Drug-Target Interaction Prediction: End-to-End Deep Learning Approach

Nelson R. C. Monteiro, Bernardete Ribeiro, Joel P. Arrais

Abstract—The discovery of potential Drug-Target Interactions (DTIs) is a determining step in the drug discovery and repositioning process, as the effectiveness of the currently available antibiotic treatment is declining. Although putting efforts on the traditional *in vivo* or *in vitro* methods, pharmaceutical financial investment has been reduced over the years. Therefore, establishing effective computational methods is decisive to find new leads in a reasonable amount of time. Successful approaches have been presented to solve this problem but seldom protein sequences and structured data are used together. In this paper, we present a deep learning architecture model, which exploits the particular ability of Convolutional Neural Networks (CNNs) to obtain 1D representations from protein sequences (amino acid sequence) and compounds SMILES (Simplified Molecular Input Line Entry System) strings. These representations can be interpreted as features that express local dependencies or patterns that can then be used in a Fully Connected Neural Network (FCNN), acting as a binary classifier. The results achieved demonstrate that using CNNs to obtain representations of the data, instead of the traditional descriptors, lead to improved performance. The proposed end-to-end deep learning method outperformed traditional machine learning approaches in the correct classification of both positive and negative interactions.

Index Terms—Drug Repositioning, Drug-Target Interaction, Deep Learning, Convolutional Neural Network, Fully Connected Neural Network, Protein Sequence, SMILES, Drug

1 Introduction

Multidrug-resistant bacterias are a rising health concern to the overall population and pharmaceutical industry as more and more drugs are becoming ineffective and unresponsive to the symptoms and diseases associated with these kinds of infections [1]. Although modern medicine is aligned with antibiotic treatment, the discovery of new and potential drugs is declining, as there is an increase of the misuse of the current available medicine, causing a resistance effect to these kinds of agents. Additionally, a reduced financial investment makes it difficult for researchers to keep up with the current population and pharmaceutical needs [2].

Traditional *de novo* drug discovery is very time-consuming, as it may take 10 to 17 years from concept to marketed drug [3], expensive, in the realm of thousands of millions, and it is associated with a low probability of success, as there is a considerable number of conditions to be met in order to be viable for human consumption. Therefore, new approaches are needed in order to reach a better time-reward trade-off. Aligning drug repositioning [4], that is, finding new clinical purposes for existing drugs, with computational methods is decisive to find potential drug-target interactions in a reasonable amount of time. As such, establishing effective computational methods is crucial to find new leads (hit compounds), which are identified as potential drugs for therapeutic use.

Computational methods for DTI prediction are divided into 3 main approaches [5]: ligand based, docking simulation and chemogenomic.

1.1 Ligand Based

Ligand based approaches are built upon the concept that similar molecules have similar properties and therefore should bind to the same group of proteins. Keiser et al. (2008) [6] developed a method, Similarity Ensemble Approach (SEA), where receptors (proteins) were quantitatively related based on the chemical similarity among their ligands. Humberto et al. (2011) [7] proposed a Multi-target QSAR (Quantitative Structure Activity Relationships) Web Server to make large scale predictions derived from chemical structures and 3D structures of target proteins. Cheng et al. (2012) [8] established multi-target quantitative structure–activity relationships (mt-QSAR) and chemogenomics methods based on substructure patterns and protein sequences descriptors to predict chemical–protein interactions.

1.2 Docking Simulation

Docking Simulation approaches are used for structure based drug design [9], where the interaction between a protein and a drug is simulated and scored, according to the intermolecular interaction energy, using 3D structures. Li et al. (2006) [10] developed an useful tool for target identification, TarFisDock, where a reverse ligand-protein docking is used to identify potential protein targets for a small molecule. Yang

- *Nelson R. C. Monteiro is with Department of Informatics Engineering (DEI), Center for Informatics and Systems of the University of Coimbra (CISUC), University of Coimbra, Coimbra, Portugal*
  E-mail: nelsonrcmonteiro@gmail.com
- *Bernardete Ribeiro is with Department of Informatics Engineering (DEI), Center for Informatics and Systems of the University of Coimbra (CISUC), University of Coimbra, Coimbra, Portugal*
  E-mail: bribeiro@dei.uc.pt
- *Joel P. Arrais is with Department of Informatics Engineering (DEI), Center for Informatics and Systems of the University of Coimbra (CISUC), University of Coimbra, Coimbra, Portugal*
  E-mail: jpa@dei.uc.pt

Manuscript received ; revised

et al. (2011) [11] established a docking method, Chemical-Protein Interactome, to mimic the differences in the drug-protein interactions across a set of human proteins. The proposed work gives information about the binding conformation and the binding strength. Additionally, it was able to identify an important biomarker, *HSPA1A*, an off-target of clozapine. Cheng et al. (2007) [12] designed a binding free energy model combined with drug like properties to predict the maximal affinity by a drug-like molecule (drugability) using the crystal structure of the target binding site.

Although the use of 3D structures is a realistic approach to model the interaction between proteins and drugs, the lack of information, the complexity of 3D structures and the amount of time it takes to simulate, makes this kind of approaches inapplicable and inefficient in most cases.

1.3 Chemogenomic

The growth of available biological and chemical data useful for prediction resulted in a higher usage of chemogenomic methods over the traditional methods. Chemogenomic approaches are based on the chemical space of compounds, genomic space of target proteins and/or the pharmacological space (interactions between proteins and drugs) to predict new potential interactions. Yamanishi et al. (2008) [13] proposed a supervised method to infer DTIs by integrating the chemical space and genomic space into a unified space defined as the pharmacological space. The proposed work uses a bipartite graph learning method to learn the correlation (similarity) between chemical/genomic space and the interaction space to infer new possible interactions (high scoring compound-protein pairs). Although the method was not validated experimentally, the major four datasets used in this work are still the base of many DTI studies. Cheng et al. (2012) [14] proposed a network based inference (NBI) approach using FDA (United States Food and Drug Administration) approved drug-target binary links to infer new predictions. This method only uses known drug-target bipartite network topology similarity to calculate predictive scores for each drug and unlinked target. Unlike Yamanishi et al. (2008), some of the predictions were validated experimentally by *in vitro* assays.

1.3.1 Machine Learning

Due to the considerable amount of available data, machine learning approaches are pursued as a result of their ability to learn relationships and patterns among the data related to proteins and drugs. Cao et al. (2014) [15] combined chemical data, MACCS (Molecular Access System) fingerprints and/or substructure fingerprints, biological data, protein descriptors, and network properties, presence or absence of association, into feature vectors to be used in a predictive random forest (RF) model, to identify new DTIs. Yu et al. (2012) [16] proposed a machine learning method, using random forest and support vector machine (SVM) as the predictive models, to infer new interactions. In that work, chemical and protein descriptors were combined to create the feature vectors. Nagamine et al. (2007) [17] used support vector machine as the predictive model to infer new interactions. Instead of using the conventional chemical and genomic descriptors, protein sequences, chemical structures and mass spectrometry, which generates information about the structure and physicochemical properties, were encoded into numerical values, based on the existence or frequency, and concatenated into features vectors. Cobanoglu et al. (2013) [18] presented a method using probabilistic matrix factorization (PMF) combined with active learning, without reliance on chemical/target similarity. This approach decomposes the connectivity matrix, related to the DTI network, as a product of 2 matrices that express each drug/target, which objective is to determine the missing interactions that are likely to exist. Yamanishi et al. (2009) [19] proposed a supervised prediction method using bipartite local models, one based on chemical structure similarity and another one based on sequence similarity between proteins. The prediction is done using two support vector machines to predict target proteins and drugs for a given drug or protein, respectively. The results are combined to give a definitive prediction for each interaction.

1.3.2 Deep Learning

Even though traditional machine learning approaches usually result in good performance, with the increased computational power and amount of available data, deep learning approaches are being used more often, resulting in even higher performance in most cases, due to the fact that they are able to identify hidden and complex patterns (representations) of the data without using any feature engineering. Tian et al. (2016) [20] proposed a deep neural network approach, based on a feedforward architecture, DL-CPI (Deep Learning for Compound-Protein Interactions prediction), to predict compound-protein interactions, where chemical fingerprints and protein domains were used as features. Peng Wei et al. (2016) [21] developed an approach known as multi-scale features deep representations inferring interactions (MFDR), where a certain type of deep neural network architecture, autoenconder, is used to extract low-dimensional representations from chemical structure and protein sequence descriptors to then be used as features in a support vector machine. Wen et al. (2017) [22] proposed a deep learning method, DeepDTIs, based on deep belief networks (DBN). This type of neural network architecture is made by stacking restricted Boltzmann machine (RBM), which is a graphical model that can learn a probability distribution from input data. The features used were extracted from chemical substructures and sequence order information (descriptors). Xie et al.(2017) [23] used transcriptome data, z-score of genome wide gene expressions, in a deep feedforward neural network to predict new drug-target interactions.

In this work we propose a deep learning approach to predict the interaction between proteins and drugs using 1D raw data, protein amino acid sequences and SMILES strings to represent the drug's chemical structure. A pipeline with two parallel Convolutional Neural Networks is used to uncover deep patterns (representations or local dependencies) from raw data instead of the conventional physicochemical and/or structural descriptors, as they are general descriptors of the whole sequence or chemical structure and therefore being non relevant, in most cases, to a possible real interaction, or 3D structures, as the amount of available

known structures is limited or highly complex. Although convolutional neural networks are known as the state-of-art for image classification (LeCun et al. (2010) [24], Krizhevsky et al. (2012) [25], Kang et al. (2014) [26], Üreten et al. (2019) [27]), some of the most recent studies applied these specific type of deep neural architectures to learn deep hidden patterns from sequences or strings (Zeng et al. (2016) [28], Ozturk et al. (2018) [29], Budach et al. (2018) [30], Kwon et al. (2018) [31]). Furthermore, convolutional neural networks have also been explored to extract useful information from graph representations of the data [32], including 2D graph representations of the molecules [33]. We use Coelho et al. (2016) [34] DTI (Drug-Target Interaction) dataset to evaluate the performance of our model and validate the whole pipeline. Additionally, we compared our model with four different approaches, specifically random forest, a fully connected neural network architecture, support vector machine and also a CNN, autoencoder and FCNN combined model. The designed setup resulted in better performance overall.

2 METHODS

2.1 Dataset

2.1.1 Drug-Target Interaction Pairs

Coelho et al. (2016) [34] DTI dataset was used as benchmark. Positive interaction dataset was obtained from DrugBank [35] and Yamanishi et al. (2008) [13], where all entries related to specific classes of protein targets and proteins with unreviewed status were removed. On the other hand, the negative interaction dataset was collected from BioLiP [36] and BindingDB [37], where a bioactivity threshold of 10 μM was used to identify weak binding interactions. Table 1 summarizes the amount of unique drugs, targets and drug-target interactions extracted from these databases.

TABLE 1: Unique drugs, targets and DTIs.

	Positive		Negative	
	DrugBank	**Yamanishi et al. (2008)**	**BioLip**	**BindingDB**
Drugs	1328	790	894	12454
Targets	706	1371	636	404
DTI	3530	7206	1223	14985

A ratio of 1.5 negative to positive was adopted, resulting in 7206 positive and 10,912 negative DTI pairs for training and 3,530 positive and 5,297 negative DTI pairs for testing (Table 2). Plus, only Yamanishi et al. (2008) [13] and DrugBank [35] positive entries were used for training and testing, respectively.

TABLE 2: Training and Testing Datasets.

	Positive	**Negative**	**Total**
Training	7206	10912	18118
Testing	3530	5297	8827

The reference work [34] ensured the discriminating power by evaluating the sequence similarity within each dataset and across all datasets and guaranteeing that less than 1 % of all possible drug pairs had a sequence similarity score greater than 0.85, excluding any possibility of redundancy between the two datatsets, training and testing.

2.1.2 Protein Data

The protein sequences were all extracted from UniProt [38] using their identifiers. Proteins are constituted by an unique amino acid sequence, hence different proteins have different sequence's lengths. Since we are using protein sequences directly and not global descriptors, each amino acid that constituted the sequence is considered as a feature. Thus, it was necessary to define a threshold based on their length, in order to guarantee that each protein is characterized by the same amount and type (order) of features.

Figures 1a and 1b show the protein sequence length distribution for the training and testing set, respectively. An information threshold of 95 % was used, resulting in a maximum length of 1205 for the protein sequences. Every protein sequence with a length superior or inferior to the threshold was removed or padded, respectively.

2.1.3 Chemical Data

The SMILES strings were collected from PubChem [39] exclusively, in their canonical format, to guarantee a consistent notation to represent the chemical structure. Each character of the SMILES string is considered as a feature, therefore if different notations were to be used to represent the chemical structures, equal segments of the compounds would be seen as different components by the model.

The dataset contains IDs from Protein Data Bank (PDB) [40], KEGG [41], ZINC [42] [43] and DrugBank [35], thus it was necessary to convert them to PubChem [39] compound IDs first in order to extract the SMILES strings. The Python packages, PyPDB [44], BioServices [45] and PubChemPy [46], were used for conversion and extraction.

Identical to the protein sequences, a threshold based on their length was applied, resulting in a maximum length of 90. Figures 1c and 1d show the SMILES string length distribution for the training and testing set, respectively.

All entries duplicated or containing missing characters in one of the datasets were also removed. Table 3 summarizes the result of elimination and Table 4 the amount of unique targets, drugs and number of targets for the training and testing datasets, respectively, after elimination.

TABLE 3: Training and Testing Datasets after elimination.

	Positive	**Negative**	**Total**
Training	5839	10172	16011
Testing	3012	4914	7926

TABLE 4: Unique targets, drugs and number of targets for the training and testing datasets.

	Unique		**Number of Targets**	
	Targets	**Drugs**	**1**	**>1**
Training	1790	9583	8026	1557
Testing	1068	5718	4884	834

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2020.2977335, IEEE/ACM Transactions on Computational Biology and Bioinformatics

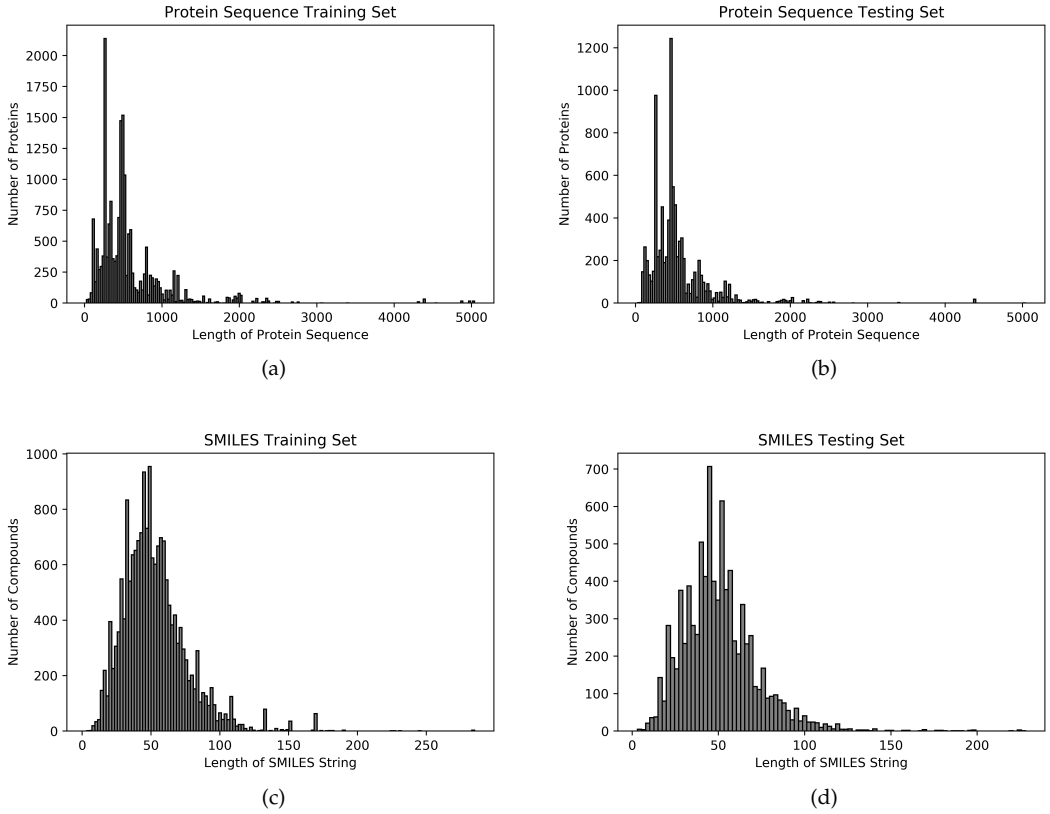

Fig. 1: Distribution of the lengths of training and testing datasets. (a) Training protein sequences. (b) Testing protein sequences. (c) Training SMILES. (d) Testing SMILES.

2.2 Data Representation

2.2.1 Protein Sequence Encoding

We used Yu et al. (2010) [47] protein substitution table (Table 5), which organizes amino acids into 7 groups according to their physicochemical properties. Each amino acid was encoded into an integer based on the corresponding group. This kind of representation allows to directly use protein sequences, preserve the sequential information and also reduce the amount of categories from 20, associated with the number of possible amino acids, to 7.

TABLE 5: Yu et al. (2010) [47] Protein Substitution.

Groups	Amino Acids
1	Ala, Gly, Val
2	Ile, Leu, Phe, Pro
3	Tyr, Met, Thr, Ser
4	His, Asn, Gln, Trp
5	Arg, Lys
6	Asp, Glu
7	Cys

2.2.2 SMILES String Encoding

A simple integer encoding, based on the number of different characters, was used to transform each character of the SMILES string into a integer. A dictionary containing 32 categories (number of different characters) was established (Table 6). This representations preserves the structural information and has a low computational cost given the amount of different characters.

TABLE 6: SMILES Char-Integer Dictionary.

Integer	Character
1	I
...	...
7	[
...	...
25	P
...	...
32	g

2.3 Model Overview

The proposed approach is based on the combination of two deep neural network architectures, Convolutional Neural Network and Fully Connected Neural Network, constituting a deep learning model to predict the interaction, positive or negative, between targets (proteins) and compounds (drugs), directly using 1D raw data, protein amino acid sequences and SMILES strings.

Protein sequences and SMILES strings are initially processed based on the length, as mentioned in Section 2.1.2 and 2.1.3, and then encoded into integer values according to the encoding scheme, Section 2.2.1 and 2.2.2, respectively.

These integers values are still recognized as categorical variables, therefore an one-hot encoding layer was applied to normalize the importance of each categorical value, since higher categorical values would have more influence than the others in the training process, leading to possible errors and misclassifications by the model. One-Hot Layer assigns a binary variable for each unique integer value, converting every integer into a binary vector, which sets the corresponding integer to "1" and "0" to the rest. In particular, this is illustrated in Figure 2 with respect to myocyte-specific enhancer factor 2B (protein).

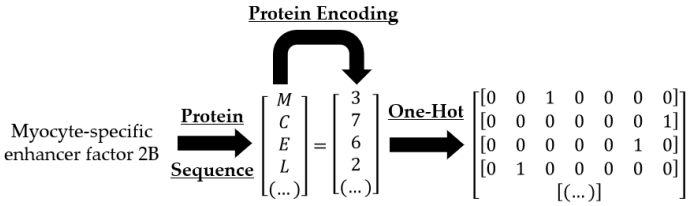

Fig. 2: One-Hot encoding applied to myocyte-specific enhancer factor 2B.

CNNs are known as motif detectors and feature extractors, capable of identifying deep patterns from the data by moving from low level features to abstract concepts. The convolutional layer is composed by filters, which are identified as arrays of weights that slide over the entire input. These filters work as feature identifiers and convolute at each particular location, originating activation maps, which are learnable feature maps composed by all the single convolution outputs and used as the input of the next layer. Convolution is a specialized kind of linear operation, described as an element-by-element multiplication between a particular location of the input (matrix) and the filter, followed by the sum of the results. Similar to the traditional neural networks, an activation function is applied to every value of the feature maps.

Two parallel series of 1D convolutional layers were used, one for the protein sequences and another for the SMILES strings, to uncover deep patterns (representations or local dependencies). A global max pooling layer was applied, after each series of convolutional layers, to reduce the spatial size of each feature map to its maximum representative feature. The obtained deep representations were concatenated into a single feature vector, characterizing a DTI pair.

The resulting features vectors were then used as the input of a FCNN architecture. This type of neural network is similar to the traditional neural networks, where all the neurons are interlinked and the output is the result of the weighted sum of all the outputs given by the previous connected neurons and to which an activation function is applied. Dropout was applied between each fully connected layer to reduce the overfitting. Deep neural network architectures have many non-linear hidden layers, therefore there are many complex relationships to be learned between inputs and outputs, which can lead to training noise. Dropout is seen as a regularization strategy, which helps reducing learning inter-dependency and improve the generalization of the model. It works by deactivating a given percentage of neuron which develop co-dependency amongst each other during training.

This architecture was followed by an output layer, which is essentially composed by one neuron that returns the type of interaction, 0 or 1, as it is a binary classification problem, classifying the interaction as negative or positive, respectively.

The proposed end-to-end deep learning approach to predict DTIs is illustrated in Figure 4.

2.4 Hyperparameter Optimization Approach

The most common approach to determine the best model architecture and set of parameters is grid search with cross-validation, where the dataset is divided into training to train the model, validation to evaluate the model architecture and parameters and testing to evaluate the performance and generalization of the model. However, another strategy was applied for hyperparameter optimization (Figure 3) due to the fact that dividing the training set into training and validation led to high scores for every model architecture and set of parameters in both training and validation. Therefore, it was not possible to select the best model using this approach, as every model was supposedly good in the validation set but the results were inconsistent when applied to the testing set.

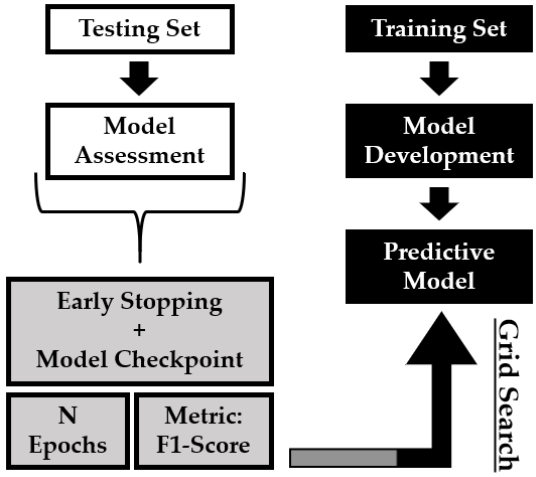

Fig. 3: Hyperparameter optimization model based on grid search.

Two simultaneous methods, combined with grid search were used to determine the best model, early stopping and model checkpoint. Early stopping allows to interrupt the training process if, after a chosen number of epochs, there is no improvement of the evaluation metric. On the other hand, model checkpoint saves the best model, including the parameters, for that training run, independently of the finishing epoch.

Considering that splitting the training set into training and validation was not relevant for the discovery of the best model, we used the whole training set for training and the testing set to evaluate the model performance at each epoch. Since the testing set is highly imbalanced, F1-score, which is an harmonic mean that considers both the precision and recall and therefore an overall goodness of the classification, was used for this evaluation.

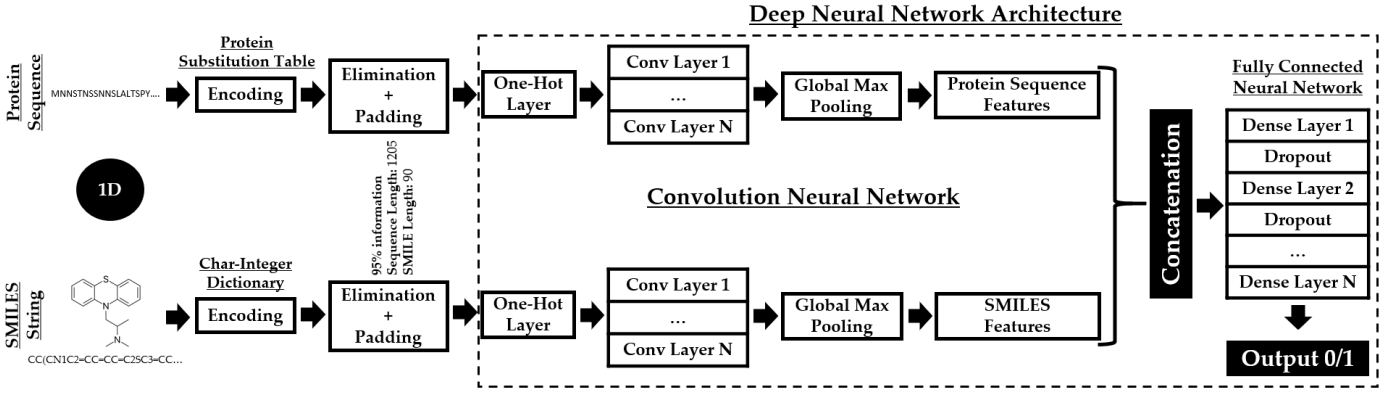

Fig. 4: Drug-Target Interaction Model Architecture.

3 Experimental Setup

We propose a deep neural network architecture to predict the interaction between drugs and targets based on protein sequences and SMILES strings. Conversely to the reference work, which performs grid search applying 5-fold cross-validation, we used the testing set to evaluate the model performance, based on F1-score, in each epoch, as it was explained in Section 2.4, to find the best model and set of parameters. Although the model has several parameters possible to hyperoptimize, we only selected six: number of filters for proteins and compounds, filter length for proteins, filter length for compounds, number of neurons for each dense layer, dropout rate and optimizer learning rate. A wide range of possible values was given for each hyperparameter and the number of convolutional layers and dense layers was fixed at three.

Rectified Linear Unit (ReLU) was selected as the activation function for each convolutional and dense layers. This function is normally used in deep learning architectures [48] and returns zero if it receives any negative input or the value itself if any positive input.

$$f(x) = max(0, x) \tag{1}$$

Loss functions are used to measure the inconsistency between predicted and real values and therefore play an important role in any classification problem. Binary cross entropy was selected as the loss function and measures the divergence between two probability distributions, in which y is the label and p(y) is the predicted probability:

$$L(\theta) = -\frac{1}{n}\sum_{i=1}^{n}[y_i log(p_i) + (1 - y_i) log(1 - p_i)] \tag{2}$$

Adam (Adaptive Moment Estimation) [49], known as an extension to stochastic gradient descent, was used as the optimization algorithm and it is responsible to update the network weights in each iteration of the training process. Considering that this is a binary classification task, sigmoid function was used in the output layer.

$$S(x) = \frac{1}{1 + e^{-x}} \tag{3}$$

Additionally, taking in account the existing class imbalance of the training set (64% and 36 % for the negative and positive class, respectively) we decided to switch class weights, giving special attention to the positive class, as the primary focus is around positive interactions.

Table 7 summarizes the hyperparameters obtained from grid search.

TABLE 7: Parameter settings for the proposed model.* Initial number of epochs to allow convergence of the model, however early stopping and model checkpoint were used.

Parameters	Value
Number of Convolutional Layers	3
Number of Dense Layers (FC)	3
Number of Filters	[128, 256, 384]
Filter Length (Proteins)	[3,4,5]
Filter Length (Compounds)	[3,4,5]
Epochs*	500
Hidden Neurons	[128,128,128]
Batch Size	256
Dropout Rate	0.5
Optimizer	Adam
Learning Rate	0.0001
Loss Function	Binary Cross Entropy
Activation Function (CNN)	ReLU
Activation Function (FC)	ReLU
Activation Function Output)	Sigmoid
Class Weights (imbalanced classes)	{0: 0.36, 1: 0.64}

We compared our proposed model performance with random forest approach, a fully connected neural network architecture, a support vector machine approach and also a CNN, autoencoder and FCNN combined model.

We used Python 3.6.6 and Keras [50] with Tensorflow [51] back-end to develop our proposed model. Our experiments were run on 2.20GHz Intel i7-8750H and GeForce GTX 1060 6GB.

3.1 Random Forest (RF)

Coelho et al. (2016) [34] uses a random forest approach, which is an ensemble learning method that generates a chosen number of decision trees and returns the class that is the mode of the classes across the output of each individual decision tree, to make predictions on drug-target interactions. The parameters used in the random forest model, 150 n_estimators and 100 max_features, were the same as the original work. Scikit-learn [52] was used to implement the random forest method.

Additionally, we decided to use both Coelho et al. (2016) [34] descriptors and protein and SMILES deep representations extracted from the proposed setup, to evaluate the performance of this approach.

3.2 Fully Connected Neural Network (FCNN)

The proposed approach already uses a FCNN as a binary classifier, however we decided to evaluate the performance of this method using descriptors. The parameters setting for the architecture were obtained by grid search using the hyperparameter optimization approach mentioned in Section 2.4. Table 8 summarizes the parameter settings for this architecture.

TABLE 8: Parameter settings for the FCNN model using descriptors as input. *Initial number of epochs to allow convergence of the model, however early stopping and model checkpoint were used.

Parameters	Value
Number of Dense Layers (FC)	3
Epochs*	500
Hidden Neurons	[128,1024,256]
Batch Size	256
Dropout Rate	0.2
Optimizer	Adam
Learning Rate	0.001
Loss Function	Binary Cross Entropy
Activation Function (FC)	ReLU
Activation Function (Output)	Sigmoid
Class Weights (imbalanced classes)	{0: 0.36, 1: 0.64}

3.3 Support Vector Machine (SVM)

Support vector machine defines a hyperplane that maximizes the separation margin between different classes and gives a penalty term for misclassifications. In the case of non-linearly separable problems, it maps data to high dimensional spaces, using kernels, where it is possible to classify with linear decision surfaces. Scikit-learn [52] was used to implement this classifier.

To define the hyper-parameters, namely penalty parameter C, kernel and/or degree (poly kernel), for this model, grid search with 5-fold stratified cross-validation was applied. Contrarily to k-fold cross-validation, stratified ensures that each fold contains roughly the same proportion of each class. Thus, the parameters used were C=1.0 and radial basis function (RBF) kernel.

Similar to the random forest approach, we used both descriptors and protein and SMILES deep representations to evaluate the performance.

3.4 CNN, Autoencoder and FCNN Combined Model

In order to determine the influence of specific descriptors to the overall prediction of DTIs, we decided to evaluate our setup with a model based on Convolution Neural Networks, Autoenconders and Fully Connected Neural Networks. (Figure 5).

Identical to the proposed model, we used two parallel convolutional neural networks to extract deep representations from protein sequences and SMILES strings, where the pre-trained proposed setup was used for this purpose.

Autoencoders are a specific type of neural network architecture, where the learning process is done in an unsupervised manner. The main objective of this architecture is to compress data (dimensionality reduction combined with some data "denoising") and uncompress into something that closely matches the original data. Hence, it allows to extract a smaller set of features that represent the input data [53].

We decided to apply an autoencoder on a particular group of descriptors, namely CTD (Composition, Transition and Distribution) descriptors for proteins and charge, molecular property and molecular connectivity descriptors for compounds. The CTD descriptors represent several structural and physicochemical properties, specifically hydrophobicity, polarity, charge, polarizability, normalized van der Waals volume, secondary structures and solvent accessibility. On the other hand, charge descriptors express electronic features and molecular property and connectivity descriptors represent a handful of physicochemical properties. The main reason behind the choice of these descriptors was that they represent specific and intrinsic properties of the proteins and compounds. The Python package PyDPI [54] was used to extract all the descriptors, resulting in a total of 147 CTD (21 Composition, 21 Transition and 105 Distribution) descriptors, 44 molecular connectivity descriptors, 25 charge descriptors and 6 molecular property descriptors.

Our model uses a stack of dense layers, three for encoding and decoding, respectively. We applied early stopping and model checkpoint based on the loss value to find the best set of weights for the network. This resulted in a dimensionally reduction of 222 descriptors to 32 deep representations. Keras [50] with Tensorflow [51] back-end was used to built this architecture. Table 9 summarizes the parameter settings for the autoencoder model.

TABLE 9: Parameter settings for the autoencoder model. * Initial number of epochs, yet early stopping and model checkpoint were applied.

Parameters	Value
Number of Encoding Dense Layers	3
Number of Decoding Dense Layers	3
Encoding Hidden Neurons	[128, 64, 32]
Decoding Hidden Neurons	[64,128,222]
Epochs*	500
Batch Size	256
Optimizer	Adam
Learning Rate	0.0001
Loss Function	Mean Squared Error
Activation Function	ReLU
Activation Function (Output)	Sigmoid

The obtained features from the two pre-trained models were concatenated into a single feature vector and used as the input of a fully connected neural network. Grid search based on the hyperparameter optimization approach of Section 2.4 was performed. Table 10 summarizes the parameter settings for the fully connected neural network architecture.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/TCBB.2020.2977335, IEEE/ACM Transactions on Computational Biology and Bioinformatics

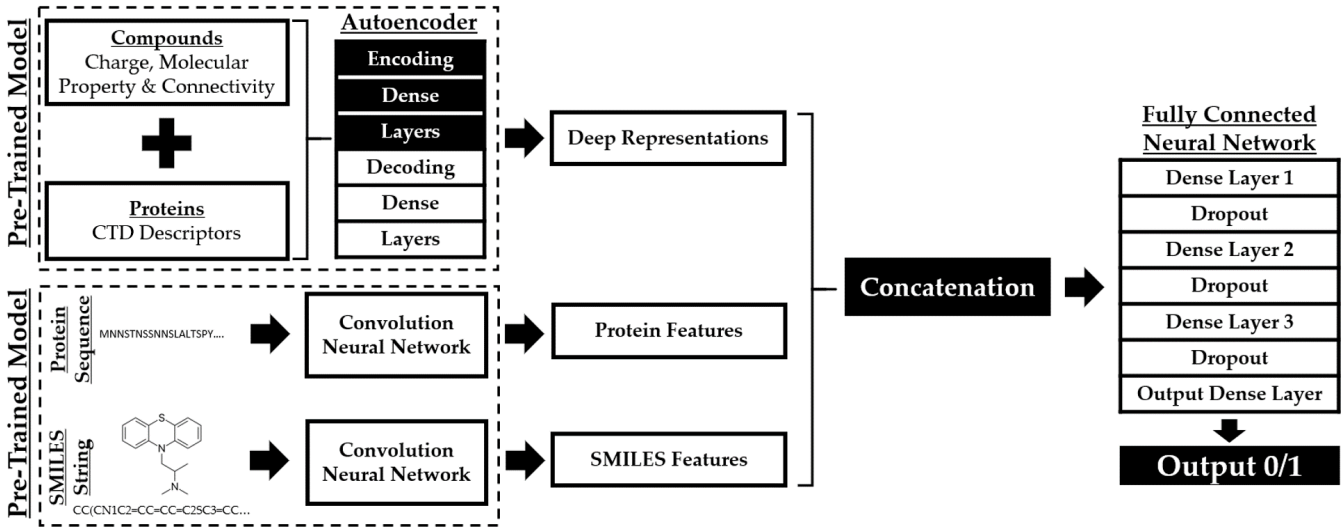

Fig. 5: CNN, Autoencoder and FCNN Combined Model.

TABLE 10: Parameter settings for the fully connected neural network. * Initial number of epochs, however early stopping and model checkpoint were applied.

Parameters	Value
Number of Dense Layers (FC)	3
Epochs*	500
Hidden Neurons	[512,256,1024]
Batch Size	256
Dropout Rate	0.2
Optimizer	Adam
Learning Rate	0.001
Loss Function	Binary Cross Entropy
Activation Function (FC)	ReLU
Activation Function Output)	Sigmoid
Class Weights (imbalanced classes)	{0: 0.36, 1: 0.64}

4 Results

4.1 Evaluation Metrics

For performance comparison, we use the following evaluation metrics:

1) **Accuracy**: rate of predictions correctly classified.

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (4)$$

2) **Sensitivity**: rate of positives correctly classified.

$$Sensitivity = \frac{TP}{TP + FN} \quad (5)$$

3) **Specificity**: rate of negatives correctly classified.

$$Specificity = \frac{TN}{TN + FP} \quad (6)$$

4) **F1-Score**: harmonic mean between precision and recall.

$$F1 - Score = 2 * \frac{precision * recall}{precision + recall} \quad (7)$$

5) **Area Under Receiver Operating Characteristic Curve (AUROC)**: measure of the trade-off between the TP rate and FP rate.

6) **Area Under Precision-Recall Curve (AUPRC)**: measure of the trade-off between the precision and recall.

7) **Confusion Matrix**: two-dimensional table that allows visualization of the performance of the algorithm.

TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative

4.2 Discussion

In the context of drug repositioning and finding new leads, identifying correctly positive interactions should be the central focus, as negative interactions are not normally registered and therefore based on possible hypotheses or absence of information. The differences in performance between all models can be interpreted as a result of the difference between using deep representations, obtained using protein sequences and SMILES strings, and global descriptors. Besides, it is also possible to highlight the difference between applying traditional machine learning and deep learning approaches. We applied grid search for all the models in order to accurately compare and evaluate the performance. Table 11 shows the overall experimental results in terms of seven metrics, including the mean and standard deviation for a total of 30 runs. Supplementary Material contains the predictions results of testing set for all the 30 runs for each model.

Our approach is based on the concept of using an end-to-end deep learning process, capable of extracting deep representations from data and then use them as input of another deep learning architecture. We used two parallel convolutional neural networks to extract deep representations from protein sequences and SMILES strings and then fed them into a fully connected neural network. The results obtained validate the effectiveness of convolutional neural networks as feature extractors and their capacity to automatically surmise and identify important sequential and structural regions for drug-target interactions, as

TABLE 11: Prediction Results of Testing Set.

		Model						
		CNN+FCNN	CNN+Autoencoder+FCNN	FCNN	Random Forest		SVM RBF	
		CNN Representations	CNN+Descriptors Representations	Descriptors	Descriptors	CNN Representations	Descriptors	CNN Representations
Metric	Sensitivity	0.861 (0.841,0.011)	0.880 (0.844,0.013)	0.827 (0.800,0.015)	0.819 (0.816,0.003)	0.833 (0.827,0.003)	0.720 (0.720,0.000)	0.765 (0.765,0.000)
	Specificity	0.961 (0.971,0.007)	0.948 (0.972,0.008)	0.963 (0.963,0.010)	0.989 (0.989,0.001)	0.992 (0.992,0.000)	0.983 (0.983,0.000)	0.993 (0.993,0.000)
	F1-Score	0.895 (0.890,0.003)	0.896 (0.893,0.002)	0.876 (0.860,0.006)	0.892 (0.890,0.002)	0.902 (0.899,0.002)	0.824 (0.824,0.000)	0.861 (0.861,0.000)
	Accuracy	0.923 (0.921,0.002)	0.922 (0.923,0.001)	0.911 (0.901,0.004)	0.925 (0.923,0.001)	0.931 (0.929,0.001)	0.883 (0.883,0.000)	0.906 (0.906,0.000)
	AUROC	0.966 (0.969,0.004)	0.972 (0.964,0.003)	0.955 (0.940,0.010)	0.988 (0.987,0.000)	0.988 (0.988,0.000)	0.915 (0.915,0.000)	0.965 (0.965,0.000)
	AUPRC	0.960 (0.960,0.003)	0.966 (0.958,0.003)	0.949 (0.933,0.008)	0.982 (0.982,0.001)	0.983 (0.983,0.001)	0.918 (0.918,0.000)	0.958 (0.958,0.000)
	Inference Time (s)	5.024	0.453	0.419	0.137	0.108	18.198	9.089

Note: The mean and standard deviation for a total of 30 runs are given in parenthesis.

they outperform completely the results achieved using a fully connected neural network with global descriptors. Another observation is that using an end-to-end deep learning method resulted in a high sensitivity (0.861) and specificity (0.961) when compared to the other models, which obtained a high specificity and a low sensitivity, with the exception of the CNN, autoencoder and FCNN combined model that resulted in a high sensitivity (0.880) and a low specificity (0.948). Being the testing set imbalanced, 62% negatives and 38% positives, our approach exceeds other models in its capability to correctly classify both positive and negative drug-target interactions. Thus, it is possible to conclude that our model achieved better results than the one used in reference work. The confusion matrix is shown in Figure 6.

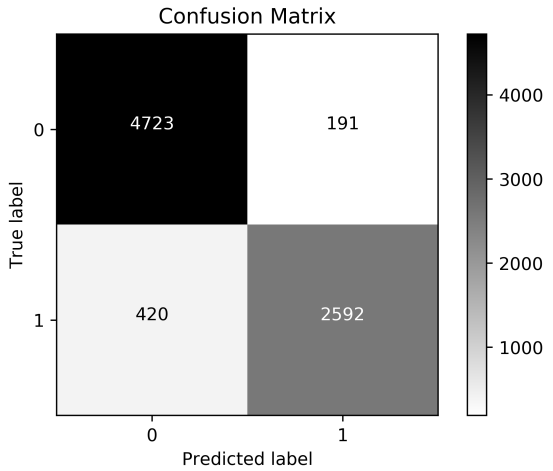

Fig. 6: Confusion matrix of testing set classification for the proposed model.

Random forest method evidences that using deep representations outperforms conventional global descriptors in every evaluation metric. Additionally, this is also manifested when using a support vector machine. The results indicate that protein and compound representations learned from sequential information with convolutional neural networks are more discriminating for classification than global descriptors. Furthermore, these representations are extracted from sequential raw data, hence the CNNs are automatically learning which sequential regions are relevant for a drug-target interaction. Conversely, conventional descriptors are general information about the whole sequence or structure and not specific to the binding regions. Lastly, random forest surpasses support vector machine in both configurations, which is in agreement to the notion that this method usually runs adequately on large datasets and is less susceptible to overfitting.

The model based on a fully connected neural network architecture with conventional descriptors as input, shows that deep learning in its essence is not enough to completely outperform traditional machine learning approaches. This is illustrated when comparing the evaluation metrics, which are higher, specifically the sensitivity (0.827), F1-score (0.876) and accuracy (0.911), than the support vector machine approach but overall lower than random forest method in both configurations, respectively. Moreover, it highlights the inefficiency of using global descriptors over deep representations extracted from CNNs. Inevitably, the quality and discriminatory power of the input data have a great influence in the performance achieved.

Although the efficiency of using CNNs to extract deep representations over global descriptors is verified, we decided to evaluate the influence of specific descriptors encoded as deep representations by an autoencoder combined with proteins and SMILES deep representations. The results demonstrate that using additional information may be useful to correctly identify positive interactions, which is verified by achieving the highest sensitivity (0.880). However, it has the lowest specificity of all models (0.948), meaning it has more difficulty to accurately classify negative (non existing) interactions. Nonetheless, the majority of the input is obtained from the CNN model, 768 protein and SMILES deep representations and 32 descriptors deep representations, which proves again the capability of this deep feature extractor model. Moreover, it reinforces the fact that using end-to-end deep learning approaches result in better performance overall.

5 Conclusion

In this paper we proposed an end-to-end deep learning approach for drug-target interaction prediction, capable of automatically feature (deep representations) extraction from sequential raw data, protein sequences and SMILES strings, using two parallel convolution neural networks. We compared the performance of this model with traditional machine learning methods, random forest and support vector machine, using both descriptors and deep representations, a deep learning approach based on a fully connected neural network with global descriptors as the input and also a convolution neural network, autoencoder and fully connected neural network combined model, that uses as input a combination of deep sequential representations obtained from the CNNs and deep descriptors representations from

the autoencoder. Our approach yielded better results in the correct classification of both positive and negative interactions, demonstrating its viability for practical use.

Deep learning has shown an overwhelming success in many classification studies for its capacity to learn deep hidden patterns from the data. Additionally, our model illustrates the remarkable ability of applying these approaches, specifically convolution neural networks, to automatically extract deep representations, identified as local patterns or dependencies, and use them to describe drug-target interactions. The results obtained showed that using these representations outperformed completely global descriptors in every model applied, proving the importance and relevance of the features extracted and also the capacity to identify and learn particular sequential regions meaningful to the interaction. Nonetheless, deep learning does not always surpass traditional machine learning approaches, as demonstrated when comparing the FCNN model and random forest.

In addition, we also evaluated the influence of particular descriptors, encoded into deep representations, combined with sequential deep representations extracted from protein sequences and SMILES strings. The results demonstrated that additional information may prove to be useful to correctly identify positive interactions, as this model obtained the highest sensitivity (0.880). On that account, as future work we will focus on building an effective ensemble of meaningful information for interaction that will further be integrated in our end-to-end deep learning model.

Acknowledgment

This research has been funded by the Portuguese Research Agency FCT, through D4 - Deep Drug Discovery and Deployment (CENTRO-01-0145-FEDER-029266).

Data Availability

The proposed classification model and related data files are available at https://github.com/larngroup/DTI-End-to-End-DL.

References

[1] B. Aslam, W. Wang, M. I. Arshad, M. Khurshid, S. Muzammil, M. H. Rasool, M. A. Nisar, R. F. Alvi, M. A. Aslam, M. U. Qamar, M. K. F. Salamat, and Z. Baloch, "Antibiotic resistance: a rundown of a global crisis," *Infect Drug Resist*, vol. 11, pp. 1645–1658, Oct 2018, 30349322[pmid]. [Online]. Available: https://doi.org/10.2147/IDR.S173867

[2] S. M. Paul, D. S. Mytelka, C. T. Dunwiddie, C. C. Persinger, B. H. Munos, S. R. Lindborg, and A. L. Schacht, "How to improve R&D productivity: the pharmaceutical industry's grand challenge," *Nature Reviews Drug Discovery*, vol. 9, pp. 203 EP –, Feb 2010. [Online]. Available: https://doi.org/10.1038/nrd3078

[3] T. T. Ashburn and K. B. Thor, "Drug repositioning: identifying and developing new uses for existing drugs," *Nature Reviews Drug Discovery*, vol. 3, pp. 673 EP –, Aug 2004, review Article. [Online]. Available: https://doi.org/10.1038/nrd1468

[4] A. J. Butte, J. T. Dudley, and T. Deshpande, "Exploiting drug–disease relationships for computational drug repositioning," *Briefings in Bioinformatics*, vol. 12, no. 4, pp. 303–311, 06 2011. [Online]. Available: https://dx.doi.org/10.1093/bib/bbr013

[5] R. Chen, X. Liu, S. Jin, J. Lin, and J. Liu, "Machine learning for drug-target interaction prediction," *Molecules*, vol. 23, no. 9, 2018. [Online]. Available: http://www.mdpi.com/1420-3049/23/9/2208

[6] M. J. Keiser, B. L. Roth, B. N. Armbruster, P. Ernsberger, J. J. Irwin, and B. K. Shoichet, "Relating protein pharmacology by ligand chemistry," *Nature Biotechnology*, vol. 25, pp. 197 EP –, Feb 2007. [Online]. Available: https://doi.org/10.1038/nbt1284

[7] H. González-Díaz, F. Prado-Prado, X. García-Mera, N. Alonso, P. Abeijón, O. Caamaño, M. Yáñez, C. R. Munteanu, A. Pazos, M. A. Dea-Ayuela, M. T. Gómez-Muñoz, M. M. Garijo, J. Sansano, and F. M. Ubeira, "MIND-BEST: Web server for drugs and target discovery; design, synthesis, and assay of MAO-B inhibitors and theoretical-experimental study of G3PDH protein from trichomonas gallinae," *Journal of Proteome Research*, vol. 10, no. 4, pp. 1698–1718, Apr 2011. [Online]. Available: https://doi.org/10.1021/pr101009e

[8] F. Cheng, Y. Zhou, J. Li, W. Li, G. Liu, and Y. Tang, "Prediction of chemical-protein interactions: multitarget-QSAR versus computational chemogenomic methods," *Molecular BioSystems*, vol. 8, no. 9, pp. 2373–2384, 2012. [Online]. Available: http://dx.doi.org/10.1039/C2MB25110H

[9] G. Pujadas, M. Vaqué, A. Ardèvol, C. Bladé, M.-J. Salvadó, M. Blay, J.-B. Fernandez-Larrea, and L. Arola, *Protein-ligand Docking: A Review of Recent Advances and Future Perspectives*, ser. Current Pharmaceutical Analysis, Feb 2008, vol. 4. [Online]. Available: http://doi.org/10.2174/157341208783497597

[10] H. Zhang, H. Li, H. Jiang, J. Shen, K. Chen, K. Yang, K. Yu, L. Kang, W. Zhu, X. Luo, X. Wang, and Z. Gao, "TarFisDock: a web server for identifying drug targets with docking approach," *Nucleic Acids Research*, vol. 34, no. suppl_2, pp. W219–W224, Jul 2006. [Online]. Available: https://dx.doi.org/10.1093/nar/gkl114

[11] L. Yang, K. Wang, J. Chen, A. G. Jegga, H. Luo, L. Shi, C. Wan, X. Guo, S. Qin, G. He, G. Feng, and L. He, "Exploring off-targets and off-systems for adverse drug reactions via chemical-protein interactome – clozapine-induced agranulocytosis as a case study," *PLOS Computational Biology*, vol. 7, no. 3, p. e1002016, Mar 2011. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1002016

[12] A. C. Cheng, R. G. Coleman, K. T. Smyth, Q. Cao, P. Soulard, D. R. Caffrey, A. C. Salzberg, and E. S. Huang, "Structure-based maximal affinity model predicts small-molecule druggability," *Nature Biotechnology*, vol. 25, pp. 71 EP –, Jan 2007. [Online]. Available: https://doi.org/10.1038/nbt1273

[13] A. Gutteridge, M. Araki, M. Kanehisa, W. Honda, and Y. Yamanishi, "Prediction of drug-target interaction networks from the integration of chemical and genomic spaces," *Bioinformatics*, vol. 24, no. 13, pp. i232–i240, Jul 2008. [Online]. Available: https://dx.doi.org/10.1093/bioinformatics/btn162

[14] F. Cheng, C. Liu, J. Jiang, W. Lu, W. Li, G. Liu, W. Zhou, J. Huang, and Y. Tang, "Prediction of drug-target interactions and drug repositioning via network-based inference," *PLOS Computational Biology*, vol. 8, no. 5, p. e1002503, May 2012. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1002503

[15] D.-S. Cao, L.-X. Zhang, G.-S. Tan, Z. Xiang, W. Zeng, Q. Xu, and A. Chen, *Computational Prediction of DrugTarget Interactions Using Chemical, Biological, and Network Features*, ser. Molecular Informatics, Oct 2014, vol. 33. [Online]. Available: https://doi.org/10.1002/minf.201400009

[16] H. Yu, J. Chen, X. Xu, Y. Li, H. Zhao, Y. Fang, X. Li, W. Zhou, W. Wang, and Y. Wang, "A systematic prediction of multiple drug-target interactions from chemical, genomic, and pharmacological data," *PLOS ONE*, vol. 7, no. 5, p. e37608, May 2012. [Online]. Available: https://doi.org/10.1371/journal.pone.0037608

[17] N. Nagamine and Y. Sakakibara, "Statistical prediction of protein-chemical interactions based on chemical structure and mass spectrometry data," *Bioinformatics*, vol. 23, no. 15, pp. 2004–2012, May 2007. [Online]. Available: https://dx.doi.org/10.1093/bioinformatics/btm266

[18] M. C. Cobanoglu, C. Liu, F. Hu, Z. N. Oltvai, and I. Bahar, "Predicting drug-target interactions using probabilistic matrix factorization," *Journal of Chemical Information and Modeling*, vol. 53, no. 12, pp. 3399–3409, Dec 2013. [Online]. Available: https://doi.org/10.1021/ci400219z

[19] K. Bleakley and Y. Yamanishi, "Supervised prediction of drug-target interactions using bipartite local models," *Bioinformatics*, vol. 25, no. 18, pp. 2397–2403, Jul 2009. [Online]. Available: https://dx.doi.org/10.1093/bioinformatics/btp433

[20] K. Tian, M. Shao, Y. Wang, J. Guan, and S. Zhou, "Boosting compound-protein interaction prediction by deep learning," *Methods*, vol. 110, pp. 64–72, 2016. [Online]. Available: https://doi.org/10.1016/j.ymeth.2016.06.024

[21] Peng-Wei, K. Chan, and Z.-H. You, *Large-scale prediction of drug-target interactions from deep representations*, Jul 2016. [Online]. Available: https://doi.org/10.1109/IJCNN.2016.7727339

[22] M. Wen, Z. Zhang, S. Niu, H. Sha, R. Yang, Y. Yun, and H. Lu, "Deep-learning-based drug-target interaction prediction," *Journal of Proteome Research*, vol. 16, no. 4, pp. 1401–1409, Apr 2017. [Online]. Available: https://doi.org/10.1021/acs.jproteome.6b00618

[23] L. Xie, S. He, X. Song, X. Bo, and Z. Zhang, "Deep learning-based transcriptome data classification for drug-target interaction prediction," *BMC Genomics*, vol. 19, no. 7, p. 667, 2018. [Online]. Available: https://doi.org/10.1186/s12864-018-5031-0

[24] Y. Lecun, K. Kavukcuoglu, and C. Farabet, *Convolutional Networks and Applications in Vision*, ser. ISCAS 2010 - 2010 IEEE International Symposium on Circuits and Systems: Nano-Bio Circuit Fabrics and Systems, May 2010. [Online]. Available: https://doi.org/10.1109/ISCAS.2010.5537907

[25] A. Krizhevsky, I. Sutskever, and G. E. Hinton, *ImageNet Classification with Deep Convolutional Neural Networks*, ser. Neural Information Processing Systems, Jan 2012, vol. 25. [Online]. Available: https://doi.org/10.1145/3065386

[26] L. Kang, P. Ye, Y. Li, and D. Doermann, *Convolutional Neural Networks for No-Reference Image Quality Assessment*, Jun 2014. [Online]. Available: https://doi.org/10.1109/CVPR.2014.224

[27] K. Üreten, H. Erbay, and H. H. Maras, "Detection of rheumatoid arthritis from hand radiographs using a convolutional neural network," *Clinical Rheumatology*, 2019. [Online]. Available: https://doi.org/10.1007/s10067-019-04487-4

[28] H. Zeng, M. D. Edwards, G. Liu, and D. K. Gifford, "Convolutional neural network architectures for predicting dna-protein binding," *Bioinformatics*, vol. 32, no. 12, pp. i121–i127, Jun 2016, 27307608[pmid]. [Online]. Available: https://doi.org/10.1093/bioinformatics/btw255

[29] H. Öztürk, A. Özgür, and E. Ozkirimli, "DeepDTA: deep drug-target binding affinity prediction," *Bioinformatics*, vol. 34, no. 17, pp. i821–i829, Sep 2018. [Online]. Available: https://dx.doi.org/10.1093/bioinformatics/bty593

[30] S. Budach and A. Marsico, "pysster: classification of biological sequences by learning sequence and structure motifs with convolutional neural networks," *Bioinformatics*, vol. 34, no. 17, pp. 3035–3037, Apr 2018. [Online]. Available: https://dx.doi.org/10.1093/bioinformatics/bty222

[31] S. Kwon and S. Yoon, "End-to-end representation learning for chemical-chemical interaction prediction," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, pp. 1–1, 2018. [Online]. Available: https://doi.org/10.1109/TCBB.2018.2864149

[32] Q. Feng, E. Dueva, A. Cherkasov, and M. Ester, "Padme: A deep learning-based framework for drug-target interaction prediction," 2018.

[33] W. Torng and R. B. Altman, "Graph convolutional neural networks for predicting drug-target interactions," *bioRxiv*, 2018. [Online]. Available: https://www.biorxiv.org/content/early/2018/11/19/473074

[34] E. D. Coelho, J. P. Arrais, and J. L. Oliveira, "Computational discovery of putative leads for drug repositioning through drug-target interaction prediction," *PLOS Computational Biology*, vol. 12, no. 11, pp. 1–17, 11 2016. [Online]. Available: https://doi.org/10.1371/journal.pcbi.1005219

[35] A. C. Guo, B. Gautam, C. Knox, D. Tzur, D. Cheng, M. Hassanali, S. Shrivastava, and D. S. Wishart, "DrugBank: a knowledgebase for drugs, drug actions and drug targets," *Nucleic Acids Research*, vol. 36, no. suppl_1, pp. D901–D906, Nov 2007. [Online]. Available: https://dx.doi.org/10.1093/nar/gkm958

[36] J. Yang, A. Roy, and Y. Zhang, "BioLiP: a semi-manually curated database for biologically relevant ligand-protein interactions," *Nucleic Acids Res*, vol. 41, no. Database issue, pp. D1096–D1103, Jan 2013, 23087378[pmid]. [Online]. Available: https://doi.org/10.1093/nar/gks966

[37] T. Liu, Y. Lin, X. Wen, R. N. Jorissen, and M. K. Gilson, "BindingDB: a web-accessible database of experimentally determined protein-ligand binding affinities," *Nucleic Acids Res*, vol. 35, no. Database issue, pp. D198–D201, Jan 2007, 17145705[pmid]. [Online]. Available: https://doi.org/10.1093/nar/gkl999

[38] T. U. Consortium, "UniProt: the universal protein knowledgebase," *Nucleic Acids Research*, vol. 45, no. D1, pp. D158–D169, Nov 2016. [Online]. Available: https://dx.doi.org/10.1093/nar/gkw1099

[39] S. Kim, P. A. Thiessen, E. E. Bolton, J. Chen, G. Fu, A. Gindulyte, L. Han, J. He, S. He, B. A. Shoemaker, J. Wang, B. Yu, J. Zhang, and S. H. Bryant, "PubChem substance and compound databases," *Nucleic Acids Res*, vol. 44, no. D1, pp. D1202–D1213, Jan 2016, 26400175[pmid]. [Online]. Available: https://doi.org/10.1093/nar/gkv951

[40] H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov, and P. E. Bourne, "The protein data bank," *Nucleic Acids Res*, vol. 28, no. 1, pp. 235–242, Jan 2000, 10592235[pmid].

[41] M. Kanehisa and S. Goto, "KEGG: kyoto encyclopedia of genes and genomes," *Nucleic Acids Res*, vol. 28, no. 1, pp. 27–30, Jan 2000, 10592173[pmid].

[42] J. J. Irwin and B. K. Shoichet, "ZINC–a free database of commercially available compounds for virtual screening," *J Chem Inf Model*, vol. 45, no. 1, pp. 177–182, 2005, 15667143[pmid]. [Online]. Available: https://doi.org/10.1021/ci049714+

[43] T. Sterling and J. J. Irwin, "ZINC 15–ligand discovery for everyone," *J Chem Inf Model*, vol. 55, no. 11, pp. 2324–2337, Nov 2015, 26479676[pmid]. [Online]. Available: https://doi.org/10.1021/10.1021/acs.jcim.5b00559

[44] W. Gilpin, "PyPDB: a Python API for the protein data bank," *Bioinformatics*, vol. 32, no. 1, pp. 159–160, Sep 2015. [Online]. Available: https://dx.doi.org/10.1093/bioinformatics/btv543

[45] D. Pultz, J. Serra-Musach, J. Saez-Rodriguez, L. M. Harder, and T. Cokelaer, "BioServices: a common Python package to access biological web services programmatically," *Bioinformatics*, vol. 29, no. 24, pp. 3241–3242, Sep 2013. [Online]. Available: https://dx.doi.org/10.1093/bioinformatics/btt547

[46] M. Swain *et al.*, "PubChemPy," https://github.com/mcs07/PubChemPy, 2014.

[47] C.-Y. Yu, L.-C. Chou, and D. T.-H. Chang, "Predicting protein-protein interactions in unbalanced data using the primary structure of proteins," *BMC Bioinformatics*, vol. 11, no. 1, p. 167, 2010. [Online]. Available: https://doi.org/10.1186/1471-2105-11-167

[48] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, pp. 436 EP –, May 2015. [Online]. Available: https://doi.org/10.1038/nature14539

[49] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *CoRR*, vol. abs/1412.6980, 2015.

[50] F. Chollet *et al.*, "Keras," https://github.com/fchollet/keras, 2015.

[51] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, and X. Zhang, *TensorFlow: A system for large-scale machine learning*, May 2016.

[52] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, and G. Louppe, *Scikit-learn: Machine Learning in Python*, ser. Journal of Machine Learning Research, Jan 2012, vol. 12.

[53] G. E. Hinton and R. R. Salakhutdinov, "Reducing the dimensionality of data with neural networks," *Science*, vol. 313, no. 5786, p. 504, Jul 2006. [Online]. Available: https://doi.org/10.1126/science.1127647

[54] D.-S. Cao, Y.-Z. Liang, J. Yan, G.-S. Tan, Q.-S. Xu, and S. Liu, "PyDPI: Freely available Python package for chemoinformatics, bioinformatics, and chemogenomics studies," *Journal of Chemical Information and Modeling*, vol. 53, no. 11, pp. 3086–3096, Nov 2013. [Online]. Available: https://doi.org/10.1021/ci400127q

Nelson R. C. Monteiro received the BSc and MSc degree in Biomedical Engineering, specializing in Clinical Informatics and Bioinformatics, from Faculty of Science and Technology of the University of Coimbra in 2017 and 2019, respectively. He is currently a research fellow at the Institute of Electronics and Informatics Engineering of Aveiro and a PhD student at the Department of Informatics Engineering of the University of Coimbra. His research interests include machine learning, deep learning, pattern recognition and data science applied to biology, genetics, biochemistry and pharmaceutics.

Bernardete Ribeiro Bernardete Ribeiro (SM'15) received the Ph.D. and Habilitation degrees in informatics engineering from University of Coimbra. She is currently a Full Professor with University of Coimbra, also the Director of the Center of Informatics and Systems, University of Coimbra, also the President of the Portuguese Association of Pattern Recognition, and also the Founder and the Director of the Laboratory of Artificial Neural Networks for over 20 years. Her research interests are in the areas of machine learning, and pattern recognition and their applications to a broad range of fields. She has been responsible/participated in several research projects both in international and national levels in a wide range of application areas. She is an IEEE SMC Senior Member, a member of the IARP International Association of Pattern Recognition, a member of the International Neural Network Society, a member of the APCA Portuguese Association of Automatic Control, a member of the Portuguese Association for Artificial Intelligence, a member of the American Association for the Advancement of Science, and a member of the Association for Computing Machinery. She received several awards and recognitions.

Joel P. Arrais received the BS and PhD degree in Computer Science from University of Aveiro, Portugal in 2004 and 2010 respectively. Since 2012 He is a Assistant Professor at University of Coimbra, Portugal and has been responsible for lecturing several courses on the subjects of Computer Science and Computational Biology. His research interests include pattern recognition and machine learning applied to computational biology.