

# LigityScore: Convolutional Neural Network for Binding-affinity Predictions

Joseph Azzopardi<sup>1</sup><sup>a</sup> and Jean Paul Ebejer<sup>1,2</sup><sup>b</sup>

<sup>1</sup>Department of Artificial Intelligence, University of Malta, Msida, MSD 2080, Malta

<sup>2</sup>Centre for Molecular Medicine and Biobanking, University of Malta, Msida, MSD 2080, Malta

**Keywords:** Virtual Screening, Structure-based Virtual Screening, Scoring Function, Pharmacophoric Interaction Point, Machine Learning, Deep-Learning, Convolutional Neural Networks, LigityScore.

**Abstract:** Scoring functions are at the heart of structure-based drug design and are used to estimate the binding of ligands to a target. Seeking a scoring function that can accurately predict the binding affinity is key for successful virtual screening methods. Deep learning approaches have recently seen a rise in popularity as a means to improve the scoring function having as a key advantage the automatic extraction of features and the creation of a complex representation without feature engineering and expert knowledge. In this study we present LigityScore1D and LigityScore3D, which are rotationally invariant scoring functions based on convolutional neural networks. LigityScore descriptors are extracted directly from the structural and interacting properties of the protein-ligand complex which are input to a CNN for automatic feature extraction and binding affinity prediction. This representation uses the spatial distribution of Pharmacophoric Interaction Points, derived from interaction features from the protein-ligand complex based on pharmacophoric features conformant to specific family types and distance thresholds. The data representation component and the CNN architecture together, constitute the LigityScore scoring function. The main contribution for this study is to present a novel protein-ligand representation for use as a CNN based SF for binding affinity prediction. LigityScore models are evaluated for scoring power on the latest two CASF benchmarks. The Pearson Correlation Coefficient, and the standard deviation in linear regression were used to compare and rank LigityScore with the benchmark model, and also to other models recently published in literature. LigityScore3D has achieved better overall results and showed similar performance in both CASF benchmarks. LigityScore3D ranked 5<sup>th</sup> place for the CASF-2013 benchmark, and 8<sup>th</sup> for CASF-2016, with an average R-score performance of 0.713 and 0.725 respectively. LigityScore1D ranked 8<sup>th</sup> place for the CASF-2013 and 7<sup>th</sup> place for CASF-2016 with an R-score performance of 0.635 and 0.741 respectively. Our methods show relatively good performance when compared to the Pafnucy model (one of the best performing CNN based scoring functions), on the CASF-2013 benchmark using a less computationally complex model that can be trained 16 times faster.

## 1 INTRODUCTION

Structure-based virtual screening (SBVS) employs the known 3D protein structure to apply computational methods that measure the ability of a small molecule to bind to the target protein. Docking is one of the most popular SBVS methods and is used to validate the ability of small molecules to bind to the target structure in a typical 'lock and key' fashion (Ching et al., 2018). During the docking process many possible ligand conformers, or docking poses, are iteratively tested at the binding site to find a suit-

able ligand pose yielding the best binding affinity. The binding affinity of a particular pose is determined by the scoring function (SF) of the docking program. The SF is therefore crucial for docking programs in SBVS and can be defined as "Estimating how strongly the docked pose of a ligand binds to the target" (Ain et al., 2015). Scoring functions are typically used for fast evaluation of protein-ligand interactions, so building an efficient and powerful SF is a means of accelerating the virtual screening (VS) process. The SF is considered the foundation in SBVS and used in the following areas for hit discovery and optimization (Ragoza et al., 2017):

1. **Pose Prediction:** Predict the shape of the ligand

<sup>a</sup> <https://orcid.org/0000-0001-9058-5361>

<sup>b</sup> <https://orcid.org/0000-0003-0888-2637>

which gives the best binding affinity.

2. **Ranking:** Ranking of molecules with known binding pose in order of the binding affinity for a given protein target.
3. **Classification:** Given the binding-pose, classify whether a small molecule is active or inactive for a given 3D structure of a target protein.

Intense research has been carried out over the years on this problem, however improving the accuracy for binding affinity prediction has proven to be a non-trivial task (Ain et al., 2015). Despite several advances in binding affinity prediction, the current binding affinity estimates are still not accurate enough leading to high false positive rates (Zheng et al., 2019). The work described in this paper focuses directly on the SF layer of SBVS and tackles this problem by finding an alternative data representation of the protein-ligand complex and then applies a Convolutional Neural Network (CNN) regression model to implement the SF. Deep Learning (DL) models have achieved remarkable success in various areas such as computer vision (Szegedy et al., 2015) and natural language processing (Goldberg, 2016). Inspired by this success, the use of DL and in particular CNNs have naturally become an obvious strategy to apply for computer aided drug design. Conventional ML techniques, such as Random Forests and Support Vector Machines, are limited to process raw data, requiring careful feature engineering and expert domain knowledge (LeCun et al., 2015). Deep learning methods aims to reduce feature engineering and automatically extract the salient feature information from the input data using multiple hidden layers, provided it has a suitable representation of the molecular interactions between the protein and ligand (Pérez-Sianes et al., 2019). Therefore, our strategy is to allow deep learning models to *learn* the underlying molecular interactions so that this learned information can be reapplied to other protein targets for exploration of novel ligands, without the need to incorporate expert chemical knowledge. Our work is evaluated using the CASF-2013 (Li et al., 2014) and CASF-2016 (Su et al., 2018) benchmarks and is also compared to other recently-published SF methods that use the same benchmarks.

The first deep neural network (DNN) used for VS was introduced by the winners of the 2012 Kaggle Merck Molecular activity challenge (Kaggle, 2012) where the team applied a multi-task deep feed forward network for quantitative structure-activity. This work was later published by Ma et al. (2015) and generated a lot of interest and excitement around the use of DL in this field. Ma et al. (2015) have achieved an

average Pearson correlation coefficient of 0.496 using a multi-task DNN compared to the 0.423 obtained when using a Random Forest model.

The Convolutional Neural Network (CNN) is one of the most common DL architectures used for SFs (Rifaioglu et al., 2018; Pérez-Sianes et al., 2019). The CNN uses a number of sequential layers of convolutions and pooling modules to encode the hidden features of the data, and then use a fully connected feed-forward neural network for classification or regression. One of the advantages of CNNs in the area of structure-based drug design is its ability to capture local spatial information interactions between protein-ligand complexes. In recent years, many CNN models have been applied to SF development (Ragoza et al., 2017; Stepniewska-Dziubinska et al., 2017; Jiménez et al., 2018; Zheng et al., 2019; Liu et al., 2019).

Ragoza et al. (2017) were the first to use CNNs to implement a DL scoring function that predicts the docking score for a drug target interaction which was then used for SBVS and pose prediction. However, Sieg et al. (2019) later showed that their model was effected by non-causal bias. One of the most promising DL SF models, Pafnucy, was proposed by Stepniewska-Dziubinska et al. (2017) where the authors achieved a Pearson Coefficient,  $R$ , for the predicted versus experimental binding affinity of 0.70 on the CASF-2013, and 0.78 on the CASF-2016 benchmarks respectively. Pafnucy uses a 3D CNN model with a 4D tensor to represent 19 protein-ligand features in 3D. The 4D tensor includes discretised atom location in the first three dimensions, whilst the features for the particular atom are encoded in the fourth dimension.

Jiménez et al. (2018) also utilise a 3D CNN model termed  $K_{\text{deep}}$  and achieved state of the art results on the CASF-2016 test with an  $R$  value of 0.82. Their input features use a 3D voxel representation where each channel encodes a particular property of the atom. Each protein-ligand complex is represented by a 4D tensor, where each 3D hyperplane represents the protein-ligand complex with respect to a particular property only. The eight properties chosen for  $K_{\text{deep}}$  include: hydrophobic, aromatic, hydrogen bond acceptor (HBA), hydrogen bond donor (HBD), cation, anion, metallic, and excluded volume. A recent study proposed by Zheng et al. (2019) compares their model to Stepniewska-Dziubinska et al. (2017) and criticize the Pafnucy model that the protein-ligand interactions in a 3D grid box of 20Å are not sufficient to capture all the protein-ligand interactions, and suggest that other long-range electrostatic interactions outside the 20Å, termed *non-local* electrostatic interactions are also important. To capture all the interactions between protein-ligand

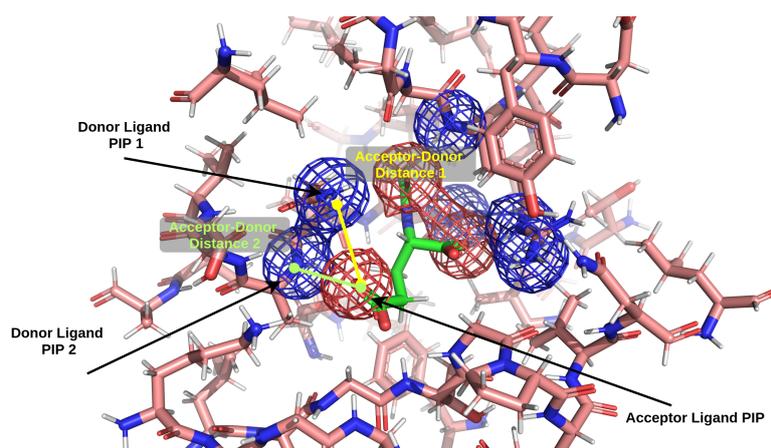


Figure 1: PIP pair interaction between two hydrogen bond donor protein PIPs (blue mesh), and a hydrogen bond acceptor ligand PIP (red mesh). Other PIPs not shown for clarity. For each PIP interaction the distance between the geometric centres of the PIPs is calculated.

complexes, Zheng *et al.* (2019) divide all the 3D space of the binding site into a number of shells or zones and count the number of different element-to-element interactions within each shell. Their experiments show that the shells closer to the ligand are more important, as was intuitively expected, however they also show that non-local interactions have significant importance. Zheng *et al.* (2019) compare their method OnionNet to another recent model, AGL-Score by Nguyen and Wei (2019a). Both methods use CASF as an evaluation benchmark. Zheng *et al.* comment that OnionNet provides a more *complete* local environment and improves the affinity prediction performance with an  $R$  of 0.833. To date this represents the best performing ML scoring function. The better results are achieved by adding novel features relating to the physical and biological information of the complex using graph theory.

One of the limitations of Pafnucy and  $K_{\text{deep}}$  is the dependency on the coordinate frame. The representation can be thought of as one snapshot of the structure. However, if the orientation from where the snapshot is changed, a different representation of the *same* protein-ligand complex is obtained. The authors have worked around this limitation by introducing different systematic rotations of the same input during training. However, these might present additional challenges when testing novel complexes that can take different orientations. This limitation has led us to explore methods that are inherently rotationally invariant. One such model that is not dependent on the coordinate frame is *Ligity* developed by Ebejer *et al.* (2019). *Ligity* is a hybrid VS technique that collects key interaction features within the protein-ligand complexes. These key interaction points are known as *hot-spots* and are defined by considering specific

pharmacophoric features that lie within a predetermined distance threshold between the protein and ligand feature pairs. Each of these pharmacophoric features that *interact* together are termed *Pharmacophoric Interaction Points* or PIPs. Once these pairs are extracted, the *Ligity* descriptor for the ligand is created by considering only the PIPs from the ligand space. Three or four PIP combinations are considered in the original *Ligity* method. The *Ligity* descriptor is built using the spatial distribution of PIPs (*i.e.* the distance between PIPs), and is, therefore, rotationally invariant.

Pafnucy has motivated us to use the CNN models for automatic feature extraction, and to find an alternative representation for the protein-ligand complex based on its structural and interacting properties. On the other hand, *Ligity* was used as the basis of our study and has inspired us to build a feature representation using *both* the protein and ligand PIPs, as opposed to *Ligity* that uses only ligand PIPs. In this study we present *LigityScore* — *LigityScore* is a novel rotationally invariant CNN based scoring function that utilises the interaction of pharmacophoric features of the protein and ligand for its data representation. In our approach we have therefore hypothesised that these pharmacophoric interactions across different feature types contain key information to suitably represent the protein ligand structure and their binding properties. We have further hypothesised that this representation would be suitable to train a CNN model for binding affinity prediction. Our approach introduces two techniques, *LigityScore1D* and *LigityScore3D*, that make use of important structural features of both the protein and ligand to create a suitable data representation of the protein-ligand complex. *LigityScore* uses distance between PIPs, which

remain the same irrespective of the structure’s orientation hence making the representation rotationally invariant. The PIP pair interactions from the protein and ligand are illustrated in Figure 1. Other methods such as OnionNet have inspired us to consider Pharmacophoric Interaction Points (PIPs) that are further apart, and to use InstanceNorm and ReLU to enhance our CNN models.

The LigityScore method considers six pharmacophore features: hydrophobic, hydrogen bond acceptor, hydrogen bond donor, aromatic, cation, and anion. Gund (1977) describes a *pharmacophore* as “a set of structural features in a molecule that are recognised at the binding site and is responsible for that molecule’s biological activity”. Therefore a pharmacophore model represents a number of general structural features such as aromatic or hydrophobic regions within the molecule, which are used to identify the features at the binding site that are responsible for molecular binding and biological activity. These features at the binding site may be used to find strong molecular binding interactions or *hot-spots* in the protein-ligand complex which are used to extract descriptors that represent the protein-ligand attributes. These features can be used in a 3D pharmacophore model and the spatial relationship between these pharmacophoric features can also be used to represent the protein-ligand complex (Leach et al., 2010).

The novel protein-ligand representation for use in a CNN based scoring function for binding affinity prediction is our major contribution in the SBVS domain. The source code for LigityScore is available at <https://gitlab.com/josephazzopardi/ligityscore>.

## 2 MATERIALS AND METHODS

LigityScore is a CNN based scoring function that utilises a rotationally invariant data representation extracted from interacting pharmacophoric features in protein-ligand complexes. An overview of LigityScore is illustrated in Figure 2, highlighting the parameters that can be changed for each module. The major functional parts for both LigityScore1D and LigityScore3D are detailed below:

1. **Pre-processing.** PDBbind (Liu et al., 2017a) files are processed to build a dataset of the complexes with their respective binding affinity values. At this stage the molecular files are validated to check that the protein-ligand complexes listed have corresponding molecular files, and also to fix any errors that occur whilst loading the files using the RDKit cheminformatics library (Landrum, 2020).
2. **PIP (Hot-spots) Generation.** This module loads the complexes and searches for the pharmacophoric features using the RDKit *BuildFeatureFactory* class. All the possible pairs of pharmacophoric features across the protein pocket and the ligand are built, and are then run against a number of constraints (see Table 1). The resultant feature pairs represent the PIPs or interaction hot-spots for a particular complex.
3. **Generation of LigityScore Descriptors.** The LigityScore descriptors module utilises the hot-spots dataset that includes both the ligand and protein PIPs to generate a feature descriptor for each complex. LigityScore1D considers two hot-spots at a time that correspond to a particular pharmacophoric feature family pair (e.g. HBA-HBA). For each possible family pair a feature vector is constructed, hence the name LigityScore1D. On the other hand, LigityScore3D uses three PIPs at a time, and the spatial information for the family set (e.g. HBA-HBA-HBA) is encoded in a feature cube. A *family set* is a combination of three PIP types. The names of our models are derived from the dimensionality of the spatial information used to generate the features.
4. **CNN Training.** This module is built using the Pytorch library (Paszke et al., 2019) and includes a dynamic model to construct a CNN in order to facilitate the testing and evaluation of different CNN architectures. The CNN module is used to train the network as a scoring function using the LigityScore descriptors. The module tackles a regression type of problem and therefore the output is a continuous value predicting the binding affinity of the protein-ligand complex. This output is compared with the experimental binding affinity so that the network parameters are updated using stochastic gradient descent. Each epoch is validated against the validation set, composed of 1,000 randomly sampled complexes from the PDBbind Refined set, and the model with the lowest root means squared error (RMSE) is stored to disk for use for predicting unseen complexes.
5. **CNN Predictions.** The Predictions module is used to load the best performing model and to compute results for the Training, Validation, and Test Sets.
6. **Experiments Pipeline.** This module combines the Training, Validation and Testing in one pipeline. This step uses a CSV file to describe a series of experiments with different CNN parameters.

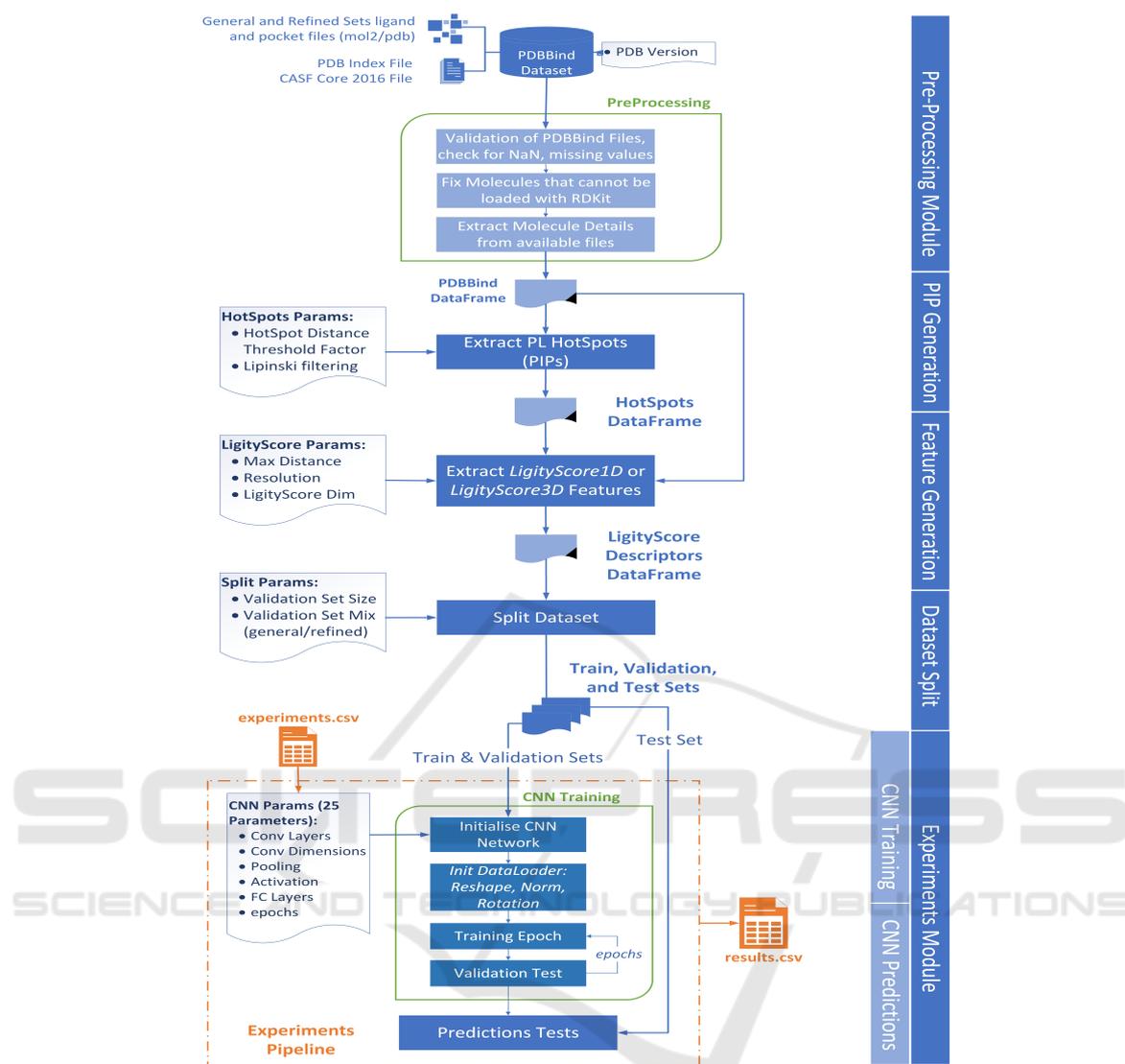


Figure 2: LigityScore schematic representation of the major functional components used in our approach to develop a CNN based scoring function for virtual screening. The parameters used in each functional block are included as reference. For example PIP Hot-spots extraction can take two parameters – Lipinski filtering, and the hot-spots distance threshold factor.

## 2.1 Evaluation Dataset

The PDBbind dataset (Liu et al., 2017a) is regarded as a golden dataset for the development of scoring functions (Liu et al., 2017b), and was therefore used for training and testing of LigityScore. The PDBbind v2018 was also used as an additional experiment for data augmentation since it has around 2,700 additional protein-ligand complexes. The PDB-Bind dataset is manually curated and includes records of experimentally measured binding affinity data for biomolecular complexes taken from the Protein Data Bank (PDB) (Berman et al., 2003). Their binding affinity is expressed in terms of dissociation ( $K_d$ ), in-

hibition ( $K_i$ ) or half-concentration ( $IC_{50}$ ) constants. No distinction is made between these constants and they were converted into a negative log;  $pK_a = -\log_{10}K_x$ , where  $K_x$  can be  $K_i$ ,  $K_d$  or  $IC_{50}$ , and  $pK_a$  is the binding affinity (Stepniewska-Dziubinska et al., 2017; Zheng et al., 2019). The PDBbind dataset is split into the *General* and *Refined* sets.

The *Core* set v2013 and v2016 were established as part of the CASF-2013 and CASF-2016 benchmarks. These benchmarks are meant to provide an objective platform to assess scoring functions, using high-quality protein-ligand complexes selected from the refined set, through a systematic and non-redundant sampling procedure (Su et al., 2018). The CASF

benchmarks were used to assess the *Scoring Power* of LigityScore1D and LigityScore3D. The scoring power is quantitatively measured for evaluation using the Pearson’s correlation coefficient,  $R$ , and the standard deviation in linear regression (SD). The Scoring Power measures the ability of the model to map a linear correlation of the predicted and known experimental affinity values. This study is focused to predict the binding affinity and the scoring power CASF benchmarks will be used for objective assessment and evaluation of the proposed scoring function.

A validation set, composed of 1,000 randomly chosen complexes from the refined set, was selected to evaluate the training progress after each epoch and select the CNN model with the smallest error. The validation set was also used for Early Stopping functionality in the training module so that training is stopped after a number of epochs with no loss improvements. The complexes in the validation set are chosen entirely from the refined set, as these provide higher quality protein-ligand complexes and are more reliable for the development of scoring functions (Liu et al., 2017b). Each of the core sets (2013 and 2016) were used entirely as the two test sets to simulate new and unseen protein-ligand complexes during the prediction stage. None of these test structures were used during training and validation. The remainder of the protein-ligand complexes were used as the training set. In our study, training was not performed on individual protein families but a generic CNN model was developed for all protein families which is a common approach in ML-based scoring functions (Stepniewska-Dziubinska et al., 2017; Jiménez et al., 2018; Ragoza et al., 2017; Zheng et al., 2019; Öztürk et al., 2018).

## 2.2 LigityScore Implementation

The LigityScore scoring function consists of the feature generation process to extract a representation of the protein ligand complex in LigityScore space, and the CNN model for automatic feature extraction and representation for binding affinity predictions. The two components are described next.

### 2.2.1 Feature Descriptors

The protein-ligand feature representation is split into two phases. In the first phase the PIPs of each individual protein-ligand complex are extracted to create the *PIP dataset* using all the PDBBind complexes. The algorithm used for PIP generation is based on the Ligity methodology described in Ebejer et al. (2019). The PIPs are extracted from the query protein-ligand complex using the open-source

cheminformatics package *RDKit BuildFeatureFactory* class (Landrum, 2020). The *BuildFeatureFactory* uses SMARTS patterns to identify these pharmacophoric features within the molecule. These PIPs, from both the protein and the ligand, are then filtered by a set of rules constraining feature family-pairs at a specific distance threshold so as to capture only the stronger interactions between the protein’s and ligand’s pharmacophoric features. The allowed feature family pairs and their corresponding distance threshold are listed in Table 1. The euclidean distances between the features are calculated using the centre of the atoms making up the feature. For example in a six-membered aromatic ring PIP, only the centre of the atomic structure is considered. In order to extract all conformant PIPs from the protein-ligand complex a cartesian product of all PIPs from the protein and ligand is performed, followed by the filtering of the allowed family pairs, and further filtering by the maximum distances allowed. PIP interactions are illustrated in Figure 1 showing the calculated distances between centres of PIPs.

In our approach we have also considered using longer distance thresholds than those stated in Table 1 which varies from the approach of Ebejer et al. (2019), and was inspired by the non-local electrostatic interactions used in Zheng et al. (2019). The PIP generation module provides a distance *threshold-factor* argument that can be used to multiply this baseline distance. A number of experiments were carried out using a varying distance threshold-factor between 1.0 and 1.6 in order to capture additional PIP interactions in our feature representation. This additional information includes also other weaker interactions, since the protein and ligand features are further apart, which can lead to a more information-rich representation of the protein-ligand complex.

Table 1: Pharmacophoric features and distance thresholds used to extract PIPs from the protein-ligand complex, reproduced from Ebejer et al. (2019).

Interacting Protein-Ligand PIP Family Pairs	Distance Threshold (Å)
hydrophobic, hydrophobic	4.5
hydrogen bond acceptor, donor	3.9
cation, anion	4.0
aromatic, aromatic	4.5
cation, aromatic	4.0

The second phase of the protein-ligand complex representation uses the *PIP dataset* described in the previous section to create a feature matrix, or a *feature cube collection* for every complex for LigityScore1D or LigityScore3D respectively. The *feature cube collection* generation process for LigityScore3D is highlighted in Figure 3. Each *feature cube collection* is

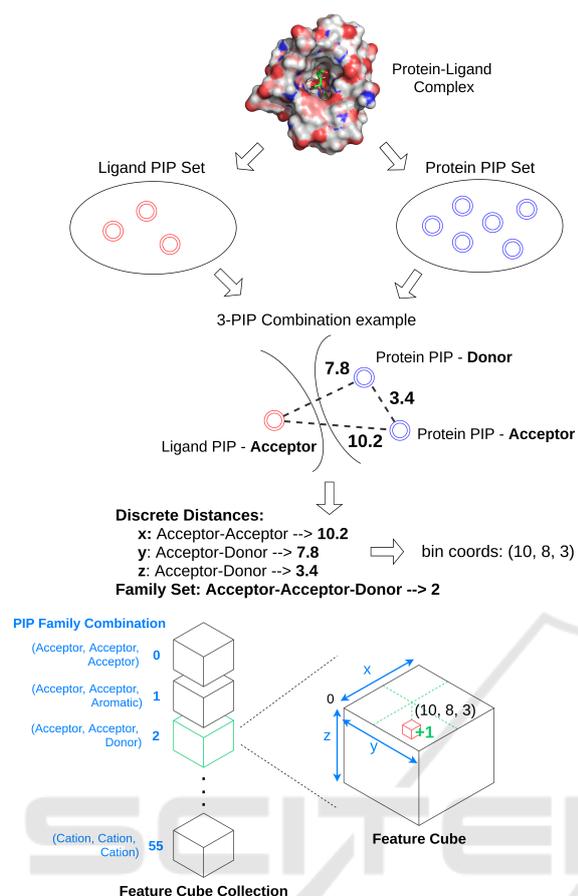


Figure 3: LigityScore3D *Feature Cube Collection* Generation. A protein-PIP set and a ligand-PIP set are extracted from a protein-ligand complex. From each 3-PIP combination taken from the PIP pool, the family-set, and their discrete distances are used to update the binning count in the Feature Cube. A Feature Cube is built for every available family-set.

calculated using the PIPs from the *PIP-dataset* related to the particular protein-ligand complex. The PIPs for the ligand side and those of the protein side are extracted to obtain two separate sets – the ligand PIP set, and the protein PIP set.

The *feature cube collection* is constructed by considering all the possible combinations when choosing one PIP from the ligand-PIP set and two PIPs from the protein-PIP set, and vice-versa. This 3-PIP combination creates a triangular structure amongst the PIPs as shown in Figure 3 and generates a set of three distances. The three distances are discretised using a 1Å resolution to extract a binning coordinate in 3D space. Additionally, each 3-PIP family combination represents a unique feature cube within the *feature cube collection*. Taking three, out of six pharmacophoric families considered with replacement creates a total of 56 possible 3-family set combinations.

The unique family set combination is used to index the particular feature cube to update the binning count using the coordinates from the three discretised distances. The voxel, or bin at this location is then incremented by one. Considering the example in Figure 3, one ligand PIP and two protein PIPs are considered. These generate a PIP-family combination of *HBA-HBA-HBD*, so that its cube will be updated at the (10, 8, 3) voxel location. The family combinations were sorted using both names and distance, to ensure the correct feature cube is updated.

Considering a maximum distance of 20Å in each dimension, each feature cube has a dimension of  $21 \times 21 \times 21$ . Since each 3-PIP family set has its own feature cube, 56 features cubes are stacked together to create a protein-ligand LigityScore3D representation of size  $1176 \times 21 \times 21$ . As indicated by Figure 3, the PIP distances are calculated by using combinations across both the ligand PIPs and the protein PIPs. In LigityScore3D, a combination of 3-PIPs is considered at a time. All the possible combinations using two PIPs for the protein, and one PIP from the ligand, plus the combinations where two ligand PIPs and one protein PIP are considered. This contrasts with the approach used in Ebejer *et al.* (2019) where only the ligand PIP pool was considered to take 3-PIP and 4-PIP combinations. Our hypothesis is that since the protein structure is essential for SBVS, considering also the protein PIPs in the feature generation process strengthens our model.

The method used for LigityScore1D is similar to LigityScore3D but considers a combinations of 2-PIPs at a time, and hence one inter-PIP distance. Each PIP family pair (example HBA-HBD) represents a different row in the feature matrix for the complex. Therefore the PIP-family pair is used to index the row of the feature matrix, whilst the discretised distance is used to index the column of the PIP. These coordinates in the feature matrix are then used to increment the bin count of that location. A total of 21 family combinations are possible, where each vector has 21 discrete locations corresponding to a feature matrix of  $21 \times 21$  per protein-ligand complex.

### 2.2.2 CNN Architecture

The architecture used for LigityScore is a deep convolutional neural network with a single regression output neuron used for prediction of binding affinity. The patterns extracted should differentiate the spatial information between different complexes captured from the PIP interactions. The CNN architecture used for LigityScore3D is illustrated in Figure 4. The input is normalised and size for LigityScore3D is transformed to  $98 \times 98 \times 54$ , whilst LigityScore1D is

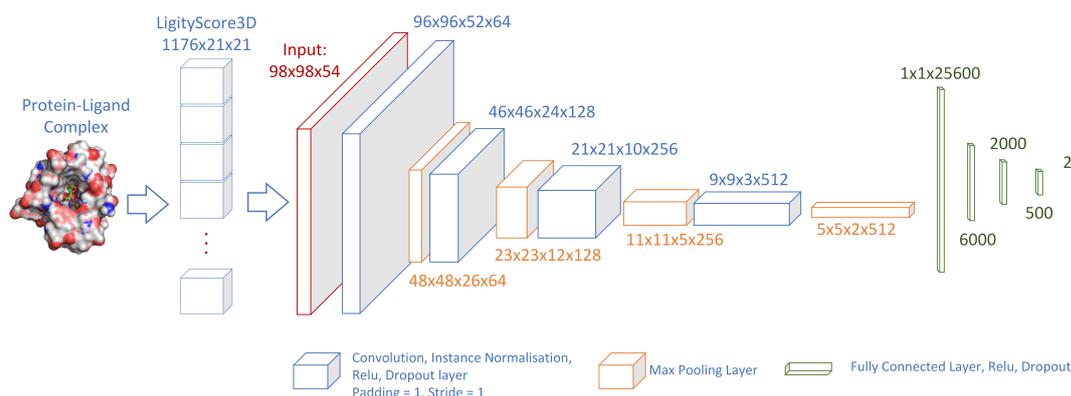


Figure 4: CNN Architecture for LigityScore3D. The input is reshaped to  $(98 \times 98 \times 54)$ . Four convolutional layers are used with instance normalisation, RELU activation, and spatial dropout. The output of the last convolution layers is flattened to feed a fully connected network with 4 hidden layers. The output is a single neuron predicting the binding affinity.

kept at  $21 \times 21$ . These inputs are treated as 2D and 3D tensors respectively, and our approach treats them similar to a greyscale and colour image respectively. This analogy allowed us to explore and use image processing techniques to optimise the scoring function model such as *InstanceNorm* (Ulyanov et al., 2017).

LigityScore3D inputs are processed using four convolutional layers with filter dimensions of 64, 128, 256, and 512 respectively, initialised using the Kaiming method (He et al., 2015). This initialising method is well suited for use with the RELU activation function as it keeps the standard deviation of the layer’s activations close to one. Correctly initialising the weights of the network is important for the training of deep neural networks as this prevents the output of the activation layers from exploding or vanishing. The PyTorch *Conv2D* module is used for each of the convolutional blocks. A convolutional kernel size of  $5 \times 5$  is used, with a padding and stride of one. Each convolution layer includes InstanceNorm, RELU activation, and spatial dropout (Tompson et al., 2015) components, which is then followed by a maxpooling layer with a patch of two to reduce the dimensions by half. LigityScore1D uses a similar CNN architecture but uses three convolution layers (64, 128, 256), and a padding of two.

The output of the last convolution layers is flattened to be used to feed four fully connected layers. LigityScore1D had a dimension of  $3 \times 3 \times 256$  at the input of the fully connected layers whilst LigityScore3D had an  $5 \times 5 \times 2 \times 512$  input. To cater for the difference in dimensionality the fully connected layers were assigned dimensions of (2000, 1000, 500, 200) and (6000, 1000, 200) respectively. Stochastic gradient descent with the Adam optimisation (Kingma and Ba, 2014) is used with default parameters for momentum scheduling ( $\beta_1 = 0.99$ ,  $\beta_2 = 0.999$ ) to train the network with a learning rate

of  $10^{-5}$  and L2 weight decay of  $\lambda = 0.001$ , using a mini-batch size of 20. Various experiments were carried out to tune hyperparameters.

### 3 RESULTS AND DISCUSSION

Several experiments were performed to find the best performing CNN architecture and LigityScore data representation. A considerable improvement in prediction performance was achieved when PIP distance threshold factors greater than one were applied to the values listed in Table 1. This implies that pharmacophoric *hot-spots* that are further apart are also considered during the PIP generation. The PIP threshold factor of 1.4 showed an improvement in the R-score of 19% over the baseline model. This may indicate that long-range interactions also play a role in protein-ligand binding.

LigityScore1D achieved best results when using InstanceNorm at the convolution layers, a PIP threshold factor of 1.4, and spatial dropout of 0.1 and obtained an R-score of 0.725 for CASF-2016 and 0.695 for CASF-2013 test sets. Spatial dropout was applied after the second convolution layers (middle layer with 128 channels) similar to the usage described in Tompson *et al.* (2015). The best results for LigityScore3D were achieved with spatial dropout probability of 0.2 on all the convolution layers and obtained a prediction performance on the Core-2016 R-score of 0.739, and a Core-2013 R-Score of 0.745. Spatial dropout improved the CNN as it made it more resilient to overfitting allowing the network to achieve higher predictions scores.

The mean and standard deviation of 10 tests of the best performing models were taken, to remove any bias that might be caused from testing using a

Table 2: Performance of LigityScore1D when trained with PDBbind v2016 and v2018, and LigityScore3D trained with PDBbind v2016, showing average and standard deviation for 10 tests using different validations sets taken from the refined set. LigityScore3D has the better overall performance for Core2013 and Core2016 test sets.

Set	RMSE ( $\pm$ std)	MAE ( $\pm$ std)	SD ( $\pm$ std)	R( $\pm$ std)
<i>LigityScore1D (v2016)</i>				
Training	0.406 (0.151)	0.323 (0.118)	0.393 (0.157)	0.974 (0.027)
Validation	1.438 (0.038)	1.144 (0.031)	1.432 (0.032)	0.698 (0.020)
Core2016	1.556 (0.039)	1.234 (0.031)	1.555 (0.038)	0.699 (0.018)
Core2013	1.861 (0.076)	1.485 (0.051)	1.701 (0.042)	0.657 (0.021)
<i>LigityScore1D (v2018)</i>				
Training	0.964 (0.295)	0.764 (0.237)	0.947 (0.287)	0.845 (0.076)
Validation	1.447 (0.037)	1.158 (0.033)	1.436 (0.029)	0.684 (0.017)
Core2016	1.516 (0.066)	1.223 (0.058)	1.461 (0.038)	0.741 (0.016)
Core2013	1.831 (0.072)	1.472 (0.064)	1.743 (0.054)	0.635 (0.028)
<i>LigityScore3D (v2016)</i>				
Training	0.621 (0.077)	0.490 (0.059)	0.531 (0.116)	0.957 (0.021)
Validation	1.479 (0.020)	1.182 (0.013)	1.435 (0.021)	0.692 (0.009)
Core2016	1.509 (0.034)	1.224 (0.031)	1.497 (0.034)	<b>0.725 (0.015)</b>
Core2013	1.676 (0.050)	1.335 (0.040)	1.583 (0.044)	<b>0.713 (0.019)</b>

 Table 3: LigityScore evaluation on the **CASF-2013** Scoring Power benchmark ranked using the Pearson Correlation Coefficient, R. Our results are in bold achieving 5<sup>th</sup> and 8<sup>th</sup> placings from the scoring functions listed in the benchmark, as well as other literature marked with (\*) where authors also used the CASF-2013 benchmark for evaluation. Entries without an (\*) are taken directly from Li *et al.* (2014a) – only the top 10 are included.

Scoring Function	Rank	SD	R
AGL* (Nguyen and Wei, 2019a)	1	1.45	0.792
LearningLigand* NNScore+RDkit (Boyles et al., 2020)	2	-	0.786
OnionNet* (Zheng et al., 2019)	3	1.45	0.782
EIC-Score* (Nguyen and Wei, 2019b)	4	-	0.774
PLEC-nn* (Wójcikowski et al., 2019)	4	1.43	0.774
<b>LigityScore3D</b>	<b>5</b>	<b>1.58</b>	<b>0.713</b>
Pafnucy* (Stepniewska-Dziubinska et al., 2017)	6	1.61	0.700
DeepBindRG* (Zhang et al., 2019)	7	-	0.639
<b>LigityScore1D</b>	<b>8</b>	<b>1.74</b>	<b>0.635</b>
X-Score	9	1.77	0.622
X-ScoreHS	10	1.77	0.620
X-ScoreHM	11	1.78	0.614
X-ScoreHP	12	1.79	0.607
dSAS	13	1.79	0.606
ChemScore@SYBYL	14	1.82	0.592
ChemPLP@GOLD	15	1.84	0.579
PLP1@DS	16	1.86	0.568
PLP2@DS	17	1.87	0.558
GScore@SYBYL	18	1.87	0.558

\* other literature using CASF-2013 benchmark

single holdout validation set. Table 2 summarises these average results for LigityScore1D trained using PDBbind v2016 and PDBbind v2018, and for LigityScore3D using PDBbind v2016. LigityScore3D shows a significant performance improvement for the Core-2013 model with an average of 0.713 R-score that is well above the 0.657 and 0.635 achieved for LigityScore1D trained on PDBBind v2016 and PDBBind v2018 respectively. On the other hand the results for Core-2016 for LigityScore3D shows comparable

 Table 4: LigityScore evaluation on the **CASF-2016** Scoring Power benchmark ranked using the Pearson Correlation Coefficient, R. Our results are in bold achieving 7<sup>th</sup> and 8<sup>th</sup> placings from the scoring functions listed in the benchmark, as well as other literature marked with (\*) where authors also use the CASF-2016 benchmark. Entries without an (\*) are taken directly from Su *et al.* (2018) – only the top 10 are included.

Scoring Function	Rank	SD	R
AGL* (Nguyen and Wei, 2019a)	1	-	0.830
EIC-Score* (Nguyen and Wei, 2019b)	2	-	0.826
LearningLigand NNScore+RDkit (Boyles et al., 2020)	2	-	0.826
K <sub>deep</sub> * (Jiménez et al., 2018)	3	-	0.820
PLEC-nn* (Wójcikowski et al., 2019)	4	1.26	0.817
OnionNet* (Zheng et al., 2019)	5	1.26	0.816
$\Delta$ VinaRF20	5	1.26	0.816
Pafnucy* (Stepniewska-Dziubinska et al., 2017)	6	1.37	0.780
<b>LigityScore1D</b>	<b>7</b>	<b>1.46</b>	<b>0.741</b>
<b>LigityScore3D</b>	<b>8</b>	<b>1.50</b>	<b>0.725</b>
X-Score	9	1.69	0.631
X-ScoreHS	10	1.69	0.629
$\Delta$ SAS	11	1.70	0.625
X-ScoreHP	12	1.70	0.621
ASP@GOLD	13	1.71	0.617
ChemPLP@GOLD	14	1.72	0.614
X-ScoreHM	15	1.73	0.609
Autodock Vina	16	1.73	0.604
DrugScore2018	17	1.74	0.602

\* other literature using CASF-2016 benchmark

performance to the LigityScore1D (PDBBind v2018) models with only 0.01 difference in R-score. Due to the similarity in results obtained for both CASF-2013 and CASF-2016, LigityScore3D is chosen as the best performing model with R-score of 0.725 and 0.713. The additional scoring power of approximately 10% for CASF-2013 comes at the expense of a more complex network. The LigityScore3D model has 94M learnable parameters, whilst the best model for LigityScore1D has only 3.9M parameters.

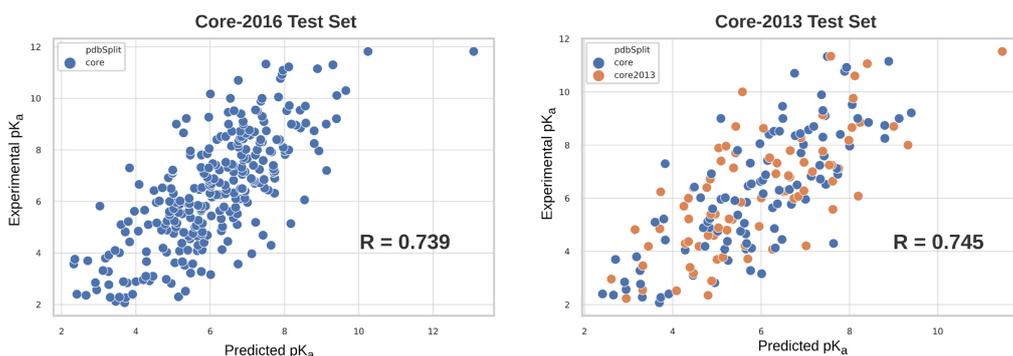


Figure 5: Experiment vs Predicted Binding Affinity in  $pK_a$  for best LigityScore3D model. The left scatter plot represents the core 2016 set, whilst the right scatter plot represents the core 2013 set.

The scatter plots for the predicted affinity versus the experimental affinity for best performing LigityScore3D model is shown in Figure 5. The scatter plots represent the Core-2016 (left) and the Core-2013 (right) sets showing good correlation between predicted and experimental affinities. The ideal model would produce a plot of function  $y = x$ , as the predicted affinity should be equal to the experimental value. The ranking of LigityScore for CASF-2013 and CASF-2016 are presented in Tables 3 and 4. Apart from the scoring function evaluated directly in CASF, Tables 3 and 4 include other scoring functions, marked with an asterisk (\*), that represent results reported in literature (in individual publications) that also utilise the CASF benchmarks. Tables 3 and 4 thus provide, to the best of our knowledge, a comprehensive list of the scoring functions developed in recent years to date, that compare and rank the different scoring functions available. LigityScore3D achieved 5<sup>th</sup> place in the CASF-2013 benchmark with an average R-score of 0.713, and exceeds the reported CASF-2013 score for Pafnucy. On the CASF-2016 benchmark, LigityScore models achieve the 7<sup>th</sup> and 8<sup>th</sup> places.

## 4 CONCLUSIONS

In this study we explored the use of CNNs to develop a scoring function, called LigityScore, for binding affinity prediction. Machine learning scoring functions have been developed to address the limitations of classical models, such as the use of linear models, imposed functional form, and their inability to learn from new data. However, conventional ML based scoring functions still rely on a degree of feature engineering that requires expert knowledge to preprocess the data. This, in turn, led to the introduction of deep learning methods. To this effect we have developed

two different protein-ligand representations that are extracted directly from the 3D structure of both the protein and ligand using pharmacophoric features.

The choice of representation of the protein-ligand structure determines the flexibility and expressiveness that the model is able to learn and ultimately its scoring power. Although deep learning methods extract features automatically during training, correct representation of the complex is critical for the feature extraction ability of the DL model. As a point for improvement for LigityScore performance, future work would focus on the data representation component of the protein-ligand complex to build on the existing representation and possibly seek ways to incorporate alternate types of features within LigityScore. In this regard one of the research tasks would be to look into additional pharmacophoric feature families (or types) that could help create an enriched descriptor. Other features such as the spatial distribution count for distances between key atom combinations could be considered as another dimension to the LigityScore representation. Additionally, since CNNs are difficult to interpret, in future work we would apply techniques such as SHAP (Lundberg and Lee, 2017) to determine critical features used for predictions.

The major contribution for this study is in the presentation of a novel protein-ligand representation for use as a CNN scoring function for binding affinity prediction adapted from Ebejer *et al.* (2019). Representation engineering is required when using CNN for SBVS as the data needs to represent the protein-ligand structure. Representation engineering is necessary since the protein-ligand complex cannot be input directly into the CNN as in the case of an image. In our approach we use spatial distances between key pharmacophoric features which is simpler than creating a mathematical model to describe the protein-ligand interactions. LigityScore still relies on the automatic feature extraction of CNNs for feature

extraction. Since LidityScore is based on distances between pharmacophoric features, it also presents a rotationally invariant representation. Additionally, the method shows relatively good performance that marginally exceed the Pafnucy R-score performance on the CASF-2013 benchmark by 0.01 on average, using a less computationally complex model that can be trained 16 times faster. The LidityScore models can potentially be used for affinity predictions for novel molecules, and as a scoring function for docking in virtual screening.

A recent paper by Shen *et al.* (2020) highlights the importance of assessing the scoring function in all four powers (scoring, ranking, docking, and screening) of the CASF benchmark for a 360 degree performance evaluation. Due to the recent release of Shen *et al.* (2020) it was not possible to extend evaluation of LidityScore on the rest of the powers. This is a limitation in the sense that these results are not known, and future work would consider testing LidityScore for the other powers in the CASF benchmark. Recent literature for deep learning scoring functions also focused on only the scoring power aspect such as Stepniewska-Dziubinska *et al.* (2017), Jiménez *et al.* (2018), and Zheng *et al.* (2019), and therefore a similar approach was taken. Shen *et al.* (2020) has shown that Pafnucy and OnionNet do not perform well on the rest of the benchmark powers, and even report performance lower than the classical functions.

Although the ideal scoring function should perform well in all CASF benchmark powers, we argue that this is not necessarily the case and a particular ML scoring function may not be suited for every scenario. Therefore a different version, trained for a particular power, may be better suited. As an example, the screening power would require the scoring function model to differentiate between actives and inactives. However, the models trained with the PDB-bind dataset do not include any inactive information. Due to the lack of experimentally-validated inactives there are no evaluation datasets that include inactive molecules highlighting the need for better and more complete datasets (Sieg *et al.*, 2019). ML models, including DL, use learning by representation to extract the underlying function in the data. If the dataset does not include the inactive class it is intuitive that the model may not respond well when presented with inactive molecules. A ML scoring function can be developed to cater for the particular power, leveraging on the flexibility they provide to adjust and derive their parameters from the given training data automatically.

Finding a suitable representation of the protein-ligand complex is a major challenge when building a

scoring function, and is key for accurate predictions using deep learning techniques. In our work we have successfully found a suitable representation that to the best of our knowledge was never used for binding affinity prediction, which provides good results and ranked 5<sup>th</sup> in the CASF-2013 benchmark. Therefore, although our work did not outperform the top scoring function we deem it is still a valid contribution to the area and may be further enhanced in future work, or may also serve as motivation and inspiration for other researchers to seek out alternative methods that increase the effectiveness of scoring functions and virtual screening in general.

We believe a deeper understanding of CNN in the domain of SBVS is still required, and a breakthrough like the work of Krizhevsky *et al.* (2012) in the computer vision domain is still being sought after in this challenging domain. Nonetheless, we also believe that ML and DL techniques will lead the future of the development of scoring functions.

## ACKNOWLEDGEMENTS

We would like to thank the AWS Research Credits Team for supporting our research with AWS credits to develop our models.

## REFERENCES

- Ain, Q. U., Aleksandrova, A., Roessler, F. D., and Ballester, P. J. (2015). Machine-learning scoring functions to improve structure-based binding affinity prediction and virtual screening. *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 5(6):405–424.
- Berman, H., Henrick, K., and Nakamura, H. (2003). Announcing the worldwide protein data bank. *Nature Structural & Molecular Biology*, 10(12):980–980.
- Boyles, F., Deane, C. M., and Morris, G. M. (2020). Learning from the ligand: using ligand-based features to improve binding affinity prediction. *Bioinformatics*, 36(3):758–764.
- Ching, T., Himmelstein, D. S., Beaulieu-Jones, B. K., Kalinin, A. A., Do, B. T., Way, G. P., Ferrero, E., Agapow, P.-M., Zietz, M., Hoffman, M. M., *et al.* (2018). Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society Interface*, 15(141):20170387.
- Goldberg, Y. (2016). A primer on neural network models for natural language processing. *Journal of Artificial Intelligence Research*, 57:345–420.
- He, K., Zhang, X., Ren, S., and Sun, J. (2015). Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *2015 IEEE In-*

- ternational Conference on Computer Vision (ICCV)*, pages 1026–1034.
- Jiménez, J., Skalic, M., Martínez-Rosell, G., and De Fabritiis, G. (2018). K deep: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. *Journal of chemical information and modeling*, 58(2):287–296.
- Kaggle, M. (2012). Kaggle: Merck molecular activity challenge. <https://www.kaggle.com/c/MerckActivity>, Accessed Feb 8, 2019.
- Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. [arxiv:1412.6980](https://arxiv.org/abs/1412.6980) Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.
- Landrum, G. (2020). Rdkit: Open-source cheminformatics. Accessed April, 2020.
- Leach, A. R., Gillet, V. J., Lewis, R. A., and Taylor, R. (2010). Three-dimensional pharmacophore methods in drug discovery. *Journal of medicinal chemistry*, 53(2):539–558.
- LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436–444.
- Li, Y., Liu, Z., Li, J., Han, L., Liu, J., Zhao, Z., and Wang, R. (2014). Comparative assessment of scoring functions on an updated benchmark: 1. compilation of the test set. *Journal of chemical information and modeling*, 54(6):1700–1716.
- Liu, Z., Cui, Y., Xiong, Z., Nasiri, A., Zhang, A., and Hu, J. (2019). Deepseqpan, a novel deep convolutional neural network model for pan-specific class i hla-peptide binding affinity prediction. *Scientific reports*, 9(1):794.
- Liu, Z., Su, M., Han, L., Liu, J., Yang, Q., Li, Y., and Wang, R. (2017a). Forging the basis for developing protein–ligand interaction scoring functions. *Accounts of chemical research*, 50(2):302–309.
- Liu, Z., Su, M., Han, L., Liu, J., Yang, Q., Li, Y., and Wang, R. (2017b). Forging the basis for developing protein–ligand interaction scoring functions. *Accounts of chemical research*, 50(2):302–309.
- Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.
- Nguyen, D. D. and Wei, G.-W. (2019a). Agl-score: Algebraic graph learning score for protein–ligand binding scoring, ranking, docking, and screening. *Journal of chemical information and modeling*, 59(7):3291–3304.
- Nguyen, D. D. and Wei, G.-W. (2019b). Dg-gl: Differential geometry-based geometric learning of molecular datasets. *International journal for numerical methods in biomedical engineering*, 35(3):e3179.
- Öztürk, H., Özgür, A., and Ozkirimli, E. (2018). Deepdta: deep drug–target binding affinity prediction. *Bioinformatics*, 34(17):i821–i829.
- Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.
- Pérez-Sianes, J., Pérez-Sánchez, H., and Díaz, F. (2019). Virtual screening meets deep learning. *Current computer-aided drug design*, 15(1):6–28.
- Ragoza, M., Hochuli, J., Idrobo, E., Sunseri, J., and Koes, D. R. (2017). Protein–ligand scoring with convolutional neural networks. *Journal of chemical information and modeling*, 57(4):942–957.
- Rifaioğlu, A. S., Atas, H., Martin, M. J., Cetin-Atalay, R., Atalay, V., and Dogan, T. (2018). Recent applications of deep learning and machine intelligence on in silico drug discovery: methods, tools and databases. *Brief Bioinform*, 10.
- Sieg, J., Flachsenberg, F., and Rarey, M. (2019). In need of bias control: Evaluating chemical data for machine learning in structure-based virtual screening. *Journal of chemical information and modeling*, 59(3):947–961.
- Stepniewska-Dziubinska, M. M., Zielenkiewicz, P., and Siedlecki, P. (2017). Pafnucy—a deep neural network for structure-based drug discovery. *stat*, 1050:19.
- Su, M., Yang, Q., Du, Y., Feng, G., Liu, Z., Li, Y., and Wang, R. (2018). Comparative assessment of scoring functions: the casf-2016 update. *Journal of chemical information and modeling*, 59(2):895–913.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1–9.
- Tompson, J., Goroshin, R., Jain, A., LeCun, Y., and Bregler, C. (2015). Efficient object localization using convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 648–656.
- Ulyanov, D., Vedaldi, A., and Lempitsky, V. (2017). Improved texture networks: Maximizing quality and diversity in feed-forward stylization and texture synthesis. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.
- Wójcikowski, M., Kukielka, M., Stepniewska-Dziubinska, M. M., and Siedlecki, P. (2019). Development of a protein–ligand extended connectivity (plec) fingerprint and its application for binding affinity predictions. *Bioinformatics*, 35(8):1334–1341.
- Zhang, H., Liao, L., Saravanan, K. M., Yin, P., and Wei, Y. (2019). Deepbindrg: a deep learning based method for estimating effective protein–ligand affinity. *PeerJ*, 7:e7362.
- Zheng, L., Fan, J., and Mu, Y. (2019). Onionnet: a multiple-layer intermolecular-contact-based convolutional neural network for protein–ligand binding affinity prediction. *ACS omega*, 4(14):15956–15965.