Multi-task convolutional neural networks for predicting in vitro clearance endpoints from molecular images

Andrés Martínez Mora  
Imaging and Data Analytics, Clinical Pharmacology & Safety Sciences, R&D, AstraZeneca, Pepparedsleden 1, SE-43183 Gothenburg, Sweden

Vigneshwari Subramanian  
Imaging and Data Analytics, Clinical Pharmacology & Safety Sciences, R&D, AstraZeneca, Pepparedsleden 1, SE-43183 Gothenburg, Sweden

Filip Miljković (filip.miljkovic@astrazeneca.com)  
Imaging and Data Analytics, Clinical Pharmacology & Safety Sciences, R&D, AstraZeneca, Pepparedsleden 1, SE-43183 Gothenburg, Sweden

Research Article

Keywords: Metabolic stability, compound clearance, molecular images, convolutional neural networks, deep learning, pharmacokinetics.

Posted Date: March 29th, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1482632/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License. Read Full License
Abstract

Optimization of compound metabolic stability is a highly topical issue in pharmaceutical research. Accordingly, application of predictive in silico models can potentially reduce the number of design-make-test-analyze iterations and consequently speed up the progression of novel candidate molecules. Herein, we have investigated the question if multiple in vitro clearance endpoints could be accurately predicted from image-based molecular representations. Thus, compound measurements for four commonly investigated clearance endpoints were curated from AstraZeneca internal sources, providing a sound basis for building multi-task convolutional neural network models. Application of several increasingly challenging data splitting strategies confirmed that convolutional neural network models were successful at capturing implicit chemical relationships contained in training and test data, similar to what is commonly observed for structural fingerprints. Furthermore, model benchmarking against state-of-the-art machine learning methods, including deep neural networks, trained with standard structure-based representations revealed on par or increased accuracy of convolutional neural networks with clear benefit of multi-task learning across all clearance endpoints. Our findings indicate that image-based molecular representations can be applied to predict multiple clearance endpoints, suggesting a potential follow-up to investigate model interpretability from molecular images.

Introduction

Prospective drug candidates are typically required to exhibit not only desirable biochemical and physiological effects, but also optimal absorption, distribution, metabolism, and elimination (ADME) characteristics to ensure adequate exposure at the site of action [1, 2]. Drug exposure is a focal component of pharmacokinetics (PK), which plays an essential role in drug discovery and consequently a part in compound attrition during clinical development [3, 4].

A suite of in vitro assays and animal PK studies is commonly applied to evaluate ADME properties of drugs [2]. Among these, drug metabolism describes the rate of a molecule modified by degrading or metabolizing enzymes (i.e., metabolic stability). Metabolic stability can be quantified using intrinsic clearance (CL) which together with volume of distribution defines other, so-called secondary PK parameters. As these parameters eventually define pharmacological and toxicological profiles of compounds, optimization of drug metabolic stability has long been considered a fundamental objective in drug discovery [3]. Hepatic matrices, such as liver microsomes and hepatocytes, are regularly employed to estimate hepatic metabolic stability of compounds for subsequent extrapolation to in vivo [2, 4]. Accurate prediction of CL from hepatocytes and microsomes increases confidence in estimating total in vivo CL in animals, and eventually humans [6]. In addition, early application of in silico tools, such as machine learning, has the potential to triage down the number of compounds proceeded to synthesis and experimental evaluation [7-10]. Traditionally, these models learn from chemical structure which can be encoded as either qualitative (e.g., presence or absence of carbonyl group) or quantitative (e.g., number of hydrogen bond acceptors) descriptors/fingerprints [8, 9]. Due to high experimental variability, but also intrinsic limitations of conventional structural fingerprints, a number of ADME in silico models (e.g.,
different clearance endpoints) suffer from low predictability which ultimately increases the number of design-make-test-analyze (DMTA) iterations, causing the delay in candidate progression [9-11]. While the experimental error is inherently attributed to the conducted experimental procedure, different chemical structure representations could further be explored to potentially increase the information gain responsible for successful predictions.

In recent years, several chemoinformatics studies have investigated use of alternative, low-resolution image-based molecular representations to train predictive convolutional neural network (CNN) models for different tasks. For example, Fernandez et al. explored few two-dimensional representations of molecules with CNNs using known benchmarking data sets that comprised seven nuclear receptor and five toxic stress response endpoints [12]. They concluded that predictability of CNN models was on par with state-of-the-art methodologies, which was further prospectively validated using an external test set from which several compounds with predicted antiandrogen activity were experimentally verified [12]. In another study, Cortés-Ciriano and Bender used compound images to build CNN regression models for the prediction of compound potency against 25 different protein targets and eight citotoxicity endpoints [13]. Different CNN architectures were evaluated against conventional machine learning methods and deep neural networks (DNNs) trained with structural representations. Once more, benchmarked models displayed comparable performance which greatly improved once random forest and CNN model outputs were averaged. Such averaged outputs yielded lower prediction errors compared to individual estimates, which demonstrated complementarity of features extracted from orthogonal methodological frameworks and representations [13]. In a proof-of-concept study, Iqbal et al. evaluated activity cliffs – pairs of structurally similar compounds with large differences in biological activity – against structural analogs that did not form activity cliffs, both of which were represented as image-based representations modelled with CNNs [14]. Molecular images displayed structures of shared cores and exchanged substituents that were assigned one of the two class labels (i.e., ‘forms an activity cliff’ or ‘does not form an activity cliff’). Surprisingly, high prediction accuracy was achieved which enabled mapping of gradient weights from CNN layers back onto the images of test compounds, revealing structural features essential for successful predictions [14]. In a study by Yoshimori, so-called Molecular Topographic Map (MTM) approach was developed, which applied generative topographic mapping of atomic features to derive two-dimensional maps for each chemical compound [15]. In a subsequent benchmarking study, performance of MTM-based CNN models was shown to be equal to or better than state-of-the-art machine learning methods trained on different structural representations [15]. In yet another study by Iqbal et al., CNNs were applied to predict structure-activity relationship characteristics of activity landscapes [16]. Activity landscapes were depicted as visuals of both similarity and activity relationships of compounds. Once again, high prediction accuracy was achieved that distinguished between activity landscapes having significantly different topographies [16]. Evidently, recent advances in deep learning have enabled investigation of novel molecular representations with ultimate aim to further increase model predictability for different molecular endpoints, particularly those with significant impact on DMTA cycle with yet limited prediction accuracy.
Herein, we systematically explored several molecular image representations and configurations to build predictive multi-task CNN models for different in vitro clearance endpoints coming from AstraZeneca internal sources. These models were thoroughly evaluated using a number of increasingly difficult data splitting strategies that are commonly applied in chemoinformatics. Lastly, we benchmarked performance of our single- and multi-task CNN models against the respective configurations of existing machine learning approaches that were trained with established structural fingerprints.

**Methods And Materials**

**Intrinsic clearance**

Experimental data for four clearance endpoints, namely human hepatocyte intrinsic clearance (HH CLint), human liver microsome intrinsic clearance (HLM CLint), rat hepatocyte intrinsic clearance (RH CLint), and dog hepatocyte intrinsic clearance (DH CLint), were extracted from AstraZeneca internal sources.

Standard operating procedure for hepatocyte intrinsic clearance assays (HH CLint, RH CLint, and DH CLint) follows similar protocol, which is outlined as follows:

- Test compound is prepared in 10 mM dimethyl sulfoxide (DMSO) solution at 37°C and then diluted in acetonitrile to 100 μM concentration
- Frozen hepatocytes are warmed up in a 37°C bath and incubated in medium until reaching an approximate viable cell concentration of 1.0x10⁶ cells/mL, before the solution is transferred to incubation well plates
- 2.5 μL of test compound solution is combined with the hepatocyte solution
- At time points of 0.5, 5, 15, 30, 45, 60, 80, 100, and 120 minutes, 20 μL of the solution is transferred to a separate well, centrifugated for 20 minutes at 4000g, and then a percentage of disappearance of the test compound is estimated
- Linear correlation between the percentage of disappearance (in natural logarithm units) and time is drawn from which a slope of the curve \( k \) is extracted to calculate intrinsic clearance with the following equation:

\[
CLint = k \frac{V}{N}
\]

where \( V \) is the volume of incubation (μL) and \( N \) is the number of hepatocytes in the well. Hepatocyte intrinsic clearance is measured in μL/(min x 10⁶ cells) units.

For HLM CLint, following standard operating procedure was applied:
- Test compound is prepared in 10 mM DMSO solution and then diluted in acetonitrile to 100 μM concentration
- Solution containing liver microsomes is prepared at a concentration of 1.1236 mg/mL in 100 mM phosphate-buffered saline solution at pH 7.4
- 222.5 μL of the microsomal solution is mixed with 2.5 μL of the test compound solution and 25 μL of a 10 mM nicotinamide adenine dinucleotide phosphate (NADPH) solution in a separate plate at 37°C
- At time points of 0.5, 5, 10, 15, 20, and 30 minutes, 20 μL of the solution is sampled, centrifuged for 15 minutes at 3000g, and then a percentage of disappearance of the test compound is estimated
- Linear correlation between the percentage of disappearance (in natural logarithm units) and time is drawn from which a slope of the curve $k$ is extracted to calculate intrinsic clearance with the following equation:

$$CL_{int} = k \frac{V}{m}$$

where $V$ is the volume of incubation in μL and $m$ is the amount of microsomal protein in milligrams. Microsome intrinsic clearance is measured in μL/(min x mg) units.

**Data curation**

Compound annotations given as SMILES strings were standardized using RDKit[17] and ChEMBL Structure Pipeline[18] packages in Python, which included removal of salts and solvent molecules, neutralization of acids and bases, conversion of isotopes, removal of stereoisomers and canonicalization of aromatic structures. Only compounds with molecular weight between 200 and 900 Da were retained.

Only measurements within the range of 1 and 300 μL/(min x 10$^6$ cells) were kept for HH, RH, and DH CLint, whereas for HLM CLint this limit was between 3 and 300 μL/(min x mg). Decadic logarithm transformation was applied to convert qualifying experimental values. If a compound was measured several times for a particular endpoint then its mean and standard deviation of measurements were calculated. Standard deviation distribution for each endpoint was used to derive experimental error estimates, which were set at 95th percentile value of distributions [19]. Such derived experimental errors were applied to investigate model strength where root-mean-square error values of model performances were compared to the estimates. To ensure that the compounds with high experimental variability were excluded from the model building, the linear correlation between the mean and standard deviation of measurements was explored. If a compound’s standard deviation was larger than two times the expected standard deviation of the linear fit, then this measurement would be removed [19]. Following these steps, clearance data were proceeded to modelling.

**Image representations**
Canonical SMILES of each compound was converted to a two-dimensional molecular representation using OpenEye library in Python [20]. Image size was fixed at 240 x 240 pixels to accommodate depiction of larger molecules present in the data set, but also to preserve the information gain from smaller compounds. Aromatic bonds were represented using ring-centered circles to maximize the information content and number of pixels associated with chemical bonds.

Default pixel range was between 0 (black) and 255 (white) where pixels of value of zero contained no chemical information. Atoms and bonds were encoded using two alternative molecular image representations. In the first representation, which we termed color coded, different color values were assigned to different atoms (e.g., black for carbon, red for oxygen, blue for nitrogen, etc.). This information was converted in the pixel space so that the atom and bond values retained their original color but were normalized between 0 and 1. Second image representation, named property gradient, consisted of atom and bond properties depicted as grayscale gradients (atomic number, MMFF94 [21] partial charge, bond order, or valence). Pixel values (from 0 to 255) were mapped back to original feature values whereas the empty pixels were assigned the value of zero. Presence of property gradients reduced the number of pixels devoid of chemical information. Images from different properties were then combined in a multi-channel image. Fig. 1 shows an example of image representations for drug imatinib in both color and property gradient representations.

Two different configurations of molecular sizes were explored for image derivation: fixed and scaled. In fixed configuration, consistent resolution of 0.03Å/pixel was applied which ensured equally sized renditions of atoms and bonds for explored molecules. This consequently introduced greater number of pixels devoid of chemical information for small-sized molecules. For the scaled configuration, molecules were adapted to fit the image dimensions so that both atom and bond contents were maximized. Here, small-sized molecules were depicted as relatively larger, as to minimize the number of empty pixels. Fig. 2 depicts both fixed and scaled configurations for drug clozapine.

Following the definition of representations and configurations, different data augmentation strategies were applied to reduce the model bias towards single two-dimensional poses rendered for each molecule. Here, we applied 90° or 180° rotation, vertical flip, and horizontal flip to expand the number of molecular poses and allow better model generalization independent of rendered orientations. During the training process, for each SMILES provided per batch, an augmented version of the original molecular representation was submitted for learning, efficiently carried out by a PyTorch [22] data loader.

**Structural representations**

State-of-the-art structural representations were used to benchmark CNN models built on images. Here, different configurations of extended-connectivity fingerprints (ECFPs) [23] were applied, which included diameters four (ECFP4) and six (ECFP6) hashed to either 1024- or 2048-bit versions. ECFPs were obtained using RDKit [17] package in Python.

**Data splitting strategies**
Prior to model building, clearance data set was divided into the training and test sets using three different strategies given in an increasingly challenging order of predictions: random split, scaffold-based split, and time-based split. For the random split, compounds were randomly divided in respective training and test sets, which may have shared common chemical space given the stochastic nature of this approach. On the other hand, scaffold-based split ensured that the compounds containing same Bemis-Murcko scaffolds [24] were made part of the same (training or test) set. This further reduced chemical similarity between the training and test sets and thus increased prediction difficulty. Further up the scale, time-based split defines a time point threshold by which the data are divided into the training and test set. As our internal data were all associated with sampling dates, the latest date of clearance measurement was defined as the time stamp for each compound. Time thresholds of thirty, twelve, six, and four months were established to evaluate different time split scenarios.

Training sets were used for the model cross-validation and hyperparameter optimization whereas the test sets were employed to independently evaluate final model performances. Three-fold cross-validation was applied for all three data splitting strategies. For both random and scaffold-based splits, 80% of the data set were dedicated for model building while the remaining 20% were defined as a test set. For time-based split, different time thresholds produced different ratios of training-test data with time point of thirty months being the closest to the 80/20 split.

**Loss functions and performance metrics**

Models were optimized using either Huber [25] loss or root-mean-square error (RMSE). Huber loss balances between L1 and L2 losses, where L1 minimizes the sum of all absolute differences between true and predicted values while L2 minimizes the sum of all squared differences between true and predicted values. For individual predictions, it can be calculated using the following:

$$\text{Huber} (y, \hat{y}) = \begin{cases} 
0.5(y - \hat{y})^2 & \text{if } |y - \hat{y}| < \delta \\
\delta(|y - \hat{y}| - 0.5\delta) & \text{if } |y - \hat{y}| \geq \delta
\end{cases}$$

where parameter $\delta$ regulates the application domains of L1 and L2 penalties, set to a default of 1.

The RMSE presents the measure of spread of prediction errors around the line of best fit and can be calculated using the following equation:

$$\text{RMSE}(y, \hat{y}) = \sqrt{\frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)^2}{n}}$$

where $y_i$ is the true value for $i$-th sample, $\hat{y}_i$ is the predicted value for $i$-th sample, and $n$ is the number of samples. Model configuration with lowest RMSE value was proceeded forward.
Model performance was evaluated using both RMSE and coefficient of determination ($R^2$) score. The $R^2$ presents the proportion of variance of dependent variable explained by independent model variables. It provides an indication of goodness of fit and can be defined as:

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

where $\bar{y}$ is the arithmetic average of all true values.

**Convolutional neural networks**

Different CNN architectures, coded in PyTorch [22], were explored to learn *in vitro* clearance endpoints from molecular images. Basic concept of CNNs is as follows: a series of convolutional layers comprised of convolution and pooling operators are repeatedly applied to extract features from different scales before being processed by several fully connected layers (activated by a non-linear function) which end in an output layer corresponding to the number of tasks (e.g., clearance endpoints) [26]. For each training epoch, prediction outputs are evaluated against corresponding experimental values to calculate the training loss which is then minimized using a pre-defined loss function.

Following hyperparameters were optimized: batch size (32, 64, or 128), learning rate ($10^{-3}$ to $10^{-5}$), number of convolutional layers (2-6), number of fully connected layers (1-3). Other hyperparameters remained consistent: number of epochs was 150 with early stopping [27], gradient descent optimization performed with Adam optimizer [28], non-linear activation function set to ReLU [29], batch normalization [30], and dropout [31] rate of 0.5 in fully connected layers.

Hyperparameters were initialized either randomly or from pre-trained networks that were built using a commonly applied ImageNet [32] data set. The latter approach, commonly known as transfer learning, is often practiced in computer vision to speed up convergence and improve predictability of novel tasks. Weight imports from these architectures could be partial, as in a simple configuration which we termed SimpleNet, where only the convolutional operators were acquired from a pre-trained VGG [33], or total, where all the pre-trained weights from architectures Chemception [34], ResNet [35], and EfficientNet [36] were applied. As we primarily focused on multi-task predictions, CNNs were optimized in a multi-task mode to derive optimal configuration which was later used to train single-task CNN variants to enable fair comparison between single and multi-task learning.

**Machine learning for structural representations**

Machine learning methods frequently employed in ECFP-learning tasks, such as random forest [37], support-vector regressor [38], and deep neural network [39], were optimized and validated using these
structural representations. To enable fair comparison between image- and structure-based representations, the training-test splits were kept consistent across different splitting strategies.

Random forest (RF) is an ensemble of decision trees each built from a randomly sampled subset of training data, procedure commonly known as bootstrapping. Herein, we optimized number of decision trees (50, 100, or 150) and minimum number of samples left in terminal nodes (1-5) using Scikit-learn [40] implementation.

Support vector regressor (SVR) maps training data (or features) in high-dimensional space (often using a non-linear kernel function), as close as possible to their quantitative output (or label) to derive a regression function. Hyperplane margin (ε) controls tolerable deviation between the observed and predicted values, and larger errors are penalized. Hyperparameter C, also known as regularization slack variable, also regulates the relaxation of error minimization problem by penalizing large deviations from the ε tube. The SVRs were optimized for hyperparameters ε (0.001, 0.01, and 0.1), C (0.1, 1, and 10) and a kernel function (radial basis or Tanimoto [41]). These as well were implemented via Scikit-learn [40].

Deep neural network (DNN) applies a series of learned matrix multiplications and non-linear activation functions in units known as neurons, commonly organized in several fully connected layers. Loss function (such as one of those described previously) evaluates the difference between predicted and experimental values and minimizes it via an iterative learning process. The DNNs were implemented in PyTorch [22] and optimized for batch size (32, 64, 128, 256, and 512), learning rate ($5 \times 10^{-5}, 10^{-4}, 5 \times 10^{-4}$), neuron and layer configuration ([1000, 500], [1500, 750], [2000, 1000], [4000, 2000], [4000, 2000, 1000], or [5000, 2500]), and dropout rate (between 0.25 and 0.5).

Models were optimized with Ray Tune [42] library which implements Bayesian Optimization with HyperBand (BOHB) [43] algorithm, for our purposes set at 30 sampled experiments. Methodological framework of RFs and SVRs allows only for single-task learning, whereas DNNs can be trained in both single- and multi-task fashion.

**Results And Discussion**

**Clearance data**

All compounds associated with experimental data for HH CLint, HLM CLint, RH CLint, and DH CLint, were obtained from internal AstraZeneca sources and curated to combine assay data which followed equivalent experimental procedures (vide supra). Basic statistics for the curated intrinsic clearance data are summarized in Tab. 1.

As we were interested to explore multi-task modelling for the four clearance endpoints, we derived a [compound, endpoint] matrix where each compound corresponded to its curated experimental value for a particular endpoint. In total, we identified 139,907 compounds with at least a single clearance endpoint measurement that qualified for our analysis. As common in experimental sciences, not all the
compounds were investigated for all the clearance endpoints, resulting in only 36.91% of the potential compound-endpoint pairs available for model building and evaluation. Out of 139,907 compounds, only 922 (0.66%) had clearance measured for all four endpoints, whereas 79,470 (56.70%), more than half of the curated compounds, had only a single clearance value reported. This revealed that our data set consisted of compounds spanning different stages of drug development. This is further corroborated by the individual clearance statistics in Tab. 1. Most measured clearance experiments upon compound synthesis are HLM CLint and RH CLint, which were present for 101,913 and 79,043 in our data set, respectively. This is followed by HH CLint (21,616 compounds) and DH CLint (6,747 compounds) which are evaluated comparatively later.

Our curated clearance set was also characterized with high chemical diversity. In total, we detected 64,262 Bemis-Murcko scaffolds, with an average of 2.18 compounds per scaffold. Compound-to-scaffold ratio for each individual endpoint (Tab. 1) corresponded to global average. About 34.18% (or 47,821) of compounds were singletons, i.e., they contained a scaffold that was not shared with any other compound. Value distribution for all clearance endpoints had a wide range and was typical of compounds present at different stages of drug development. The HH CLint and DH CLint had overall the lowest distribution average, main characteristic of compounds with optimized clearance characteristics. Errors estimated from clearance measurements were consistent with the standard two-fold margin expected for clearance experiments.

Altogether, qualified compounds and clearance measurements were of adequate quality to warrant further exploration via machine learning.

**Evaluation of transfer learning, image configurations, and representations**

Prior to evaluating different CNN architectures and hyperparameters, we first investigated the impact of transfer learning, image configurations, and representations on model accuracy using cross-validation results.

Fig. 3 details the performance of SimpleNet models trained on either scaled or fixed image configuration, with or without transfer learning. To fairly compare the effect of transfer learning and different image configurations, other parameters were kept consistent: color coded images, maximum of 150 training epochs, four convolutional layers, two fully connected layers with 0.5 dropout rate, batch size of 128, learning rate of $10^{-3}$, RMSE as a loss function, Adam optimizer, and batch normalization. As shown on the top (Fig. 3), average $R^2$ scores for all clearance endpoints were greater for transfer learning models than for those initialized with random weights (without transfer learning). This held true for both scaled and fixed image configurations where transfer learning increased $R^2$ score by 0.36 and 0.40, respectively. This difference was less evident for RMSE (Fig. 3, bottom) where transfer learning models displayed slightly lower values ($\Delta$RMSE = 0.10 for scaled and $\Delta$RMSE = 0.11 for fixed). Model accuracy was only marginally better for fixed configuration. Based on these results, we further focused our investigation on models trained using transfer learning and fixed image configuration.
Next, we analyzed the performance of SimpleNet models trained with different molecular representations (Fig. 4). Once more, other parameters remained unchanged for just comparison: fixed image configuration trained with transfer learning, maximum of up to 150 training epochs, four convolutional layers, two fully connected layer with dropout rate of 0.5, batch size of 128, learning rate of $10^{-3}$, RMSE as a loss function, Adam optimizer, and batch normalization. Although we initially defined two image representations, color coded and property gradient, we also evaluated individual components of multi-channel property gradient representation: atomic number, partial charge, bond order, and valence. As shown in Fig. 4, color coded representation showed a slight advantage over individual property gradients, with an average of $DR^2 = 0.08$ and $DRMSE = 0.03$. On the other hand, multi-channel property gradient, a representation which combines all four property gradients, scored similarly to color coded images. Taken all together, differences between individual property gradients and multi-channel representation were minor. This demonstrated that plain, single channel images were often a good substitute for more complicated, multi-channel representations. Once again, RMSE scores revealed very minor differences among evaluated representations. Given their simplicity and initial performance, we further proceed with color coded images for CNN architecture assessment.

**Evaluation of CNN architectures and hyperparameters**

Now that we identified transfer learning models with fixed configuration and color coded image representation as a preferential setup, we continue by investigating various, hyperparameter optimized CNN architectures using cross-validation results.

**Tab. 2** shows best performing hyperparameter sets per evaluated CNN architecture presented as average $R^2$ and RMSE values across all clearance endpoints. We note that there was a minimal difference among the optimized architectures of different complexity and depth. For instance, SimpleNet as an architecture featuring the lowest number of convolutional layers (five layers with two convolutional operators per layer) had only moderately lower performance ($R^2 = 0.48$, RMSE = 0.36) compared to 152-layered ResNet ($R^2 = 0.53$, RMSE = 0.34). However, compared to unoptimized SimpleNet models shown in Fig. 3 and Fig. 4, hyperparameter optimization provided a clear benefit. In addition, for the same CNN architectures, greater number of hidden layers did not necessarily provide an increase in model performance. While SimpleNet and ResNet performance correlated with depth, Chemception and EfficientNet did not, and here shallower variants often provided comparable performances. In case of EfficientNet, not all batch sizes were tested for all available model depths (from B0 to B7, going from the shallowest to the deepest) due to memory issues, thus potentially limiting the advantage of deeper models. Taking this into account, EfficientNet B2 was preferred over much deeper variants for the upper batch size limit that was possible to test across all model depths. Interestingly, a certain number of optimized hyperparameter variables was consistent for the majority of different CNN architectures. Here, Huber loss, learning rate of $2 \times 10^{-4}$, batch size of 128, and one fully connected layer were generally preferred. Altogether, ResNet with hyperparameter setup given in **Tab. 2** was chosen to further assess the performance of hold-out test sets originating from different data splitting strategies.
Model performance

Next, we built and evaluated ResNet models using previously defined data splitting strategies: random, scaffold-, and time-based splits. Here, we first divided complete clearance data into the training and test sets for each split. Then, we built multi-task models using the designated training sets. Finally, we validated each model using the respective hold-out test sets.

Fig. 5 shows model accuracy for individual clearance endpoints as evaluated on 20%-worth of test data for random, scaffold-, and time-based splits (last 30 months). Designated order of splitting strategies corresponds to an increase in prediction difficulty, where training-test data for random split is expected to have greater chemical overlap than time-based training-test data of equivalent ratio. As shown in Fig. 5, model performances decline in the expected order of prediction difficulty. On average, only a minor accuracy decrease was noted between the random and scaffold-based splits ($\Delta R^2 = 0.05$ and $\Delta RMSE = 0.03$), where RH CLint recorded the greatest ($\Delta R^2 = 0.07$ and $\Delta RMSE = 0.03$) and DH CLint the smallest drop ($\Delta R^2 = 0.02$ and $\Delta RMSE = 0.02$). Significant performance decrease was observed for time-base splits with an average decline of $\Delta R^2 = 0.29$ and $\Delta RMSE = 0.11$ compared to the random split. Here, HH CLint and RH CLint showed the greatest change in $R^2$ and RMSE, respectively ($\Delta R^2 = 0.43$ and $\Delta RMSE = 0.14$), whereas DH CLint and HLM CLint showed the smallest change in $R^2$ and RMSE, respectively ($\Delta R^2 = 0.13$ and $\Delta RMSE = 0.09$). It should be noted that the time threshold of 30 months (2.5 years) is not commonly evaluated in practice. Yet, for the purpose of comparison between different data splitting strategies, 20% of the data were chosen to form hold-out test sets, which corresponds to a 30-month threshold for the time-based split. Interestingly, the model performance correlated with the expected increase in prediction difficulty, imposed by different data splitting strategies. This indicates that CNN models could extract implicit chemical information from molecular images that were relevant for predictions. As chemical overlap with each strategy narrows, it becomes more difficult for CNNs to exhibit good performance.

This is further supported by the performance results coming from time-based splits at different time thresholds. Fig. 6 depicts changes in the model accuracy as the time threshold becomes gradually relaxed, from initially displayed thirty months (Fig. 5) to twelve, six, and four months. As the time threshold decreases so does the model accuracy improve across all four clearance endpoints. Clearance models at a 30-month threshold displayed overall the lowest accuracy, which was followed by the 12-, six-, and four-month thresholds. Minimal differences in the performance of six- and four-month threshold models were due to highly similar composition of their data sets. As these models only differ by two-months’ worth of data (either in the training set as for the four-month threshold, or in the test set as for the six-month threshold), model performances may display either slight increase or decrease. For example, we observed a decrease in performance for the four-month threshold models for HH CLint and DH CLint, which are commonly the least frequently measured endpoints. We note that the trend displayed in Fig. 6 reflects the common observation made for structure-based models, that in practice require continuous updates to retain reasonable model performance for the latest chemical data.
Benchmarking with structural representations

Finally, we compared the performance of single- and multi-task CNN models against a number of different machine learning methods (RF, SVR, DNN) that were trained using structure-based representations. Provided that RF and SVR support only single-task and DNNs both single- and multi-task configuration, single- and multi-task CNNs were evaluated against respective, equivalent setups.

Fig. 7 compares the model performance (hold-out test set from the same random split) of multi-task DNNs, trained with a structural representation chosen via hyperparameter optimization, and multi-task CNNs trained on molecular images. Optimal DNN models were trained using 2048-bit ECFP4 fingerprints with the following set of hyperparameters: batch size of 64, learning rate of $10^{-4}$, two hidden layers with [4000, 2000] neuron configuration, RMSE as a loss function, and dropout rate of 0.4. This setup was used to train both single- and multi-task DNNs. Both CNN and DNN models displayed overall good performance, in line with the observations made in practice. We note that molecular images demonstrated a minor advantage over conventional structure-based fingerprints (Fig. 7). An average improvement in model performance was only $\Delta R^2 = 0.03$ and $\Delta \text{RMSE} = 0.01$ with HLM CLint showing the greatest and HH CLint the lowest increase in accuracy. This shows that the multi-task CNNs trained on molecular images are, the very least, on par with the multi-task DNNs which use structural fingerprints commonly applied in predicting similar tasks. Further, we looked at the estimated experimental errors for each of the clearance endpoints. We conceived experimental errors as the upper bound limits of potentially achievable model performances, given that the models should not be able to predict more accurately than the error margins imposed by the experiments. As shown in Fig. 7, three out of four clearance endpoints (HH CLint, HLM CLint, and RH CLint) had experimental errors that were still considerably lower than multi-task model performances. In the case of DH CLint, multi-task CNN model achieved accuracy that was only marginally worse than its estimated experimental error ($\Delta \text{RMSE} = 0.03$). This suggests that further modelling of HH CLint, HLM CLint, and RH CLint data could potentially result in the reduction of performance-error margin, whereas for DH CLint this would not yield tangible improvement.

However, a different picture emerged for the comparison of single-task models. Fig. 8 shows model performance for RF, SVR, single-task DNN, and single-task CNN models as evaluated on hold-out test sets from the same random split. The optimal RF was trained on 2048-bit ECFP4 fingerprints, using an ensemble of 100 bootstrapped decision trees with a minimum of one sample per terminal node. On the other hand, optimal SVR models used 2048-bit ECFP6 fingerprints, radial basis kernel, hyperplane margin $\varepsilon$ of 0.01, and regularization slack variable C of 1. We applied these optimal configurations (RF and SVR) across all individual clearance endpoint models. As depicted in Fig. 8, single-task CNNs demonstrated equivalent or inferior performance compared to the other machine learning methods trained on structural representations. Here, SVR models displayed the best performance for three out of four clearance endpoints, except HLM CLint where single-task CNN showed only a minor advantage. Of the three machine learning methods that utilized structural fingerprints, RF showed overall the lowest performance that was on average weaker by $\Delta R^2 = 0.08$ and $\Delta \text{RMSE} = 0.03$ compared to the best performing models.
Clear disadvantage of single-task CNNs was observed for HH CLint ($R^2 = 0.45$, RMSE = 0.35) and DH CLint ($R^2 = 0.37$, RMSE = 0.40) models which were among the lowest performing. Compared to multi-task CNNs (Fig. 7), single-task equivalents for HH CLint and DH CLint recorded an $R^2$ decrease of 0.13 and 0.15, respectively. Similarly, the RMSE increased by 0.03 and 0.04, respectively. This demonstrates the advantage of multi-task learning for endpoints with a limited number of data points for model building. For the other two clearance endpoints (HLM CLint and RH CLint), multi-task learning provided only a minor performance increase. Although SVRs showed overall the best accuracy in a single-task setting, methodological framework of neural networks enables knowledge transfer between the tasks, which ultimately enhances performance of individual endpoints, especially for those with fewer data points. Given the high similarity among the clearance endpoints, their joint knowledge transfer becomes even more emphasized. Altogether, CNN models learning from molecular images showed comparable performance to standard machine learning methods trained on structural representations. Moreover, multi-task learning showed greater advantage over single-task variants, suggesting it as preferential setup for modelling multiple clearance endpoints.

**Conclusion**

In this study, we have investigated the use of multi-task convolutional neural networks to predict a series of *in vitro* clearance endpoints using different molecular image representations. Prediction of *in vitro* ADME parameters, including different clearance endpoints, is commonly achieved through the use of structure-based fingerprints that encode chemical structure as fixed-length binary vectors which indicate presence or absence of particular substructural patterns. With the recent advent of novel deep learning architectures, end-to-end representation learning together with iterative model refinement have expanded capabilities of predictive models beyond pre-defined, knowledge-based structural representations.

To evaluate this, we have trained convolutional neural networks using low-level two-dimensional image representations of molecules to predict four *in vitro* clearance endpoints commonly applied in AstraZeneca. First, we evaluated the impact of transfer learning, different image configurations, and representations to choose appropriate modelling setup for the follow-up evaluation of different CNN architectures and hyperparameters. We then applied different data splitting strategies to assess the performance of selected model configuration. Finally, we benchmarked our single- and multi-task CNN models to respective machine learning methods that were trained using conventional structural fingerprints, typically employed in learning tasks such as clearance prediction. Based on these results, we conclude that CNNs were successful at deriving implicit chemical information from raw molecular images, resulting in models with on par accuracy as state-of-the-art machine learning approaches using standard structure-based fingerprints. Interestingly, performance of CNN models reflected changes in prediction difficulty associated with designated data splitting strategies, similar to the observations made for structure-based machine learning models typically applied in practice. This was further supported by model performances estimated at different time thresholds. While experimental error estimates revealed
that model accuracy could still be potentially improved, it remains questionable whether further in-depth investigation of different modelling aspects would yield measurable increase in performance. In addition, we demonstrated that multi-task learning greatly improved predictability of our models, particularly those with limited number of accessible data points.

Although structure-based representations are generally less complex than molecular images, several potential advantages of image representations should be mentioned. First, CNN models could potentially extract plausible chemical features that extend beyond fixed-length structural rules. Derivation of novel molecular features via imaging could improve the learning capability of chemoinformatics-based models where conventional structural features show only limited success. However, such models often require a large corpus of data to confidently engineer learnable features. Secondly, in cases where insufficient modelling data exists, transfer learning could be applied to fine-tune pre-trained models for increased endpoint predictability. This approach has been successfully employed for a variety of deep learning applications, including the study discussed herein. Lastly, features extracted from molecular images could be mapped back onto two-dimensional structures of molecules to investigate their relevance for compound design. Applied to clearance endpoints, feature contribution maps could suggest potential structural modifications to lower metabolic clearance and consequently increase stability of compounds. These, and other potential applications of CNNs to molecular images, will be in our focus for the coming period.

**Declarations**

**Notes**

The authors declare no competing financial interest.

**References**

1. Kola I (2008) The state of innovation in drug development. Clin Pharmacol Ther 83:227–230. https://doi.org/10.1038/sj.clpt.6100479
2. Williamson B, Colclough N, Fretland AJ, Jones BC, Jones RDO, McGinnity DF (2020) Further considerations towards an effective and efficient oncology drug discovery DMPK strategy. Curr Drug Metab 21:145–162. https://doi.org/10.2174/1389200221666200312104837
3. Masimirembwa CM, Bredberg U, Andersson TB (2003) Metabolic stability for drug discovery and development: Pharmacokinetic and biochemical challenges. Clin Pharmacokinet 42:515-528. https://doi.org/10.2165/00003088-200342060-00002
4. Davies M, Jones RDO, Grime K, Jansson-Löfmark R, Fretland AJ, Winiwarter S, Morgan P, McGinnity DF (2020) Improving the accuracy of predicted human pharmacokinetics: Lessons learned from the AstraZeneca drug pipeline over two decades. Trends Pharmacol Sci 41:390–408. https://doi.org/10.1016/j.tips.2020.03.004
5. Morgan P, Brown DG, Lennard S, Anderton MJ, Barrett JC, Eriksson U, Fidock M, Hamrén B, Johnson A, March RE, Matcham J, Mettetal J, Nicholls DJ, Platz S, Rees S, Snowden MA, Pangalos MN (2018) Impact of a five-dimensional framework on R&D productivity at AstraZeneca. Nat Rev Drug Discov 17:167–181. https://doi.org/10.1038/nrd.2017.244

6. Williamson B, Harfinger S, McGinnity DF (2020) Evaluation of the disconnect between hepatocyte and microsome intrinsic clearance and in vitro in vivo extrapolation performance. Drug Metab Dispos 48:1137–1146. https://doi.org/10.1124/dmd.120.000131

7. Oprisiu I, Winiwarter S (2020) In: Wolkenhauer O (ed) Systems medicine: Integrative, qualitative and computational approaches, 1st edn. Academic Press

8. Lombardo F, Desai PV, Arimoto R, Desino KE, Fischer H, Keefer CE, Petersson C, Winiwarter S, Broccatelli F (2017) In silico absorption, distribution, metabolism, excretion, and pharmacokinetics (ADME-PK): Utility and best practices. An industry perspective from the international consortium for innovation through quality in pharmaceutical development. J Med Chem 60:9097–9113. https://doi.org/10.1021/acs.jmedchem.7b00487

9. Winiwarter S, Ahlberg E, Watson E, Oprisiu I, Mogemark M, Noeske T, Greene N (2018) In silico ADME in drug design - Enhancing the impact. ADMET DMPK 6:15-33. https://doi.org/10.5599/admet.6.1.470

10. Miljković F, Martinsson A, Obrezanova O, Williamson B, Johnson M, Sykes A, Bender A, Greene N (2021) Machine learning models for human in vivo pharmacokinetic parameters with in-house validation. Mol Pharm 18: 4520-4530. https://doi.org/10.1021/acs.molpharmaceut.1c00718

11. Winiwarter S, Middleton B, Jones B, Courtney P, Lindmark B, Page KM, Clark A, Landqvist C (2015) Time dependent analysis of assay comparability: A novel approach to understand intra- and intersite variability over time. J Comput Aided Mol Des 29:795–807. https://doi.org/10.1007/s10822-015-9836-5

12. Fernandez M, Ban F, Woo G, Hsing M, Yamazaki T, LeBlanc E, Rennie PS, Welch WJ, Cherkasov A (2018) Toxic Colors: The use of deep learning for predicting toxicity of compounds merely from their graphic images. J Chem Inf Model 58:1533-1543. https://doi.org/10.1021/acs.jcim.8b00338

13. Cortés-Ciriano I, Bender A (2019) KekuleScope: Prediction of cancer cell line sensitivity and compound potency using convolutional neural networks trained on compound images. J Cheminform 11:e41. https://doi.org/10.1186/s13321-019-0364-5

14. Iqbal J, Vogt M, Bajorath J (2021) Prediction of activity cliffs on the basis of images using convolutional neural networks. J Comput Aided Mol Des 35:1157-1164. https://doi.org/10.1007/s10822-021-00380-y

15. Yoshimori A (2021) Prediction of molecular properties using molecular topographic map. Molecules 26:e4475. https://doi.org/10.3390/molecules26154475

16. Iqbal J, Vogt M, Bajorath J (2020) Activity landscape image analysis using convolutional neural networks. J Cheminform 12:e34. https://doi.org/10.1186/s13321-020-00436-5
17. RDKit (2022) RDKit: Open-Source Cheminformatics and Machine Learning Software. https://www.rdkit.org/. Accessed 14 Feb 2022

18. Bento AP, Hersey A, Félix E, Landrum G, Gauton A, Atkinson F, Bellis LJ, De Veij M, Leach AR (2020) An open source chemical structure curation pipeline using RDKit. J Cheminform 12:e51. https://doi.org/10.1186/s13321-020-00456-1

19. Wenlock MC, Carlsson LA (2015) How experimental errors influence drug metabolism and pharmacokinetic QSAR/QSPR models. J Chem Inf Model 55:125−134. https://doi.org/10.1021/ci500535s

20. OEChem Toolkit, version 2.0.0, OpenEye Scientific Software: Santa Fe, NM

21. Halgren TA (1996) Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94. J Comput Chem 17:490-519. https://doi.org/10.1002/(SICI)1096-987X(199604)17:5/6<490::AID-JCC1>3.0.CO;2-P

22. Paszke A, Gross S, Massa F, Lerer A, Bradbury J, Chanan G, Killeen T, Lin Z, Gimelshein N, Antiga L, Desmaison A, Xamla AK, Yang E, DeVito Z, Raison Nabla M, Tejani A, Chilamkurthy S, Ai Q, Steiner B, Fang L, Bai J, Chintala S (2019) PyTorch: An imperative style, high-performance deep learning library. arXiv:1912.01703

23. Rogers D, Hahn M (2010) Extended-connectivity fingerprints. J Chem Inf Model 50:742–754. https://doi.org/10.1021/ci100050t

24. Bemis GW, Murcko MA (1999) Properties of known drugs. 2. Side chains. J Med Chem 42:5095-5099. https://doi.org/10.1021/jm9903996

25. Huber PJ (1964) Robust estimation of a location parameter. Ann Math Statist 35:73-101. https://doi.org/10.1214/aoms/1177703732

26. LeCun Y, Bengio Y, Hinton G (2015) Deep Learning. Nature 521:436–444. https://doi.org/10.1038/nature14539

27. Prechelt L (2012) In: Montavon G, Orr GB, Müller K-R (eds) Neural Networks: Tricks of the trade, 2nd edn. Springer, Berlin, Heidelberg

28. Kingma DP, Ba JL (2014) Adam: A Method for stochastic optimization. arXiv:1412.6980

29. Agarap AF (2018) Deep learning using rectified linear units (ReLU). arXiv:1803.08375

30. Ioffe S, Szegedy C (2015) Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv:1502.03167

31. Srivastava N, Hinton G, Krizhevsky A, Salakhutdinov R (2014) Dropout: A simple way to prevent neural networks from overfitting. J Mach Learn Res 15:1929–1958.

32. Deng J, Dong W, Socher R, Li L-J, Li K, Fei-Fei L (2009) ImageNet: A large-scale hierarchical image database. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit: 248–255. https://doi.org/10.1109/CVPR.2009.5206848

33. Simonyan K, Zisserman A (2014) Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556
34. Goh GB, Siegel C, Vishnu A, Hodas NO, Baker N (2017) Chemception: A deep neural network with minimal chemistry knowledge matches the performance of expert-developed QSAR/QSPR models. arXiv:1706.06689

35. He K, Zhang X, Ren S, Sun J (2015) Deep residual learning for image recognition. arXiv:1512.03385

36. Tan M, Le QV (2019) EfficientNet: Rethinking model scaling for convolutional neural networks. arXiv:1905.11946

37. Breiman L (2001) Random forests. Mach. Learn. 45:5–32. https://doi.org/10.1023/A:1010933404324

38. Vapnik VN (2000) The Nature of Statistical Learning Theory. Springer, New York

39. Bengio Y (2009) Learning Deep Architectures for AI. Found Trends Mach Learn 2:1-127. http://dx.doi.org/10.1561/2200000006

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

41. Ralaivola L, Swamidass SJ, Saigo H, Baldi P (2005) Graph kernels for chemical informatics. Neural Netw 18:1093–1110. https://doi.org/10.1016/j.neunet.2005.07.009

42. Liaw R, Liang E, Nishihara R, Moritz P, Gonzalez JE, Stoica I (2018) Tune: A research platform for distributed model selection and training, arXiv:1807.05118

43. Falkner S, Klein A, Hutter F (2018) BOHB: Robust and efficient hyperparameter optimization at scale. arXiv:1807.01774

Tables

Tab. 1. Post-curation clearance statistics. For each intrinsic clearance endpoint, number of compounds qualifying for the analysis, number of Bemis-Murcko scaffolds, distribution of measured values, and experimental error estimate are provided. Sign # denotes ‘Number of…’.

| Endpoint  | #Compounds | #Scaffolds | Value distribution (log_{10}) | Experimental error (log_{10}) |
|-----------|------------|------------|-------------------------------|-----------------------------|
| HH CLint  | 21,616     | 10,121     | 0.99±0.48                     | 0.22                        |
| HLM CLint | 101,913    | 49,501     | 1.47±0.48                     | 0.25                        |
| RH CLint  | 79,043     | 38,537     | 1.36±0.56                     | 0.27                        |
| DH CLint  | 6,747      | 3,526      | 1.05±0.51                     | 0.33                        |

Tab. 2. Evaluation results for optimized CNN architectures. For each investigated CNN architecture, a set of optimized hyperparameter variables and resulting averaged performance across all endpoints (R^2 and RMSE) scores is provided.
Figures

**Figure 1**

**Molecular image representations of imatinib.** Two image representations used in convolutional neural network model building are exemplified using marketed drug imatinib: **(A)** color coded representation of atoms and bonds and **(B)** separate images with gradient representation of atomic number, partial charge, bond order, and valence channels that are subsequently combined to form a multi-channel image.
Figure 2

**Image configurations for clozapine.** Shown are both fixed (A) and scaled (B) image configurations of marketed drug clozapine that were applied in convolutional neural network model building.

Figure 3
Impact of transfer learning and different image configurations. Bar plots show average $R^2$ (top) and RMSE (bottom) performance across all clearance endpoints using SimpleNet models trained on scaled and fixed image configurations, with (scaled in blue, fixed in orange) or without transfer learning (scaled in grey, fixed in yellow).
Figure 4

**Evaluation of different image representations.** Bar plots show average $R^2$ (top) and RMSE (bottom) performance across all clearance endpoints using SimpleNet models trained on different molecular representations: color coded, atomic number, partial charge, bond order, valence, and multi-channel property gradient.

Figure 5

**Model performances on different data splits.** Shown are bar plots of model performances ($R^2$, top; RMSE, bottom) for each clearance endpoint, as evaluated on 20% test sets from random (blue), scaffold- (orange), and time-based (grey) splits. Time-based test set consists of compounds measured in the last 30 months.

Figure 6

**Model performances on different time-based splits.** Shown are the line charts of $R^2$ (left) and RMSE (right) model performances for HH CLint (blue), HLM CLint (orange), RH CLint (grey), and DH CLint (yellow) evaluated at different time thresholds, from earliest to latest (left to right).
Benchmarking of multi-task models. Bar plots show $R^2$ (top) and RMSE (bottom) model accuracies for four clearance endpoints as evaluated by multi-task DNNs (blue) and CNNs (green). Experimental error bars (black) are provided for each endpoint alongside RMSE model performances.
Figure 8

Benchmarking of single-task models. Bar plots show $R^2$ (top) and RMSE (bottom) model accuracies for four clearance endpoints as evaluated by RF (blue), SVR (orange), single-task DNN (grey), and single-task DNN (yellow).
