JCIM JOURNAL OF CHEMICAL INFORMATION AND MODELING

Article

Cite This: J. Chem. Inf. Model. 2019, 59, 1163−1171

pubs.acs.org/jcim

Accurate Prediction of Biological Assays with High-Throughput Microscopy Images and Convolutional Networks

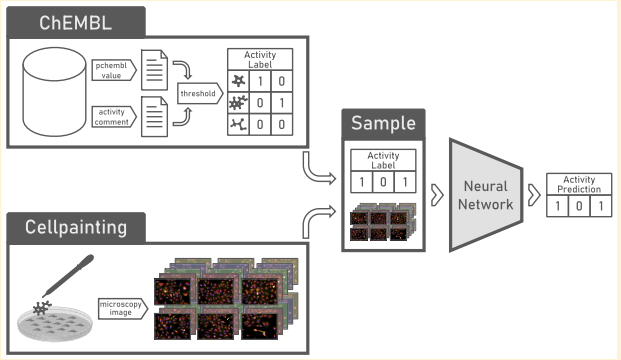
Markus Hofmarcher,[†] Elisabeth Rumetshofer,[†] Djork-Arné Clevert,[‡] Sepp Hochreiter,[†] and Günter Klambauer*,[†]

[†]LIT AI Lab & Institute for Machine Learning, Johannes Kepler University, Linz 4040, Austria

[‡]Bayer AG, Berlin 13353, Germany

Supporting Information

ABSTRACT: Predicting the outcome of biological assays based on high-throughput imaging data is a highly promising task in drug discovery since it can tremendously increase hit rates and suggest novel chemical scaffolds. However, end-to-end learning with convolutional neural networks (CNNs) has not been assessed for the task biological assay prediction despite the success of these networks at visual recognition. We compared several CNNs trained directly on high-throughput imaging data to a) CNNs trained on cell-centric crops and to b) the current state-of-the-art: fully connected networks trained on precalculated morphological cell features. The comparison was performed on the Cell Painting data set, the largest publicly available data set of microscopic images of cells with approximately 30,000 compound treatments. We found that CNNs perform significantly better at predicting the outcome of assays than fully connected networks operating on precomputed morphological features of cells. Surprisingly, the best performing method could predict 32% of the 209 biological assays at high predictive performance (AUC > 0.9) indicating that the cell morphology changes contain a large amount of information about compound activities. Our results suggest that many biological assays could be replaced by high-throughput imaging together with convolutional neural networks and that the costly cell segmentation and feature extraction step can be replaced by convolutional neural networks.



INTRODUCTION

High-throughput fluorescence microscopy imaging (HTI) is an increasingly important biotechnology in the field of drug discovery.[1,2] It captures morphological changes induced by chemical compounds on cell cultures in a very cost- and time-efficient way[3] and is viewed as a potential remedy for the current drug discovery crisis.[4] However, exploiting the wealth of information contained in microscopy images for drug discovery is still a challenge,[5] and it is important to lower entry barriers for the analysis of HTI data.[6]

HTI analysis has been successfully used in various settings in drug discovery[7] using conventional image analysis. A typical step of conventional image analysis pipelines is to extract features from these images by segmenting them into single cells and subsequently extracting cell-level feature-vectors.[8] Individual steps in the pipeline of such approaches usually require optimization of the segmentation and feature extraction procedure to the specific cell culture or assay. This is a time-consuming process in which each step potentially introduces errors and uncertainty. Furthermore, parameters for each step are typically adjusted independently of subsequent steps.[9,10] Thus, learning a model in an end-to-end fashion directly from images would be highly desirable since it would remove the difficult, time-consuming, and computationally demanding segmentation and feature extraction steps.

Predicting biological assays on the basis of high-throughput microscopy data has first been undertaken on a large scale by Simm et al.[5] and has led to a tremendous increase of hit rates, 250-fold and 60-fold, in two ongoing drug discovery projects. In principle, the authors replaced the chemical features of quantitative structure−activity relationship (QSAR) models with image-derived features. In their work, several machine learning methods have been compared, and deep fully connected networks with image-derived features performed best. However, convolutional neural networks using the raw images as input were not tested.

Since 2012, convolutional neural networks (CNNs) have been shown in several applications to outperform conventional methods in the field of image analysis especially where large data sets are available. CNNs can have a vastly better performance than expert systems at classifying images into thousands of categories,[11−13] recognizing traffic signs[14] and

Special Issue: Machine Learning in Drug Discovery

Received: October 1, 2018
Published: March 6, 2019

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171

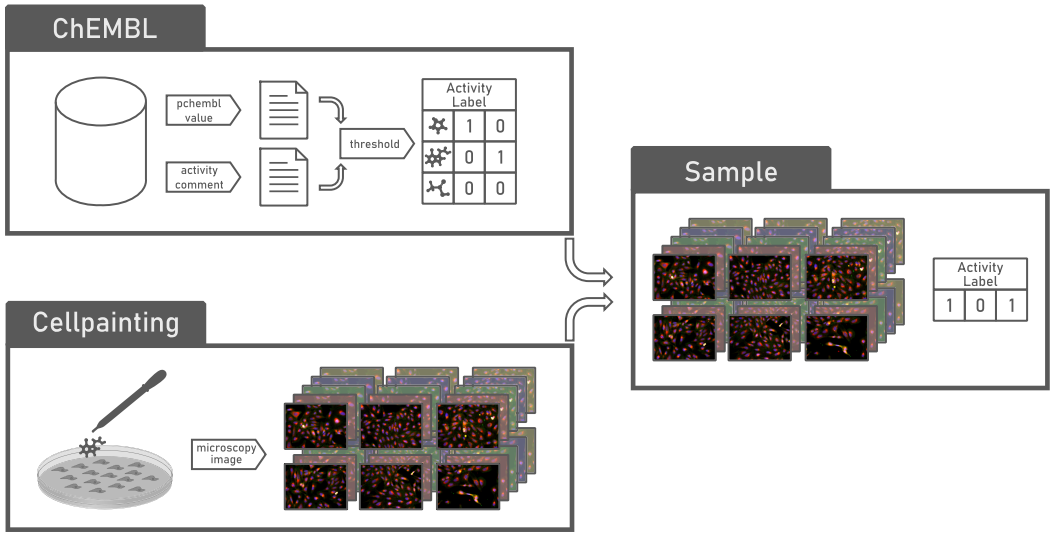


Figure 1. Label extraction and data preparation pipeline of our study. Activity labels are extracted from ChEMBL via two separate queries, *pChEMBL* and *activity comment*, then thresholded, filtered, and combined for all compounds matching with those in the Cell Painting data set. Images and corresponding activity labels are then combined and used for training our models.

even pixel-accurate segmentation of images.[15] As CNNs learn features from images automatically they can easily be adapted to other domains and indeed have been applied to cellular image data, for example, to segment cells achieving higher accuracy than human experts.[16] Therefore, CNNs are a promising method for extracting biological knowledge from high-throughput imaging data for the purpose of drug design.

However, applying CNNs to HTI data in the field of drug design poses unique challenges. State-of-the-art CNN architectures are designed for and evaluated on benchmark data sets such as ImageNet[17−19] in which images are of a much lower resolution than high-throughput microscopy images. This poses a problem as increased image resolution immediately results in vastly increased memory consumption, especially for very deep architectures, making training difficult with full resolution microscopy images (Kraus et al.,[20] Godinez et al.[21]) while scaling or cropping such images results in loss of information. Another challenge arises from the fact that labels for microscopy images are often noisy, as typically whole images are assigned a label by experts but not all individual cells conform to this label. In Kraus et al.[20] the authors combine methods from Multiple Instance Learning and convolutional neural networks to alleviate this problem and in the process learn to focus on correctly labeled cells. Overall, the main challenges of high-throughput image analysis of cells are handling the high resolution of the images and that typically the whole image rather than a single cell is labeled.

We use the largest public data set, Cell Painting,[22] which consists of high-throughput fluorescence microscopy images of cells treated with chemical compounds, to benchmark and compare convolutional neural networks against each other and against the best feature-based method. This data set comprises 919,265 five-channel microscopy images across 30,610 tested compounds as well as single-cell features extracted using a CellProfiler[8] pipeline. We augmented this data set with drug activity data for 10,574 compounds integrated from ChEMBL.[23] With this data, HTI images from the Cell Painting data set and activity data from ChEMBL, we train networks to classify a compound as either active or inactive against an entire panel of targets given a single HTI image. However, we do not restrict activity information from ChEMBL to specific assays as that would reduce the amount of data available drastically. Therefore, the resulting labels must be viewed as noisy, and learning meaningful patterns from this data set is no easy task.

We introduce a novel network architecture that is able to cope with the characteristics of typical HTI data, GapNet. Our architecture extracts features from full resolution images such that there is no need for scaling, cropping, or segmenting the input images. It then combines features from different levels of abstraction before feeding the resulting features into a fully connected classification network. We combine the features in such a way that the network can handle arbitrarily large input images similar to Kraus et al.[20]

In the section Methods, we provide details on the data sets, we describe competing state-of-the-art methods previously used for similar data and the evaluation criteria, and we introduce our novel network architecture. Finally, in sections Results and Discussion we present and discuss results.

METHODS

Data Set. To assess our method, we used a data set released by Bray et al.[22] which we refer to as the Cell Painting data set. This data set contained 919,265 five-channel microscopy images (using the U2OS cell-line) across 30,610 tested compounds as well as single-cell features extracted using a CellProfiler[8] pipeline. For our comparison we used the most recent version of the extracted features, where for each image 1,783 features were extracted.

The fact that the cells in each image were treated with a specific chemical compound allowed us to automatically obtain labels for this data set without the need of labeling by a human expert. We identified each chemical compound in the large bioactivity database ChEMBL.[23] In this way, we obtained labels for this data set as drug activity data for 10,574 compounds across 209 assays (we only included assays for which at least 10 active and 10 inactive measurements exist). Consequently, we created a large data set of high-throughput images together with chemical activity data.

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171

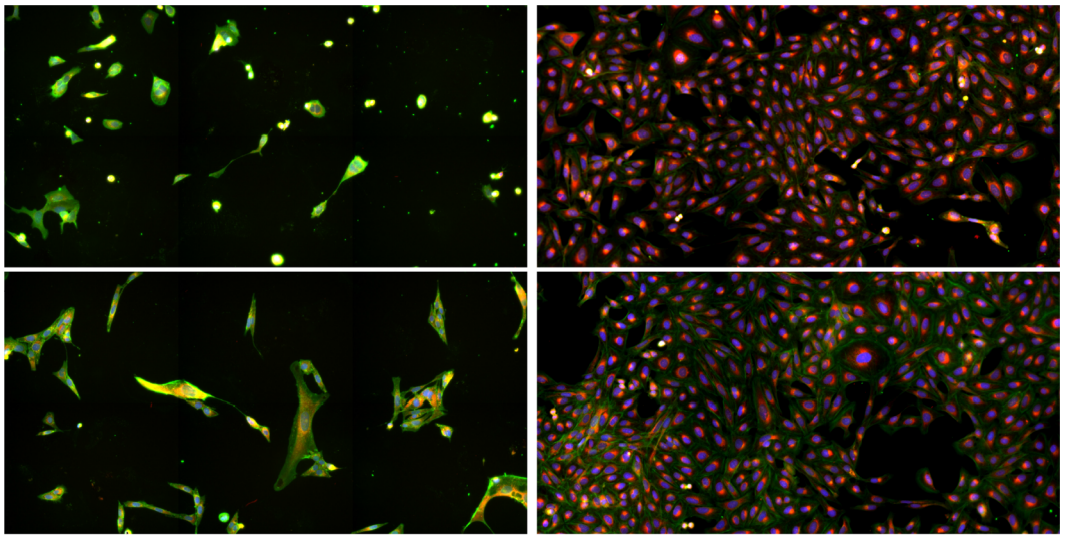


Figure 2. Illustrative examples from the assay "Gametocytocidal compounds screen" that are (a) labeled as active and (b) labeled as inactive. Cells treated with *active* compounds are decreased in number and show a distinct morphology, which is clearly visible on the images. However, cell segmentation and feature extraction can be hampered because of the strong morphological changes as optimizing parameters to be robust to these rare outliers is hard and time-consuming. The samples shown here are reconstructed by combining six view images.

Figure 1 illustrates the pipeline we used to create bioactivity labels for the images of the Cell Painting data set.

We first converted the SMILES representation of the compounds of the Cell Painting data set into the InChiKey[24] format. We then queried the ChEMBL database for compounds matching these InChiKeys resulting in 11,585 hits. For these compounds we extracted two values from the database, namely *pChEMBL*, a numerical value on log scale indicating bioactivity, and *activity comment*, where the researcher or lab technician creating the entry on ChEMBL marked a compound as either active or inactive.

We obtained the first part of our label matrix by extracting all measurements with a pChEMBL value between 4 and 10 for IC50 (inhibitory effect) or EC50 (stimulatory effect). We resolved duplicate entries by averaging over all measurements. However, due to the high amount of noise in these labels,[25] we aimed at binary prediction tasks (active/inactive) and therefore thresholded this matrix similar to other works on assay or target prediction.[26] We applied not only one but three increasing thresholds, namely 5.5, 6.5, and 7.5. As the pChEMBL value is on a log scale, these thresholds represent increasingly higher activity indicators. Applying all three thresholds allowed us to obtain more labeling information, hence we concatenated the three resulting matrices along the assay dimension. This means that an assay can occur multiple times in the final label matrix but at different thresholds.

The second part of our label matrix we obtained by extracting all compounds with a valid *activity comment*. We only allowed a fixed set of comments (such as "active" or "inactive" with slight variations in spelling and casing). Here, we resolved duplicate entries via majority voting. We combined the results of both the thresholded pChEMBL values as well as the *activity comment* along the assay dimension. Then, we filtered this matrix to only allow assays, or in case of the first part assay/threshold combinations, where at least 10 active and 10 inactive compounds are present and removed compounds without any measurements in the remaining assays. The final label matrix consisted of 10,574 rows corresponding to compounds, 209 columns corresponding to binary prediction tasks with 0.87% positive labels (active), 1.64% negative labels (inactive), and 97.49% missing labels (NA). Supporting Information Figure S4 shows the distribution of the number of data points per assay over all splits. We released our label matrix containing bioactivity data for the Cell Painting data set, for which images and precalculated features are publicly available, as Supporting Information.

From the Cell Painting data set we extracted images corresponding to compounds for which we had activity information in our label matrix. As the data set contained multiple instances per compound, we had several images per row in the label matrix. Furthermore, each instance comprised six adjacent images, called *views*, with a resolution of 692 × 520 pixels and 5 channels, each channel corresponding to a stain used for the microscopy screen. These images have been imaged using an ImageXpress Micro XLS automated microscope at 20× magnification. We convert the raw 16bit TIFF files to 8bit to reduce data loading time. In doing so we also remove extreme outlier pixel values by removing the 0.0028% highest pixels prior to this conversion. Furthermore, we normalize each image individually to mean zero and a standard deviation of one. This strategy can be viewed as illumination correction. Since individual views already contained a large number of cells, we did not combine these images to obtain one large image per screen but rather used each view image individually for training and only combined the network outputs by averaging predictions. Figure 2 shows examples of images from the assay "Gametocytocidal compounds screen" labeled as (a) active and (b) inactive (for illustrative purposes, each image is a full screen image comprised of six views).

The final data set consisted of 284,035 view images which we split into training, validation, and test sets, making sure that multiple images from the same sample are in the same fold. We generated three different data set splits using 70% of compounds for training, 10% as a validation set, and the remaining 20% as held out test set for our experiments.

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171

Due to the sparseness of the label matrix, the majority of output units for a given sample image had missing labels and should not receive an error signal. Therefore, the loss for all output units for unlabeled assays for a given sample were masked by multiplying it with zero before performing back-propagation to update the parameters of the network during training.

We compared four convolutional neural network architectures that (a) were designed for processing HTI data (MIL-Net[20] and M-CNN[21]) or (b) that we could adapt to our task (ResNet,[13] DenseNet[27]) with a novel architecture we designed specifically for HTI data as well as a fully connected neural network (FNN[5]) baseline and a CNN processing not the full microscopy images but cropped images containing single cells. The FNN received as input the precalculated features included in the Cell Painting data set based on single cells. For all compared methods, we manually optimized their most important hyperparameters on the validation set. The hyperparameters we tuned are described for each model in its respective section below, while hyperparameters such as kernel sizes and number of layers from the original publications are reported in Supporting Information Section S4. We considered this setting as a multitask problem of 209 binary prediction tasks; therefore, all networks comprised 209 output units with sigmoid activations. All methods are optimized using the SGD optimizer as several methods could not be trained successfully using adaptive optimizers. Furthermore, given sufficient resources, SGD converges to solutions on par, often even better, than other optimizers as it finds more robust flat minima.[28] The batch size was chosen such that the video memory of an Nvidia GTX 1080 TI was fully utilized during training. We used the framework PyTorch[29] to implement all models. We released the code to reproduce our experiments on https://github.com/ml-jku/hti-cnn.

Convolutional Multiple Instance Learning (MIL-Net). In Kraus et al.[20] the authors introduced a CNN designed specifically for HTI data with a focus on the problem of noisy labels, i.e. that microscopy images not only contain cells of the target or labeled class but also outliers. The authors proposed to tackle this problem with *multiple instance learning* (MIL), where cells belonging to the class label of an image are identified automatically while the influence of other cells on the result of the model is down-weighted by using a special pooling function called *noisyAND*. The authors implemented their model using a fully convolutional approach (FCN) allowing them to train on full images with noisy labels and applied this model to images of single-cell crops. The model consists of several convolutional and max pooling layers that encode an input image and the MIL pooling layer which directly outputs class predictions. We used a learning rate of 0.01, SGD optimizer with momentum of 0.9, L2 weight decay of 0.0001, and a batch size of 64. The model specific parameter a for the *noisyAND* pooling function was set to 10 as suggested by the authors.

Multi-Scale Convolutional Neural Network (M-CNN). The approach of Godinez et al.[21] processes the input at several different resolutions simultaneously and fuses the resulting feature maps late in the network. Specifically the input is processed by several convolutional layers in parallel at seven scales, from original resolution to downscaled by a factor of 64. Then, the intermediate features are pooled to be of equal size and concatenated before a final convolutional layer with 1×1 kernel to combine them. This architecture was designed specifically for phenotype prediction directly from microscopy images. We used a learning rate of 0.001, SGD optimizer with momentum of 0.9, L2 weight decay of 0.0005, and a batch size of 100.

ResNet. The work of He et al.[13] enabled training of very deep networks with hundreds of layers. This is achieved by using *residual connections*, which are identity connections bypassing several convolutional layers allowing gradients to flow unencumbered through the network. Networks based on this design are state-of-the-art today. We used the variant of ResNet with 101 layers for our comparisons. We used a learning rate of 0.001, SGD optimizer with momentum of 0.9, L2 weight decay of 0.0001, and a batch size of 24. However, this batchsize was only possible on an Nvidia Quadro GV100 with 32GB of video memory. Even with this powerful graphics card, the training time of this model was approximately 13 days.

DenseNet. Densely Connected Convolutional Networks[27] are state-of-the-art for various image processing tasks. The basic idea of DenseNet is to reuse features learned on early layers of a network later in the network. This is achieved by passing output feature maps of a layer to all consecutive layers within a block. A benefit of this design is that it does not have to relearn features several times throughout the network. Hence, the individual convolutional layers have a relatively small number of learned filters. In fact, the variant we used adds only 32 feature maps per layer to this collection of layer outputs. We used the variant DenseNet-121 with a learning rate of 0.01, SGD optimizer with momentum of 0.9, L2 weight decay of 0.0001, and a batch size of 12.

Baseline Fully Connected Network (FNN). In Simm et al.,[5] the best performing method was the fully connected deep multitask neural network trained on precalculated morphological features of cells. We reimplemented this architecture together with the best hyperparameters and used it as a baseline model to compare the convolutional networks against. This network has three hidden layers with 2048 units and the ReLU activation function. We used parameters given by the authors, without dropout schedule because using a fixed dropout rate improved the results.

GapNet. While abstract features from deep convolutional layers are semantically strong, they lack spatial information necessary for detecting or taking into account smaller objects or input features. In Lin et al.[30] the authors leveraged features from multiple scale levels of the convolutional network by combining feature maps via scaling and subsequent 1×1 convolution, while Godinez et al.[21] processed the input at different scales and combined the resulting feature maps late in the network. In Huang et al.,[27] feature maps from all scale levels were carried over to subsequent levels within so-called Dense-Blocks and thus can be reused by higher layers of the network.

We introduce a novel convolutional neural network architecture designed for the unique challenges of analyzing microscopy images for drug design. Since we make heavy use of *Global Average Pooling*, we refer to it as *GapNet*. It consists of an encoder part using convolutional and pooling layers in a way that allows it to process high-resolution images efficiently while not losing information due to downscaling operations in a preprocessing step. Additionally, we use *dilated convolutions* in the deepest layers of the encoder network tuned for a receptive field roughly equal to the dimensions of the input to the network to enable those layers to gather global information

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171

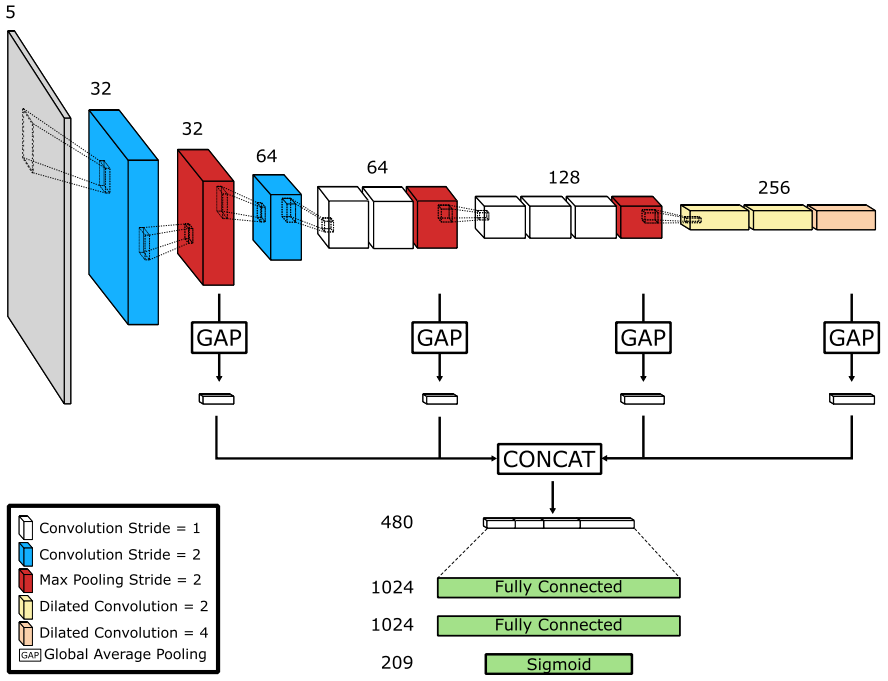


Figure 3. Schematic representation of the GapNet architecture. A standard CNN architecture with a sequence of 2D convolutions and max pooling is combined with global average pooling operations of particular feature maps. The resulting feature vectors are concatenated and fed into two fully connected layers and an output layer.

Table 1. Model Performances in Terms of Different Performance Metrics: Mean AUC, Mean F1 Score, and Number of Tasks (Assays) That Can Be Predicted with an AUC Better than 0.9, 0.8, and 0.7[a]

model	type	training	AUC	F1	AUC > 0.9	AUC > 0.8	AUC > 0.7
ResNet	CNN	end-to-end	**0.731 ± 0.19**	0.508 ± 0.30	68	94	119
DenseNet	CNN	end-to-end	**0.730 ± 0.19**	0.530 ± 0.30	61	98	121
GapNet	CNN	end-to-end	**0.725 ± 0.19**	0.510 ± 0.29	63	94	117
MIL-Net	CNN	end-to-end	0.711 ± 0.18	0.445 ± 0.32	61	81	105
M-CNN	CNN	end-to-end	0.705 ± 0.19	0.482 ± 0.31	57	78	105
SC-CNN	CNN	cell-centric	0.705 ± 0.20	0.362 ± 0.29	61	83	109
FNN	FNN	cell-centric	0.675 ± 0.20	0.361 ± 0.31	55	71	90

[a]CNNs operating on full high-resolution images significantly outperform SC-CNN which operates on cell-centric crops and the FNN operating on precalculated features. The three best methods perform on par. All CNN-based methods outperform the previous state-of-the-art methods: FNNs trained on precalculated morphological features of cells. Performance values marked in bold indicate that the best performing method does not significantly outperform the respective method.

which has been shown to be beneficial for some tasks. Features extracted by this convolutional encoder at different scales are then reduced via averaging over the individual feature maps, in effect producing feature statistics from different layers of abstraction and spatial resolution. These pooled features are then concatenated and processed by several fully connected layers. Figure 3 shows the architecture with the parameters of the variant used for our experiments. We used the SELU[31] activation function throughout the network except for the output layer where we used the *sigmoid* activation function. As for hyperparameters, we used a learning rate of 0.01, SGD optimizer with momentum of 0.9, L2 weight decay of 0.0001, and a batch size of 128.

Single-Cell CNN (SC-CNN). We included a convolutional neural network operating on small crops centered on single cells, SC-CNN. This method depends on a segmentation algorithm that had already identified the cell centers.[22] We extracted cropped images of size 96 × 96 centered on individual cells, whose coordinates were taken from this precalculated cell-segmentation included in the Cell Painting data set. These dimensions were chosen such that the majority of cells is contained within a cropped image regardless of orientation. Due to this greatly reduced input size, we modified the GapNet architecture slightly such that the dilated convolutions in the last block of the encoder have been replaced by regular convolutions as the large receptive field is detrimental for such small images. We used the same data set split as for all other methods, resulting in 12.7 million crops for training, with 1.8 million and 3.6 million crops in the validation and test sets, respectively. During training of the cell-centric model we randomly sampled a subset of 20% from the training-set each epoch. We used a learning rate of 0.01, SGD optimizer with momentum of 0.2, L2 weight decay of 0.0001, and a batch size of 2048.

Evaluation Criteria. We used the area under ROC curve (AUC) as main evaluation criterion. This is the most relevant criterion in drug discovery, since compounds are selected from a ranked list for subsequent lab tests. The AUC was calculated per task, such that each method is characterized by 209 performance values across 209 prediction tasks. In addition to

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171

Table 2. p-Values of a Paired Wilcoxon Test with the Alternative Hypothesis That the Column Method Has Outperformed the Row Method[a]

	ResNet	DenseNet	GapNet	MIL-Net	M-CNN	SC-CNN
DenseNet	7.18e-01					
GapNet	8.36e-02	4.09e-02				
MIL-Net	**1.08e-05**	**4.47e-03**	4.90e-02			
M-CNN	**1.17e-08**	**3.01e-05**	**1.13e-03**	6.37e-01		
SC-CNN	**4.70e-08**	**1.20e-05**	**1.29e-04**	9.07e-01	3.22e-01	
FNN	**1.71e-14**	**1.20e-09**	**3.23e-08**	**2.39e-10**	**2.66e-07**	**8.98e-07**

[a]p-values are reported without correction for multiple testing. The test is performed by comparing the 209 task AUCs of two architectures. Significant values with $\alpha = 0.01$ are marked in bold.

AUCs, we also report F1 scores that require that a method provides binary predictions rather than continuous predictions.

The difference between two methods was tested by a paired Wilcoxon test across these 209 AUC values, where the null hypothesis is that the two methods have equal performance.

RESULTS

Method Comparison Across All Assays. We trained three models for each method and averaged all metrics per task. In Table 1 we report mean AUC values and F1 scores over all 209 tasks. Additionally, we calculated the standard deviation of AUC values and F1 scores over the averaged tasks. Additional performance metrics are available in the Supporting Information (Table S1 and Figure S1). Furthermore, we report percentile confidence intervals for each task and method calculated using the bootstrap method in Supporting Information Table S6. We calculated 1,000 bootstrap replicates on the model with median performance from the three replicates per architecture.

The compared methods yielded mean AUCs from 0.675(±0.20) for FNNs to 0.731(±0.19) for ResNet and mean F1 scores ranging from 0.361 ± 0.31 for FNN to 0.530 ± 0.30 for DenseNet, see Table 1 and Supporting Information Table S5. In addition to AUC, we also report the F1 score to account for situations in which a method has to provide a binary prediction for such unbalanced classification tasks. All end-to-end and all convolutional networks, including SC-CNN, significantly outperformed the baseline FNN. ResNet, DenseNet, and GapNet significantly outperformed SC-CNN and FNN. There were no significant differences between the top three performing methods ResNet, DenseNet, and GapNet (see Table 2).

However, GapNet exhibits the lowest training time of all convolutional neural networks, and MIL-Net has the lowest number of parameters (see Table 3). The fast training (and also inference) time can be explained by the drastically lower number of MACs, multiply accumulate operations, required to calculate the model output compared to other architectures.

Table 3. Number of Parameters and GMACs (Giga Multiply Accumulate Operations) of the CNN Architectures[a]

architecture	parameters	time per epoch	GMACs
GapNet	3,694,545	662.04	1.72
MIL-Net	726,180	1344.18	4.46
M-CNN	11,230,929	4252.47	7.55
DenseNet	7,174,353	8519.91	21.19
ResNet	42,934,673	13208.64	58.48

[a]Runtime (in seconds per epoch) was measured on a Nvidia GTX 1080 TI.

Method Comparison at the Assay Level. We also investigated whether the difference in predictive performance of the image-based GapNet and the feature-based FNN is related to the type of the modeled assay. For this comparison we picked the first experimental replicate for these two architectures. Overall, we found that the predictive performance of these two approaches is highly correlated across assays (Pearson correlation of 0.72, see Figure 4). Oftentimes, there is no significant difference in performance, which we determined using Venkatraman's test for difference in ROC curves.[32] This test is a distribution-free permutation test based on rank statistics to determine the significance of the difference of ROC curves. The method tests the hypothesis that two curves are identical for all operating points. However, for 29 assays, GapNet is significantly better than the FNN, indicating that CNNs detect morphological characteristics that are not captured by the precalculated features.

DISCUSSION

We have investigated the difference in predictive performance between Convolutional Neural Networks operating on raw images and Fully connected Neural Networks operating on features extracted from images using a traditional image-processing pipeline. Our results indicate that CNNs are able to extract morphological changes of cells from images since all CNN models outperformed the FNN baseline which is based on segmenting cells and subsequent feature extraction. Our results demonstrate that the complicated and costly cell segmentation and feature extraction step is not necessary but rather should be skipped to obtain better predictive performance. Since even the previous state-of-the-art method by Simm et al.[5] has led to a strong increase in hit rates, we envision that our suggested approach could even further improve these.

With CNNs and high-throughput fluorescence imaging, many new relations between cell morphology and biological effects could be detected and used to annotate chemical compounds at low cost. Figure 5 shows examples treated with compounds predicted as active from the assay "Gametocytocidal compounds screen" (see A28 in Figure 4) where the CNN performs significantly better than the FNN. Here we see that—while the overall cell density is not significantly different from samples treated with inactive compounds—some cells show clear morphological changes which might be indicative for the classification of the compound as *active* in the respective assay. Such drastic changes can be a problem for segmentation and feature extraction algorithms, which might result in missed detection of these features. Even if these few abnormal cells are detected, their signal may be lost when

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171

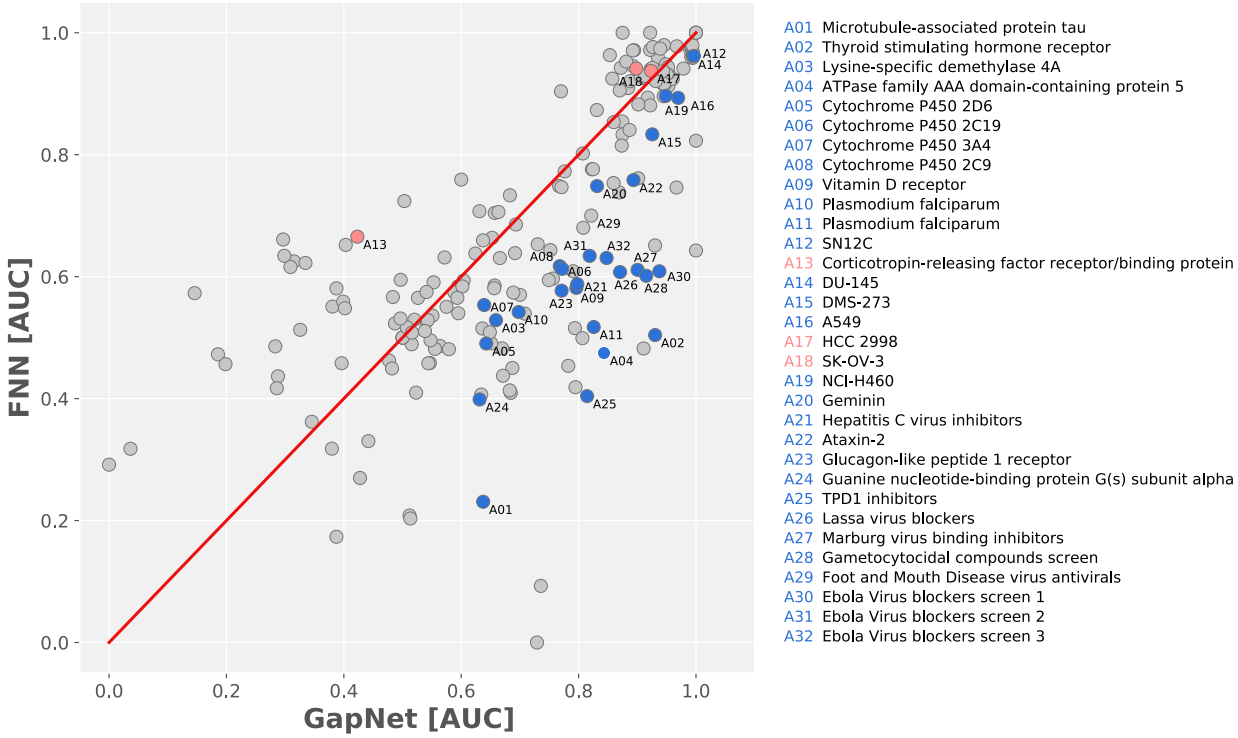


Figure 4. Comparison of FNN performance with GapNet over all tasks. The *x*-axis displays the performance in terms of AUC of GapNet at the 209 assays, whereas the *y*-axis shows the performance of FNN. For a large number of assays, there is no significant difference in predictive performance (gray dots), whereas for 29 assays, GapNet significantly outperforms FNN (blue dots) and for 3 assays FNN significantly outperforms GapNet (red dots). Overall, predictive performance of the two compared methods is highly correlated across assays, which indicates that if a biological effect expresses in morphological changes in the cells, both approaches capture it to a certain degree.

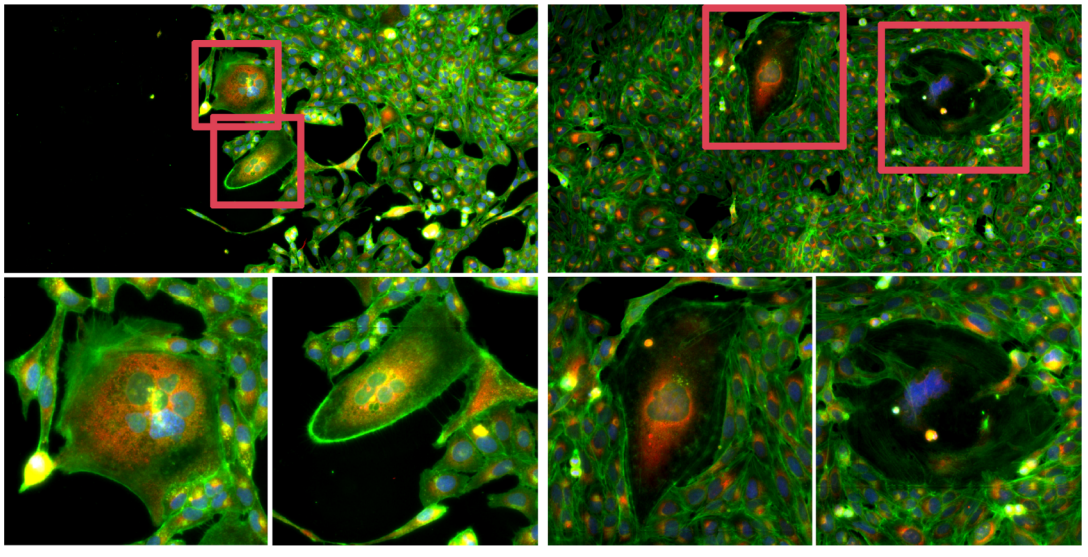


Figure 5. Examples from the assay "Gametocytocidal compounds screen" treated with compounds predicted and labeled as active. While cell density is not significantly different from untreated samples, cells show distinct morphological changes. These changes can hamper segmentation and feature extraction algorithms, thus ignoring these indicative cells.

averaging across cells.[5] We hypothesize that the increased performance of CNNs arises from problems of cell segmentation and detection of sparse signals.

The fact that multiple subimages, i.e. views, are typically available per compound, offers the possibility to incorporate the variability of the predictions as confidence measure for the prediction. Furthermore, for some compounds multiple replicates are available that could be used in a similar way. Also methods emerging from Bayesian Deep Learning, such as Monte Carlo-Dropout,[33] could be a possibility to obtain confidence intervals. We consider the incorporation, evaluation, and assessment of confidence measures for our approach as a promising line of future research.

We also investigated the relationship between the number of data points per task as well as the ratio of active vs inactive compounds per task and the predictive performance of our

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171

models (see Supporting Information Figures S2 and S3). There are mainly two effects: first, that the variability of the performance metric is high when there are few data points and, second, an increase of performance with the number of data points. However, this effect is not as pronounced as with models based on chemical structure.[26] We hypothesize that the reason for this is that there is another main factor influencing the predictive performance of the models: the cellular system. Our approach relies on morphological changes of cells upon compound treatment that can be imaged. Certain compound-induced biological effects do not lead to morphological changes of the imaged cells and therefore limit the predictive performance of image-based models.

The fact that the Cell Painting assay protocol has been published and similar images could be produced in many laboratories across the world opens the opportunity that our trained network could be used to automatically annotate these images. With the currently available data, we were able to annotate ≈30,000 compounds in approximately 60 assays at high predictive performance (AUC > 0.9), which amounts to approximately 1.8 million lab tests. We envision that this work could make worldwide drug discovery efforts faster and cheaper.

■ ASSOCIATED CONTENT

Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jcim.8b00670.

Label data generated from ChEMBL, additional performance metrics (Table S1 and Figure S1), plots showing the relation of model performance and label data (Figure S2, Figure S3), distribution of available data points per task (Figure S4), architecture configuration (Figure S5), and detailed network descriptions for benchmarked architectures (Section S4) (PDF)

AUC values per task and method (Table S5) and bootstrap confidence intervals for all tasks and methods (Table S6) (ZIP)

■ AUTHOR INFORMATION

Corresponding Author

*E-mail: klambauer@ml.jku.at.

ORCID

Günter Klambauer: 0000-0003-2861-5552

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This project was supported by Bayer AG with Research Agreement 09/2017, Audi Electronic Venture GmbH with Research Agreement 12/2016, and by the LIT grant LIT-2017-3-YOU-003. We thank the NVIDIA Corporation for the GPU donations, IWT research grant IWT150865 (Exaptation), and the Audi.JKU Deep Learning Center.

■ REFERENCES

(1) Pepperkok, R.; Ellenberg, J. High-throughput Fluorescence Microscopy for Systems Biology. *Nat. Rev. Mol. Cell Biol.* **2006**, *7*, 690−696.

(2) Starkuviene, V.; Pepperkok, R. The Potential of High-content High-throughput Microscopy in Drug Discovery. *Br. J. Pharmacol.* **2007**, *152*, 62−71.

(3) Yarrow, J.; Feng, Y.; Perlman, Z.; Kirchhausen, T.; Mitchison, T. Phenotypic Screening of Small Molecule Libraries by High Throughput Cell Imaging. *Comb. Chem. High Throughput Screening* **2003**, *6*, 279−286.

(4) Dorval, T.; Chanrion, B.; Cattin, M.-E.; Stephan, J. P. Filling the Drug Discovery Gap: Is High-content Screening the Missing Link? *Curr. Opin. Pharmacol.* **2018**, *42*, 40−45.

(5) Simm, J.; Klambauer, G.; Arany, A.; Steijaert, M.; Wegner, J. K.; Gustin, E.; Chupakhin, V.; Chong, Y. T.; Vialard, J.; Buijnsters, P.; Velter, I.; Vapirev, A.; Singh, S.; Carpenter, A. E.; Wuyts, R.; Hochreiter, S.; Moreau, Y.; Ceulemans, H. Repurposing High-throughput Image Assays Enables Biological Activity Prediction for Drug Discovery. *Cell Chem. Biol.* **2018**, *25*, 611−618.

(6) Scheeder, C.; Heigwer, F.; Boutros, M. Machine Learning and Image-based Profiling in Drug Discovery. *Curr. Opin. Syst. Biol.* **2018**, *10*, 43−52.

(7) Wawer, M. J.; Li, K.; Gustafsdottir, S. M.; Ljosa, V.; Bodycombe, N. E.; Marton, M. A.; Sokolnicki, K. L.; Bray, M.-A.; Kemp, M. M.; Winchester, E.; Taylor, B.; Grant, G. B.; Hon, C. S.-Y.; Duvall, J. R.; Wilson, J. A.; Bittker, J. A.; Dančík, V.; Narayan, R.; Subramanian, A.; Winckler, W.; Golub, T. R.; Carpenter, A. E.; Shamji, A. F.; Schreiber, S. L.; Clemons, P. A. Toward Performance-diverse Small-molecule Libraries for Cell-based Phenotypic Screening using Multiplexed High-dimensional Profiling. *Proc. Natl. Acad. Sci. U. S. A.* **2014**, *111*, 10911−10916.

(8) Carpenter, A. E.; Jones, T. R.; Lamprecht, M. R.; Clarke, C.; Kang, I. H.; Friman, O.; Guertin, D. A.; Chang, J. H.; Lindquist, R. A.; Moffat, J.; Golland, P.; Sabatini, D. M. CellProfiler: Image Analysis Software for Identifying and Quantifying Cell Phenotypes. *Genome Biol.* **2006**, *7*, R100.

(9) Finkbeiner, S.; Frumkin, M.; Kassner, P. Cell-Based Screening: Extracting Meaning from Complex Data. *Neuron* **2015**, *86*, 160−174.

(10) Sommer, C.; Gerlich, D. W. Machine Learning in Cell Biology - Teaching Computers to Recognize Phenotypes. *J. Cell Sci.* **2013**, *126*, 5529−5539.

(11) Krizhevsky, A.; Sutskever, I.; Hinton, G. E. ImageNet Classification with Deep Convolutional Neural Networks. *Proceedings of the 25th International Conference on Neural Information Processing Systems (NeurIPS)*; USA, 2012; pp 1097−1105.

(12) Simonyan, K.; Zisserman, A. Very Deep Convolutional Networks for Large-Scale Image Recognition. 2014, arXiv preprint arXiv:1409.1556. https://arxiv.org/abs/1409.1556 (accessed Feb 25, 2019).

(13) He, K.; Zhang, X.; Ren, S.; Sun, J. Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2015; DOI: 10.1109/CVPR.2016.90.

(14) Ciresan, D.; Meier, U.; Schmidhuber, J. A Committee of Neural Networks for Traffic Sign Classification. *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*; 2011; pp 1918−1921, DOI: 10.1109/IJCNN.2011.6033458.

(15) Long, J.; Shelhamer, E.; Darrell, T. Fully Convolutional Networks for Semantic Segmentation. *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2015; pp 3431−3440.

(16) Ronneberger, O.; Fischer, P.; Brox, T. U-Net: Convolutional Networks for Biomedical Image Segmentation. *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*; 2015; pp 234−241, DOI: 10.1007/978-3-319-24574-4_28.

(17) Everingham, M.; Van Gool, L.; Williams, C. K. I.; Winn, J.; Zisserman, A. The Pascal Visual Object Classes (VOC) Challenge. *International Journal of Computer Vision* **2010**, *88*, 303−338.

(18) Lin, T.-Y.; Maire, M.; Belongie, S.; Hays, J.; Perona, P.; Ramanan, D.; Dollár, P.; Zitnick, C. L. Microsoft COCO: Common Objects in Context. *European Conference on Computer Vision (ECCV)*; 2014; pp 740−755, DOI: 10.1007/978-3-319-10602-1_48.

(19) 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. *International Journal of Computer Vision* **2015**, *115*, 211−252.

(20) Kraus, O. Z.; Ba, J. L.; Frey, B. J. Classifying and Segmenting Microscopy Images with Deep Multiple Instance Learning. *Bioinformatics* **2016**, *32*, i52−i59.
(21) Godinez, W. J.; Hossain, I.; Lazic, S. E.; Davies, J. W.; Zhang, X. A Multi-scale Convolutional Neural Network for Phenotyping High-content Cellular Images. *Bioinformatics* **2017**, *33*, 2010−2019.
(22) Bray, M.-A.; Gustafsdottir, S. M.; Rohban, M. H.; Singh, S.; Ljosa, V.; Sokolnicki, K. L.; Bittker, J. A.; Bodycombe, N. E.; Dančík, V.; Hasaka, T. P.; Hon, C. S.; Kemp, M. M.; Li, K.; Walpita, D.; Wawer, M. J.; Golub, T. R.; Schreiber, S. L.; Clemons, P. A.; Shamji, A. F.; Carpenter, A. E. A Dataset of Images and Morphological Profiles of 30 000 Small-molecule Treatments using the Cell Painting Assay. *GigaScience* **2017**, *6*, giw014.
(23) Gaulton, A.; Hersey, A.; Nowotka, M.; Bento, A. P.; Chambers, J.; Mendez, D.; Mutowo, P.; Atkinson, F.; Bellis, L. J.; Cibrián-Uhalte, E.; Davies, M.; Dedman, N.; Karlsson, A.; Magariños, M. P.; Overington, J. P.; Papadatos, G.; Smit, I.; Leach, A. R. The ChEMBL Database in 2017. *Nucleic Acids Res.* **2017**, *45*, D945−D954.
(24) Heller, S.; McNaught, A.; Stein, S.; Tchekhovskoi, D.; Pletnev, I. InChI - The Worldwide Chemical Structure Identifier Standard. *J. Cheminf.* **2013**, *5*, 7.
(25) Cortés-Ciriano, I.; Bender, A. How Consistent are Publicly Reported Cytotoxicity Data? Large-Scale Statistical Analysis of the Concordance of Public Independent Cytotoxicity Measurements. *ChemMedChem* **2016**, *11*, 57−71.
(26) Mayr, A.; Klambauer, G.; Unterthiner, T.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Clevert, D.-A.; Hochreiter, S. Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chem. Sci.* **2018**, *9*, 5441−5451.
(27) Huang, G.; Liu, Z.; van der Maaten, L.; Weinberger, K. Q. Densely Connected Convolutional Networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2017; pp 2261−2269, DOI: 10.1109/CVPR.2017.243.
(28) Chaudhari, P.; Soatto, S. Stochastic gradient descent performs variational inference, converges to limit cycles for deep networks. *2018 Information Theory and Applications Workshop (ITA)*; 2018; pp 1−10, DOI: 10.1109/ITA.2018.8503224.
(29) Paszke, A.; Gross, S.; Chintala, S.; Chanan, G.; Yang, E.; DeVito, Z.; Lin, Z.; Desmaison, A.; Antiga, L.; Lerer, A. Automatic Differentiation in PyTorch. NeurIPS Autodiff Workshop; 2017.
(30) Lin, T.-Y.; Dollár, P.; Girshick, R. B.; He, K.; Hariharan, B.; Belongie, S. J. Feature Pyramid Networks for Object Detection. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*; 2017; pp 936−944, DOI: 10.1109/CVPR.2017.106.
(31) Klambauer, G.; Unterthiner, T.; Mayr, A.; Hochreiter, S. Self-Normalizing Neural Networks. *Advances in Neural Information Processing Systems (NeurIPS)*; 2017; pp 971−980.
(32) Venkatraman, E. S.; Begg, C. B. A Distribution-free Procedure for Comparing Receiver Operating Characteristic Curves from a Paired Experiment. *Biometrika* **1996**, *83*, 835−848.
(33) Gal, Y.; Ghahramani, Z. Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning. *Proceedings of the 33rd International Conference on Machine Learning (ICML)*; 2016.

DOI: 10.1021/acs.jcim.8b00670
J. Chem. Inf. Model. 2019, 59, 1163−1171