Computational Biology and Chemistry 93 (2021) 107529

Contents lists available at ScienceDirect

Computational Biology and Chemistry

journal homepage: www.elsevier.com/locate/cbac

Ensembling machine learning models to boost molecular affinity prediction

Maksym Druchok [a,b,*], Dzvenymyra Yarish [a], Sofiya Garkot [a,c], Tymofii Nikolaienko [a,d], Oleksandr Gurbych [a,e]

[a] SoftServe, Inc., 2d Sadova Str., 79021 Lviv, Ukraine
[b] Institute for Condensed Matter Physics, NAS of Ukraine, 1 Svientsitskii Str., 79011 Lviv, Ukraine
[c] Ukrainian Catholic University, 17 Svientsitskii Str., 79011 Lviv, Ukraine
[d] Taras Shevchenko National University of Kyiv, 64/13, Volodymyrska Str., 01601 Kyiv, Ukraine
[e] Lviv Polytechnic National University, 5 Kniazia Romana Str., 79005 Lviv, Ukraine

ARTICLE INFO

Keywords:
Binding affinity
Human thrombin
Ensembled prediction
Machine learning
Deep neural networks

ABSTRACT

This study unites six popular machine learning approaches to enhance the prediction of a molecular binding affinity between receptors (large protein molecules) and ligands (small organic molecules). Here we examine a scheme where affinity of ligands is predicted against a single receptor – human thrombin, thus, the models consider ligand features only. However, the suggested approach can be repurposed for other receptors. The methods include Support Vector Machine, Random Forest, CatBoost, feed-forward neural network, graph neural network, and Bidirectional Encoder Representations from Transformers. The first five methods use input features based on physico-chemical properties of molecules, while the last one is based on textual molecular representations. All approaches do not rely on atomic spatial coordinates, avoiding a potential bias from known structures, and are capable of generalizing for compounds with unknown conformations. Within each of the methods, we have trained two models that solve classification and regression tasks. Then, all models are grouped into a pipeline of two subsequent ensembles. The first ensemble aggregates six classification models which vote whether a ligand binds to a receptor or not. If a ligand is classified as active (i.e., binds), the second ensemble predicts its binding affinity in terms of the inhibition constant K_i.

1. Introduction

Initial stages of drug discovery require localization of what causes a disease, understanding of the molecular mechanism, then suggesting and testing drug leads. After a disease target has been identified, a list of drug candidates is drafted and screened for the target-candidate affinities, also known as a drug-target binding affinity (DTBA). DTBA reflects the strength of an interaction between targets and ligands. In particular, it can be quantified by the inhibition constant K_i. The smaller K_i value is, the stronger the ligand obstructs the target active site, and the greater therapeutic effect can be achieved with the lower dose of the drug. There is a large variety of experimental methods for measuring DTBA but they are costly in terms of human efforts, time, and resources. Thus, computational approaches are needed to help shrink the pool of *in vitro* tests by eliminating weakly scored candidates. One of such techniques is molecular docking (Shoichet et al., 2002; Pagadala et al., 2017; de Azevedo, 2019) that explores a binding conformational space between different molecules and solves the task of optimal docking conformations. The estimated binding energies during docking scenarios might not be predictive because docking modes with low estimated binding energies not always correspond to the experimentally observed binding modes (Frimurer et al., 2003). It is worthy of note that molecular docking might suffer from imprecise detection of ligand spots/poses or even dock completely inactive compounds (Chen, 2015). Along with classical force field simulations and molecular docking, machine learning (ML) techniques became a powerful tool in the field of virtual screening. One of the pioneering studies in machine learning methods for binding affinity prediction is reported by King et al. (1995) where the performances of the feed-forward neural networks, k-Nearest Neighbors, and Decision Trees models are compared on the set of about 200 ligands and two target receptors. The list of ligand descriptors there includes molecular size, flexibility, polarity, polarizability, numbers of donors/acceptors, etc. Jorissen and Gilson (2005) rated compounds by a DTBA using Support Vector Machine approach. Their dataset consisted of several hundreds of ligand-receptor pairs. More than 500 molecular descriptors were generated but only ≈50 of them were used – the ones

* Corresponding author at: SoftServe, Inc., 2d Sadova Str., 79021 Lviv, Ukraine.
E-mail address: maksym@icmp.lviv.ua (M. Druchok).

https://doi.org/10.1016/j.compbiolchem.2021.107529
Received 20 April 2021; Received in revised form 1 June 2021; Accepted 8 June 2021
Available online 12 June 2021
1476-9271/© 2021 Elsevier Ltd. All rights reserved.

with the highest discrimination scores. The importance of the feature selection is also highlighted in a study by Yugandhar and Gromiha (2014) that focuses on the protein-protein binding affinity.

A concept of the interaction-averaged space is presented in a study by Li et al. (2019a). In this study a set of 439 features is suggested for the multi-receptor active/inactive ligand binary classification: 107 features describe receptors, 166 features are MACCS fingerprints (Durant et al., 2002) that describe ligands, the rest 166 features are fingerprints averaged over ligands within the same receptor. The method of choice there is Bayesian Additive Regression Trees, while the following approaches are used as a baseline: Support Vector Machine, Random Forest, Decision Trees, and Logistic Regression. The authors report the accuracy of ≈95% on the binary classification to actives or inactives. A similar concept of the interaction-averaged space is also used in a study by Heck et al. (2017) who built a number of regression models for DTBA over an aggregated scoring space. This space is based on ligand scores that are averaged in a per-receptor manner.

It is worth to mention a number of papers (Pahikkala et al., 2014; He et al., 2017; Öztürk et al., 2018, 2019; Nguyen et al., 2020; Shim et al., 2021) by different research groups, but benchmarking on the same datasets (Davis et al., 2011; Tang et al., 2014) that cover a niche of kinases. In particular, the prediction in KronRLS (Pahikkala et al., 2014) is based on a similarity score for each drug-target pair where the similarity of drug-target pairs is defined through the Kronecker product of drug-drug and target-target similarity matrices. Given a set of the among-drugs and among-targets similarities, SimBoost (He et al., 2017) uses gradient boosting to predict DTBAs and yields MSE of 0.28 on the Davis dataset (Davis et al., 2011). A deep neural network for the DTBA prediction DeepDTA (Öztürk et al., 2018) takes the receptor's FASTA (Lipman and Pearson, 1985; Pearson and Lipman, 1988) and ligand's SMILES (Weininger, 1988; Weininger et al., 1989) strings as inputs, encodes, and zero-pads them. The encodings are passed through two separate convolutional networks, concatenated, and sent to fully connected layers, yielding binding affinity. DeepDTA achieved MSE in the range of 0.26–0.66 (depends on the encoding setup) on the Davis dataset. WideDTA (Öztürk et al., 2019) uses four textual inputs: the protein sequence, ligand SMILES, protein domains and motifs, and maximum common substructure words to predict DTBA. These inputs are fed into four separate sleeves with convolutional layers, concatenated, and passed through a set of fully connected layers to predict a DTBA value. The model scored MSE of 0.26 on the Davis dataset. The GraphDTA approach (Nguyen et al., 2020) also relies on separate input sleeves – for ligand and receptor. The outputs of the sleeves are concatenated and regressed towards DTBA. The ligands are represented as graphs with atoms as vertices and bonds as edges. Four implementations of graph neural networks are tested for the ligand part, whereas the receptor is encoded from FASTA notation and processed by a set of convolution networks. The reported MSE values range from 0.23 to 0.25 in pK_d units from the Davis dataset. Similar to the KronRLS approach, SimCNN-DTA by Shim et al. (2021) is based on chemical similarities. In particular, for a given ligand-receptor pair two vectors are calculated: first one consists of Tanimoto similarities between molecular fingerprints of ligands, the second one – of Smith-Waterman similarities between FASTA sequences of receptors. The outer product of these two vectors constitutes a 2D matrix, which serves as an input for a 2D convolutional network to predict binding affinities. SimCNN-DTA is benchmarked on both Davis and KIBA datasets, showing equal or better performance, than the studies from this paragraph.

A study by Kundu et al. (2018) compared a performance of a number of ML methods (Random Forest, Support Vector Machine, Gaussian Process, feed-forward neural network) on DTBA prediction on the PDBbind (v.2015) dataset. Within all the considered approaches, the receptors and ligands were featurized with a set of structural and physico-chemical properties, constituting a single input vector per receptor-ligand pair. The predicted outputs were K_i and K_d values. A feature engineering and fine-tuning of the models revealed the best prediction results were achieved with the Random Forest approach.

DGraphDTA approach (Jiang et al., 2020) uses a structural information of molecules and proteins. Two graphs for drug molecules and proteins are built up respectively, regressing to predict DTBAs. Notably, protein graphs are constructed out of the protein contact maps which are predicted from FASTA sequences by a contact predictor PconsC4 (Michel et al., 2018).

Jiménez and co-authors (Jiménez, Škalič et al., 2018) introduced K_{deep} – a three-dimensional convolutional neural network (3D-CNN) for DTBA prediction. Each protein and ligand pair is featurized via a voxelized 24 Å representation of the binding site, characterizing each voxel by eight pharmacophoric-like properties. The authors achieved root mean square error (RMSE) of 1.27 in pK units between experimental and predicted DTBA on the standard PDBbind (v.2016) core test-set. DeepAtom (Li et al., 2019b) is another 3D-CNN framework that extracts binding-related atomic interaction patterns from the voxelized complex structure. With the Astex Diverse dataset (Hartshorn et al., 2007) as training set, DeepAtom achieves RMSE of 1.23 in pK units on the PDBbind (v.2016) as a test set.

A study by Kwon et al. (2020) discusses an approach for DTBA prediction based on the ensemble of 3D-CNNs. The ensemble has the lowest accuracy metric values with mean absolute error (MAE) of 1.01 kcal/mol and an RMSE of 1.29 kcal/mol. Using the ensemble of the networks improved a prediction quality by 0.1 kcal/mol as compared to a single network performance. Ensemble of Random Forest, AdaBoostRegressor, Gradient Boosting Regressor, and feed-forward neural network models by Chen et al. (2019) predicts activities of peptide ligands to several tumor-related proteins. Chen et al. (2019) reported quite high coefficients of determination R^2 of 0.81/0.9 on the training/test set and, as an additional validation step, performed a set of computer simulations for selected protein-ligand pairs. However, the selected ligands did not reveal an expected strong attachment. An integrated approach that uses ligand docking on multiple structural ensembles to reflect receptor flexibility is proposed by Schneider et al. (2019). The approach unites the ligand docking stage with the stage of DTBA prediction for a docked complex by engaging a Random Forest model. To enrich the regression solution, various sets of descriptors are examined for effectiveness.

Since their introduction Transformer-like neural networks have been successfully adopted for mapping the raw chemical sequences to representations of biological functions and properties. In a study by Schwaller et al. (2019), self-attention models are used for the task of predicting the products of chemical reactions formulated as a machine translation problem between SMILES strings of reactants, reagents, and the products. Payne et al. (2020) analyze applications of BERT model and its attention to learn useful contextualized representations of chemical compounds that are used for problems of toxicity, solubility, drug-likeness, and synthesis accessibility prediction. In study by Rives et al. (2019), encodings of protein sequences are learned by BERT model from a large-scale unlabeled dataset and thoroughly tested on the subject of encoding diverse protein aspects. MT-DTI model (Shin et al., 2019) combines the sequence modelling capacities of two representation learners – CNNs for FASTA and self-attention mechanisms for SMILES for the task of DTBA prediction. Such coupling of neural architectures allowed it to achieve state-of-the-art results on the above-mentioned KIBA and Davis datasets.

In the last two years, a special attention is also paid to the search of effective inhibitors against SARS-CoV-2-related receptors. The epidemic pace of the infection spread urges the use of ML methods as fast and lightweight approaches to screen millions of compounds (see, for example, a recent review by Mottaqi et al. (2021)). A gradient boosting regression approach is discussed by Gao et al. (2020) where active inhibitors against the SARS-CoV-2 3CL protease (Jin et al., 2020) are sought from a list of FDA approved drugs. As the protease is the single receptor in this task, there is no need to parameterize it. With the training set of 314 inhibitors, the ligands were parameterized with a

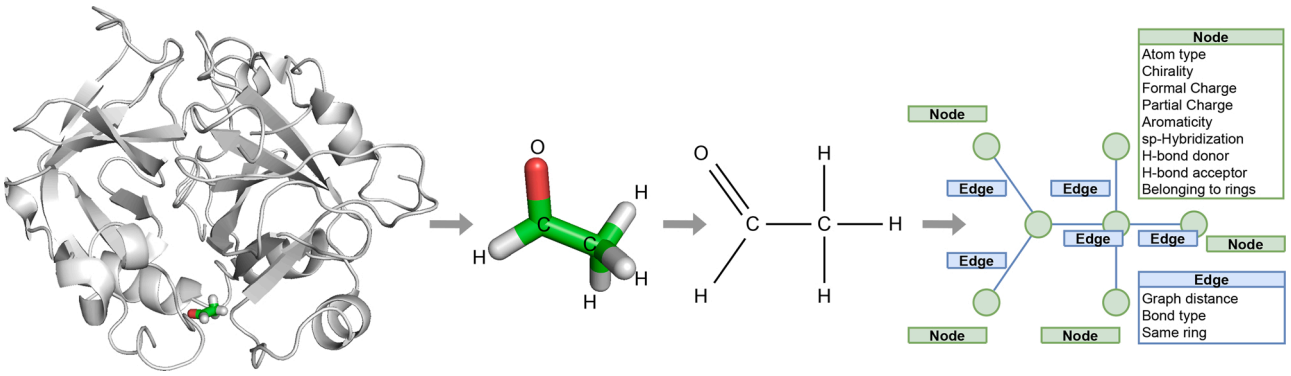

Fig. 1. An example ligand, acetaldehyde, residing in the human thrombin's binding site. All ligands are represented as molecular graphs with nodes (atoms) and edges (bonds). Each node and edge is assigned with a set of features.

consensus of three kinds of fingerprints. A study by Nand et al. (2020) adopted a multistage pipeline uniting activity classification, drug-likeness filtering, docking, and binding affinity predictions to search inhibitors against the protease. Since a single-receptor approach was considered, only ligands were featurized with a set of descriptors and served as inputs. Finally, selected inhibitors were simulated by molecular dynamics within the protease active site. Santana and Silva-Jr (2020) screened a list compounds from the ChEMBL database (Mendez et al., 2019) by a recurrent neural network to classify them for inhibitory action against the protease. The compounds predicted as active were further analyzed using molecular docking. Kowalewski and Ray (2020) suggested a pipeline to identify drug candidates against multiple SARS-CoV-2-related receptors with a special focus on a feature selection. The trained pipeline was further used to screen a list of thousands of known drugs and millions of purchasable chemicals for binding affinities, toxicity, and volatility. The MT-DTI model by Shin et al. (2019) was applied to predict binding affinities of known antiviral drugs to six SARS-CoV-2-related receptors (Beck et al., 2020). In a study by Kadioglu et al. (2021), known drugs and purchasable chemicals are examined for interaction ability with three receptors (spike, capsid, and transferase proteins) by means of an AutoDock Vina and ML combined approach.

Concluding the introductory part, we would like to mention a recent review by Ellingson and co-authors (Ellingson et al., 2020) that discusses most popular machine learning approaches and emerging obstacles in studies of DTBA. Besides the typical problems with representations mentioned above, difficulties with datasets and data consistency are discussed. The quality of the binding affinity prediction might suffer from inconsistent data, affected by experimental noise. Notably, some of the discussed studies are focused on one certain receptor, thus, no receptor representation is needed in such approaches. Also, the use of multistage pipelines helps enrich a scope of predictions through a combination of task-specific solutions.

2. Methods

In this study, we evaluate six ML approaches with regard to their capability to predict molecular binding affinity. The approaches are wrapped into a pipeline with two subsequent ensembles, as outlined in the next subsection. We used the following methods: Support Vector Machine (SVM) (Cortes and Vapnik, 1995), Random Forest (RF) (Breiman, 2001), Catboost (Dorogush et al., 2018), feed-forward neural network (FFNN), graph neural network (GNN) (Scarselli et al., 2009), and Bidirectional Encoder Representations from Transformers (BERT) (Devlin et al., 2018). The first four approaches – SVM, RF, CatBoost, FFNN – use extended-connectivity fingerprints (Rogers and Hahn, 2010) in the ECFP4 formulation as inputs. The fifth approach – GNN – uses graph representations of ligands as inputs by treating atoms as graph nodes, whereas graph edges are bonds that connect atoms (nodes). Graphs with edge information (Beck et al., 2018) were proposed to

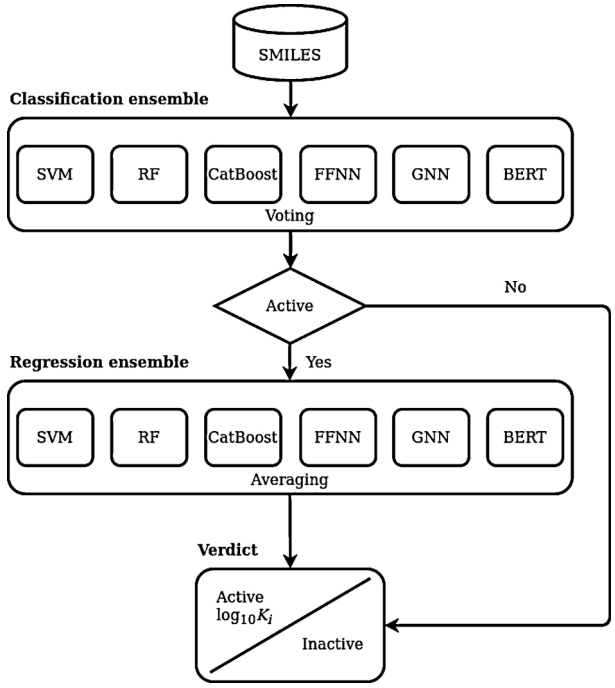

Fig. 2. A scheme of the prediction pipeline with classification and regression ensembles.

extend the representation capability of the original GNN formalism (Scarselli et al., 2009). In our study, both the nodes and edges are ascribed with physico-chemical properties which distinguish different atoms and bonds (see Section 2.6 for more details) – a schematic graph representation for acetaldehyde molecule as an example is shown in Fig. 1. In contrast to the above approaches dealing with purely physico-chemical properties, BERT works directly with string representations of ligands and thus discards the need for feature engineering. The mentioned methods are examined for the capability to predict binding affinities of various ligands to human thrombin, however, all the models can be retrained for a case of other receptors. Such a per-target paradigm is a common approach in ML applications for cheminformatics (see, for example, Gao et al. (2020), Nand et al. (2020), Santana and Silva-Jr (2020), Chupakhin et al. (2013)).

2.1. Pipeline

A principal scheme of the pipeline is shown in Fig. 2. It is comprised of two stages: the first stage decides whether a ligand is active or inactive, then the second stage predicts the inhibition constants for active

ligands. The predictions within these two stages are interpreted in an ensemble-like manner by voting (for classification) and averaging (for regression) strategies. The modular structure within the ensembles allows one to add or remove models and implement various ensembling scenarios.

2.2. Support Vector Machine

Support Vector Machine in a transformed feature space, viz. with Gaussian radial basis function (RBF) used as a dot-product kernel. SVM models were engaged in both classification and regression ensembles. In both cases, they were used as implemented in Scikit-learn library (Pedregosa et al., 2011). Due to the use of RBF kernel$k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \| \mathbf{x}_i - \mathbf{x}_j \|^2)$ on feature vectors $\mathbf{x}_i$ and $\mathbf{x}_j$ with binary-valued components, the obtained model basically compared the input molecules by similarity of their ECFP4 fingerprints at a bit-wise level. A regularization parameter C of the model was selected basing on the results of a grid search. The search revealed that the default value $C = 1$ is the optimal choice among the values 0.01, 0.1, 0.2, 0.5, 1.0, 2.0, 5.0, 10.0, and 100.0 in terms of the highest accuracy metric. When training the model, class_weight = "balanced" option was set to account for the prevailing number of molecules with inactive class in the input dataset. For the regression task, the Epsilon-Support Vector Regression model was used, as implemented in the Scikit-learn library.

2.3. Random Forest

We use the Random Forest implementation from the Scikit-learn, as well. RF maintains internally the balance between active and inactive samples on the classification task by switching the class_weight = "balanced" option. At the initial stage, we applied a grid-search procedure to tune model hyper-parameters. During this optimization all combinations of the following parameter choices were tested: the number of trees in ensemble (n_estimators) – one of 200, 500, 1000, 2000, the maximum depth of the tree (max_depth) – one of 2, 5, 7, 8, 10, the minimum number of samples required to split an internal node (min_samples_split) – one of 1, 2, 4, 8, 10, 20. A five-fold cross-validation was used to find the combination of parameters that yield the best averaged accuracy. As a result, parameter combination n_estimators = 200, max_depth = 10, min_samples_split = 10 appeared the most appropriate.

Regression setting was similar, but this time the mean squared error of the regression model was minimized within the five-fold cross-validation. As a result, the following parameter values were found optimal for regression: n_estimators = 2000, max_depth = 10, min_samples_split = 2.

2.4. CatBoost

This section describes the gradient boosting over decision trees approach. We used the CatBoost implementation (Dorogush et al., 2018) of this algorithm. The models were fitted on the ECFP4 fingerprints of ligands. Hyperparameters of the corresponding classification and regression models were firstly tuned by a grid search over the following grid: learning rate – 0.1, 0.03, 0.01; L2 leaf regularization – 1, 3, 5, 7, 9; maximum depth – 6, 8, 10. The metrics for choosing the optimal hyperparameters are the accuracy for classification and the MSE for regression, computed on the test set. As a result, the chosen set of hyperparameters for the classification task includes the learning rate of 0.1, the L2 leaf regularization parameter of 3, and the maximum depth of 10. The same set of hyperparameters, except for the L2 leaf regularization of 1, is used for the regression task. The evaluation metrics typically stopped to improve after about 200 epochs of training for classification and 500 epochs for regression. Early-stopping scenario was applied during the training with the patience parameter set to 100 epochs.

2.5. Feed-forward neural network

This subsection describes the approach based on the formalism of feed-forward neural networks. Similar to the above methods, the FFNN inputs are ECFP4 fingerprints, thus, the size of the input layer is 2048 neurons. Further, this input is passed to a set of custom fully-connected layers. During the optimization procedure, we tested various architectures that varied the width and depth of the network, tested skip-connections, the ReLU, Softplus, and PReLU activation functions, and a set of training parameters, as well. Despite our expectation, the skip-connections did not improve the performance on both the classification and regression tasks. In case of classification task, the final architecture consists of five layers total: input layer of size 2048, three hidden layers with corresponding 512, 256 and 64 neurons, and the output layer with 2 neurons. The activation function on the first, second, and third layer is ReLU. In the case of regression task, the architecture consists of five layers: the input layer is of size 2048, the hidden layers with 1024, 256 and 64 neurons, and the output layer with 1 neuron. The activation function between the layers is Softplus. The training is performed by using Stochastic Gradient Descent technique with Adam optimizer and batch size of 32. During the training, we also used a learning rate scheduler that decreased the initial learning rate of 0.001 every 50 training epochs by a factor of 0.9. On the classification task, the trained model was optimized with the cross entropy loss, while the best model was chosen according to the best accuracy score on the test subset. On the regression task, the MSE was used as the optimization loss. The model with the lowest MSE on the test subset was chosen as the optimal one.

2.6. Graph neural network

As we have already stated, the representations of ligands within the GNN approach are constructed in form of molecular graphs. Here we used the AttentiveFP GNN (Veličković et al., 2017; Xiong et al., 2020) in the DGL-LifeSci implementation (https://lifesci.dgl.ai/) for both the classification and regression tasks. The philosophy of Attentive FP is based on the message passing between nodes and extracting non-local effects: the node state vectors are processed by attentive layers that allow atoms to progressively aggregate state features from neighbors and propagate their features back to neighbors. Thus, the individual atomic states contribute to a molecular state vector. The output of Attentive FP is passed to a fully connected layer with two or one neuron (s) for either classification or regression tasks. Here we list some of the architecture parameters: node and edge feature sizes were set to 27 and 12 (being equal to the WeaveAtomFeaturizer and WeaveEdgeFeaturizer sizes), the number of layers in Attentive FP – to 2, the graph feature size – to 200, the number of readout timesteps – to 2, and the dropout rate – to 0.2. The entire network is trained in an end-to-end manner by using the Adam optimizer for gradient descent technique with the batch size of 30, L2 regularization rate of 0.0002, and early stopping callback with patience of 40 epochs. As in the case of feed-forward NN, the initial learning rate of 0.001 was gradually decreased by a factor 0.9 every 10 epochs during the training routine. On the classification task, we used the focal loss (Lin et al., 2020) to optimize the trained model and to pick the one with the best accuracy score on a test subset. On the regression task, MSE was used for both optimizing and picking the best model.

2.7. Bidirectional Transformers

This subsection presents the Transformer-based approach to the DTBA prediction. We chose BERT (Devlin et al., 2018, 2019) as the representation learner because it attends both to left and right contexts when constructing token representations, which is crucial to fully capture the intricate inter-dependencies in chemical structures. BERT training routinely consists of two stages. The first stage – pre-training on large-scale unlabeled dataset with the masked language modeling

objective – predicts the randomly masked token(s) in a sequence. The second stage – fine-tuning on the particular downstream task with the corresponding labeled (small) dataset and tailored for this task supervised learning loss function.

For the classification task, the BERT architecture is as follows: 4 layers, 12 self-attention heads, hidden size 768. For each input token from tokenized SMILES string, its embedding is constructed by summing learnable token and position embedding. Pre-trained weights were taken from Wolf et al. (2019) where they had been optimized on about 155 000 SMILES sequences from the PubChem database (Kim et al., 2019). Maximal sequence length was set to 128, vocabulary was constructed via BytePair encoding and its size is ≈52 000. During the fine-tuning stage, we experimented with several loss functions. The best accuracy and precision/recall tradeoff were achieved with the weighted focal loss for gamma of 2 and alpha of 0.81. To weaken the overfitting and aid in better generalization during the training, the dropout of 0.3 in BERT was applied to the input embeddings, attention probabilities, and hidden states outputted by each layer of the model.

The regression setting differs from the classification one in terms of the model scale – we used smaller BERT architecture with 3 layers, 6 attention heads, hidden size 768, and vocabulary size 2229. Pre-trained weights for such configuration were also taken from Wolf et al. (2019) that had been optimized on about 400 000 SMILES sequences from the ChEMBL (Mendez et al., 2019) database.

Training for both models was run over 8 and 14 epochs respectively, with the batch size of 32. The learning rates of $9 \cdot 10^{-6}$ and $1 \cdot 10^{-4}$ for the classification and regression respectively, were linearly decreased over the course of training with warm up steps proportion 0.06, parameters of the models were updated by Adam optimizer.

3. Data

A number of publicly available datasets provide data for inhibition constants K_i, as well as for a classification to active/inactive ligands towards different receptors. The inhibition constant K_i reflects how potent an inhibitor is: it is the concentration of a ligand in solution, required to produce half-maximum inhibition of a target receptor. Thus, the lower K_i is, the stronger (more active) inhibitor is, and vice versa. There is no strict threshold for the inhibition constant discriminating active and inactive ligands, however, K_i of 10 000 nM is often used as such a delimiter (He et al., 2017; Öztürk et al., 2019; Kowalewski and Ray, 2020; Shim et al., 2021; Rampogu et al., 2018).

Because our pipeline consists of the classification and regression ensembles, we prepared two datasets to train the corresponding models. The classification and regression datasets were both combined out of three databases: BindingDB (Gilson et al., 2016), DUD-E (Mysinger et al., 2012), and ChEMBL (Mendez et al., 2019). Ligands were represented with the SMILES string notation, during pre-processing they were canonicalized with RDKit (http://www.rdkit.org) with the isometry information removed.

We collected ≈30 000 ligand samples of two activity classes with respect to human thrombin which constituted our classification dataset. The fractions of active and inactive ligands are 18% and 82%. A ligand was considered inactive if any of the affinity records for it (K_i, K_d, IC50, EC50) had value >10 000 nM. If there were any conflicting labels for the same ligand in different databases, the bind class as indicated by the majority was prescribed (if it was impossible to yield the majority, the ligand was dropped from the dataset). During the training, we used a five-fold validation of the classification models and kept the same train-test split for all of them. A special attention was paid here to maintain the constant 18%/82% ratio of actives to inactives all over the folds. The fixed five-fold split assures that all models are tuned and compared on the same footing, while the constant rate between actives and inactives guarantees that all folds are equally populated with both classes.

The collected dataset for regression consists of ≈4000 unique ligand samples with corresponding values of inhibition constants. Only records

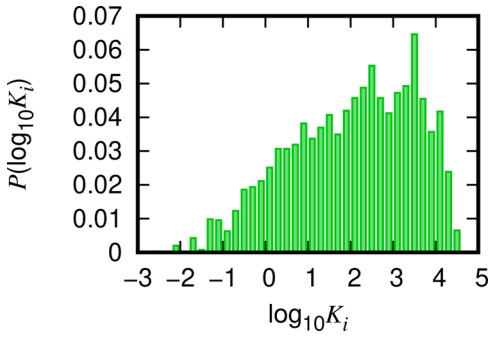

Fig. 3. Normalized distribution of binding affinities $\log_{10}K_i$ within the regression dataset. The K_i units are nanomoles.

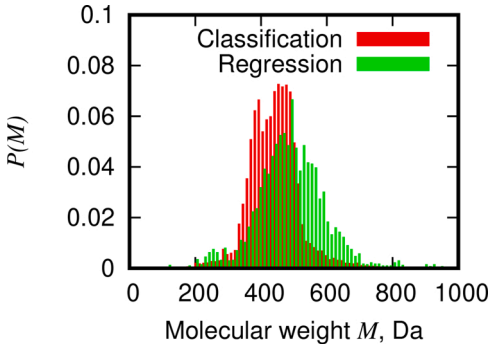

Fig. 4. Normalized distributions of molecular weights of compounds within the classification (shown in red) and regression (green) subsets.

with the precise values of K_i were included in the dataset. If there were several different values of K_i for the same ligand, the value with the biggest count was taken (or median if all counts were equal). An additional filtering rule for regression samples originates from the pipeline construction: the regression ensemble comes into play only if the classification one claims a ligand active. Therefore, on the regression task we need to predict K_i values only within the "active" range and filter out the samples with high inhibition constants. As a result, the labels in regression dataset lie within the range of $K_i \in [0 : 30\,000\,\text{nM}]$. A slight exceed of K_i range over 10 000 nM is aimed to allow the regression models to predict the inhibition constants for very weak inhibitors. In concentration measurements the error increases proportionally to the concentration itself, therefore, it is convenient to work with decimal logarithm of inhibition constants $\log_{10}K_i$. Such a conversion helps balance error contributions to the loss function at different concentration ranges. The histogram showing the distribution of samples with $\log_{10}K_i$ values is shown in Fig. 3, where K_i's are expressed in nanomoles. Similar to the classification case, the regression dataset was split into five folds.

We also considered several ways to augment the data. Commonly used tricks of SMILES augmentations include various alternative annotations, but all of them would be dropped during the canonization preprocessing. Moreover, both the ECFP4 fingerprints and molecular graphs are invariant with regard to those permutations. Therefore, we proceeded with the above-mentioned dataset with no further augmentations.

As we highlighted in the introduction, the feature engineering for the DTBA prediction is often a subject of thorough investigation. The ligands in our study are encoded in three principally different ways. The first way – utilizing ECFP4 fingerprints – is used in Support Vector Machine, Random Forest, CatBoost, and feed-forward neural network approaches. We generated these fingerprints with RDKit. The second way creates molecular graphs which are used in the graph neural network approach. The third way tokenizes the textual SMILES notations within the BERT

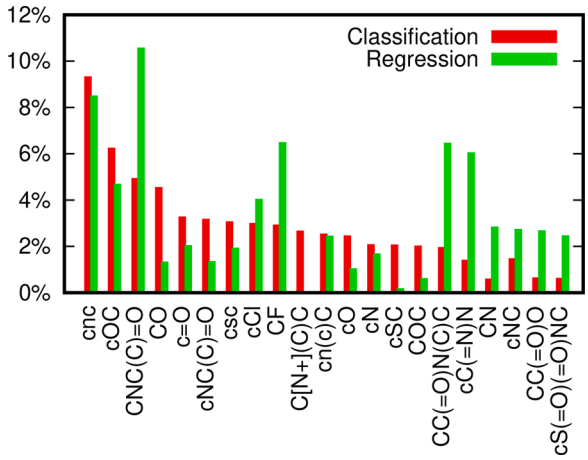

Fig. 5. Distributions over molecular fragments in compounds belonging to the classification (shown red) and regression (green) subsets.

approach, therefore, no extra pre-processing is needed – BERT digests SMILES directly. None of these featurizations use spatial coordinates, which was one of our essential requirements. Such a coordinate-free formalism allows one to predict binding affinities for novel ligands with unknown conformations.

Having two subsets, it is important to characterize their composition and diversity. The more diverse subset is, the wider chemical subspace it covers, and the broader knowledge to ML it brings. The simplest measure of such diversity might be a distribution of molecular weights of compounds within both subsets, which we present in Fig. 4. The distribution for the classification subset is mainly localized within the range of [300:500], while the plot for the regression subset demonstrates a broader shape with a solid representation of compounds with molecular weights above 500. To remind the reader, the regression subset consists of compounds which bind to the receptor, while the classification subset includes both binding and non-binding examples. Probably, one might attribute the "heavier" composition of the regression subset to a presence of certain molecular fragments necessary for binding. No specific structural requirements are imposed on non-binding compounds from the classification subset. To continue the analysis, we also made a split onto molecular fragments according to a recipe suggested by Ertl (2017). We already used this approach in Druchok et al. (2021) to characterize compounds belonging to different subsets. Such differences for the top populated fragments within the classification and regression subsets are shown in Fig. 5, where the fragments in a pseudo-SMILES notation are listed over the *X*-axis and the *Y*-axis denotes the corresponding percentages.

Before proceeding with the next section, we also wanted to discuss molecular graphs in more detail. In general, it is expected that the key interactions among nuclei and electrons in a molecule can be implicitly captured by a graph and provide a source of insight into the functions and properties of the molecule. Moreover, as for many drug discovery applications, an active conformation of a small molecule upon binding can be unknown, thus, graph-based molecular representations are more suitable than spatial coordinates.

To encode a molecular graph, node (atom) and edge (bond) features need to be defined. For this purpose we used Weave atom and bond featurizers (Kearnes et al., 2016) that produce a total of nine types of atomic features and three types of bond features to characterize atoms and their local environment. The atomic features are – atom type ('H', 'C', 'N', 'O', 'F', 'P', 'S', 'Cl', 'Br', 'I', 'other'), formal and partial charges, chirality, aromaticity, *sp* hybridization, capability of being a hydrogen bonding donor or acceptor, and ring size. The bond features – bond type ('single', 'double', 'triple', or 'aromatic'), distance within the graph, and belonging to same ring. Most of these features are encoded in a one-hot manner, except for charges and the number of rings an atom belongs to, being integers due to their additive nature.

Table 1
Performance metrics of the methods on the classification task. Presented accuracy, precision, recall, area under the receiver operating characteristic curve score, and average precision score are means over five folds.

	SVM	RF	CatBoost	FFNN	GNN	BERT
Accuracy	0.95	0.93	0.94	0.95	0.88	0.87
Precision	0.84	0.80	0.87	0.85	0.80	0.69
Recall	0.91	0.82	0.77	0.86	0.45	0.55
ROC AUC	0.98	0.97	0.97	0.91	0.94	0.90
Average precision	0.89	0.83	0.87	0.76	0.75	0.70

Table 2
Performance scores of the methods on the regression task. Mean squared error MSE, mean absolute error MAE, and coefficient of determination R^2 are averaged over five validation folds.

	SVM	RF	CatBoost	FFNN	GNN	BERT
MSE	0.56	0.71	0.62	0.66	0.66	0.82
MAE	0.55	0.65	0.59	0.60	0.59	0.66
R^2	0.74	0.66	0.71	0.69	0.69	0.61

4. Results

4.1. Classification

In this part, we analyze the performance of the classification models by a five-fold validation. We decided to compare the models on a set of scores which include accuracy, precision, recall, area under the receiver operating characteristic curve (ROC AUC), and average precision. Thus, each of the models was retrained five times, yielding five models and five sets of metrics. The means over folds of these metrics are presented in Table 1. We do not discuss the different arrangement of methods within these "leaderboards", rather present them for the reference purposes. In our view, the most important score here is the precision because it relies on true and false positives – only positives (either true or false) are passed to wet-lab experiments. Thus, the "leaders" in the precision race are CatBoost (with 0.87), FFNN (0.85), and SVM (0.84) which briefly resemble the list of accuracy successors.

4.2. Regression

Here we compare the performance of the regression models. Table 2 lists averaged scores of mean squared error, mean absolute error, and coefficient determination R^2. As one can notice, the list of regression "leaders" differs from the classification list: the lowest errors and the highest R^2 scores are achieved within the SVM approach, then followed by CatBoost, GNN, and FFNN. The weakest scores are demonstrated by RF and BERT.

In addition to the performance scores on the regression task, we decided to illustrate the reconstruction quality on one of the test folds. This can be done by plotting the predicted $\log_{10}K_i$ values versus experimental ones, so the ideal reconstruction is achieved if such a relation fits the $y = x$ function. In Fig. 6, six separate plots show the reconstruction for each of the methods. One can see the overall tendency across the methods: the predictions are rather cloud-shaped around the ideal reconstruction $y = x$, while the quality of prediction worsens at the ends of the interval. This can be attributed to the data distribution: the dataset has less samples for the very low and very high concentrations. Especially RF and FFNN seem to be prone to this issue – their bottom left and top right tails shift towards the moderate values. SVM, CatBoost, and GNN overcome this problem for the high concentrations, experiencing the shift only for the low concentrations. BERT is the least affected method, however, shows a high scatter of the predicted data around the

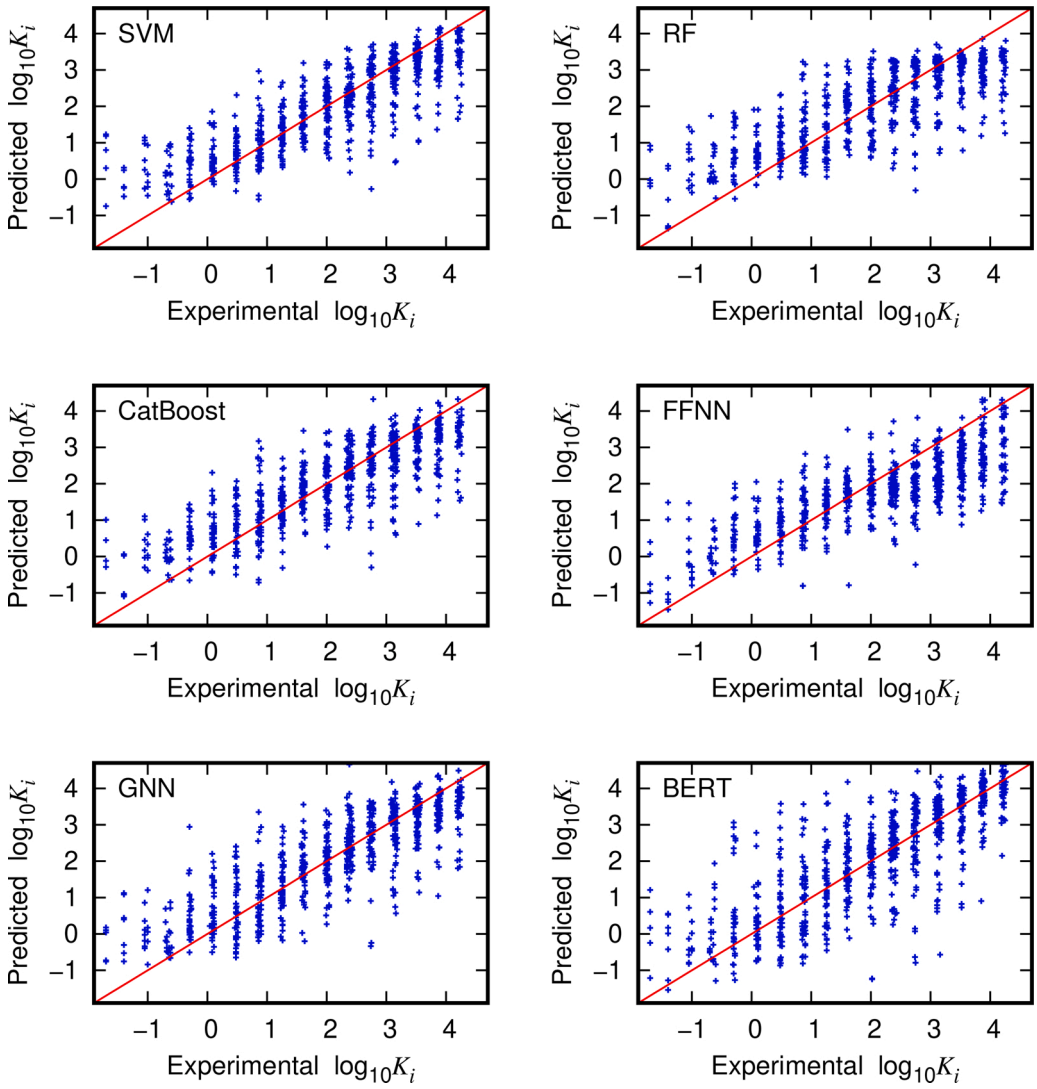

Fig. 6. The reconstruction quality of the binding affinities $\log_{10}K_i$ within different methods. The red lines denote the $y = x$ bisector to guide the eye.

$y = x$ line. This observation agrees with a relatively low R^2 coefficient of 0.61 for BERT.

4.3. Voting and averaging

We have already discussed the performance of the considered methods if they act separately. Next, we proceed with ensembling of the methods to correct their individual errors and increase the confidence on combined predictions. It is worth reminding the reader, our study focuses on the application of the ML methods to the high-throughput screening. Therefore, we are rather concerned about the prediction of positives – active ligands. The price for a false positive misclassification is much higher than the price for false negative because only the ligands predicted as active are further passed to the wet-lab experiments. Thus, to reduce the rate of erroneous positive classifications, we are going to engage an ultimately strict strategy: a ligand will be classified as active only if all the ensembled models vote it as active, otherwise a ligand will be claimed inactive. Naturally, such a scenario will increase the rate of false negatives (active ligands might be classified as inactive ones), however, it complies with the idea of selecting only confident predictions for the wet-lab stage.

The suggested strategy succeeds, if the error distribution between the methods differs: less common errors produce less ensembled false positives. Thus, to quantify the ensembling performance, we need to assess the rates of overlaps on false positives over the methods. One can do this in terms of intersection over union (IOU) between lists of false positives FP_A and FP_B from methods A and B:

Table 3
Intersection over union on false positives.

	SVM	RF	CatBoost	FFNN	GNN	BERT
SVM	–	0.72	0.36	0.51	0.32	0.53
RF	0.72	–	0.32	0.42	0.30	0.63
CatBoost	0.36	0.32	–	0.34	0.23	0.30
FFNN	0.51	0.42	0.34	–	0.18	0.32
GNN	0.32	0.30	0.23	0.18	–	0.30
BERT	0.53	0.63	0.30	0.32	0.30	–

Table 4
Intersection over union on true positives.

	SVM	RF	CatBoost	FFNN	GNN	BERT
SVM	–	0.88	0.86	0.92	0.61	0.81
RF	0.88	–	0.82	0.84	0.60	0.80
CatBoost	0.86	0.82	–	0.86	0.61	0.78
FFNN	0.92	0.84	0.86	–	0.60	0.77
GNN	0.61	0.60	0.61	0.60	–	0.59
BERT	0.81	0.80	0.78	0.77	0.59	–

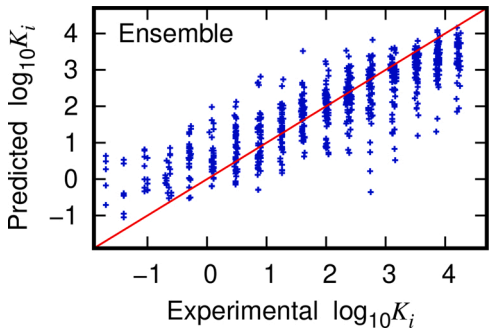

Fig. 7. The reconstruction quality of the binding affinities $\log_{10}K_i$ averaged over the ensemble of methods. The red line denotes the $y = x$ bisector to guide the eye.

$$\mathrm{IOU}_{\mathrm{AB}}(\mathrm{FP}) = \frac{\mathrm{FP_A} \cap \mathrm{FP_B}}{\mathrm{FP_A} \cup \mathrm{FP_B}} \tag{1}$$

We present the corresponding IOUs for false positives in Table 3. One can distinguish there CatBoost and GNN as the most contributors to the error-exclusion, while the least contributors are SVM and RF. It is worth noting that the conceptually different SVM and RF approaches show a relatively high mutual IOU of 0.72.

Eq. (1) can be also rewritten for IOUs of true positives which we present in Table 4. The highest IOU on true positives of 0.92 is achieved for the SVM-FFNN pair, whereas the lowest IOUs of 0.59–0.61 – for the combinations of GNN with other methods.

Of course, the interpretation of the results would be incomplete without discussion of the bare numbers on positives and negatives. The chosen test subset consists of 4949 inactive and 1097 active ligands. We apply the classification ensemble and claim ligands active only if all models classify them as active. Otherwise, ligands are claimed inactive. According to such voting scenario, we end up with only 24 false positives out of 4949. The number of true positives is 507 out of 1097. About a half of active ligands are misclassified as inactive, however, it is compensated by the extremely low rate of false positives and high precision score of 0.95.

It is worth comparing the performance scores on the test subset with the corresponding numbers on the train subset. The train subset consists of 19,781 inactive and 4376 active ligands. The sieving by the classification ensemble yields 11 false positives and 2354 true positives. Again, roughly a half of positives are correctly classified by all models, while a very low fraction of false positives is spotted.

Next, we illustrate the performance within the regression ensemble. The test subset in this case consists of 796 ligands. Having six regression models, for each test ligand we did six predictions of $\log_{10}K_i$ and then averaged them. Such averaged reconstruction is shown in Fig. 7. One can see that the cloud of ensembled predictions is thinner than the clouds of individual models (compare with Fig. 6). We also assessed the scores for averaged predictions, resulting in MSE of 0.54, MAE of 0.56, and R^2 of 0.74. These scores are better than the ones from Table 2, that speaks in favor of the ensembled approach. For the sake of comparison, we also calculated the scores on ensembled predictions on the train subset, that yields MSE of 0.15, MAE of 0.29, and R^2 of 0.93.

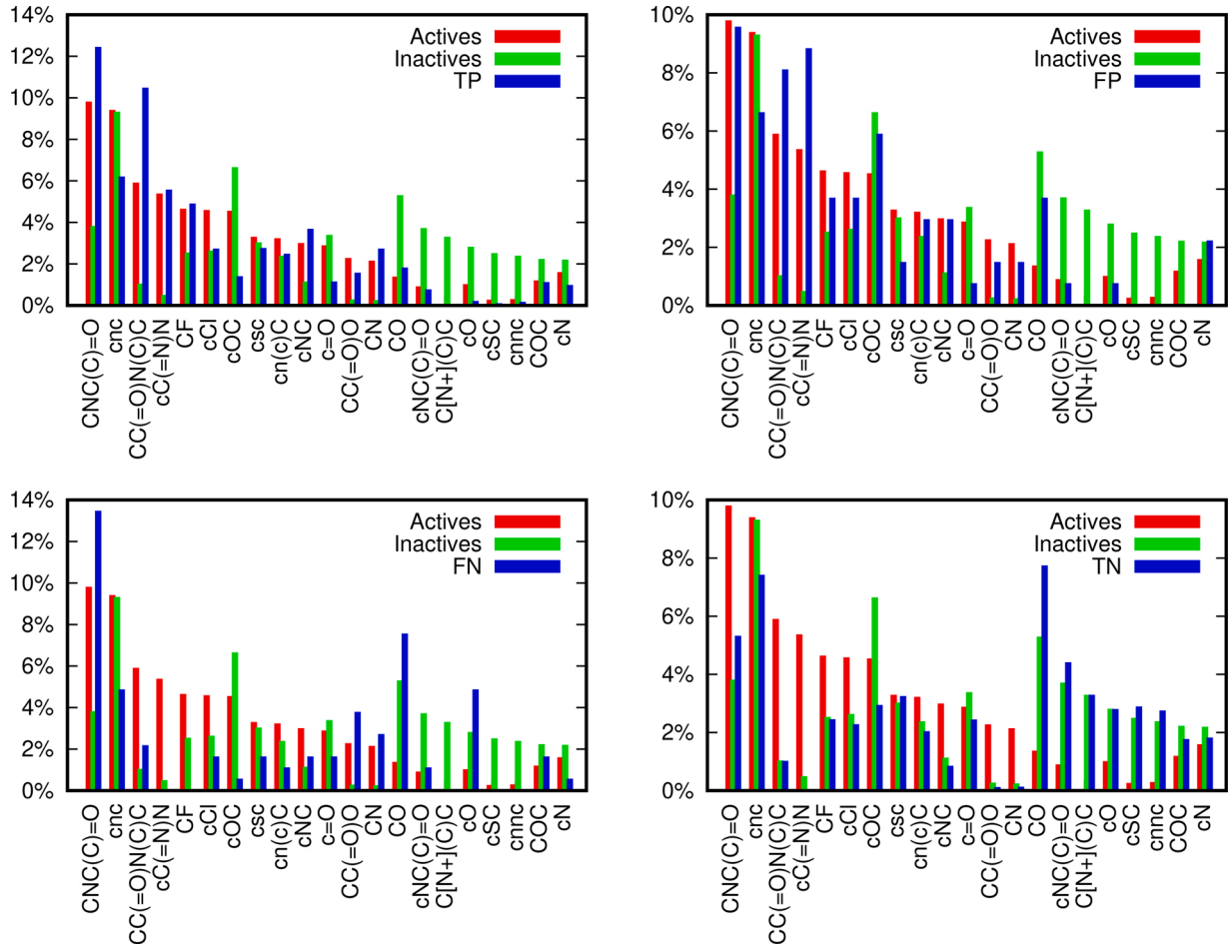

Fig. 8. Distributions over molecular fragments in compounds belonging to the classification subset. Distributions for actives and inactives are used as a reference for the comparison with true negatives (top left), false positives (top right), false negatives (bottom left), and true negatives (bottom right) distributions.

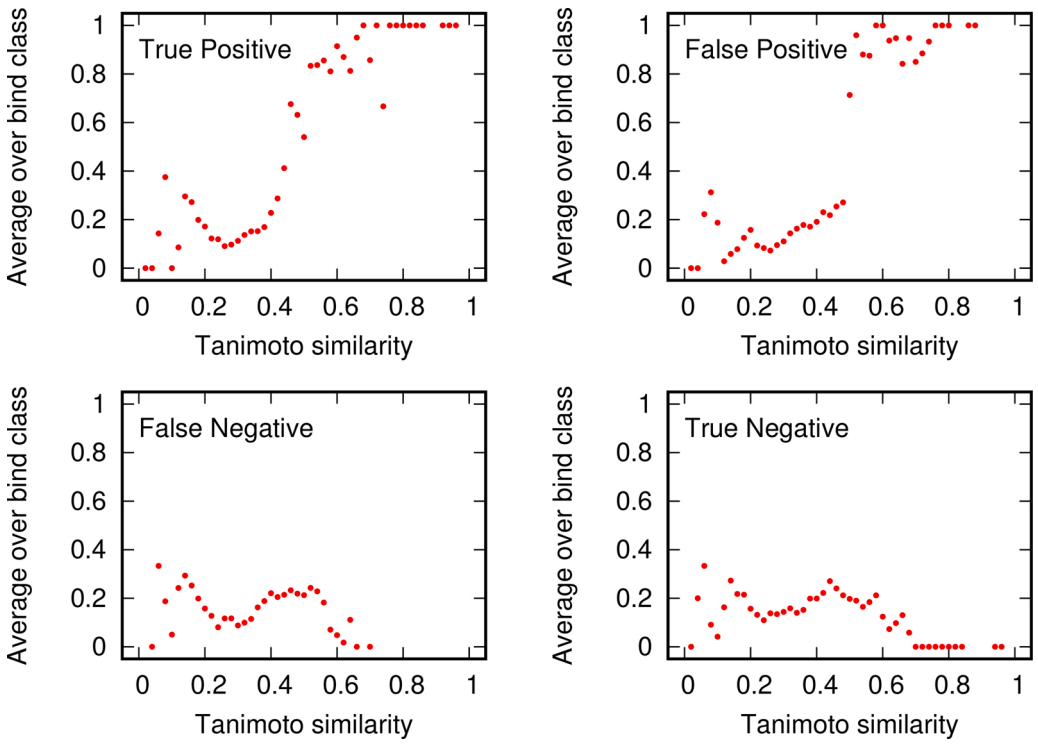

Fig. 9. Averages over bind class of train ligands as a function of Tanimoto similarity with a test ligand. Four plots show typical examples for true negatives (top left), false positives (top right), false negatives (bottom left), true negatives (bottom right).

4.4. Error analysis

Having assessed the pipeline performance, we made an attempt to sort the errors. It is assumed that the characteristics and behaviour of substances are partially conditioned by their structure, thus, the chemical similarity is often substituted by the structural similarity. On the other hand, as ML methods make use of data and, especially, of data similarities, one might assume that a prediction for a test ligand will be preferably determined by most similar ligands from a train subset. The ML models within both ensembles operate on different subsets, but take same inputs and have similar architectures. Therefore, we limited the analysis to the classification ensemble only, as the interpretation of errors is more straightforward in this case. The idea here is to use active and inactive compounds from the classification train subset as references for the comparison with true positives (TP), false positives (FP), false negatives (FN), and true negatives (TN) from the classification test subset. For this purpose we split the above-mentioned compounds onto molecular fragments (as we did for the analysis in Fig. 5) and grouped the corresponding distributions as shown in Fig. 8. The top left plot shows the distributions for the active (binding) and inactive (non-binding) compounds with distribution for the TP compounds. True positives are correctly classified binding compounds, thus, we expect this distribution to be similar to the distribution over active (binding) compounds. Indeed, one can see a good agreement between distributions for active (red bars) and TP (blue bars) compounds, while the distribution for inactives (green) significantly differs. The top right plot shows the comparison between reference distributions (actives and inactives) and false positives. False positives are inactive compounds misclassified as actives. If compared to the two references, the FP distribution reveals more similarity with the distribution for actives, not with the inactives distribution as it should be. The bottom left plot stands for false negatives – the active compounds misclassified as inactives. It is difficult to find a clear similarity pattern here. Probably in such uncertain cases the ensemble decision leans to declare compounds as inactives, since the classification subset is imbalanced in favor of inactive samples. The bottom right plot compares true negatives – correctly classified inactives. As expected, the TN distribution (blue bars) agrees with the distribution for inactives.

Thus far we examined the similarity of compounds on the level of fragments. Although being good in catching of atomic sequences, this kind of analysis does not consider the molecules as a whole. An alternative approach here could be a comparison of molecular fingerprints (Baldi and Nasr, 2010; Bajusz et al., 2015; Lo and Torres, 2016) as they aggregate an information about presence or absence of particular substructures on the molecular level. Various types of fingerprints can be used, as well as fingerprint-fingerprint distance criteria. Having ECFP4 fingerprints already calculated, we used them for this purpose, while for a distance criterion we chose the Tanimoto similarity in the RDKit's implementation. The Tanimoto similarity index varies within the range of [0:1] – the more similar chemical compounds are, the closer their similarity index approaches 1, and vice versa. Thus, for any compound from the test subset, one can bucketize train compounds by their Tanimoto similarity with the test compound and average their bind classes. To remind the reader, we label active ligands with class 1 and inactives – with class 0. In Fig. 9 we show four typical examples of such class-similarity averages. The *X*-axis in the plots in Fig. 9 denotes a Tanimoto similarity between example test ligand and all ligands from the train subset. The *Y*-axis stands for the averaged bind class of neighbors. In some plots the red dots might appear irregularly in vicinity of similarity = 1, indicating the lack of train ligands with a certain value of similarity. The top left plot in Fig. 9 shows a distribution for one of the ligands correctly classified as active (true positive). Indeed, the expected class 1 coincides with the average over classes of similar ligands – see the plateau at value 1 for similarities above 0.7. The top right plot demonstrates an example for a ligand with false positive prediction. Its actual class is 0, however, the majority of similar ligands voted for the class 1. A similar picture can be seen in the bottom left plot with an example of false negative prediction: an active ligand is misclassified because of the majority of similar train samples of the class 0. The bottom right plot shows a true negative example with clear prediction of the class 0. One

might conclude that predictions are mostly governed by train ligands with high similarity to the tested sample. Important to note, the above analysis is made over fingerprints, but other molecular representations are considered in the study, as well. We also wanted to remind the reader, the ML methods in hand mostly work in a non-linear fashion, so the plain average over bind classes here is rather a simplification, aimed to provide a qualitative interpretation.

5. Summary

Typical tasks of high-throughput screening might require billions of *in silico* experiments (Ton et al., 2020), thus, promoting the family of machine learning methods, as they are much faster than traditional simulation-based approaches. Here we present an ML-based pipeline for high-throughput screening of binding affinities between small organic molecules and proteins. The methods in hand are Support Vector Machine, Random Forest, CatBoost, feed-forward neural network, graph neural network, and Bidirectional Encoder Representations from Transformers. The pipeline unites two subsequent ensembles – classification and regression. This two-ensemble setup is mostly motivated by the data-specific reasons – the majority of the binding affinity data comes as two-class records, while the data with binding strength is often imbalanced towards active ligands. Thus, the classification ensemble decides whether a particular ligand binds to a protein or not. We used an ultimate voting scenario – a ligand is claimed active (affine to a protein) only in case all models predict it as active. Such a strict rule is aimed to reduce the rate of false positives and, therefore, the number of wet-lab experiments with weakly scored ligands. If a ligand is recognized as active, it is then passed to the regression ensemble that predicts its binding affinity expressed as the inhibition constant $\log_{10}K_i$. The output of the second ensemble is the mean over six regression predictions. With classification followed by regression, not only drug candidate class (active/inactive) can be assessed, but also the strength of association. The inhibition constant is of the high interest for the pharma industry because it allows a selection of candidates with the highest estimated activity.

The results of the study show that the two-ensemble pipeline makes use of all available affinity data, thus, reducing the error-rate and being fast enough for the high-throughput screening. We also demonstrate how the diversity of the methods allows one to exclude (on the classification) or compensate (on the regression) the errors made by their ensemble mates. We witnessed this in the Results section where the performance scores of individual models are weaker than the scores of ensembled predictions. The analysis of the errors concludes the Results section. The performance of the discussed pipeline is validated on the case of human thrombin, however, the whole scheme can be applied for other proteins, as well.

Data availability statement

The data that support the findings of the study are available from the corresponding author upon reasonable request.

Conflict of interest

All authors declare no scientific or financial conflict of interest.

References

Bajusz, D., Rácz, A., Héberger, K., 2015. Why is tanimoto index an appropriate choice for fingerprint-based similarity calculations? J. Cheminform. 7, 20. https://doi.org/10.1186/s13321-015-0069-3.

Baldi, P., Nasr, R., 2010. When is chemical similarity significant?. The statistical distribution of chemical similarity scores and its extreme values. J. Chem. Inf. Model. 50 (7), 1205–1222. https://doi.org/10.1021/ci100010v.

Beck, D., Haffari, G., Cohn, T., 2018. Graph-to-sequence learning using gated graph neural networks. Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). Association for Computational Linguistics, Melbourne, Australia, pp. 273–283. https://doi.org/10.18653/v1/P18-1026.

Beck, B., Shin, B., Choi, Y., Park, S., Kang, K., 2020. Predicting commercially available antiviral drugs that may act on the novel coronavirus (sars-cov-2) through a drug-target interaction deep learning model. Comput. Struct. Biotechnol. J. 18, 784–790. https://doi.org/10.1016/j.csbj.2020.03.025.

Breiman, L., 2001. Random forests. Mach. Learn. 45 (1), 5–32. https://doi.org/10.1023/A:1010933404324.

Chen, Y.-C., 2015. Beware of docking! Trends Pharmacol. Sci. 36 (2), 78–95. https://doi.org/10.1016/j.tips.2014.12.001.

Chen, J.-Q., Chen, H.-Y., Dai, W.-j., Lv, Q.-J., Chen, C.-C., 2019. Artificial intelligence approach to find lead compounds for treating tumors. J. Phys. Chem. Lett. 10 (15), 4382–4400. https://doi.org/10.1021/acs.jpclett.9b01426.

Chupakhin, V., Marcou, G., Baskin, I., Varnek, A., Rognan, D., 2013. Predicting ligand binding modes from neural networks trained on protein-ligand interaction fingerprints. J. Chem. Inf. Model. 53 (4), 763–772. https://doi.org/10.1021/ci300200r.

Cortes, C., Vapnik, V., 1995. Support-vector networks. Mach. Learn. 20 (3), 273–297. https://doi.org/10.1007/BF00994018.

Davis, M., Hunt, J., Herrgard, S., Ciceri, P., Wodicka, L., Pallares, G., Hocker, M., Treiber, D., Zarrinkar, P., 2011. Comprehensive analysis of kinase inhibitor selectivity. Nat. Biotechnol. 29, 1046–1051.

de Azevedo, W.F.JJr. (Ed.), 2019. Docking Screens for Drug Discovery. Humana Press, New York, NY. https://doi.org/10.1007/978-1-4939-9752-7.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2018. BERT: Pre-Training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805.

Devlin, J., Chang, M.-W., Lee, K., Toutanova, K., 2019. BERT: pre-training of deep bidirectional transformers for language understanding. Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers). Association for Computational Linguistics, Minneapolis, Minnesota, pp. 4171–4186. https://doi.org/10.18653/v1/N19-1423.

Dorogush, A., Ershov, V., Gulin, A., 2018. CatBoost: Gradient Boosting With Categorical Features Support. arXiv:1810.11363.

Druchok, M., Yarish, D., Gurbych, O., Maksymenko, M., 2021. Toward efficient generation, correction, and properties control of unique drug-like structures. J. Comput. Chem. 42 (11), 746–760. https://doi.org/10.1002/jcc.26494.

Durant, J., Leland, B., Henry, D., Nourse, J., 2002. Reoptimization of mdl keys for use in drug discovery. J. Chem. Inf. Comput. Sci. 42 (6), 1273–1280. https://doi.org/10.1021/ci010132r.

Ellingson, S., Davis, B., Allen, J., 2020. Machine learning and ligand binding predictions: a review of data, methods, and obstacles. Biochim. Biophys. Acta (BBA) – General Subj. 1864 (6), 129545. https://doi.org/10.1016/j.bbagen.2020.129545.

Ertl, P., 2017. An algorithm to identify functional groups in organic molecules. J. Cheminform. 9, 36. https://doi.org/10.1186/s13321-017-0225-z.

Frimurer, T., Peters, G., Iversen, L., Andersen, H., Møller, N., Olsen, O., 2003. Ligand-induced conformational changes: improved predictions of ligand binding conformations and affinities. Biophys. J. 84, 2273–2281. https://doi.org/10.1016/S0006-3495(03)75033-4.

Gao, K., Nguyen, D., Chen, J., Wang, R., Wei, G.-W., 2020. Repositioning of 8565 existing drugs for COVID-19. J. Phys. Chem. Lett. 11 (13), 5373–5382. https://doi.org/10.1021/acs.jpclett.0c01579.

Gilson, M., Liu, T., Baitaluk, M., Nicola, G., Hwang, L., Chong, J., 2016. BindingDB in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. Nucleic Acids Res. 44, D1045–D1053.

Hartshorn, M., Verdonk, M., Chessari, G., Brewerton, S., Mooij, W., Mortenson, P., Murray, C., 2007. Diverse, high-quality test set for the validation of protein-ligand docking performance. J. Med. Chem. 50 (4), 726–741. https://doi.org/10.1021/jm061277y.

He, T., Heidemeyer, M., Ban, F., Cherkasov, A., Ester, M., 2017. SimBoost: a read-across approach for predicting drug-target binding affinities using gradient boosting machines. J. Cheminform. 9, 24. https://doi.org/10.1186/s13321-017-0209-z.

Heck, G., Pintro, V., Pereira, R., de Avila, M., Levin, N., de Azevedo Jr., W., 2017. Supervised machine learning methods applied to predict ligand-binding affinity. Curr. Med. Chem. 24 (23), 2459–2470. https://doi.org/10.2174/0929867324666170623092503.

Jiang, M., Li, Z., Zhang, S., Wang, S., Wang, X., Yuan, Q., Wei, Z., 2020. Drug-target affinity prediction using graph neural network and contact maps. RSC Adv. 10, 20701–20712. https://doi.org/10.1039/D0RA02297G.

Jiménez, J., Škalič, M., Martínez-Rosell, G., De Fabritiis, G., 2018. K_{DEEP}: protein-ligand absolute binding affinity prediction via 3d-convolutional neural networks. J. Chem. Inf. Model. 58 (2), 287–296. https://doi.org/10.1021/acs.jcim.7b00650.

Jin, Z., Du, X., Xu, Y., Deng, Y., Liu, M., Zhao, Y., Zhang, B., Li, X., Zhang, L., Peng, C., Duan, Y., Yu, J., Wang, L., Yang, K., Liu, F., Jiang, R., Yang, X., You, T., Liu, X., Yang, X., Bai, F., Liu, H., Liu, X., Guddat, L., Xu, W., Xiao, G., Qin, C., Shi, Z., Jiang, H., Rao, Z., Yang, H., 2020. Structure of M^{pro} from SARS-CoV-2 and discovery of its inhibitors. Nature 582, 289–293. https://doi.org/10.1038/s41586-020-2223-y.

Jorissen, R., Gilson, M., 2005. Virtual screening of molecular databases using a support vector machine. J. Chem. Inf. Model. 45 (3), 549–561. https://doi.org/10.1021/ci049641u.

Kadioglu, O., Saeed, M., Greten, H., Efferth, T., 2021. Identification of novel compounds against three targets of sars cov-2 coronavirus by combined virtual screening and supervised machine learning. Comput. Biol. Med. 133, 104359. https://doi.org/10.1016/j.compbiomed.2021.104359.

Kearnes, S., McCloskey, K., Berndl, M., Pande, V., Riley, P., 2016. Molecular graph convolutions: moving beyond fingerprints. J. Comput.-Aided Mol. Des. 30, 595–608. https://doi.org/10.1007/s10822-016-9938-8.

Kim, S., Chen, J., Cheng, T., Gindulyte, A., He, J., He, S., Li, Q., Shoemaker, B., Thiessen, P., Yu, B., Zaslavsky, L., Zhang, J., Bolton, E., 2019. PubChem 2019 update: improved access to chemical data. Nucleic Acids Res. 47, D1102–D1109.

King, R., Hirst, J., Sternberg, M., 1995. Comparison of artificial intelligence methods for modeling pharmaceutical QSARs. Appl. Artif. Intell. 9 (2), 213–233. https://doi.org/10.1080/08839519508945474.

Kowalewski, J., Ray, A., 2020. Predicting novel drugs for sars-cov-2 using machine learning from a >10 million chemical space. Helion 6, e04639. https://doi.org/10.1016/j.heliyon.2020.e04639.

Kundu, I., Paul, G., Banerjee, R., 2018. A machine learning approach towards the prediction of protein-ligand binding affinity based on fundamental molecular properties. RSC Adv. 8, 12127–12137. https://doi.org/10.1039/C8RA00003D.

Kwon, Y., Shin, W.-H., Ko, J., Lee, J., 2020. Ak-score: accurate protein-ligand binding affinity prediction using an ensemble of 3d-convolutional neural networks. Int. J. Mol. Sci. 21 (22), 8424. https://doi.org/10.3390/ijms21228424.

Li, L., Koh, C., Reker, D., Brown, J., Wang, H., Lee, N., Liow, H.-h., Dai, H., Fan, H.-M., Chen, L., Wei, D.-Q., 2019a. Predicting protein-ligand interactions based on bow-pharmacological space and Bayesian additive regression trees. Sci. Rep. 9, 7703. https://doi.org/10.1038/s41598-019-43125-6.

Li, Y., Rezaei, M., Li, C., Li, X., 2019b. DeepAtom: a framework for protein-ligand binding affinity prediction. 2019 IEEE International Conference on Bioinformatics and Biomedicine (BIBM) 303–310.

Lin, T.-Y., Goyal, P., Girshick, R., He, K., Dollár, P., 2020. Focal loss for dense object detection. IEEE Trans. Pattern Anal. Mach. Intell. 42, 318–327.

Lipman, D., Pearson, W., 1985. Rapid and sensitive protein similarity searches. Science 227 (4693), 1435–1441. https://doi.org/10.1126/science.2983426.

Lo, Y.-C., Torres, J., 2016. Chemical similarity networks for drug discovery. In: Chen, T., Chai, S. (Eds.), Special Topics in Drug Discovery. IntechOpen. https://doi.org/10.5772/65106. Ch. 3.

Mendez, D., Gaulton, A., Bento, A., Chambers, J., Veij, M., Felix, E., Magariños, M., Mosquera, J., Mutowo-Meullenet, P., Nowotka, M., Gordillo-Marañón, M., Hunter, F., Junco, L., Mugumbate, G., Rodriguez-Lopez, M., Atkinson, F., Bosc, N., Radoux, C., Segura-Cabrera, A., Hersey, A., Leach, A., 2019. ChEMBL: towards direct deposition of bioassay data. Nucleic Acids Res. 47, D930–D940.

Michel, M., Menéndez Hurtado, D., Elofsson, A., 2018. PconsC4: fast, accurate and hassle-free contact predictions. Bioinformatics 35 (15), 2677–2679. https://doi.org/10.1093/bioinformatics/bty1036.

Mottaqi, M., Mohammadipanah, F., Sajedi, H., 2021. Contribution of machine learning approaches in response to sars-cov-2 infection. Inform. Med. Unlocked 23, 100526. https://doi.org/10.1016/j.imu.2021.100526.

Mysinger, M., Carchia, M., Irwin, J., Shoichet, B., 2012. Directory of useful decoys, enhanced (DUD-E): better ligands and decoys for better benchmarking. J. Med. Chem. 55, 6582–6594.

Nand, M., Maiti, P., Joshi, T., Chandra, S., Pande, V., Kuniyal, J., Ramakrishnan, M., 2020. Virtual screening of anti-hiv1 compounds against sars-cov-2: machine learning modeling, chemoinformatics and molecular dynamics simulation based analysis. Sci. Rep. 10, 20397. https://doi.org/10.1038/s41598-020-77524-x.

Nguyen, T., Le, H., Quinn, T., Nguyen, T., Le, T., Venkatesh, S., 2020. GraphDTA: predicting drug-target binding affinity with graph neural networks. Bioinformatics. https://doi.org/10.1093/bioinformatics/btaa921.

Öztürk, H., Özgür, A., Ozkirimli, E., 2018. DeepDTA: deep drug-target binding affinity prediction. Bioinformatics 34 (17), i821–i829. https://doi.org/10.1093/bioinformatics/bty593.

Öztürk, H., Ozkirimli, E., Özgür, A., 2019. WideDTA: Prediction of Drug-Target Binding Affinity. arXiv:1902.04166.

Pagadala, N., Syed, K., Tuszynski, J., 2017. Software for molecular docking: a review. Biophys. Rev. 9, 91–102. https://doi.org/10.1007/s12551-016-0247-1.

Pahikkala, T., Airola, A., Pietilä, S., Shakyawar, S., Szwajda, A., Tang, J., Aittokallio, T., 2014. Toward more realistic drug-target interaction predictions. Brief. Bioinform. 16 (2), 325–337. https://doi.org/10.1093/bib/bbu010.

Payne, J., Srouji, M., Yap, D., Kosaraju, V., 2020. BERT Learns (and Teaches) Chemistry. arXiv:2007.16012.

Pearson, W., Lipman, D., 1988. Improved tools for biological sequence comparison. Proc. Natl. Acad. Sci. USA 85 (8), 2444–2448. https://doi.org/10.1073/pnas.85.8.2444.

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.

Rampogu, S., Zeb, A., Baek, A., Park, C., Son, M., Lee, K.W., 2018. Discovery of potential plant-derived peptide deformylase (pdf) inhibitors for multidrug-resistant bacteria using computational studies. J. Clin. Med. 7 (12) https://doi.org/10.3390/jcm7120563.

Rives, A., Goyal, S., Meier, J., Guo, D., Ott, M., Zitnick, C., Ma, J., Fergus, R., 2019. Biological Structure and Function Emerge from Scaling Unsupervised Learning to 250 Million Protein Sequences. https://doi.org/10.1101/622803 arXiv:bioRxiv.

Rogers, D., Hahn, M., 2010. Extended-connectivity fingerprints. J. Chem. Inf. Model. 50 (5), 742–754. https://doi.org/10.1021/ci100050t.

Santana, M., Silva-Jr, F., 2020. De novo design and bioactivity prediction of sars-cov-2 main protease inhibitors using recurrent neural network-based transfer learning. BMC Chem. 15, 8. https://doi.org/10.1186/s13065-021-00737-2.

Scarselli, F., Gori, M., Tsoi, A., Hagenbuchner, M., Monfardini, G., 2009. The graph neural network model. IEEE Trans. Neural Netw. 20 (1), 61–80. https://doi.org/10.1109/TNN.2008.2005605.

Schneider, M., Pons, J.-L., Bourguet, W., Labesse, G., 2019. Towards accurate high-throughput ligand affinity prediction by exploiting structural ensembles, docking metrics and ligand similarity. Bioinformatics 36 (1), 160–168. https://doi.org/10.1093/bioinformatics/btz538.

Schwaller, P., Laino, T., Gaudin, T., Bolgar, P., Hunter, C., Bekas, C., Lee, A., 2019. Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. ACS Central Sci. 5, 1572–1583.

Shim, J., Hong, Z.-Y., Sohn, I., Hwang, C., 2021. Prediction of drug-target binding affinity using similarity-based convolutional neural network. Sci. Rep. 11, 4416. https://doi.org/10.1038/s41598-021-83679-y.

Shin, B., Park, S., Kang, K., Ho, J., 2019. Self-Attention Based Molecule Representation for Predicting Drug-Target Interaction. arXiv:1908.06760.

Shoichet, B., McGovern, S., Wei, B., Irwin, J., 2002. Lead discovery using molecular docking. Curr. Opin. Chem. Biol. 6 (4), 439–446. https://doi.org/10.1016/S1367-5931(02)00339-3.

Tang, J., Szwajda, A., Shakyawar, S., Xu, T., Hintsanen, P., Wennerberg, K., Aittokallio, K., 2014. Making sense of large-scale kinase inhibitor bioactivity data sets: a comparative and integrative analysis. J. Chem. Inf. Model. 54 (3), 735–743.

Ton, A.-T., Gentile, F., Hsing, M., Ban, F., Cherkasov, A., 2020. Rapid identification of potential inhibitors of SARS-CoV-2 main protease by deep docking of 1.3 billion compounds. Mol. Inform. 39 (8), 2000028. https://doi.org/10.1002/minf.202000028.

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y., 2017. Graph Attention Networks. arXiv:1710.10903.

Weininger, D., 1988. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. J. Chem. Inf. Comput. Sci. 28 (1), 31–36. https://doi.org/10.1021/ci00057a005.

Weininger, D., Weininger, A., Weininger, J., 1989. SMILES. 2. Algorithm for generation of unique SMILES notation. J. Chem. Inf. Comput. Sci. 29 (2), 97–101. https://doi.org/10.1021/ci00062a008.

Wolf, T., Debut, L., Sanh, V., Chaumond, J., Delangue, C., Moi, A., Cistac, P., Rault, T., Louf, R., Funtowicz, M., Davison, J., Shleifer, S., von Platen, P., Ma, C., Jernite, Y., Plu, J., Xu, C., Le Scao, T., Gugger, S., Drame, M., Lhoest, Q., Rush, A., 2019. HuggingFace's Transformers: State-of-the-Art Natural Language Processing. arXiv: 1910.03771.

Xiong, Z., Wang, D., Liu, X., Zhong, F., Wan, X., Li, X., Li, Z., Luo, X., Chen, K., Jiang, H., Zheng, M., 2020. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. J. Med. Chem. 63 (16), 8749–8760. https://doi.org/10.1021/acs.jmedchem.9b00959.

Yugandhar, K., Gromiha, M., 2014. Feature selection and classification of protein-protein complexes based on their binding affinities using machine learning approaches. Proteins Struct. Funct. Bioinform. 82 (9), 2088–2096. https://doi.org/10.1002/prot.24564.