Machine Learning Tools For off-Target Early Safety Assessment of Small Molecules In Drug Discovery (Single Task Neural Networks Vs Automated Machine Learning)

Doha Naga

F. Hoffmann-La Roche Ltd. <https://orcid.org/0000-0002-1679-5333>

Wolfgang Muster

F. Hoffmann-La Roche Ltd. <https://orcid.org/0000-0002-9832-6257>

Eunice Musvasva

F. Hoffmann-La Roche Ltd. <https://orcid.org/0000-0003-1581-3349>

Gerhard F. Ecker (✉ gerhard.f.ecker@univie.ac.at)

Department of Pharmaceutical Sciences, Austria. <https://orcid.org/0000-0003-4209-6883>

Research article

Keywords: Drug discovery, Safety screening, Off-target panel, Class imbalance, Deep learning, Automated machine learning (AutoML), Ensembling methods.

Posted Date: October 11th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-957525/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Abstract

Unpredicted drug safety issues constitute the majority of failures in the pharmaceutical industry according to several studies[1-3]. Some of these preclinical safety issues could be attributed to the non-selective binding of compounds to targets other than their intended therapeutic target, causing undesired adverse events. Consequently, pharmaceutical companies including Roche, routinely run in-vitro safety screens to detect off-target activities prior to preclinical and clinical studies.

Hereby we present a machine learning framework aiming at the prediction of our in-house 50 off-target panel[4] activities for ~ 4000 compounds, directly from their structure. This framework is intended to guide chemists in the drug design process prior to synthesis and accelerate drug discovery. It incorporates different ML approaches such as deep learning and automated machine learning. Outcomes from different methods are compared in terms of efficiency and efficacy. The most important challenges and factors impacting model construction and performance in addition to suggestions on how to overcome such challenges are also discussed.

Introduction

In-depth exploration of drug candidates' attrition through analysis of public or individual in-house data has been an ongoing trend in recent years[1, 3, 5, 6]. Preclinical toxicity has been identified as one of the major reasons behind the failures of candidates in the early phases of drug development[5, 7]. Big efforts have been exerted to minimize undesired effects attributed to one of the key contributors to non-clinical toxicities, poly-pharmacology[8–10]. Poly-pharmacology or off-target activity is the interaction of the drug with targets other than the intended therapeutic target. This interaction may lead to adverse reactions, which can be seen in pre-clinical animal studies and clinical trials. Adverse events might also manifest in later stages such as post-marketing, eventually leading to drug withdrawal from the market[11].

Consequently, pharmaceutical companies routinely run safety in-vitro assays in early stages of drug development where the lead compounds are screened against off-targets and thus assessing the promiscuity of these compounds. A minimal panel of the most significant off targets (~ 44 targets) were presented in the work of Bowes et al, where the relation between their activation/inhibition and undesired physiological effect has been extensively described[12].

Our in-house off- target optimized panel (composed of 50 targets) was presented along with the target's selection methodology and assay protocol by Bendels et al[4]. These routinely conducted safety screens (dating back to 2004) have resulted in an enriched off-target interaction database. Exploiting this historical data and using it to predict the off-target activities of compounds directly from their chemical structure is a key objective of this work. Anticipating early toxicities of chemical structures through in silico models prior to their synthesis could be a useful guide to medicinal chemists. Additionally, using such models in lieu of in vitro assays could reduce in vitro testing and consequently accelerate drug discovery. However, choosing the appropriate computational method is not a simple task due to the

presence of many approaches that vary in algorithms, time consumption, availability, resources, and required skills. In recent years, deep learning approaches have demonstrated success in prediction of bioactivity data over other shallow approaches (e.g. random forest or support vector machine), especially when applied to large datasets, and has ranked among the top performing method for QSAR predictions in the Kaggle and Tox21 challenge [13–15]. It has thus been considered in this contribution.

Another emerging approach in the field of ML is automated machine learning (autoML) which has recently shown a high impact in healthcare research and drug discovery[16]. The major difference between autoML and other machine learning methods is the automatic feature selection and hyper parameter optimization, which are among the trickiest steps of building a ML model. With regard to our project, the 50 off-targets had varying active/inactive ratios and varying number of samples, and therefore different model architecture and hyper parameters tuning were required and should be accounted for, which is automatically established in autoML tools. Furthermore, constructing a neural network often requires considerable computer science skills and experience, while several autoML approaches might require little or no programming expertise and are available as a user interface (e.g H2O driverless AI) or open-source packages, where the models can be executed through few command lines (e.g AutoGluon Tabular). Another advantage of AutoML is the stacking of several models together to obtain an overall best accuracy.

The major challenge of this work is handling the data imbalance, understanding its impact on the model performance, and taking the necessary precautions while preparing the data and constructing the models, regardless of the machine learning method used. In this study, we explore and compare the capability of neural networks and autoML in constructing prediction models for fifty off-targets of different protein classes, different dataset sizes, and high-class imbalance.

Methods

Data extraction The off-target compounds interaction data (in terms of percentage of inhibition-[PCT]) for the 50 targets listed in Table 1 were extracted from our in-house databases according to the following criteria: (a) Percentage of inhibition was measured at 10 μ m concentration (b) Assay types were either radio ligand binding displacement or enzymatic assays (c) Only human recombinant receptors were used in these assays (except for GABA A, Glutamate, Glycine and CA^{2+} channel receptors, where rat brain was used). These safety in-vitro assays were conducted over a period of ~15 years (time interval for the assays collected was between January 2020 and September 2004). The panel comprised diverse target classes: 22 GPCRS, 6 Ion-channels, 5 Kinases, 4 Nuclear-receptors, two Transporters, two other receptors, and 9 further enzymes.

Table 1

Roche optimized off-target panel (50 targets) arranged in a descending order according to total number of compounds screened per target.

Target	Symbol	Compounds screened	Positives	Hit percent
Adenosine A3	ADORA3	2834	832	29.35
Dopamine D2S	D2S	2606	346	13.27
Mu-type opioid	MOP	2573	364	14.14
Muscarinic M1	M1	2533	916	36.16
Seritonin 5-HT2A	5-HT2A	2391	526	21.99
Adenosine A1	ADORA1	2378	102	4.28
NE transporter	NET	2360	393	16.65
Seritonin 5-HT2B	5-HT2B	2316	823	35.53
Dopamine D1	D1	2301	241	10.47
Seritonin -5HT1A	5HT1A	2290	290	12.66
Histamine H1	H1	2286	214	9.36
Muscarinic M2	M2	2253	513	22.76
Adrenergic β 1	ADRB1	2252	60	2.66
Acetylcholineesterase	ACHE	2245	418	18.61
5HT transporter	SERT	2241	799	35.65
GABA A (Cl ⁻ channel)	GABA A (Cl ⁻ channel)	2238	448	20.01
Adrenergic α 1A	ADRA1A	2210	380	17.19
Monoamine oxidase	MAOA	2204	20	0.90
Seritonin 5-HT3	5-HT3	2130	50	2.34
HIV-1 Protease	HIV1-PR	2110	46	2.18
Adrenergic α 2A	ADRA2A	2020	196	9.70
Adrenergic β 2	ADRB2	1973	76	3.85
PPARgamma	PPARG	1955	194	9.92
Ca2+ channel (Diltiazem site)	CACNA1C	1942	473	24.35
Nicotinic muscle-type	CHRNA1	1899	173	9.11

Target	Symbol	Compounds screened	Positives	Hit percent
Prostaglandin F	FP	1897	73	3.84
Histamine H3	H3	1864	215	11.53
Xanthine oxidase	XDH	1831	49	2.67
Glucocorticoid	GR	1816	76	4.18
Cyclooxygenase 2	COX2	1799	192	10.67
Cholestykinin 1	CCK1	1792	139	7.75
Matrix metalloproteinase 9	MMP-9	1787	38	2.12
Angiotensin receptor II	AT1	1762	45	2.55
GABA-A (Benzo)	GABA-A (Benzo)	1645	329	20
Histamine H2	H2	1644	123	7.48
Cannabinoid CB1	CB1	1609	67	4.16
Phosphodiesterase 3B	PDE3B	1563	66	4.22
ZAP70 Kinase	ZAP70	1484	25	1.68
CDK2 Kinase	CDK2	1465	61	4.16
GSK3 beta	GSK3B	1394	65	4.66
Estrogen alpha	ERalpha	1389	7	0.50
ABL1 Kinase	ABL1	1328	236	17.77
GSK3 alpha	GSK3A	1285	97	7.54
Androgen	AR	1234	86	6.96
Glutamate (PCP)	PCP	1187	6	0.50
Glycine receptor (Strychnine insensitive)	Glycine	1145	2	0.17
Nicotinic neuronal-type (alpha-BGTx insens.)	CHRNA4	951	9	0.94
Angiotensin converting enzyme	ACE	832	58	6.97
Kappa-type opioid	KOP	428	89	20.79
Phosphodiesterase 4D2	PDE4D2	133	15	11.27

Hit percent calculation and binary classification A cut-off of 50% inhibition at 10 μ m was drawn in order to classify the compounds' inhibition percentages of the targets into active ($\geq$ 50% binding) or inactive

(<50% binding) upon each target. The hit percent was then calculated according to Eq. 1 and a cut-off of 20% was specified, where a hit percent >20% was defined as high and hit percent <20% as low.

$$\frac{\text{No. of active compounds}}{\text{Total screened}} \times 100 \text{ Eq. 1}$$

Targets with hit percent less than 1 (Glutamate, Glycine and Nicotinic receptor-neuronal type) were removed, since constructing models for such extremely imbalanced data sets would be unfeasible. The final assembled dataset thus consisted of 3928 unique compounds and 47 targets.

Compounds' descriptors Structural features of the compounds were used as descriptors. Extended Circular FingerPrints with radius 4 (ECFP4)[17] were selected in this study since they are considered amongst the best performing features in bioactivity predictions[13, 18–20]. Standardization of canonical smiles and duplicates removal was performed in Pipeline Pilot (version 9.1.0)[21]. Calculations were done using the rcdk package in R version 3.5.1[22] where canonical smiles were parsed into the 'mol' format and the 1024 bits of the ECFP4 fingerprints were generated for each compound. A matrix of 3928 compounds vs 1024 structural binary bits was obtained.

Since compounds are usually screened against the suspected off-targets (not necessarily the full panel), the compounds' off-target interaction matrix was not fully populated and comprised missing values. Thus, a single task modelling approach was adopted, where one model was constructed for each target separately.

Train-Test splits Due to the high imbalance observed in the datasets, and to ensure uniform presence of positive samples in both training and test sets, a stratified activity splitting approach was used, using the caret package in R (version 3.5.1). A ratio of 80:20 was used for the training and test splits, respectively. The 80 percent training set was automatically split by keras into 60% training and 20% validation (i.e. 60% training, 20% validation and 20% test set). Test sets were used as external held out sets and were completely excluded from the training procedure. To allow a comparison between the modelling approaches, the same training and test sets were used for each approach.

Machine learning approaches A deep learning (Feedforward neural network) and two automated machine learning approaches (H2O driverless AI[23] and AutoGluon Tabular[24]) were used to construct the models. Subsequently, a comparative analysis was performed between the three methods used, with respect to their robustness, resources consumed, and expertise needed to build the models.

Deep learning with feed forward neural networks

A feed forward neural network was employed using tensorflow[25] (version 2.3) and keras[26] packages (version 2.3) in R studio (version 1.1.456) using R (version 3.5.1). As explained above, the ECFP4 fingerprints were used as an input layer to the network to predict the binary activity of the molecules in the output layer (inactive = 0, active = 1). A sigmoid activation function was applied to the output layer while the rectified linear unit (ReLU) function was used to activate the input and hidden layers. A drop out

was applied to the input layer and the hidden layers to avoid overfitting, and a penalty was also applied to the input and hidden layer by using a kernel regularizer. The number of hidden units and dropout rate varied according to Table 2. The list of fixed parameters used in the network is presented in Table 3

Table 2 Values of the grid search parameters used in the neural network.

Parameter	Value
Hidden units	[256,512,1024,2048]
Dropout input	[0,0.1,0.2]
Dropout hidden	[0.2,0.3,0.4]
Learning rate	[0.01,0.001,0.0001]
Batch size	[64,128,256]

Table 3 Fixed parameters of the neural network.

Parameter	Value
Optimizer	Adam
Loss	Binary cross entropy
Hidden layers	2
Activation	ReLu
Kernel regularizer l2	0.001
Activation output	Sigmoid

In addition to the binary accuracy function

provided by keras, a custom metric function (balanced accuracy) was created to monitor the training statistics to avoid any misleading outcomes that could be caused by the unbalanced nature of the data. Adam optimizer was employed to minimize the selected loss function (binary cross entropy), with a varying learning rate. The learning rate was further reduced when no improvement was shown in the validation balanced accuracy over 10 epochs, in order to help to reach a global minimum and to minimize random noise.

Since fitting the same neural network for 47 datasets of varying size and hit percent is a complex task, the grid search was performed on a selection of hyper parameters and a network architecture that could fit with these diverse datasets. The variable flags used are indicated in Table 2 and were searched using the tuning function from the tfruns library provided by tensorflow (version 2.3).

A 50% randomized sample of the grid search combinations were trained for 250 epochs (full Cartesian was not performed due to time and resource limits). However, to overcome overfitting, early stopping was applied and training was stopped when no improvement in the validation balanced accuracy was observed for 20 epochs.

In addition to the previous callbacks used during fitting the model (early stopping, reduce learning rate on plateau), model checkpoint call back was also implemented. This means that all the models were saved after each epoch, and only the model with the best validation balanced accuracy was retained. In order to ensure model generalization, a 20% random validation split was applied during the training (therefore the 80% training sets were further divided into 60% training and 20% validation).

Finally, for each target, three models were preserved, with best validation loss, validation balanced accuracy, and validation binary accuracy. Afterwards, the models were further evaluated on the external held out test sets.

AutoML with H2O Driverless Artificial Intelligence (DAI)

H2O autoML models were implemented using the R client driverless artificial intelligence (DAI) library (version 1.8.5.1) in R studio (version 1.1.456) using R (version 3.5.1). A license is required for the access of H2O driverless AI.

Datasets in autoML models require less complex handling than tensorflow models. As mentioned previously, the same data sets for each target were used, except for the difference in the shape of the dataset. Unlike neural network models, where the chemical structures (input) and the binary activities (output) are fed separately to the network, one dataset comprising the structures and the binary activities is directly fed to the H2O system. For each target we uploaded the train and test sets as follows:

```
Train <- dai.upload_dataset ("filepath/MuscarinicM2_train.csv")
```

```
Test <- dai.upload_dataset ("filepath/MuscarinicM2_test.csv")
```

A "target_col" is specified, which is the column to predict; in our case this refers to the 'binary activity' column. H2O automatically detects the column named 'ID' as the identifier column of the dataset (in our case compounds IDs). No feature engineering or hyper parameter optimization is required in H2O autoML. Only few parameters require configuration during the training procedure, namely: accuracy, time, interpretability, and model scorers. As there is no implementation of balanced accuracy within H2O, area under the precision recall curve (AUCPR) is used for training the models. AUCPR is also considered as a suitable metric for unbalanced problems, since it is not affected by the true negatives and has higher sensitivity towards true positives, true negatives, and false positives. Further elaboration on the use of different evaluation metrics and how to overcome evaluation bias is given in the next section.

Models were set to be reproducible and the default expert settings were used.

```
model <- dai.train(training_frame = train, testing_frame = test, target_col = "binary_activity", is_timeseries = FALSE, is_classification = T, accuracy = 10, time = 6, interpretability = 4, seed = 30, enable_gpus = T, config_overrides = "make_mojo_scoring_pipeline = 'off' \n min_num_rows = 50 \n make_python_scoring_pipeline = 'off' \n make_autoreport = false \n ", scorer = "AUCPR")
```

For less complex tasks (e.g. models for targets with high hit percent), different configurations were explored, such as setting a low accuracy threshold (accuracy = 1). The different algorithms implemented in H2O are: XGBoost GBM models, Light GBM models, XGBoost GLM models, TensorFlow models and RuleFit models. A final ensemble of the top performing models is created in order to optimize the overall performance. More details on the models integrated in H2O and the model stacking process can be found in the H2O booklet documentation.

The final ensemble conformation matrices (TP, FP, TN, FN), ensemble rocs (TPR, FPR, Precision, Recall) and ensemble scores (Accuracy, MCC, AUCPR) of the held-out test sets were retrieved for all the 47 target

models in form of .json files and analyzed in R studio 3.5.1. Balanced accuracy was calculated according to Eq. 4.

The process was automated to train and retrieve the results for the models of the 47 targets.

AutoML with AutoGluon Tabular

Similar to the neural networks and H2O autoML, we utilized open source AutoGluon (version 0.0.13) in Jupyter notebook (version 7.12.0) with python (version 3.6.5) to construct 47 models for the 47 off targets. AutoGluon implements several algorithms such as neural networks, LightGBM boosted trees, CatBoost boosted trees, Random Forest, Extremely Randomized Trees, and k-Nearest Neighbors. A multilayer stacking strategy is adopted, where AutoGluon individually fits various models and creates an optimized model ensemble that outperforms all individual models. Further details on AutoGluon's strategy in models training, tuning and ensembling can be found in the work of Erickson et al[24].

Similar to H2O, AutoGluon does not require any data processing. The same train and test csv files (containing the compounds' fingerprints and binary activities) utilized for H2O were used for AutoGluon and are imported as follows:

```
train_data = task.Dataset(file_path=train_filename)
```

```
test_data = task.Dataset(file_path=test_filename)
```

A 'label' column (containing the binary activities) and an 'id_column' (containing compounds ids) are defined. Allocation of the predictive accuracy, disk usage and time required for model training are defined by a list of presets. Several presets are available, such as 'best quality', which trains the models to obtain the best accuracy regardless of training time or disk usage, 'best quality with high quality refit', which is identical to 'best quality' but with slightly lower accuracy and higher efficiency (10x less disk space and 10x faster training times). Other presets with different time and accuracy specifications are available and are explained in detail in the AutoGluon Tabular documentation. We selected the option 'best quality with high quality refit' to ensure a reasonable balance between accuracy and efficiency. No time limit specification is needed for this option (assigned automatically by AutoGluon). Best models are also automatically stacked in this preset to produce the final model (autostack = TRUE). The parameter 'optimize for deployment' was also enabled in order to delete the unused/unstacked models to save disk space with no effect on the final model accuracy. The training command lines can be summarized as follows:

```
presets = ['best_quality_with_high_quality_refit',  
'optimize_for_deployment']
```

```
predictor =  
task.fit(train_data=train_data,label=label_column,  
id_columns=['ID'],output_directory=output_directory,
```

```
presets=presets, problem_type='binary'
eval_metric='balanced_accuracy', verbosity=1, random_seed
= 42 )
```

AutoGluon models were evaluated on the external test sets using performance metrics implemented in the sci-kit learn library (version 0.22.2).

Evaluation of models and overcoming assessment bias

Classification models are assessed through different metrics applied to the confusion matrix of the test set, which is composed of four quadrants summarizing actual and predicted values: True positives (TP), true negatives (TN), false positives (FP) and false negatives (FN). To ensure a fair evaluation of machine learning models, the performance metrics must be carefully selected and adapted to the nature of the dataset. For example, in our study most of the targets have highly imbalanced datasets. Using the accuracy metric (calculated by Eq. 2) could be misleading, where high values could be attributed to correct prediction of the true negatives and not the true positives. Area under receiving operator curve (AUC), which measures the area under the plot of true positive rate (TPR) (calculated by Eq. 3) vs false positive rate (FPR) (calculated by Eq. 4) might also produce erroneous conclusions for the same reason.

Capturing both the true positives and true negatives is essential in off-target modeling and thus the balanced accuracy (calculated by Eq. 5), which accounts for both outcomes, was utilized as the primary metric to assess the performance of all the models and methods. Other metrics adapted to the data imbalance were also investigated through case studies, such as Mathew's correlation coefficient (MCC) (calculated by Eq. 6), which equally weighs the four quadrants of the confusion matrix, and area under the precision-recall curve (AUCPR), which excludes the true negatives from the calculation by measuring the area under the plot of the positive predicted value (ppv) or precision (calculated by Eq. 7) vs the recall (calculated by Eq. 3). An overall comparison between the balanced accuracy of all the three methods (Neural networks, H2O and AutoGluon) and is presented in Additional file (Table S1). Furthermore, a detailed analysis of all the mentioned performance metrics of the neural networks is provided in Additional file 2 (Table S2).

$$Accuracy = \frac{\sum TN + \sum TP}{\sum P + N} \quad \text{Eq. 2} \quad TPR(recall) = \frac{TP}{TP + FN} \quad \text{Eq. 3} \quad FPR = \frac{FP}{FP + TN} \quad \text{Eq. 4}$$

$$Balanced accuracy = \frac{\sum \frac{TP}{P} + \sum \frac{TN}{N}}{2} \quad \text{Eq. 5} \quad MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad \text{Eq. 6}$$

$$Precision(ppv) = \frac{TP}{TP + FP} \quad \text{Eq. 7}$$

Results

Overview of the dataset activity and population

A summary of the list of targets, their symbols, total number of compounds screened (which resembles the dataset size of each target), number of active compounds, and the hit percent for each target are given in the Table 1 As illustrated in Fig. 1, not all compounds were screened against each target and a high variability is shown within the targets in terms of dataset size and hit percent.

Targets with hit percent less than 1 (Glutamate, Glycine and Nicotinic receptor-neuronal type) were removed since constructing models for such extremely imbalanced data sets would be unfeasible. The final assembled dataset consisted of 3928 unique compounds and 47 targets.

Most of the datasets had a considerable number of compounds, for example, among the 47 targets, 43 targets had >1000 compounds and only 4 targets had <1000 compounds. Only two datasets were rather small compared to the other targets: Kappa opioid receptor (KOP) = 428 compounds and Phosphodiesterase D2 (PDE2) = 133 compounds.

Only 10 targets showed a hit percent higher than 20 (e.g. Serotonin transporter (SERT) 35.65%, Muscarinic M1 receptor 36.16%, Serotonin 5HT2B receptor 35.53%), while 37 targets showed a hit percent lower than 20%. A few amongst these 37 targets exhibited extremely low hit percent (e.g. HIV protease 2.18%, Adrenergic beta 1 receptor (ADRB1) 2.66%, Angiotensin I receptor II (AT1) 2.53%).

The impact of both factors (dataset size and hit percent) on the model performance is discussed in the “Technical and biological factors affecting model performance” section.

Comparison of models' performance between Keras, H2O and AutoGluon Tabular

Overall performance

An overall view of the highest scoring machine learning method in terms of balanced accuracy for each target model is represented in Fig. 2a. Neural network models scored a higher balanced accuracy than H2O and Autogluon for 28 targets, equal balanced accuracy for 3 targets, while both Autogluon and H2O scored a higher balanced accuracy for 8 targets each. Neural networks seem to have a better overall performance than the two autoML methods. Both Autogluon and H2O seem to have a similar overall performance.

Figure 2b illustrates the balanced accuracy range of the targets for the highest scoring method, neural networks. The majority of the neural network models were successful, whereby the balanced accuracy was between 0.7 and 0.8 for 22 of the targets, between 0.8 and 0.9 for 4 targets and exceptionally high (>0.9) for 2 targets (in total 28 targets with balanced accuracy more than 0.7). Few targets had a moderate balanced accuracy between 0.6 and 0.7 (12 targets) and 7 targets were not well predicted and exhibited a low balanced accuracy (less than 0.6).

The variability within the models' performance is investigated through a detailed insight into the balanced accuracy values for each target (via the three methods) along with the factors that might affect the performance, such as the hit percent. The values of the balanced accuracy for each target model and the

respective hit percent values are displayed in Fig. 3. All of the high hit percent targets were predicted with a balanced accuracy higher than 0.7 for all methods except for the Kappa opioid receptor (AutoGluon balanced accuracy = 0.69) and GABA A CL⁻ channel (H2O balanced accuracy = 0.69). AutoGluon models for GABA A CL⁻ gave a 0.86% balanced accuracy.

Despite the similar hit percent values between the three targets: Phosphodiesterase D2 (PDE4D2), Cyclooxygenase 2 (COX2) and Histamine (H3) receptors (11.2, 10.6 and 11.5 respectively), the three methods showed a poor performance for the Phosphodiesterase D2 (PDE4D2) target (Neural networks balanced accuracy = 0.5) while succeeded in Cyclooxygenase 2 (COX2) (H2O balanced accuracy = 0.74) and Histamine (H3) receptors (Neural networks balanced accuracy = 0.8). This could be attributed to the small sample size of the PDE4D2 used to construct the model (n = 133) versus those for COX (n = 1799) and H3 (n = 1864). Also, other targets like the Kappa opioid receptor (KOP) exhibited a small dataset size (n = 423). However, in this case the relatively high hit percent of KOP (20.79%) provided a model with at least moderate balanced accuracy (an average of 0.7 across the three methods).

18 targets with low hit percent (<20%) were predicted with a relatively good balanced accuracy (>0.7) for neural networks. Two among these targets had a significantly lower hit percent than the others: Angiotensin receptor 1 (AT1) and beta-1 adreno receptor (ADRB1) (hit percent = 2.5 and 2.6 respectively). Both targets showed a surprisingly good balanced accuracy (mean balanced accuracy of three methods ~ 0.74 for both targets). Both targets are class A GPCR receptors and have a relatively large dataset size (ADRB1, n = 2252 and AT1, n = 1762). On the other hand, the Prostaglandin receptor (FP), (which is also a class A GPCR receptor) had a low hit percent but in contrast to AT1 and ADRB1, no high balanced accuracy was observed (mean balanced accuracy = 0.53). The FP receptor is involved in allosteric binding and only a few FP inhibitors are well characterized, which in addition to the low hit percent, might be another reason for the low predictivity of the model.

Implementation time and resources

The neural networks were trained using 3 GPUs and 18 CPUs on an Intel(R) Xeon(R) Gold 6148 CPU @ 2.40GHz machine with 40 cores and 768 GB memory, AutoGluon was trained on an Intel(R) Xeon(R) CPU E5-2680 v3 @ 2.50GHz machine with 24 cores and 256 GB and finally the H2O models were trained using an amazon web services (AWS) virtual machine (m5.2xlarge instance) with 8 CPUs and 32 GB memory.

For neural networks the calculation took between 3 and 4 hours per target, resulting in a total of 188 hours for the 47 targets. H2O training ranged from 1.35 hours to 9.50 hours depending on the target, resulting in a total training time of 173.81 hours. By far the most rapid method was AutoGluon, which consumed only 20 minutes in total for all the models. An overall comparison between the three methods with respect to availability, time consumption, required skills and supported languages is shown in Table 4. The time consumption described in Table 4 corresponds to the previously described training settings of each tool and is subject to change depending on these settings.

Table 4

An overall comparison between the three machine learning tools used in the predictions.

	Neural networks (via tensorflow)	H2O driverless AI	AutoGluon
Availability	Open source	License required	Open source
Time consumption	High	High	Low
Required skills	- High data science skills required in R or python	-No data science skills required for the user- interface. - Minimum Python or R experience required to use the backend framework. (R/Python API client)	- Basic experience required in python
Supported language	R or Python	R or Python	Python only

Summary on the technical and biological factors impacting prediction performance

Technical factors (hit percent and data set size)

Figure 4a shows how the hit percent affects the model performance and the overall difference in the three methods used with respect to the hit percent. A statistically significant difference in the balanced accuracy is shown between the high and low hit percent groups (Wilcoxon rank test p-value = 0.001, 0.004 and 0.00 for AutoGluon, H2O and neural networks respectively). The three methods performed similarly in predicting high hit percent targets. Neural networks showed again the best performance (mean balanced accuracy = 0.79) followed by an equal performance of H2O and AutoGluon (mean balanced accuracy = 0.77 for both methods). Regarding the low hit percent category, the mean balanced accuracy dropped by more than 10% for each method. The same trend is observed where neural networks again showed a slightly better performance than the two other methods, followed by H2O and finally AutoGluon (mean balanced accuracy of 0.68, 0.66 and 0.64) respectively.

Biological factors

The impact of the target protein class is displayed in Fig. 4b, where little or no difference is seen between the high hit percent and the low hit percent group in the GPCR class. On the other hand, a large difference is seen between the high hit percent and low hit percent group for the Ion-channel class. The high hit percent Ion-channel Ca²⁺ channel (Diltiazem site) and GABA-A receptors (CL – channel and Benzodiazepene sites) had an average balanced accuracy of 0.808, while the low hit percent Ion-channels: Nicotinic receptors (CHRNA1, CHRNA4) and Serotonin receptor (5HT3) had a much lower balanced accuracy average = 0.57. With regards to the ‘Kinases’ class, despite that all the targets had a

low hit percent, the average balanced accuracy was equal to 0.72. On the other hand, all the targets in the Nuclear receptors class also belonged to the low hit percent group but still had a low average balanced accuracy (0.63). Targets in the enzymes class also belong to the low hit percent group and showed a wide balanced accuracy distribution (from 0.50 till 0.81). The transporter class comprised two targets, with a high balanced accuracy for the high hit percent target (Serotonin transporter, hit percent = 35.65%, BA = 0.88) and the low hit percent target (Norepinephrine transporter, hit percent = 16.65%, BA = 0.76). Both, the correlation between the hit percent and the balanced accuracy of the neural network models as well as the impact of the target type on the model performance is seen in Fig. 5. We can deduce that some target classes such as GPCRs, kinases, and transporter, might be easier to predict than e.g. the ion-channels and nuclear receptors (Fig. 5).

There was no systematic impact of the dataset size seen on the prediction accuracy of the models, as most of the datasets comprised >1000 compounds. Only target models with small dataset size showed either poor (PDE4D2) or mediocre performance (e.g. KOP).

Discussion And Next Steps

In this work we have discussed several approaches to tackle challenges arising in the attempt of predicting off-target activities directly from a chemical structure. One of the key learnings from this work is the importance of analyzing and understanding the nature of the dataset and the impact of applying certain techniques at different stages of modelling: (1) Implementation of stratified activity splitting of the datasets prior to model construction (2) Implementation of early stopping, model checkpoints and training on an unbiased performance metric (e.g. balanced accuracy), grid search on hyper parameters during model construction and finally (3) Choosing the best model according to again an unbiased metric After model construction and while assessing the model performance. Deploying such techniques showed an improvement in the models and provided realistic evaluation of the model performance.

As previously mentioned, setting up a neural network framework requires several tactics and expertise to manually implement hyper parameter optimization in order to create robust models. On the other hand, neural networks were found to be the best performing method for the majority of the targets. One drawback is that once the grid search parameters are chosen, the network has to exhaust the possible combinations, which is extremely time consuming. On the contrary, AutoML did not require hyper parameters tuning and was less time consuming, since time allocation was one of the configuration settings that a user can choose.

A difference was seen in the configuration parameters of the autoML methods, for example, in H2O one has to manually set the accuracy, time and interpretability required for the models (values range from 1 to 10). This gives quite a large flexibility in the model construction but it might be more difficult to tweak and assess the best needed for the datasets, unless through trial and error. In some cases, where the target has a large data set and high hit percent, setting time to 1 for example can achieve good predictions while for other targets H2O models require longer prediction times. The configuration setting

of AutoGluon was categorical, which makes it simpler to choose, either high quality models with limited time allocation or medium quality models with less accuracy but very rapid implementation. The option “best quality with high quality refit” we have selected achieved high prediction accuracies (close to those of H2O and the neural network models) and was very efficient in terms of time consumption, disk usage and implementation.

In general, autoML can also be used to guide other tailored machine learning approaches, for example 5HT3 was more accurately predicted with H2O autoML than neural networks, indicating that this is not a dataset issue and that the neural networks were simply not able to converge for this specific target and could be further improved.

In some cases, regardless how simple or exhaustive the machine learning method is, the prediction failed which was due to the nature of the dataset. If there is extreme imbalance in the dataset, hyperparameters optimization will likely not improve the prediction. In this case, other techniques are worth trying such as transfer learning or anomaly detection. These techniques have been widely and successfully used in e.g. detecting credit card frauds, which resembles a minority when compared to the millions of normal transactions done every day. In our case, active molecules would be considered as a minority and the model will train to predict these minorities. Another option is oversampling, through either generating multiple conformations of the scarce active molecules (COVER)[27] or creation of artificial samples (SMOTE)[28] and thus balancing out the dataset. Overcoming the challenge of data scarcity and imbalance through attempting all these mentioned approaches summarizes the next steps in our off-target prediction framework.

It was also noticed that poor model performance was not due to one single factor, but rather is an interplay of several factors, such as the data balance, the sample size, and also the target type. For example, most of the target models in the GPCR family performed well regardless of the hit percent, while target models in the ion-channel family were highly dependent on the hit percent. Some biological factors could be involved such as allosteric binding of targets and the presence of co-enzymes, which requires further investigation.

It could also be of value to exploit target models with high prediction accuracies and use it for building models for low performing targets with similar inhibition mechanisms or similar pocket sequence. This approach could be perceived as transfer learning, where we train a model on a rich target dataset (where we have already reached a good balanced accuracy) and then apply it to an imbalanced/small target dataset. It could also be applied in a wider context, where we could train our models on a large public dataset and then apply it systematically to all the low performing targets.

Conclusion

In this work we constructed in silico models that can predict the off-target activity of in-house compounds and can be deployed in the drug discovery pipeline and guide chemists throughout compounds’ design prior to synthesis. A relatively good prediction accuracy was achieved for the majority of our in-house

panel (>70% BA for 35 out of 50 targets). Reasons behind the poor performance of the rest of the target models were identified and techniques to overcome these issues were proposed, as well as our next steps for this project, which is attempting innovative techniques such as anomaly detection, transfer learning and multi task learning. Although neural networks were the top performing method, our work shows that both autoML methods, H2O and AutoGluon, achieved similar results for many of the targets and might provide a quick and user-friendly alternative to manual model building in the future.

Declarations

Acknowledgements

The authors would like to thank Torsten Schindler for his valuable input and guidance on the autoML implementation. We thank Thierry Lavé and Sonia Roberts for their valuable guidance and fruitful discussions throughout the project. We thank Carlos Fenoy and Ioannis Liabotis for their technical support and assistance.

Author's contribution

GFE and DN conceived and designed the study. DN performed the data collection, analysis and developed the models. DN wrote the manuscript under the supervision of GFE. GFE, WM and EM provided supervision and critical review of the manuscript. All authors read and approved the final manuscript.

Corresponding author

Correspondence to Gerhard F.Ecker

Competing interests

The authors declare that they have no competing interests.

Funding

This project has received funding from the Innovative Medicines 2 Joint Undertaking under grant agreement No 777365 ("eTRANSafe"). This Joint Undertaking receives support from the European Union's Horizon 2020 research and innovation programme and EFPIA companies in-kind contribution.

Disclaimer

This paper reflects only the authors' view. Innovative Medicines Initiative 2 Joint Undertaking is not responsible for any use that may be made of the information it contains.

Availability of Data and Materials

The data used in constructing the models could not be shared due to confidentiality issues.

References

1. Cook D, Brown D, Alexander R, March R, Morgan P, Satterthwaite G, Pangalos MN (2014) Lessons learned from the fate of AstraZeneca's drug pipeline: a five-dimensional framework. *Nat Rev Drug Discovery* 13(6):419–431
2. Waring MJ, Arrowsmith J, Leach AR, Leeson PD, Mandrell S, Owen RM, Pairaudeau G, Pennie WD, Pickett SD, Wang J et al (2015) An analysis of the attrition of drug candidates from four major pharmaceutical companies. *Nat Rev Drug Discov* 14(7):475–486
3. Hutchinson L, Kirk R (2011) High drug attrition rates—where are we going wrong? *Nature Reviews Clinical Oncology* 8(4):189–190
4. Bendels S, Bissantz C, Fasching B, Gerebtzoff G, Guba W, Kansy M, Migeon J, Mohr S, Peters JU, Tillier F et al (2019) Safety screening in early drug discovery: An optimized assay panel. *J Pharmacol Toxicol Methods* 99:106609
5. Waring MJ, Arrowsmith J, Leach AR, Leeson PD, Mandrell S, Owen RM, Pairaudeau G, Pennie WD, Pickett SD, Wang J et al (2015) An analysis of the attrition of drug candidates from four major pharmaceutical companies. *Nat Rev Drug Discovery* 14(7):475–486
6. Kola I, Landis J (2004) Can the pharmaceutical industry reduce attrition rates? *Nature reviews Drug discovery* 3:711–715
7. Ralston, Sea: **Pre-development attrition of pharmaceuticals: how to identify the bad actors early.** *Toxicol Sci* 2017, 150(2323)
8. Anighoro A, Bajorath J, Rastelli G (2014) Polypharmacology: challenges and opportunities in drug discovery. *J Med Chem* 57(19):7874–7887
9. Bender A, Scheiber J, Glick M, Davies JW, Azzaoui K, Hamon J, Urban L, Whitebread S, Jenkins JL (2007) Analysis of pharmacology data and the prediction of adverse drug reactions and off-target effects from chemical structure. *ChemMedChem* 2(6):861–873
10. Whitebread S, Hamon J, Bojanic D, Urban L (2005) Keynote review: in vitro safety pharmacology profiling: an essential tool for successful drug development. *Drug Discov Today* 10(21):1421–1433
11. Roy M, Dumaine R, Brown AM (1996) HERG, a primary human ventricular target of the nonsedating antihistamine terfenadine. *Circulation* 94(4):817–823
12. Bowes J, Brown AJ, Hamon J, Jarolimek W, Sridhar A, Waldron G, Whitebread S (2012) Reducing safety-related drug attrition: the use of in vitro pharmacological profiling. *Nat Rev Drug Discov* 11(12):909–922
13. Koutsoukas A, Monaghan KJ, Li X, Huan J (2017) Deep-learning: investigating deep neural networks hyper-parameters and comparison of performance to shallow methods for modeling bioactivity data. *J Cheminform* 9(1):42
14. Ma J, Sheridan RP, Liaw A, Dahl GE, Svetnik V (2015) Deep neural nets as a method for quantitative structure-activity relationships. *J Chem Inf Model* 55(2):263–274

15. Mayr A, Klambauer G, Unterthiner T, Hochreiter S: **DeepTox: Toxicity Prediction using Deep Learning.** *Frontiers in Environmental Science* 2016, 3(80)
16. Waring J, Lindvall C, Umeton R (2020) Automated Machine Learning: Review of the State-of-the-Art and Opportunities for Healthcare. *Artif Intell Med* 104:101822
17. Rogers D, Hahn M (2010) Extended-Connectivity Fingerprints. *J Chem Inf Model* 50:742–754
18. Bender A, Jenkins J, Scheiber J, Sukuru S, Glick M, Davies J (2009) How Similar Are Similarity Searching Methods? A Principal Component Analysis of Molecular Descriptor Space. *J Chem Inf Model* 49:108–119
19. O’Boyle NM, Sayle RA (2016) Comparing structural fingerprints using a literature-based similarity benchmark. *J Cheminform* 8(1):36
20. Steffen A, Kogej T, Tyrchan C, Engkvist O (2009) Comparison of molecular fingerprint methods on the basis of biological profile data. *J Chem Inf Model* 49(2):338–347
21. **BIOVIA, Dassault Systèmes, Pipeline Pilot, Version 9.1.0, San Diego: Dassault Systèmes, 2019**
22. R Core Team. R Foundation for Statistical Computing V, Austria.: **R: A language and environment for statistical computing** (2020)
23. Hall P, Kurka M, Bartz A: **Using H2O Driverless AI.** 2018
24. Erickson N, Mueller J, Shirkov A, Zhang H, Larroy P, Li M, Smola A: **AutoGluon-Tabular: Robust and Accurate AutoML for Structured Data;** 2020
25. Abadi M, Barham P, Chen J, Chen Z, Davis A, Dean J, Devin M, Ghemawat S, Irving G, Isard M et al: **TensorFlow: A system for large-scale machine learning.** 2016
26. Chollet F: **keras,Github.** 2015
27. Hemmerich J, Asilar E, Ecker GF (2020) COVER: conformational oversampling as data augmentation for molecules. *J Cheminform* 12(1):18
28. Chawla N, Bowyer K, Hall L, Kegelmeyer W (2002) SMOTE: Synthetic Minority Over-sampling Technique. *J Artif Intell Res (JAIR)* 16:321–357

Figures

Figure 1

Overview of the total number of compounds screened per target and the number of active compounds within the total screened. The number of compounds is represented on the x axis and the fifty off targets are represented on the y-axis. The color code indicates the activity of compounds: pink for active compounds and blue for the total number of compounds screened.

Figure 2

An overall comparison between the performance of the three machine learning methods used for the construction of the off-target models and an overview of the balanced accuracy range for the best performing method (neural networks). (a) A bar plot comparing neural networks, H2O and AutoGluon (x-axis) with respect to the number of targets each method scored the highest balanced accuracy (y-axis) (b) A bar plot displaying the number of off-target models (y-axis) falling under each balanced accuracy range (x-axis) for the neural networks method.

Figure 3

Lollipop chart representing the balanced accuracy of each target model for each method. Each subplot represents one of the methods used to build the models (Neural networks, H2O and AutoGluon). The abbreviated names of each target model are represented on the x-axis and the corresponding balanced accuracy values on the y axis. Lines are color coded according to hit percent (Blue = High and Turquoise = Low). Numerical values of the hit percent of targets are indicated inside the circles.

Figure 4

Box plots comparing the overall balanced accuracy for high and low hit percent target groups, within the (a) Machine learning tools used for prediction and (b) Target classes. (a) The three methods are represented on the x axis and the balanced accuracy values on the y axis. The red box plots represent the high hit percent target groups and the blue box plots represent the low hit percent target groups. The High hit percent target groups achieve a higher balanced accuracy irrespective of the method used. (b) The target classes are represented on the x axis and the balanced accuracy values on the y axis. For some target classes, no significant difference is seen in the overall balanced accuracy between the high hit percent and low hit percent groups (e.g. GPCR class) while for other classes (e.g. Ion-channels), a significant difference is seen in the overall balanced accuracy between the high and low hit percent groups.

Figure 5

Scatterplot displaying the impact of the hit percent, data set size and target class on the balanced accuracy of the individual target models. The hit percent is represented on the x axis and the balanced accuracy on the y axis. Each circle represents a target. The circles are color coded according to the target classes. The size of the circles varies according to the dataset size.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [S1.xlsx](#)
- [S2.xlsx](#)