Neural network and deep-learning algorithms used in QSAR studies: merits and drawbacks

Fahimeh Ghasemi[1], Alireza Mehridehnavi[1], Alfonso Pérez-Garrido[2] and Horacio Pérez-Sánchez[2]

[1] Department of Bioinformatics and Systems Biology, School of Advanced Technologies in Medicine, Isfahan University of Medical Sciences, Isfahan, Hezar-Jerib Ave., 81746 73461, Islamic Republic of Iran
[2] Bioinformatics and High Performance Computing Research Group (BIO-HPC), Universidad Católica de Murcia (UCAM), E30107 Murcia, Spain

Introduction

Drug discovery protocols in the pharmaceutical industry have for many years mainly relied on high-throughput screening (HTS) methods for rapidly ascertaining the biological or biochemical activity of a large number of drug-like compounds. Various problems, including the efficacy, activity, toxicity, and bioavailability of the designed compounds, are frequently encountered during the discovery process. Computational techniques, which provide options for understanding chemical systems, yield information that is difficult, if not impossible, to obtain in laboratory experiments. In recent decades, these techniques, when used in drug design procedures, have accelerated the process of HTS by using the virtual features of molecules. Among all high-throughput virtual screening (HTVS) approaches, those involving quantitative structure–activity relationships (QSARs) have proved their applicability in modern drug discovery protocols. The method depends exclusively on the physicochemical features of the ligands (molecular descriptors) when no information is available concerning the 3D structure of the target [1,2]. QSARs are fundamentally a protocol that applies a knowledge of statistics and mathematics to the prediction or classification of biological data related to designed molecules. Many linear and nonlinear statistical model-building methods have been applied in the QSAR approach.

Artificial NNs (ANNs) [3–5] are one of the most popular nonlinear modeling methods used in QSAR studies. These were first applied in drug design in 1973 by Hiller *et al.*, who indicated that NNs could be helpful for the classification of molecules into two categories: active and inactive [6]. Later, in 1990, Aoyama *et al.* successfully applied NNs in decision-making relating to compound interactions, contrasting NNs with a linear model-building method, multiple linear regression (MLR). These authors tested NNs as a multiple regression method with one neuron at the output layer for predicting molecular biological activity [7]. At the same time, NNs were widely used in QSARs, based on the 2D representation of compound similarities [8,9]. In all studies, it was confirmed that NNs are potential tools for the routine tasks of QSAR analysis, feature extraction, nonlinear modeling, classification and prediction [10,11].

The number of drug-like compounds of potential use in the pharmaceutical industry is increasing daily, and the same is true for the number of molecular descriptors describing the physicochemical features of these compounds. However, arcane descriptors can affect the results of biological activity prediction or classification, although the models cannot be interpreted, whereas simpler interpretable descriptors cannot make good models for diverse data sets. By contrast, there are two major disadvantages in QSAR studies, namely redundancy and overfitting, which makes prediction and/or classification unreliable. Over the past two decades, several algorithms have been proposed as possible solutions to these drawbacks. Moreover, in HTVS, thousands of molecules and descriptors inevitably lead to the selection of networks with more than one hidden layer and many nodes in each layer. Nevertheless, not all the proposed solutions have been successful in solving the above-mentioned problems.

Besides identifying network problems in QSARs, other fields of research have experienced the same problems with NNs. In 2006, a novel fast algorithm was introduced by Hinton *et al.* [12] based on the restricted Boltzmann machine (RBM), which represented the infrastructure of DL methods in processing fields (e.g., computer vision, speech processing, and image processing) and led to the recovery of NNs. The DL configuration comprises multiple levels of

Corresponding authors: Ghasemi, F. (f_ghasemi@amt.mui.ac.ir), Pérez-Sánchez, H. (hperez@ucam.edu)

https://doi.org/10.1016/j.drudis.2018.06.016

linear and nonlinear operations. In fact, the approach is based on a hierarchical construction, in which higher-level features are calculated over lower-level features. The depth of DL models refers to the longest path from an input to an output node [13]. Based on deep architectures, there have been various solutions to avoid the situation of getting stuck in local minima and being vulnerable to overfitting. The main proposed algorithms are: deep belief networks (DBNs) [14], convolutional NN s(CNNs) [15], dropout [16], autoencoder [17], hessian free optimization (HF) [18], and rectified linear units (ReLUs) [19] instead of the sigmoid function [20], and the conditional RBM [21].

Here, we review the advantages and disadvantages of NN algorithms, especially innovative DL techniques for use in ligand-based VS.

Overview of studies using deep-learning algorithms in chemoinformatics

The event that led to the advent of DL in drug discovery was the Kaggle competition promoted by Merck in 2012. DL inventors won the competition mainly through using DL in QSAR to capture complex statistical patterns among thousands of descriptors extracted from numerous compounds [22]. Moreover, DL is appropriate for finding the best statistical model for predicting biological activity and for classifying thousands of compounds based on their descriptors in HTVS [23]. In recent years, there have been various approaches applied in QSAR based on deep architecture.

Lusci *et al.* showed how recursive NN approaches can be applied to the problem of predicting molecular properties [24]. The RBM was used by Wang and Zeng to predict drug–target interactions [25]. Unterthiner *et al.* compared the performance of a DL approach in seven target prediction methods on the ChEMBL database, finding that DL outperformed all other methods in terms of the area under the curve (AUC) [23]. Ma *et al.* demonstrated that deep NNs (DNN) based on the procedure of dropout and ReLUs as activation function routinely made better prospective predictions compared with random forest (RF) on a large set of diverse QSAR data sets [26]. Hughes used a database of 702 epoxidation reactions to build a deep machine-learning network, and concluded that the site of epoxidations (SOEs) had 94.9% AUC, and the separation of epoxidized and non-epoxidized molecules was carried out with 79.3% AUC [22]. Tian *et al.* improved the prediction of ligand–target interactions based on DL using the DNN algorithm. In this study, chemical and protein features were extracted from ligand–target interactions as an input algorithm. The authors succeeded in improving the performance of their network by applying ReLU as the activation function [27]. Ghasemi *et al.* looked for a solution to improve the performance of DNN, focusing on primary parameter computation using the DBN called DBN-DNN, observing that the outputs of DBN-DNN appeared to outperform the models obtained with DNN [28,29] (F. Ghasemi, PhD thesis, Isfahan University of Medical Science, 2017). Koutsoukas *et al.* compared the performance of the DNN with a shallow learning algorithm for molecular classification [i.e., Naïve Bayes, k-nearest neighbor (KNN), RF, and support vector machines (SVM)]. It was shown that the average Matthews correlation coalition (MCC) of DNN was achieved more than with the shallow learning algorithm, whereas at the higher level of noise, the Naïve Bayes method performed better than DNN [30]. Xu *et al.* investigated why multitask DNNs perform differently from ANN and RF in QSAR tasks in the Kaggle data sets. The authors found that if activities between molecules in the training and test sets were uncorrelated, the predictive performance was reduced using multitask DNNs [31]. Wen *et al.* used a DBN model to effectively reduce raw input vectors and accurately predict Drug Target Interactions (DTIs), finding that the proposed algorithm achieved a high prediction performance [32]. Generative adversarial autoencoders (AAE) were applied for generating novel molecular fingerprints by Kadurin *et al.* [33]. In this study, 72 million compounds in PubChem were used to train AAE and candidate molecules were selected for their potential anticancer properties [33]. Zhang *et al.* reviewed QSAR methods and used the DL algorithm on drug discovery, and concluded that DL methods are suitable for complex tasks based on large, heterogeneous and high-dimensional data, and could be applicable in the early stages of drug design and discovery [34]. A deep CNN (DCNN), called AtomNet, was introduced to predict the biological activity of small molecules in drug discovery by Wallach *et al.* [75]. In this study, DCNN was applied as a filter to structural targets and biological activity prediction of new compounds. The authors showed that AtomNet outperformed previous algorithms and achieved an AUC greater than 0.9 on 57.8% of the targets. Winkler and Le [35] compared the test set prediction accuracies for models built with RF, DNN, and ANN using the same data Kaggle data sets and descriptors. They showed that, on average, Bayesian regularized ANNs and DNN had the same predictive performance [35].

Overview of the drawbacks of neural networks in QSAR studies

Despite all of the advantages of the proposed NN algorithms in drug discovery programs compared with other machine learning algorithms, NNs have two serious problems: (i) the existence of thousands of descriptors, as well as the correlation between them, which leads to redundancy problems and, inevitably, 'getting stuck' in local minima. By contrast, various unknown descriptors exist that affect the results of QSAR models. Thus, feature selection algorithms have been suggested for reducing the number of descriptors [36]; and (ii) it is difficult to apply optimized essential network parameters to obtain the best prediction without overfitting. To avoid this issue, various regularization learning techniques have been proposed (e.g., ReLU and dropout in DL algorithms, and Bayesian methods in shallow NNs) [37].

The problem redundancy is of significant concern in QSARs, especially in the NN design, because it negatively affects network efficiency. In NNs, the origin of redundancy is referred to as duplicate input data. Moreover, data redundancy appears when some of the input variables of the network are repeated in the input variable [38]. In QSAR studies, redundancy refers to the similarity and correlation between descriptors. Although choosing a limited number of descriptors as input neurons was found to be useful, a new question arose in QSAR as to which descriptors could influence the prediction of biological activities. Hence, two different solutions were proposed to remove the problem of redundancy: (i) manual descriptor selection; and (ii) using data-mining techniques to extract the best features of descriptors as variables

for network input. These models are called hybrid networks [10,39]. Such hybrid networks have two main parts: (i) an unsupervised step, called the preprocessing step, which is based on the molecular descriptors; and (ii) a supervised step via the NN method, which is based on both descriptors and the biological activities of compounds [40]. Conventionally recommended techniques used for descriptor reduction include stepwise descriptor selection [41], self-organization map (SOM) [10], principle component analysis (PCA) [9], genetic algorithms (GA) [42], and optimal sparse descriptor selection using Bayesian methods [43].

The second challenge in NN is to apply the appropriate technique to avoid overfitting. When the network is prone to overfitting, the accuracy of prediction in the training set is favorable, but, for the test set, it would not be acceptable. One possible solution involves the simplest network structure, universal approximation theorem, based on the one hidden layer and the few neurons in the hidden layer [44]. In addition, various learning approaches have been proposed to improve the network parameters, including: back propagation (BP) [45], radial basis function NN (RBFN) [46], and counter propagation (CP) [47] (Table 1).

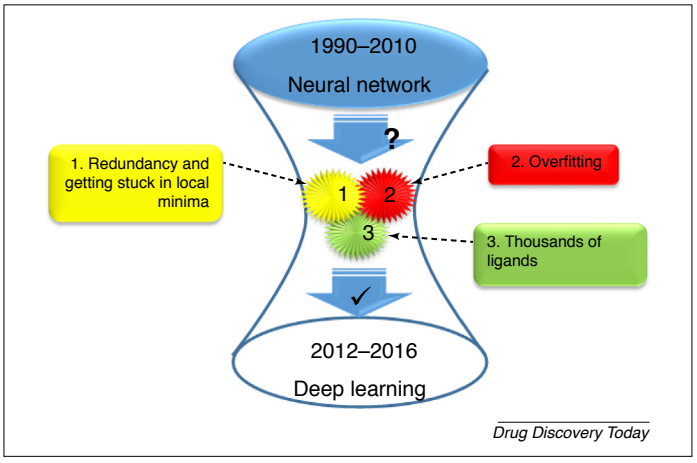

FIGURE 1

The main problems involved in using neural networks, which have led to the advent of deep learning in drug discovery.

Feature selection algorithms in QSAR studies

Feature selection is concerned with extracting the best information from an input database through data mapping in another space or discovering the most effective subset of features. Feature selection also serves to prepare the most applicable network input variables based on an input database. Furthermore, selecting an appropriate technique is crucial when the input database contains irrelevant and redundant information, although the risk of overfitting will be reduced. There are three feature selection categories that can be applied in QSAR: (i) filter; (ii) wrapper; and (iii) hybrid or embedded (Fig. 1).

Filter method

Filter methods have been used to reduce the dimensions of input data, independently of supervised algorithms, and based only on the intrinsic properties of data. In these methods, features are scored and low-scoring features are removed [48]. As a whole, filter methods can be separated into two main categories: (i) feature extraction, which is used to extract the best descriptors from among thousands; and (ii) feature reduction, which maps the input data from a higher-level dimension in another space with lower dimension. PCA, F-test, mutual information (MI), Fisher scoring, and Kolmogorov–Smirnov statistics (KS) are major filter techniques used in QSAR studies [49,50]. Simplicity, increased speed and reductions in computational time are among the advantages of filter methods, whereas the lack of interaction with classification is the main disadvantage, which sometimes leads to worse performance compared with other techniques [51].

Wrapper technique

Wrapper techniques are used to select the optimum subset of features based on the error reduction of classifier algorithms.

TABLE 1

Predicting performance between NNs and DL algorithms

Approach	Uses	Advantages	Drawbacks	Solution
Shallow NNs	For a limited number of compounds in VS	Simple computational learning	Perform poorly in HTVS	Using feature selection algorithms
		Significantly reduced training time	Getting stuck in local minima because of complex and nonlinear relationship between descriptors	Fine-tuning network initial parameters
		Slow to achieve desired results	Redundancy problems because of existence of thousands of descriptors	
		Potential tool for routine tasks of QSAR analysis	Random selection of initial parameter	
			Pruning to overfitting problem	
			Vanishing gradient in initial layers for BP network	
DL networks	In HTVS with thousands of molecules and descriptors	Capturing complex features of molecules	Redundancy problem for in-house databases	Using GPU or CPU cluster to parallel input data and reduce time required
		Facilitates drug discovery procedure	Time-consuming	
		Prevents overfitting problem for large number of inputs	Complex learning algorithm	
		Improves model performance		

Wrapper methods perform better than filter methods, but are more expensive and time-consuming. Overall, based on the algorithm strategy, wrapper approaches can be divided into deterministic methods, producing the same outputs based on the unchanged input, and randomized techniques creating different outputs based on the same input. SVM, KNN, forward selection, backward elimination, and stepwise regression are wrapper methods based on the deterministic strategy, whereas randomized strategies include GA, SOM, and RF [52,53]. Expectation maximization algorithm with a sparse prior is another proven technique to select the optimized invaluable descriptors and predict the power of subsequent models to simplify model interpretation [54,55].

Hybrid/embedded approach

Despite the advantages of both of the above approaches, each has drawbacks in dealing with complex data. Thus, it is common to merge filter and wrapper approaches as a hybrid/embedded approach, thus retaining the advantages of both approaches, while reducing their drawbacks [49].

Major neural network algorithms in chemoinformatics

The concept of NN, first proposed by McCulloch and Pitts in 1940, was founded on human brain performance [56]. However, it has two major limitations: slow convergence and unpredictable solutions during training. Several supervised and unsupervised learning algorithms were proposed based on NN that made it a powerful technique with a range of applications in drug discovery (Fig. 2).

All learning network algorithms rely on the feed-forward NN, universal approximation theorem, and a single hidden layer containing a finite number of neurons (Fig. 3). In this model, the minimum number of layers must be three: the input, hidden, and output layers. Molecular descriptors are applied as input neurons with the purpose of predicting biological activity or classification molecules. To compute hidden variables or output values, previous layer neurons are multiplied by their weights. The calculated values are applied to the activation function (Eq. (1)):

$$y_i^l = f\left(b + \sum_j w_j^l y_j^{l-1}\right) \quad (1)$$

where y_i^l is related to the i^{th} unit of l^{th} layer, b and w are bias and weight, respectively, and $f(.)$ is the activation function. Since 1990, different algorithms in computational chemistry research have been proposed based on the NN, the main ones being BP [57], CP [58], and RBFN [59].

Back propagation

BP, a short form of 'backward-propagation of errors', was the first ANN learning method used in drug design, introduced by Aoyama *et al.* [60]. It is commonly used in NN to learn parameters, weights, and biases, based on error derivative computation. To calculate the error, the predicted output must be computed in the same way as the feed-forward NN [61]. The gradient descent method is a conventional optimization method to minimize the error function to achieve the global error. Two major problems arise with the BP algorithm: (i) getting stuck in local minima is a problem when

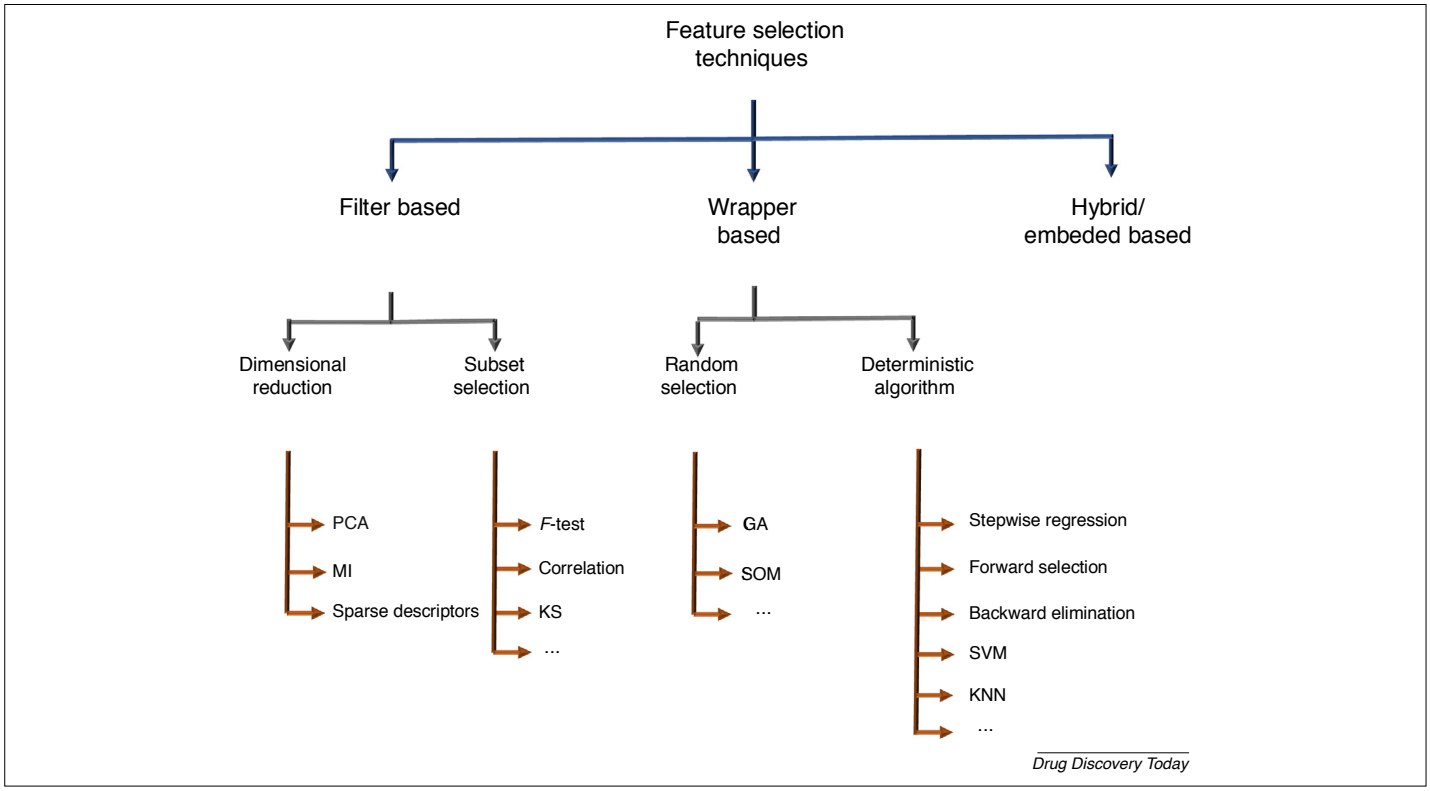

FIGURE 2

The main categories of feature selection algorithms. For definitions of abbreviations, please see the main text.

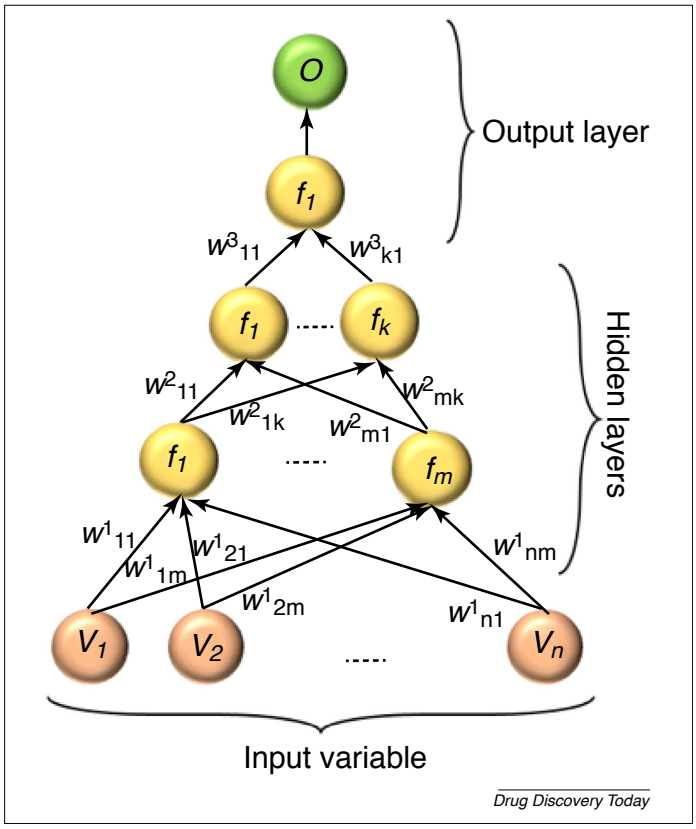

FIGURE 3

The structure of deep neural networks (DNNs) with two hidden layers. Input and output layers are related to the descriptors and predicted biological activity, respectively, of the ligands; f_i indicates the sigmoid function.

the number of network input variables is increased. By contrast, the complex and nonlinear relationship between descriptors introduces many local minima in error surface. For example, the network might be trapped in one of these local minima, and the proximity of local minima to the global minimum has a direct effect on the output of NN. Conventionally, initial network parameters are selected randomly, and getting stuck in local minima occurs as a consequence of the initial selection of the starting condition [62]; and (ii) vanishing gradients in initial layers are another problem when the parameters are updated in the BP step. Moreover, tiny portions of gradient are transmitted to the primary layers [63].

Counter propagation

CP is a kind of hybrid NN in QSAR, which was used for the first time to predict Kovats indices for substituted phenols [47]. The CP network is constructed based on a combination of the Kohenen network and the outstar structure of Grossberg [64]. The Kohenen hidden layer and Grossberg layer are used to determine winning units for input variables and to map the winners into their classes, respectively. By contrast, the learning procedure is based on two different steps: (i) discovering the winner based on the Euclidean distance between input neurons and weights; and (ii) updating the weight of the winner. CP guarantees finding the best weights and performs better than the BP algorithm [65]. The main drawback of CN is that it is a lock-up table method, and so only capable of providing a limited number of different answers.

Radial basis function network

RBFN, a kind of NN, was used to predict boiling points from structural parameters by Lohninger in 1993 [46]. In this network, Gaussian function is used as an activation function of the learned algorithm based on two main steps: (i) selecting the parameter centers, c_i, which are initialized by random values or calculated by the *k*-means clustering algorithm; and (ii) fine-tuning network parameters by means of the BP algorithm. The kernel function is commonly obtained by Gaussian distribution. Simple computational learning and a significantly reduced training time are the most significant advantages of RBFN [66], although the random selection of center points and function weights lead to unsatisfactory results [67].

Proposed deep-learning algorithms in computational chemistry

All the learning algorithms of NN mentioned above contain one or two hidden layers, with a limited number of units in each layer for feature transformation. However, although these methods are useful for solving simple problems, more complicated real-world applications (e.g., a large number of descriptors and the nonlinear relationship between them in drug discovery) has led researchers to use DL algorithms rather than shallow learning methods [68]. The main DL techniques proposed in QSAR studies are DNN (NN, with more than two layers and large number of neuron in each layer), DBN-DNN, CNN, and dropout.

Deep belief networks

DBNs were the first proposed algorithms in deep architecture to fine-tune network initial parameters, weights, and biases, rather than using the NN with random variables, especially in the case of the BP algorithm. Therefore, they are an appropriate approach to use to avoid overfitting problem. DBN is a kind of generative unsupervised learning algorithm, in which higher level features are constructed by lower level ones and comprise *l* stacks of RBM. RBM is an energy-based method used as a discriminative or generative model for labeled or unlabeled data with a single layer of hidden units and no internal layer of visible and hidden neurons [14].

Autoencoder

The same as the DBN, an autoencoder is a kind of unsupervised learning algorithm applied layer by layer. The autoencoder structure is based on two main parts, one of which encodes input data and the other decodes the hidden units to reconstruct input data again. Thus, the number of output units should be equal to the number of input variables. The training algorithm of autoencoder is used layer by layer, the same as RBM, and is widely used as a building block to pretrain the deep NN. Hence, it is appropriate to reduce redundancy problems, such as 'getting stuck' in local minima as well as improving model performance [69–72].

Convolutional neural network

CNN is a kind of artificial NN comprising neurons that have learnable weights and biases. In an in-house database with a small number of descriptors, using the fully connected network is feasible, but in a large data sets with thousands of descriptors (e.g., ChEMBL database) and network parameters (e.g., weights) the

network would rapidly lead to overfitting. Therefore, with the intention of reducing redundancy problem, CNN was proposed.

Unlike a feed-forward NN, which uses only fully connected layers, CNN architectures have three different layers: the convolutional layer, pooling layer, and fully connected layer. The convolutional layers, ConvNet, used to extract feature from input neurons, have K filters (or kernels) in which the size is smaller than the dimension of the image. By contrast, ConvNet can compute the output of neurons connected to local regions in the input. Each map is then subsampled or downsampled, typically with average, sum, or max pooling over p × p neighbor pixels, in which p changes between 2 for small input data (e.g., MINIST images) and 5 for larger inputs. The fully connected layer is the same as the feed-forward NN layers [15,73].

Dropout

Dropout is another popular and effective technique for improving the generalization error of large NNs and as a result minimizing overfitting in QSAR studies. Unlike autoencoder and DBN, which are used as unsupervised methods for fine-tuning network initial parameters, dropout is used as a supervised network with end-to-end BP [16,20]. The key point in dropout is discarding random units from visible and hidden layers during training. These neurons are temporarily removed from the network. As described in the previous section, in normal feed-forward NN, input variables are multiplied by their weights and summed by their biases. After that, the hidden variables are computed by applying the activation function to the calculated values. This procedure is repeated for other layers. With dropout, before starting the training algorithm, each input variable is multiplied the Bernoulli dropout probability. For hidden layers, these steps are repeated [74].

Concluding remarks

Here, we have provided a comprehensive review of literature dealing with NN and DL algorithms used in drug discovery, including the merits and drawbacks of each. For decades, different machine learning methods have been applied in QSAR studies, which can be divided into two categories: (i) shallow learning methods, such as NNs, based on the universal approximation theorem (the network using single hidden layer containing of the finite number of neurons); and (ii) DL algorithms. Briefly, in predicting biological activity, there are two major issues that must be taken into account: HTVS based on thousands of molecules, and the limited number of compounds, usually <1000 molecules.

Regarding the first issue, detecting drug candidates from large chemical libraries with thousands of descriptors requires a computational model with a large number of elements. Thus, QSAR focuses on finding the best model based on deep architecture to avoid getting stuck in local minima and being prone to overfitting. In other words, by using DL networks, practical ways to facilitate drug discovery procedures can be identified. By contrast, capturing complex features of molecules is the other advantage of DNN over other shallow ML methods. However, DL algorithms are naturally complex and time-consuming, such that GPU or CPU clusters must be used in a parallel procedure. By contrast, despite the advantages of deep architectures for HTVS, the drawbacks involved in using them for in-house databases, including redundancy, has led to them being replaced by shallow learning networks (e.g., BP).

Given the major problems with redundancy and overfitting in the second issue, a limited number of compounds and a large number of descriptors, NNs appear to be more suitable. To avoid the redundancy problem, a reduction in the number of molecule features by using data-mining approaches is essential. However, if the network parameters are fine-tuned before starting the learning procedure and regularization techniques are used (e.g., ReLU and dropout in DL algorithms, and Bayesian methods in shallow NNs), it might be possible to avoid the consequences of overfitting.

We conclude that both NNs and DL have a role in the future of drug discovery. Additionally, it appears that combining both approaches will be more effective in the search for the best drug candidates from libraries with large numbers of biochemical compounds.

Acknowledgment

This project was supported by the Vice Chancellery of Research, Isfahan University of Medical Sciences and by a grant from the Spanish Ministry of Economy and Competitiveness (CTQ2017-87974-R).

References

1 Consonni, V. *et al.* (2002) Structure/response correlations and similarity/diversity analysis by GETAWAY descriptors. 1. Theory of the novel 3D molecular descriptors. *J. Chem. Inf. Comput. Sci.* 42, 682–692
2 (2009) The influence relevance voter: an accurate and interpretable virtual high throughput screening method. *J. Chem. Inf. Model.* 49 (4), 756–766
3 Dahl, G.E. *et al.* (2014) Multi-task neural networks for QSAR predictions. *arXiv* 2014 14061231
4 Lowe, R. *et al.* (2011) Classifying molecules using a sparse probabilistic kernel binary classifier. *J. Chem. Inf. Model.* 51, 1539–1544
5 Erić, S. *et al.* (2012) Prediction of aqueous solubility of drug-like molecules using a novel algorithm for automatic adjustment of relative importance of descriptors implemented in counter-propagation artificial neural networks. *Int. J. Pharm.* 437, 232–241
6 Hiller, S. *et al.* (1973) Cybernetic methods of drug design. I. Statement of the problem—the perceptron approach. *Comput. Biomed. Res.* 6, 411–421
7 Aoyama, T. and Ichikawa, H. (1991) Reconstruction of weight matrices in neural networks—a method of correlating outputs with inputs. *Chem. Pharm. Bull.* 39, 1222–1228
8 Rose, V.S. *et al.* (1991) An application of unsupervised neural network methodology Kohenen topology-preserving mapping to QSAR analysis. *Quant. Struct. Act. Relat.* 10, 6–15
9 Bradbury, S.P. (1994) Predicting modes of toxic action from chemical structure: an overview. *SAR QSAR Environ. Res.* 2, 89–104
10 van Nostrum, C.F. *et al.* (1995) Supramolecular structure, physical properties, and Langmuir Blodgett film formation of an optically active liquid-crystalline phthalocyanine. *Chemistry* 1, 171–182
11 Schneider, G. (2000) Neural networks are useful tools for drug design. *Neural Netw.* 13, 15–16
12 Hinton, G.E. *et al.* (2006) A fast learning algorithm for deep belief nets. *Neural Comput.* 18, 1527–1554
13 Bengio, Y. (2009) Learning deep architectures for AI. *Found. Trends Mach. Learn.* 2, 1–127
14 Erhan, D. *et al.* (2010) Why does unsupervised pre-training help deep learning? *J. Mach. Learn. Res.* 11, 625–660
15 Krizhevsky, A. *et al.* (2012) Imagenet classification with deep convolutional neural networks. *Adv. Neural Inf. Process. Syst.* 2012, 1097–1105

16 Hinton, G.E. *et al.* (2012) Improving neural networks by preventing co-adaptation of feature detectors. *arXiv* 2012 12070580

17 Bengio, Y. and Courville, A. (2013) Deep learning of representations. In *Handbook on Neural Information Processing* (XXXXX ed), pp. 1–28, Springer

18 Martens, J. (2010) Deep learning via Hessian-free optimization. *Proc. 27th Int. Conf. Machine Learn.* 2010, 735–742

19 Dahl, G.E. *et al.* (2013) Improving deep neural networks for LVCSR using rectified linear units and dropout. *2013 IEEE Int. Conf. Acoustics Speech Signal Process.* http://dx.doi.org/10.1109/ICASSP.2013.6639346

20 (2016) Deep learning in drug discovery. *Mol. Inf.* 35 (1), 3–14

21 Mnih, V. *et al.* (2012) Conditional restricted Boltzmann machines for structured output prediction. *arXiv* 2012 12023748

22 Hughes, T.B. *et al.* (2015) Modeling epoxidation of drug-like molecules with a deep machine learning network. *ACS Cent. Sci.* 1, 168–180

23 Unterthiner, T. *et al.* (2014) *Deep Learning as an Opportunity in Virtual Screening.* NIPS

24 Lusci, A. *et al.* (2013) Deep architectures and deep learning in chemoinformatics: the prediction of aqueous solubility for drug-like molecules. *J. Chem. Inf. Model.* 53, 1563–1575

25 Wang, Y. and Zeng, J. (2013) Predicting drug-target interactions using restricted Boltzmann machines. *Bioinformatics* 29, i126–i134

26 Ma, J. *et al.* (2015) Deep neural nets as a method for quantitative structure–activity relationships. *J. Chem. Inf. Model.* 55, 263–274

27 Tian, K. *et al.* (2016) Boosting compound–protein interaction prediction by deep learning. *Methods* 110, 64–72

28 Ghasemi, F. *et al.* (2017) The role of different sampling methods in improving biological activity prediction using deep belief network. *J. Comp. Chem.* 38, 195–203. https://onlinelibrary.wiley.com/doi/abs/10.1002/jcc.24671

29 Ghasemi, F. *et al.* (2018) Deep neural network in biological activity prediction using deep belief network. *Appl. Soft Comput.* 62, 251–258. https://www.sciencedirect.com/science/article/pii/S1568494617305793

30 Koutsoukas, A. *et al.* (2017) Deep-learning: investigating deep neural networks hyper-parameters and comparison of performance to shallow methods for modeling bioactivity data. *J. Cheminf.* 9, 42

31 Xu, Y. , Ma, J. , Liaw, A. , Sheridan, R.P. and Svetnik, V. (2017) Demystifying multitask deep neural networks for quantitative structure–activity relationships. *J. Chem. Inf. Model* 57 (10), 2490–2504

32 Wen, M. *et al.* (2017) Deep-learning-based drug–target interaction prediction. *J. Proteome Res.* 16, 1401–1409

33 Kadurin, A. *et al.* (2017) The cornucopia of meaningful leads: applying deep adversarial autoencoders for new molecule development in oncology. *Oncotarget* 8, 10883

34 Zhang, L. *et al.* (2017) From machine learning to deep learning: progress in machine intelligence for rational drug discovery. *Drug Discov. Today* 22, 1680–1685. https://www.sciencedirect.com/science/article/pii/S1359644616304366

35 Winkler, D.A. and Le, T.C. (2017) Performance of deep and shallow neural networks, the Universal Approximation Theorem, Activity Cliffs, and QSAR. *Mol. Inf.* 36 (1–2), https://onlinelibrary.wiley.com/doi/abs/10.1002/minf.201600118

36 Gasteiger, J. *et al.* (2003) Neural networks as data mining tools in drug design. *J. Phys. Org. Chem.* 16, 232–245

37 Terfloth, L. and Gasteiger, J. (2001) Neural networks and genetic algorithms in drug design. *Drug Discov. Today* 6, 102–108

38 Alakari, S.T. (2017) A comparative analysis of data redundancy and execution time between relational and object oriented schema table. *Int. J. Adv. Sci. Eng. Inf. Technol.* 7, 1562–1566

39 Schneider, G. and Wrede, P. (1998) Artificial neural networks for computer-based molecular design. *Prog. Biophys. Mol. Biol.* 70, 175–222

40 Devillers, J. (1996) *Neural Networks in QSAR and Drug Design.* Academic Press

41 Miller, A. (2002) *Subset Selection in Regression.* CRC Press

42 Turner, D.B. and Willett, P. (2000) Evaluation of the EVA descriptor for QSAR studies: 3. The use of a genetic algorithm to search for models with enhanced predictive properties (EVA_GA). *J. Comput. Aided Mol. Des.* 14, 1–21

43 Burden, F.R. and Winkler, D.A. (2009) Optimal sparse descriptor selection for QSAR using Bayesian methods. *Mol. Inf.* 28, 645–653

44 Shen, Q. *et al.* (2004) Hybridized particle swarm algorithm for adaptive structure training of multilayer feed-forward neural network: QSAR studies of bioactivity of organic compounds. *J. Comput. Chem.* 25, 1726–1735

45 Shen, J. *et al.* (2014) A genetic algorithm-back propagation artificial neural network model to quantify the affinity of flavonoids toward P-glycoprotein. *Comb. Chem. High Throughput Screen.* 17, 162–172

46 Lohninger, H. (1993) Evaluation of neural networks based on radial basis functions and their application to the prediction of boiling points from structural parameters. *J. Chem. Inf. Comput. Sci.* 33, 736–744

47 Peterson, K.L. (1992) Counter-propagation neural networks in the modeling and prediction of Kovats indexes for substituted phenols. *Anal. Chem.* 64, 379–386

48 Khan, A.U. (2016) Descriptors and their selection methods in QSAR analysis: paradigm for drug design. *Drug Discov. Today* 21, 1291–1302

49 Goodarzi, M. *et al.* (2012) Feature selection methods in QSAR studies. *J. AOAC Int.* 95, 636–651

50 Shahlaei, M. (2013) Descriptor selection methods in quantitative structure–activity relationship studies: a review study. *Chem. Rev.* 113, 8093–8103

51 Saeys, Y. *et al.* (2007) A review of feature selection techniques in bioinformatics. *Bioinformatics* 23, 2507–2517

52 Yousefinejad, S. and Hemmateenejad, B. (2015) Chemometrics tools in QSAR/QSPR studies: a historical perspective. *Chemom. Intell. Lab. Syst.* 149, 177–204

53 Shahlaei, M. *et al.* (2012) Application of an expert system based on Genetic Algorithm–Adaptive Neuro-Fuzzy Inference System (GA–ANFIS) in QSAR of cathepsin K inhibitors. *Expert Syst. Appl.* 39, 6182–6191

54 Tarasova, A. *et al.* (2010) Robust modelling of solubility in supercritical carbon dioxide using Bayesian methods. *J. Mol. Graph. Model.* 28, 593–597

55 Burden, F.R. and Winkler, D.A. (2009) An optimal self-pruning neural network and nonlinear descriptor selection in QSAR. *Mol. Inf.* 28, 1092–1097

56 McCulloch, W.S. and Pitts, W. (1943) A logical calculus of the ideas immanent in nervous activity. *Bull. Math. Biophys.* 5, 115–133

57 Dearden, J.C. and Rowe, P.H. (2015) Use of artificial neural networks in the QSAR prediction of physicochemical properties and toxicities for REACH legislation. *Artif. Neural Netw.* 65–88. ZZ https://link.springer.com/protocol/10.1007%2F978-1-4939-2239-0_5

58 Ballabio, D. *et al.* (2011) Genetic algorithms for architecture optimisation of counter-propagation artificial neural networks. *Chemom. Intell. Lab. Syst.* 105, 56–64

59 Shahlaei, M. *et al.* (2012) QSAR study of some CCR5 antagonists as anti-HIV agents using radial basis function neural network and general regression neural network on the basis of principal components. *Med. Chem. Res.* 21, 3246–3262

60 Aoyama, T. *et al.* (1990) Neural networks applied to pharmaceutical problems. III. Neural networks applied to quantitative structure-activity relationship (QSAR) analysis. *J. Med. Chem.* 33, 2583–2590

61 Sun, H. (2005) A naive Bayes classifier for prediction of multidrug resistance reversal activity on the basis of atom typing. *J. Med. Chem.* 48, 4031–4039

62 Suresh, H. and Puttamadappa, C. (2008) Removal of EMG and ECG artifacts from EEG based on real time recurrent learning algorithm. *Int. J. Phys. Sci.* 3, 120–125

63 Sutskever, I. *et al.* (2013) On the importance of initialization and momentum in deep learning. *Proc. 30th Int. Conf. Machine Learn.* 28 III-1139–III-1147

64 Hecht-Nielsen, R. (1988) Applications of counterpropagation networks. *Neural Netw.* 1, 131–139

65 Wu, C. and Shivakumar, S. (1994) Back-propagation and counter-propagation neural networks for phylogenetic classification of ribosomal RNA sequences. *Nucleic Acids Res.* 22, 4291–4299

66 Schilling, R.J. *et al.* (2001) Approximation of nonlinear systems with radial basis function neural networks. *IEEE Trans. Neural Netw.* 12, 1–15

67 Chen, S. *et al.* (1991) Orthogonal least squares learning algorithm for radial basis function networks. *IEEE Trans. Neural Netw.* 2, 302–309

68 Deng, L. and Yu, D. (2014) Deep learning. *Signal Process.* 7, 3–4

69 Bengio, Y. *et al.* (2007) Greedy layer-wise training of deep networks. *Adv. Neural Inf. Process. Syst.* 19, 153

70 Deng, L. *et al.* (2010) *Binary Coding of Speech Spectrograms Using A Deep Auto-Encoder.* International Speech Communication Association

71 Gómez-Bombarelli, R. *et al.* (2018) Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.* 4, 268–276

72 Blaschke, T. *et al.* (2018) Application of generative autoencoder in de novo molecular design. *Mol. Inf.* 37, XXX–YYY https://dl.acm.org/citation.cfm?id=3043064

73 Kalchbrenner, N. *et al.* (2014) A convolutional neural network for modelling sentences. *arXiv* 2014 14042188

74 Srivastava, N. *et al.* (2014) Dropout: a simple way to prevent neural networks from overfitting. *J. Mach. Learn. Res.* 15, 1929–1958

75 Wallach, I. Dzamba, M. and Heifets, A. (2015) AtomNet: A deep convolutional neural network for bioactivity prediction in structure-based drug discovery. arXiv preprint arXiv. 1510.02855.