance adds further evidence to the understanding that binding is governed by complex relationships and the factors that determine activity are not readily summarized. Our findings that a deeper model improved performance is somewhat contrary to the optimization analysis performed by Ragoza et al., where cross-validation suggested that a shallower and narrower model led to an improved representation of the data, and differs from the much shallower CNNs applied in Jiménez et al. However, in both cases the network architecture was optimized on different,

much smaller, data sets and for different tasks (pose prediction in the work of Ragoza et al. and binding affinity prediction in that of Jiménez et al.²²). Given the similarities with virtual screening, we expect that if larger data sets were utilized, then deeper models would yield benefits for these tasks.

Hyperparameter tuning is necessary to extract maximum performance from a model, and suboptimal choices can lead to substantially weaker results.⁵⁰ Despite adopting largely the same hyperparameter settings determined by Ragoza et al. for a different network, our ablation study showed that changing the network architecture from a traditional CNN to a DenseNet resulted in a 20.8–36.5% increase in AUC PRC.

Advantage 2. Test Time Scoring. We see in Figure 6 that averaging the top n ranked poses outperformed scoring only the top ranked Vina pose or the highest scored pose for almost

Table 3. Ablation Study of the Primary Improvements in Our Final Protocol (DenseFS) Compared to Baseline CNN^a

| Model Architecture | Average Scoring | Protein Family Models | Ensemble | AUC PRC | Gain (%) | 0.5% EF | Gain (%) |
|--------------------|-----------------|-----------------------|----------|---------|----------|---------|----------|
| ✓ | ✓ | ✓ | ✓ | 0.443 | 68.4% | 79.321 | 78.2% |
| ✓ | ✓ | ✓ | | 0.421 | 60.1% | 75.351 | 69.2% |
| ✓ | | ✓ | ✓ | 0.407 | 54.8% | 72.995 | 64.0% |
| ✓ | ✓ | | ✓ | 0.394 | 49.8% | 69.647 | 56.4% |
| ✓ | | ✓ | | 0.384 | 46.0% | 67.903 | 52.5% |
| ✓ | ✓ | | | 0.368 | 39.9% | 64.888 | 45.7% |
| ✓ | | | ✓ | 0.359 | 36.5% | 62.713 | 40.9% |
| | ✓ | ✓ | ✓ | 0.357 | 35.7% | 64.040 | 43.8% |
| | ✓ | ✓ | | 0.348 | 32.3% | 62.010 | 39.3% |
| | | ✓ | ✓ | 0.336 | 27.8% | 59.262 | 33.1% |
| ✓ | | | | 0.330 | 25.5% | 56.819 | 27.6% |
| | | ✓ | | 0.326 | 24.0% | 56.771 | 27.5% |
| | ✓ | | ✓ | 0.298 | 13.3% | 52.289 | 17.4% |
| | ✓ | | | 0.285 | 8.4% | 49.484 | 11.1% |
| | | | ✓ | 0.278 | 5.7% | 48.481 | 8.9% |
| | | | | 0.263 | 0.0% | 44.521 | 0.0% |

^aEach of the four main changes had a material positive impact in any possible protocol, with the change of model architecture and use of family-specific models being the most important.

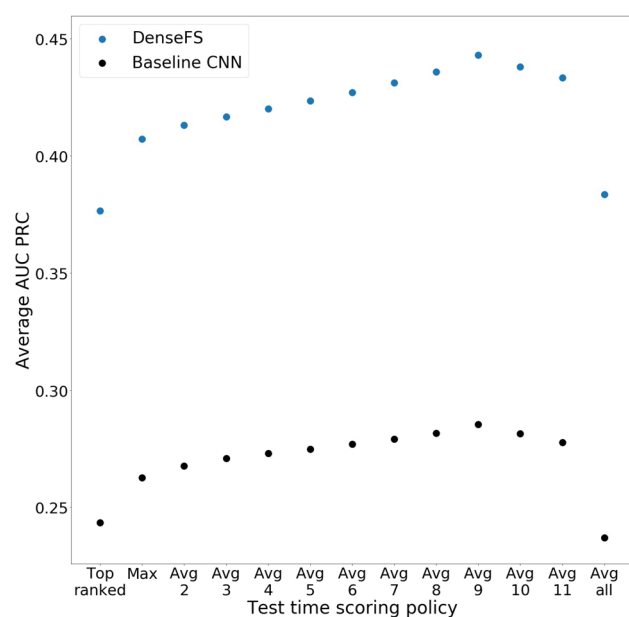


Figure 6. Average AUC PRC across targets in the DUD-E data set for different test time scoring policies. For both Baseline CNN and DenseFS, the optimum n to average across was nine.

all values of n , with the best performance exhibited by taking $n = 9$. AutoDock Vina generated 15 poses on average for each of the active molecules in DUD-E; averaging the top 9 highest scored poses corresponds to averaging the top 60% of poses. This intuitively agrees with our motivation of averaging over a group of poses that are more likely to be close to a native pose. Averaging across almost all of the complexes would be likely to include many inaccurate or unrealistic poses, whereas only averaging across a small number of poses would still be susceptible to an outlier, either with high or low score. We found that the optimum choice of n was consistent across the CNN approaches tested. Overall, adopting average scoring instead of a max scoring policy led to improvements in AUC PRC of between 7.6% and 14.4% (Table 3).

Combining these two modifications produces a universal scoring function consisting of a single model. We compared this approach, denoted as DenseU, with Baseline CNN, and

found an improvement in average AUC PRC of 39.9% (Figure 4), with DenseU achieving a higher AUC PRC for 90 of the 102 targets (88%).

Advantage 3. Protein Family-Specific Models. The benefits of finetuning a separate model for each of the four major protein families represented in DUD-E can be seen in Table 3 and Figures 7, S3, and S4. In our ablation study, adopting protein family-specific models instead of a universal model led to average improvements in AUC PRC of 18.3–24.0% (Table 3).

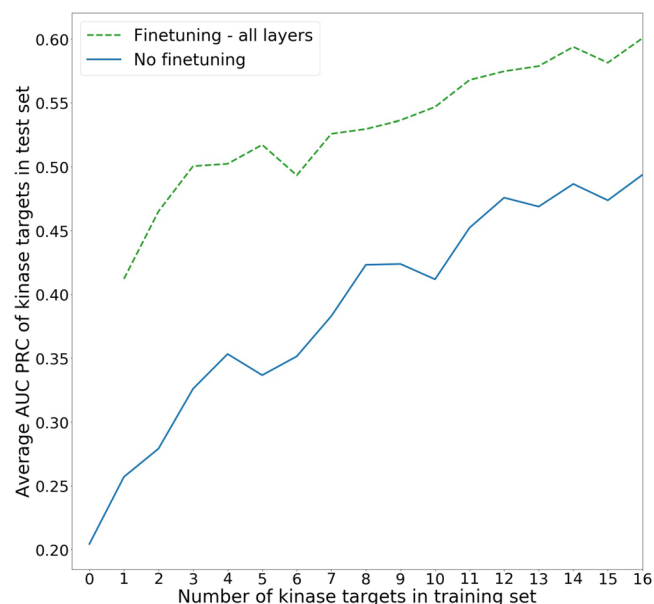


Figure 7. Average AUC PRC of kinase targets for varying number of kinases in the training set. We compared finetuning all layers to no finetuning (training protocols displayed in Figure 3). The blue line (solid) represents no finetuning, while the green line (dashed) shows the effect of finetuning all layers on only kinase data. Finetuning all layers outperformed no finetuning, even with limited data, and performance continued to improve as more targets were added to the training data.

We subdivided DUD-E into the following five groups by the classifications provided: kinase (26 targets), protease (15), nuclear (11), GPCR (5), and other (45). We found finetuning improved performance for all four families, but due to the different amounts of data available, different training regimes were optimal. Allowing all parameters to vary (Figure 3c) for kinases, proteases, and nuclear proteins led to better results over retraining solely the final layer or no finetuning at all, whereas for GPCRs, we found that freezing the convolutional filters that were trained on a mix of protein family data and only finetuning the final layer on GPCR data was optimal (Figure 3b).

We believe the different behavior is due to the small number and diversity of GPCR targets. If additional GPCR data were available, we anticipate that allowing more parameters to vary would outperform finetuning only the classifier. However, given we witnessed improvement from finetuning all layers with very limited data for the other families, it is also possible that the diversity of the GPCR targets means that it is more important to retain representations learnt from other protein families.

In order to gain a better understanding of the impact of protein family data on the construction of CNN models, we examined the effect of artificially reducing the amount of family-specific data used both in initial training and subsequent finetuning (Figures 7, S3, and S4). This allowed us to assess whether additional data had a positive impact on performance, or if further examples beyond a point were largely redundant. We trained copies of Baseline CNN with varying numbers of targets from the protein family included in the training set. We then finetuned these models as previously described in Figure 3 and compared with training from initialization.

First, we found that finetuning improved performance even with very limited number of family members for all families (Figures 7, S3, and S4). Considering the largest family, this effect persisted regardless of the number of kinases included in the training set, with a 20–50% increase in performance as a result of finetuning all layers of our model, depending on the number of targets included (Figure 7). Second, ensuring family members are present in the training set is crucially important. The average AUC PRC across kinase targets more than doubled when all available family data was present compared to including no kinase data in the training set. To confirm that this effect was not simply due to a smaller training set, we removed targets at random and did not witness the same reduction in predictive power compared to removing the same number of kinases. Finally, we saw that performance continued to improve as more targets were included in the training set, and the rate of improvement remained fairly constant as targets were added. This suggests that adding more kinases to the training set would further improve performance and that even with 16 kinase targets in the training set we have not reached learning saturation.

Advantage 4. Ensemble. Ensemble methods exploit the predictions of multiple models to improve performance. We combined the predictions of the three replicas by averaging the scores produced by each of the models. This provided a small, but appreciable, improvement, with increases in average AUC PRC of 3.4–11.0%.

Independent Test Sets. We assessed performance on the independent test sets by training models using the entirety of the DUD-E data set. We present the same version of our

method, DenseFS, and again compared to Baseline CNN and the AutoDock Vina scoring function.

ChEMBL. Consistent with cross-validation results, DenseFS substantially outperformed Baseline CNN and AutoDock Vina (Table 4). DenseFS achieved an average AUC PRC of 0.214 and 0.5% ROC enrichment factor of 47.6, representing improvements of 40.8% and 33.6%, respectively, over Baseline CNN.

Table 4. Mean AUC ROC, AUC PRC, and ROC Enrichment Across Targets in the ChEMBL Test Set for Our Method, DenseFS, Compared to Baseline CNN and the AutoDock Vina Scoring Function

| metric | Vina | Baseline CNN | DenseFS |
|---------|-------|--------------|---------|
| AUC ROC | 0.656 | 0.788 | 0.838 |
| AUC PRC | 0.039 | 0.152 | 0.214 |
| 0.5% EF | 9.293 | 35.626 | 47.587 |
| 1% EF | 7.505 | 23.244 | 30.732 |
| 2% EF | 5.324 | 14.933 | 18.940 |
| 5% EF | 4.288 | 8.417 | 9.979 |

As anticipated, the ChEMBL test set was more challenging for all methods by all metrics. However, the models retained substantial predictive power, and the results show an ability to generalize to targets and data sources beyond the training set, suggesting that the improvements seen in cross-validation were not simply a result of overfitting to DUD-E.

On a per-target basis, DenseFS outperformed Baseline CNN by the largest margin on targets belonging to one of the four specific protein families (kinase, protease, nuclear, GPCR), in particular the three protease targets. Our method demonstrated more modest improvements in performance for targets in the “other” category. Overall, DenseFS achieved a higher AUC ROC for 11 of the 14 targets (79%) and AUC PRC for 13 of the 14 targets (93%), compared to Baseline CNN.

MUV. The summary results in Table 5 show that none of the methods tested had any meaningful predictive power on MUV.

Table 5. Mean AUC ROC, AUC PRC, and ROC Enrichment Across Targets in the MUV Test Set for Our Method, DenseFS, Compared to Baseline CNN and the AutoDock Vina Scoring Function

| metric | Vina | Baseline CNN | DenseFS |
|---------|-------|--------------|---------|
| AUC ROC | 0.546 | 0.507 | 0.534 |
| AUC PRC | 0.003 | 0.003 | 0.003 |
| 0.5% EF | 0.000 | 2.018 | 2.407 |
| 1% EF | 1.204 | 1.798 | 3.207 |
| 2% EF | 1.769 | 1.521 | 2.222 |
| 5% EF | 1.323 | 1.607 | 1.338 |

The only target for which any of the methods demonstrated an ability to discriminate actives from decoys was human cAMP-dependent protein kinase (MUV target ID 548, PDB ID 3poo). All three CNN methods achieved an AUC ROC of around 0.8 and AUC PRC of ca. 0.01 (5× greater than random).

The MUV data set is constructed differently from the other data sets used, using PubChem data rather than ChEMBL. In addition, cell-based assays are used for some of the targets. These factors undoubtedly make the data set a tougher challenge, and some have even questioned the appropriateness

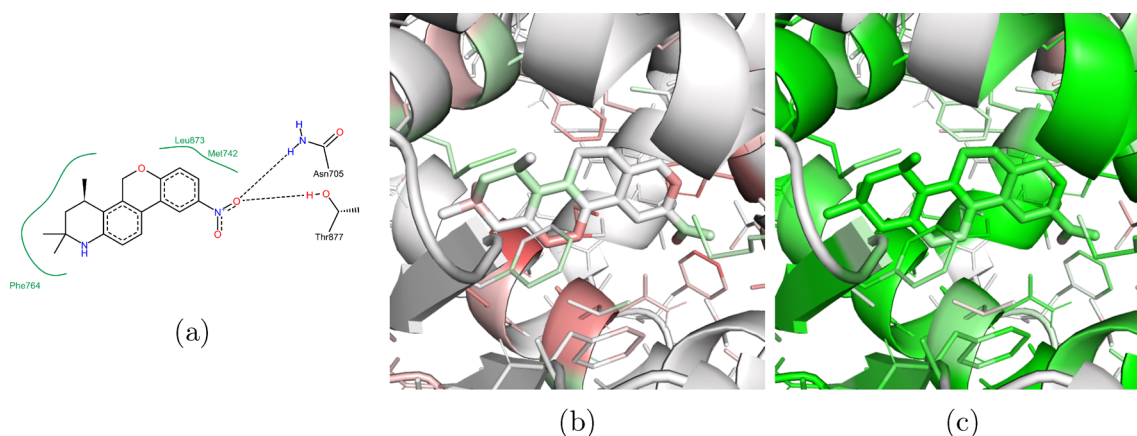


Figure 8. Visualization of the known active CHEMBL293409 ligand (a) docked against the DUD-E target ANDR. (b and c) Results of the visualization procedure for Baseline CNN and DenseFS, respectively. Areas of green indicate a score for that region above 0.5, whereas red represents a score below 0.5, with the intensity depending on the magnitude of the difference. The Baseline CNN assigned the complex an overall score of 0.34, while DenseFS scored the complex at 0.91.

of using MUV as a structure-based virtual screening benchmark.^{21,51} The only approaches that have shown meaningful predictive power on MUV are ligand-based methods.^{20,52} Furthermore, these have required including target-specific data in the training set, and have often used large external data sets without any regard to the overlap between training and test sets.

Despite the weak performance on the MUV data set, as a result of the clustered cross-validation procedure on DUD-E and the performance on the ChEMBL independent test set, we are confident that our model is learning genuinely useful information about protein–ligand interactions, as opposed to simply artifacts of the construction of DUD-E or overfitting to specific examples. However, substantial further improvements will be required in order to capture more accurately the factors determining binding.

Visualization. Understanding the components of a protein–ligand complex that govern interactions would greatly assist with both finding initial hits and lead optimization. In particular, the ability to interpret the CNN’s output beyond a simple score is crucial. In order to visualize the drivers of the CNN models’ predictions, we replaced the final fully connected layer of a trained model with a $1 \times 1 \times 1$ convolutional layer with the same parameters. This transformed the global classifier into a regional-based classifier at almost no additional computational cost.

In Figure 8, we present the analysis of our visualization procedure for an example protein–ligand complex, the ChEMBL293409 ligand docked against the human androgen receptor (DUD-E target ANDR, PDB ID 2am9), a known active. In Figure 8a, we display a 2D diagram of the complex, annotated by PoseView.⁵³ In Figures 8b and c, we colored the complex according to the regional scores, based on a threshold of 0.5. Scores below 0.5 were colored in red, whereas scores above 0.5 were colored in green, with the intensity depending on the magnitude of the difference. We analyzed the predictions of Baseline CNN and DenseFS for this complex.

Baseline CNN assigned the complex a score of 0.34. While Baseline CNN scored one of the oxygens in the nitronium ion slightly favorably due to the interaction with the alcohol on the threonine 877 residue, it scored the interaction with the amine on the asparagine 705 residue unfavorably. Baseline CNN scored the remainder of the molecule overall fairly neutrally for

binding. In contrast, DenseFS scored the complex at 0.91, scoring both of the interactions with the nitronium ion highly. In addition, the majority of the ring structure was scored favorably, due to the proximity with the phenylalanine 764, leucine 873, and methionine 742 residues. The bottom two carbons of the left-hand benzene ring (in relation to the orientation in Figure 8) were scored less favorably than the remainder of the ring structure by both models. This suggests an area of the molecule that could be altered to increase the affinity of the compound.

CONCLUSION

We have presented a deep learning approach that gives substantial improvement over previous work in virtual screening on both DUD-E and an independent test set, producing state-of-the-art results by all metrics assessed. On DUD-E, our method exhibited around a 70% improvement in AUC PRC and 0.5% ROC enrichment over the Baseline CNN of Ragoza et al., achieving a higher AUC PRC for 95 of the 102 targets (93%). On the independent ChEMBL set, our method outperformed the Baseline CNN on 13 of the 14 targets, resulting in an average increase in AUC PRC of over 40%.

The performance of our method further reinforces the power of approaches that adopt minimal input beyond a spatial representation and limited atom typing of the 3D complex structures. All improvements were obtained by better utilizing the same input structures and format. Our approach differed in four key ways from that of Ragoza et al.,²¹ each of which contributed to the improved performance in a somewhat independent manner (Table 3).

First, we showed that recent advances in computer vision can be applied to virtual screening by using a densely connected CNN architecture. This highlights the suitability of reframing virtual screening as a computer vision problem. We did not perform an extensive review of choices for our deep learning network and note that several architectures^{24,54} now report improved accuracy on ImageNet. We anticipate that further improvement could be obtained by applying the current state-of-the-art techniques. Adopting the DenseNet architecture resulted in the largest improvement in performance, and changing this alone was responsible for a 20.8–36.5% increase in AUC PRC during cross-validation (30.4–53.4% of overall improvement).

Docking introduces substantial noise to the data sets due to the inaccuracy of many docked poses (Figure S2). We demonstrated that using an average scoring protocol instead of max scoring provided a 7.6–14.4% increase in AUC PRC (11.1–21.1% of overall improvement). The benefit of this approach results from reducing the reliance on any single pose, and eliminating the ability for a molecule to be ranked highly from a single pose alone. We expect that with more accurate docked poses the impact of an average scoring protocol over a max or single pose scoring protocol would be reduced, although we would expect some benefit to persist.

Finetuning a universal model on subsets of the available data allowed us to construct protein family-specific models, as opposed to a single universal model, and resulted in average improvements to AUC PRC of 18.3–24.0% (26.7–35.1% of overall improvement). This let our models form different representations to capture physicochemical nuances exhibited by different families. In our investigation into the importance of family-specific data, we found that (i) very limited family data was required before a family-specific model outperformed a universal one; (ii) the presence of proteins from the same family in the training set is crucial for the methods tested to have high predictive power, although our models continued to exhibit predictive power even when this was not the case; and (iii) continuing to add further examples provided appreciable benefit, even for the largest family present in the training set. This suggests that more data is required to exploit CNN-based methods fully. Future work could investigate the extent to which this effect persists or if at some point learning saturation is reached and further data becomes redundant.

Finally, combining the predictions of three models (trained with different random seeds) in an ensemble improved average AUC PRC by 3.4–11.0% (5.0–16.1% of overall improvement). While this is a relatively minor benefit, this technique consistently improved predictions, despite using only three models in our ensemble. A major drawback is that the additional computation required directly scales with the number of models used; however, this is somewhat mitigated by the fully parallelizable nature of additional models, both during training and at test time.

While our methods represent a substantial improvement over the state-of-the-art on the DUD-E and ChEMBL benchmarks and show the promise of a CNN-based approach, the limited predictive power on the MUV data set underscores the challenges still faced in virtual screening.

■ ASSOCIATED CONTENT

● Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: [10.1021/acs.jcim.8b00350](https://doi.org/10.1021/acs.jcim.8b00350).

Figure S1: Illustration of connectivity in a standard convolutional neural network compared to a DenseNet. Figure S2: Docked poses of the ligand ChEMBL300406 into DUD-E target SAHH. Figure S3: Average AUC PRC of protease targets for varying number of proteases in the training set. Figure S4: Average AUC PRC of nuclear targets for varying number of nuclear proteins in the training set. Tables S1–S23: Additional statistical data (PDF)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: deane@stats.ox.ac.uk.

ORCID

Fergus Imrie: [0000-0002-6241-0123](https://orcid.org/0000-0002-6241-0123)

Charlotte M. Deane: [0000-0003-1388-2252](https://orcid.org/0000-0003-1388-2252)

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

The authors thank David Ryan Koes for providing docked protein–ligand poses. F.I. is supported by the EPSRC (Reference: EP/N509711/1). A.R.B. is employed by the SGC. The SGC is a registered charity (number 1097737) that receives funds from AbbVie, Bayer Pharma AG, Boehringer Ingelheim, Canada Foundation for Innovation, Eshelman Institute for Innovation, Genome Canada through Ontario Genomics Institute [OGI-055], Innovative Medicines Initiative (EU/EFPIA) [ULTRA-DD grant no. 115766], Janssen, Merck KGaA (Darmstadt, Germany), MSD, Novartis Pharma AG, Ontario Ministry of Research, Innovation and Science (MRIS), Pfizer, So Paulo Research Foundation–FAPESP, Takeda, and Wellcome [106169/Z/14/Z].

■ REFERENCES

- (1) Siedlecki, P.; Boy, R. G.; Musch, T.; Brueckner, B.; Suhai, S.; Lyko, F.; Zielenkiewicz, P. Discovery of two novel, small-molecule inhibitors of DNA methylation. *J. Med. Chem.* **2006**, *49*, 678–683.
- (2) Kiss, R.; Kiss, B.; Konczol, A.; Szalai, F.; Jelinek, I.; Laszlo, V.; Noszal, B.; Falus, A.; Keseru, G. M. Discovery of novel human histamine H4 receptor ligands by large-scale structure-based virtual screening. *J. Med. Chem.* **2008**, *51*, 3145–3153.
- (3) Odolczyk, N.; Fritsch, J.; Norez, C.; Servel, N.; Da Cunha, M. F.; Bitam, S.; Kupniewska, A.; Wiszniewski, L.; Colas, J.; Tarnowski, K.; Tondelier, D.; Roldan, A.; Saussereau, E. L.; Melin-Heschel, P.; Wieczorek, G.; Lukacs, G. L.; Dadlez, M.; Faure, G.; Herrmann, H.; Ollero, M.; Becq, F.; Zielenkiewicz, P.; Edelman, A. Discovery of novel potent $\Delta F508$ -CFTR correctors that target the nucleotide binding domain. *EMBO Mol. Med.* **2013**, *5*, 1484–1501.
- (4) Gau, D.; Lewis, T.; McDermott, L.; Wipf, P.; Koes, D.; Roy, P. Structure-based virtual screening identifies small molecule inhibitor of the profilin-actin interaction. *J. Biol. Chem.* **2018**, *293*, 2606–2616.
- (5) Bollini, M.; Leal, E. S.; Adler, N. S.; Aucar, M. G.; Fernández, G. A.; Pascual, M. J.; Merwaiss, F.; Alvarez, D. E.; Cavasotto, C. N. Discovery of Novel Bovine Viral Diarrhea Inhibitors Using Structure-Based Virtual Screening on the Envelope Protein E2. *Front. Chem.* **2018**, *6*, 1–10.
- (6) 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.
- (7) Li, Y.; Han, L.; Liu, Z.; Wang, R. Comparative assessment of scoring functions on an updated benchmark: 2. evaluation methods and general results. *J. Chem. Inf. Model.* **2014**, *54*, 1717–1736.
- (8) Wójcikowski, M.; Ballester, P. J.; Siedlecki, P. Performance of machine-learning scoring functions in structure-based virtual screening. *Sci. Rep.* **2017**, *7*, 46710.
- (9) Böhm, H. J. The development of a simple empirical scoring function to estimate the binding constant for a protein-ligand complex of known three-dimensional structure. *J. Comput.-Aided Mol. Des.* **1994**, *8*, 243–256.
- (10) Eldridge, M. D.; Murray, C. W.; Auton, T. R.; Paolini, G. V.; Mee, R. P. Empirical scoring functions: I. The development of a fast empirical scoring function to estimate the binding affinity of ligands in receptor complexes. *J. Comput.-Aided Mol. Des.* **1997**, *11*, 425–445.

- (11) Friesner, R. A.; Banks, J. L.; Murphy, R. B.; Halgren, T. A.; Klicic, J. J.; Mainz, D. T.; Repasky, M. P.; Knoll, E. H.; Shelley, M.; Perry, J. K.; Shaw, D. E.; Francis, P.; Shenkin, P. S. Glide: A New Approach for Rapid, Accurate Docking and Scoring. 1. Method and Assessment of Docking Accuracy. *J. Med. Chem.* **2004**, *47*, 1739–1749.
- (12) Trott, O.; Olson, A. AutoDock Vina: improving the speed and accuracy of docking with a new scoring function, efficient optimization and multithreading. *J. Comput. Chem.* **2009**, *31*, 455–461.
- (13) 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.
- (14) Gohlke, H.; Hendlich, M.; Klebe, G. Knowledge-based scoring function to predict protein-ligand interactions. *J. Mol. Biol.* **2000**, *295*, 337–356.
- (15) Huang, Y.; Niu, B.; Gao, Y.; Fu, L.; Li, W. CD-HIT Suite: A web server for clustering and comparing biological sequences. *Bioinformatics* **2010**, *26*, 680–682.
- (16) Zhou, H.; Skolnick, J. GOAP: A generalized orientation-dependent, all-atom statistical potential for protein structure prediction. *Biophys. J.* **2011**, *101*, 2043–2052.
- (17) Durrant, J. D.; McCammon, J. A. NNScore 2.0: A neural-network receptor-ligand scoring function. *J. Chem. Inf. Model.* **2011**, *51*, 2897–2903.
- (18) Li, H.; Leung, K.-S.; Wong, M.-H.; Ballester, P. J. The Importance of the Regression Model in the Structure-Based Prediction of Protein-Ligand Binding. *Computational Intelligence Methods for Bioinformatics and Biostatistics. CIBB 2014. Lecture Notes in Computer Science* **2015**, *8623*, 219–230.
- (19) Ballester, P. J.; Mitchell, J. B. O. A machine learning approach to predicting proteinligand binding affinity with applications to molecular docking. *Bioinformatics* **2010**, *26*, 1169–1175.
- (20) 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. *Chem. Sci.* **2018**, *9*, 513–530.
- (21) Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D. R. Protein-Ligand Scoring with Convolutional Neural Networks. *J. Chem. Inf. Model.* **2017**, *57*, 942–957.
- (22) Jiménez, J.; Škalič, M.; Martínez-Rosell, G.; De Fabritiis, G. KDEEP: Protein-Ligand Absolute Binding Affinity Prediction via 3D-Convolutional Neural Networks. *J. Chem. Inf. Model.* **2018**, *58*, 287–296.
- (23) Russakovsky, O.; Deng, J.; Su, H.; Krause, J.; Satheesh, S.; Ma, S.; Huang, Z.; Karpathy, A.; Khosla, A.; Bernstein, M.; Berg, A. C.; Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *Int. J. Comput. Vis.* **2015**, *115*, 211–252.
- (24) Hu, J.; Shen, L.; Sun, G. Squeeze-and-Excitation Networks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2018**, pp 7132–7141.
- (25) Drozdal, M.; Chartrand, G.; Vorontsov, E.; Shakeri, M.; Di Jorio, L.; Tang, A.; Romero, A.; Bengio, Y.; Pal, C.; Kadoury, S. Learning normalized inputs for iterative estimation in medical image segmentation. *Med. Image Anal.* **2018**, *44*, 1–13.
- (26) Li, X.; Chen, H.; Qi, X.; Dou, Q.; Fu, C.-W.; Heng, P. A. HDenseUNet: Hybrid Densely Connected UNet for Liver and Liver Tumor Segmentation from CT Volumes. *IEEE Trans. Med. Imaging* **2018**, *1*.
- (27) Yu, Y.; Lin, H.; Meng, J.; Wei, X.; Guo, H.; Zhao, Z. Deep transfer learning for modality classification of medical images. *Information* **2017**, *8*, 91.
- (28) Goh, G. B.; Siegel, C.; Vishnu, A.; Hodas, N. O.; Baker, N. Chemception: A Deep Neural Network with Minimal Chemistry Knowledge Matches the Performance of Expert-developed QSAR/QSPR Models. *arXiv.org* **2017**, 1706.06689.
- (29) Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K. Q. Densely Connected Convolutional Networks. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* **2017**, 4700–4708.
- (30) Ross, G. A.; Morris, G. M.; Biggin, P. C. One size does not fit all: The limits of structure-based models in drug discovery. *J. Chem. Theory Comput.* **2013**, *9*, 4266–4274.
- (31) Wang, Y.; Guo, Y.; Kuang, Q.; Pu, X.; Ji, Y.; Zhang, Z.; Li, M. A comparative study of family-specific protein-ligand complex affinity prediction based on random forest approach. *J. Comput.-Aided Mol. Des.* **2015**, *29*, 349–360.
- (32) Tajbakhsh, N.; Shin, J. Y.; Gurudu, S. R.; Hurst, R. T.; Kendall, C. B.; Gotway, M. B.; Liang, J. Convolutional Neural Networks for Med. Image Anal.: Full Training or Fine Tuning? *IEEE Trans. Med. Imaging* **2016**, *35*, 1299–1312.
- (33) Rumelhart, D. E.; Hinton, G. E.; Williams, R. J. Learning representations by backpropagating errors. *Nature* **1986**, *323*, 533–536.
- (34) Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T. Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv.org* **2014**, 1408.5093.
- (35) Buda, M.; Maki, A.; Mazurowski, M. A. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks* **2018**, *106*, 249–259.
- (36) Andreeva, A.; Howorth, D.; Chothia, C.; Kulesha, E.; Murzin, A. G. SCOP2 prototype: a new approach to protein structure mining. *Nucleic Acids Res.* **2014**, *42*, D310–D314.
- (37) Dawson, N. L.; Lewis, T. E.; Das, S.; Lees, J. G.; Lee, D.; Ashford, P.; Orengo, C. A.; Sillitoe, I. CATH: an expanded resource to predict protein function through structure and sequence. *Nucleic Acids Res.* **2017**, *45*, D289–D295.
- (38) Mysinger, M. M.; Carchia, M.; Irwin, J. J.; Shoichet, B. K. Directory of useful decoys, enhanced (DUD-E): Better ligands and decoys for better benchmarking. *J. Med. Chem.* **2012**, *55*, 6582–6594.
- (39) Riniker, S.; Landrum, G. A. Open-source platform to benchmark fingerprints for ligandbased virtual screening. *J. Cheminf.* **2013**, *5*, 26.
- (40) Rohrer, S. G.; Baumann, K. Maximum unbiased validation (MUV) data sets for virtual screening based on PubChem bioactivity data. *J. Chem. Inf. Model.* **2009**, *49*, 169–184.
- (41) Bento, A. P.; Gaulton, A.; Hersey, A.; Bellis, L. J.; Chambers, J.; Davies, M.; Krüger, F. A.; Light, Y.; Mak, L.; McGlinchey, S.; Nowotka, M.; Papadatos, G.; Santos, R.; Overington, J. P. The ChEMBL bioactivity database: An update. *Nucleic Acids Res.* **2014**, *42*, 1083–1090.
- (42) Heikamp, K.; Bajorath, J. Large-scale similarity search profiling of ChEMBL compound data sets. *J. Chem. Inf. Model.* **2011**, *51*, 1831–1839.
- (43) Konc, J.; Janežič, D. ProBiS algorithm for detection of structurally similar protein binding sites by local structural alignment. *Bioinformatics* **2010**, *26*, 1160–1168.
- (44) Hanley, A.; McNeil, J. The Meaning and Use of the Area under a Receiver Operating Characteristic (ROC) Curve. *Radiology* **1982**, *143*, 29–36.
- (45) Davis, J.; Goadrich, M. The relationship between Precision-Recall and ROC curves. *ICML* **2006**, 233–240.
- (46) Jain, A. N.; Nicholls, A. Recommendations for evaluation of computational methods. *J. Comput.-Aided Mol. Des.* **2008**, *22*, 133–139.
- (47) Nicholls, A. What do we know and when do we know it? *J. Comput.-Aided Mol. Des.* **2008**, *22*, 239–255.
- (48) Wallach, I.; Dzamba, M.; Heifets, A. AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery. *arXiv.org* **2015**, 1510.02855.
- (49) Gomes, J.; Ramsundar, B.; Feinberg, E. N.; Pande, V. S. Atomic Convolutional Networks for Predicting Protein-Ligand Binding Affinity. *arXiv.org* **2017**, 1703.10603.
- (50) Breuel, T. M. The Effects of Hyperparameters on SGD Training of Neural Networks. *arXiv.org* **2015**, 1508.02788.
- (51) Tiikkainen, P.; Markt, P.; Wolber, G.; Kirchmair, J.; Distinto, S.; Poso, A.; Kallioniemi, O. Critical comparison of virtual screening methods against the MUV data set. *J. Chem. Inf. Model.* **2009**, *49*, 2168–2178.

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

(53) Stierand, K.; Rarey, M. Drawing the PDB: Protein-ligand complexes in two dimensions. *ACS Med. Chem. Lett.* **2010**, *1*, 540–545.

(54) Chen, Y.; Li, J.; Xiao, H.; Jin, X.; Yan, S.; Feng, J. Dual Path Networks *Advances in Neural Information Processing Systems 30 (NIPS 2017)* 2017.