Artificial intelligence methods to repurpose and discover new drugs to fight the Coronavirus disease-2019 pandemic

Marcos V.S. Santana and Floriano P. Silva-Jr

LaBECFar—Laboratory of Computational and Experimental Biochemistry of Drugs, Oswaldo Cruz Institute, Oswaldo Cruz Foundation - FIOCRUZ, Rio de Janeiro, Brazil

24.1 Introduction

"What we want is a machine that can learn from experience." Alan Turing said such words in a public lecture about computer intelligence in 1947. One can say Turing and other pioneers foresaw the innovations that would come years later, introducing the notion of machines that could learn how to do a task by extracting information from the environment.

Artificial intelligence (AI) is widespread in the modern world, from voice and image recognition software in phones to video games, social media, and self-driving cars. The development of more powerful computers and the availability of data (Big data) made it possible for AI to become a fundamental part of many research fields, including computer vision, natural language processing, and analysis of medical diagnosis. Recently, AI started to make a bigger impact in the drug discovery process, being incorporated into pharma's industry pipelines and hundreds of biotechnology start-ups in the last 10 years. For instance, in 2020 the first AI-designed drug, developed by the UK-based company Exscientia, reached clinical trials (Burki, 2020).

In this chapter, we review the contributions of AI to repurpose drugs in the context of the ongoing Coronavirus disease-2019 (COVID-19) pandemic caused by severe acute respiratory syndrome-Coronavirus-2 (SARS-CoV-2). Despite the availability of vaccines as of December 2020, the chemical arsenal to tackle COVID-19 is still lacking in drugs.

This chapter will also cover selected studies showing different AI strategies to find approved or investigational drugs that could be used against SARS-CoV-2.

24.2 Artificial intelligence in drug discovery

AI is the study of agents that can learn to perform a task (e.g., classifying images belonging to one or more classes) by using information about the environment. When used in drug discovery, AI can come in many forms, including estimation of the affinity of small molecules against a macromolecular target, prediction of binding poses in docking simulations, de novo design of bioactive molecules, and generation of three-dimensional structures of proteins. In drug discovery, the main applications of AI consisting of the machine and deep learning (DL) algorithms that can learn useful information directly from data.

24.2.1 Machine learning overview

Machine learning (ML) is a field of AI that focuses on algorithms and models that can learn patterns from the data to solve a problem (Butler et al., 2018; Lo et al., 2018; Samuel, 1959; Valletta et al., 2017). The origins of ML in drug discovery can be traced back to early computational methods to identify bioactive molecules. For instance, molecular descriptors (e.g., molecular weight, lipophilicity, and structural fingerprints) are used to develop quantitative structure—activity relationship models (Lo et al., 2018). Another example is empirical scoring functions, which predict the binding affinity of ligands in docking simulations by adjusting experimentally calculated parameters (Ashtawy & Mahapatra, 2018; Guedes et al., 2014). Besides the applications mentioned above, ML can also be used to estimate an array of relevant quantitative parameters in drug discovery, including solubility, toxicity prediction, and plasma membrane permeability (Boobier et al., 2020; Feinberg et al., 2018; Gardiner et al., 2020; Mayr et al., 2016).

Another type of problem very common in drug discovery consists of predicting discrete or categorical values for a given task. For example, ML models can be used to predict if a given molecule is active or not in a particular protein target or if it elicits or not some toxicity endpoint (Jiménez-Luna et al., 2021; Lee & Kim, 2019; Li et al., 2020; Robinson et al., 2020; Zhang et al., 2017). Thus ML addresses both quantitative and qualitative problems. What defines which strategy to use is the available data.

Based on the type of chemical or biological data, we can classify the learning paradigm as supervised or unsupervised. The former is used when the dataset has a set of descriptors (i.e., molecular properties, fingerprints, atom types) and an associated label or response (e.g., bioactivity on a protein target, toxicity endpoint) for each data point. On the other hand, unsupervised learning deals with problems that have no explicit response associated with the data. In the unsupervised setting, the goal is to identify patterns in the data to perform tasks such as clustering of similar molecules in N groups according to the distance between them.

Supervised and unsupervised learning are the most common methods used in ML. However, other learning paradigms are also popular. For instance, semisupervised learning is halfway between supervised and unsupervised and consists of training models with

only a few labeled data points. An example of this kind of model is generative chemical models that can create new chemical matter (Bjerrum, 2017; Gupta et al., 2018; Moret et al., 2020; Santana & Silva-Jr, 2021). In reinforcement learning, the model or agent is trained to maximize a reward function; for example, to generate molecules with better binding affinity on a protein target (Blaschke et al., n.d.; Olivecrona et al., 2017; Zhou et al., 2020).

24.2.2 From machine learning to deep learning

DL is a subfield of ML that uses multilayered neural networks to learn representations from raw data to solve a problem (Esteva et al., 2019; Zhou et al., 2020; Zou et al., 2019). The foundations of DL were established decades ago, with the development of algorithms such as stochastic gradient descent (Bottou et al., 2018), the Perceptron model in the 1950s (Rosenblatt, 1958), and the backpropagation algorithm in 1986 (Rumelhart et al., 1995; Rumelhart et al., 1986).

Despite the basic algorithms used in DL being known for decades, only recently have DL models started to be widely implemented in practice, including in the drug discovery process. This surge in DL research was stimulated by the huge amounts of data available today and the ease of collecting and storing these in databases. For instance, chemical and biological databases such as ChEMBL (Bento et al., 2014) and PubChem (Kim et al., 2016) contain bioactivity data for millions of molecules across a range of Proteomes. Besides the data availability, the development of new hardware, such as the graphics processing unit, which were not available before the 21st century also made DL popular by making models extremely parallelizable and faster to train (Chen et al., 2018; Eraslan et al., 2019). In addition, there is opensource DL software available that is user-friendly, allowing models to be built and trained with just a few lines of code, such as Keras (Chollet & Others, 2015), Tensorflow (Abadi et al., 2016), PyTorch (Paszke et al., 2019, 2017), and Fastai (Howard & Fastai, 2018).

DL is becoming extremely popular in different research fields, with several successful applications, for instance in image recognition (Liang et al., 2017; Voulodimos et al., 2018), natural language processing (Alshemali & Kalita, 2020; Howard & Ruder, 2018; Vaswani et al., 2017; Young et al., 2018), and also computational chemistry (Gawehn et al., 2016; Goh et al., 2017; Shen et al., 2020; Walters & Murcko, 2020).

Since DL is a subfield of ML, it is necessary to highlight the difference between them. A common problem with ML methods is how to represent the dataset to train a model successfully. Broadly speaking, in ML it might be necessary to introduce expert knowledge to design an optimal set of features to train a model; for example, by combining the raw features into more informative types. Regardless of the feature engineering method, these transformations often require good data engineering skills as well as expertise in the specific research area.(Chen et al., 2018; Esteva et al., 2019). The main advantage of DL algorithms is their ability to automatically extract new descriptors from raw input data (Goh et al., 2017; Jiménez-Luna et al., 2021; Mater & Coote, 2019).

24.2.2.1 Neural networks

The most basic DL model consists of hierarchically organized layers of perceptrons or neurons, which perform nonlinear transformations on their inputs (Fig. 24.1). This architecture is also called a fully connected neural network (FCNN).
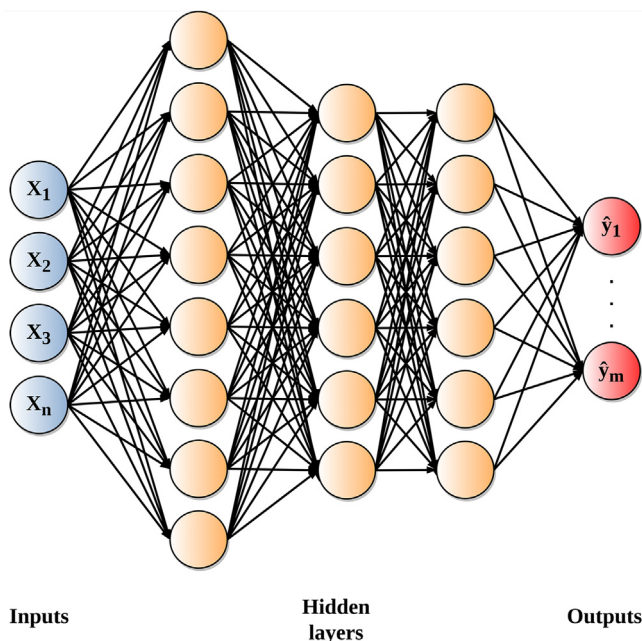
FIGURE 24.1   Fully connected neural network. Inputs are shown as *blue* circles on the left side of the image and outputs as *red* circles on the right side. The hidden layers are positioned between inputs and outputs.

Inputs   **Hidden layers**   **Outputs**

Each layer receives them as input the output of the previous layer. Given an input matrix of descriptors X, a neural network performs the mathematical transformation:

$$y = f\left(\sum_{i=1}^{N} w_i \odot x_i + b\right)$$

where y is the output, f is a nonlinear function (e.g., softmax, sigmoid, tanh, cosine, reLu), w_i the weights of a neuron, x_i represent input features to the neuron and b is a bias term. Thus in essence, a neural network consists of a set of nonlinear transformations on the input data parameterized by a set of weights. Different architectures implement these operations in specific ways, but the general process remains the same.

The nonlinear transformations on the neurons allow the neural network to learn very abstract concepts about the data (Goh et al., 2017). For instance, in computer vision, the earlier layers might learn basic shapes, such as corners in the image. As the data flows through the layers, more complex patterns start to appear, including the distinction between entities in the image (e.g., eyes, noses, different people) (Yosinski et al., 2015). In the output layer, the neural network gathers all the learned features to make a prediction. It is this property of transforming the original data into new representations that make DL models powerful and bypass the need for feature engineering. A neural network with a sufficient number of layers can approximate any abstract complex function (Cybenko, 1989; Hornik, 1991), such as those governing the interaction of proteins with small molecules.

24.2.2.2 *Examples of neural networks architectures for computational drug discovery*

Neural networks come in different flavors and the choice of what kind to use depends on the specific problem. Problems involving text data require different considerations from image data in terms of how to train the model and how the learning process works. In the next section, we will briefly review some of the most used architectures and methods that are being used to repurpose drugs for COVID-19.

24.2.2.2.1 Recurrent neural networks

The basic neural network consists of neurons of one layer connected to every neuron of the next layer. Despite the flexibility of this architecture to learn complex functions, when the problem involves sequence data the model struggles to learn the relationships between different steps of the sequence. For instance, consider the SMILES representing acetylsalicylic acid (i.e., CC(=O)OC1=CC=CC=C1C(=O)O) as input to a FCNN. As each character of the sequence flows through the layers of the network, any information about previous characters would be lost because the model does not have a *memory* of the past.

To use sequence data to solve computational chemistry problems, a common strategy consists of taking every character of the sequence as input to predict the next character. Now, the model can output the probability of sampling the next character in the sequence from a pool of individual characters available, also called a *vocabulary* (Olivecrona et al., 2017). This tweak is similar to introducing a loop during training (Fig. 24.2). The main effect is that every character is treated in the information context of characters preceding it.

In practice, basic recurrent neural networks (RNNs) maintain an internal hidden state h, which holds information about the past of the sequence. At each time step, the state is updated to predict the next character. Thus RNN's can be used for sequences of variable length to learn how each character of the sequence relates to each other.

The architecture described above is an improvement over FCNN's, but it still suffers from learning problems (Lipton et al., 2015). The main problem of basic RNN is training this kind of model. As the input moves through the layers it gets multiplied by the weights and the gradient is calculated. If the gradient is too small or too large, it will vanish or explode, respectively, as training progresses and impair learning (Lipton et al., 2015; Yu et al., 2019). Therefore basic RNNs layers are rarely used in practice today and two
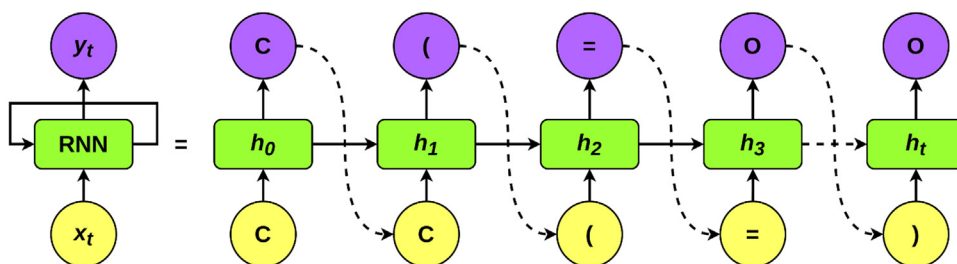

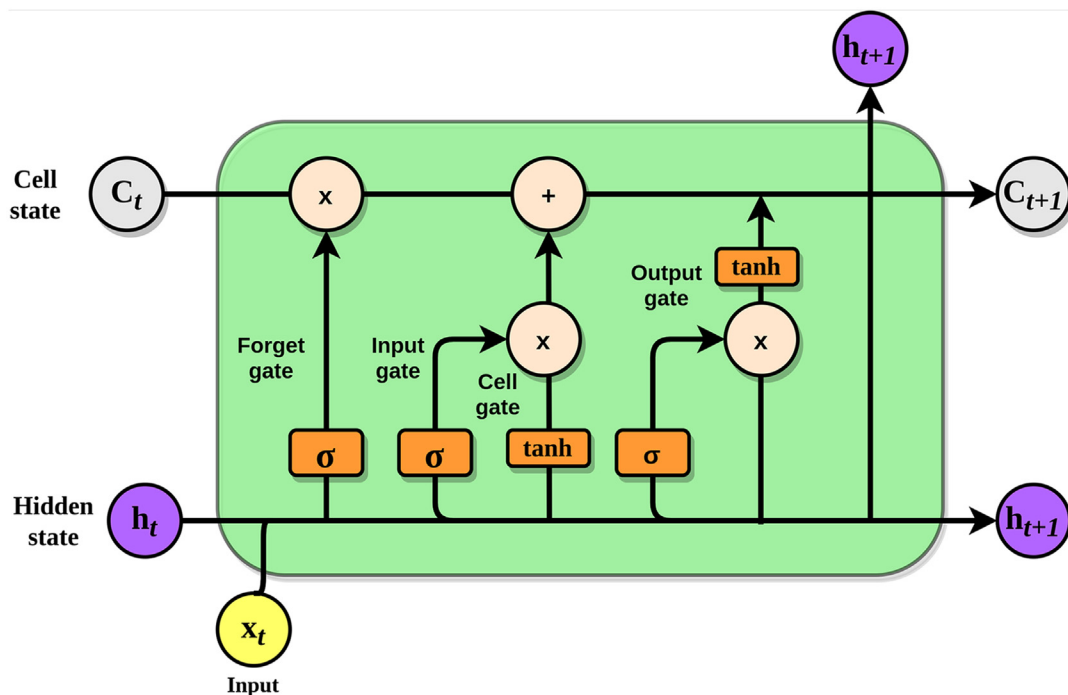**FIGURE 24.2** Recurrent neural network layer.

FIGURE 24.3   The long short-term memory unit. The inputs *xt* and hidden states *ht* are concatenated and used to update the cell state (*Ct*). The different gates in the LSTM control what information will be updated, forgotten, and output by the neural network. σ, sigmoid function; tanh, tangent function.

types of modifications are very popular: long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and gated recurrent neural network (GRU) (Cho et al., 2014).

The LSTM architecture was introduced by Hochreiter and Schimidhuber in 1997. This architecture introduced another hidden state, called cell state C_t, which gathers information from everything that happened to the sequence. In an LSTM layer, four gates modulate what will be forgotten or kept by the cell state at each time step (Fig. 24.3).

1. *Forget gate*: Concatenates the input and the hidden state of the previous time step and passes the result to a sigmoid function. The output of the sigmoid operation is a matrix of 0's and 1's, values closer to 0 are discarded, and values close to 1 are kept.
2. *Input and cell gate*: Update the cell state from the previous time step. This is useful to make the model focus on new information that can be used to solve a task.
3. *Output gate*: It determines which information of the updated cell state will actually be used for output.

24.2.2.2.2 Transformer and attention mechanism

In addition to recurrent-based networks, the transformer architecture is also widely used. The transformer is a DL architecture introduced in a seminal paper (Alshemali & Kalita, 2020; Howard & Ruder, 2018; Vaswani et al., 2017; Young et al., 2018) that uses an *attention mechanism* to focus on the most relevant parts of a sequence to predict each time step.

When dealing with sequence-to-sequence problems (seq2seq), where both input and output are sequences, a common architecture consists of two parts, an encoder, and a decoder. Given an input sequence $X = (x_2, x_2, \ldots, x_n)$, the encoder maps the sequence into an abstract representation $Z = (z_1, z_2, \ldots, z_n)$. Then, the decoder receives the modified input and transforms it into the output sequence $Y = (y_1, y_2, \ldots, y_n)$. This is a common approach when solving translation problems and is also being applied to computational chemistry problems to predict synthetic routes, generate new molecules and predict bioactivity.

As previously mentioned, LSTMs are a natural choice to work with sequence data. However, in 2017 Vaswani et al., introduced a new architecture that relies entirely on self-attention mechanisms to solve seq2seq problems. The attention mechanism consists of using key information about the sequence to make predictions (Cheng et al., 2016; Lin et al., 2017). Thus a model with attention can look into specific parts of a sequence and decide if it is important or not.

In the original transformer, the encoder consists of a stack of six identical layers. Each of these layers has two sublayers: a multihead attention layer followed by a fully connected linear layer. In addition, layer normalization and residual connection between the sublayers were introduced, allowing the data to flow through undisturbed (He et al., n.d.). The decoder follows a similar structure, but with the addition of another multihead attention (encoder attention) layer connected to the output of the encoder (Fig. 24.4).

To assign attention to different parts of a sequence, the transformer uses a set of matrix multiplications between query (Q), key (K) and value (V) matrices according to:

$$Attention\ (Q, K, V) = softmax \left(\frac{QK^t}{\sqrt{d_k}} \right) V$$

where softmax is a function to scale values between 0 and 1, and d_k is the dimension of the input embedding. The Q, K, and V matrices are generated for each word or character in the input embeddings. During training, these matrices are multiplied together to generate weighted vectors (i.e., self-attention) representing the relative importance of each word during inference. In practice, self-attention is calculated multiple times using the multihead mechanism, allowing Q and V to be represented in different ways. In the multihead layer, the different heads are concatenated into a single matrix and self-attention is calculated jointly for different representations at different positions in the sequence. The authors of the original implementation found that multihead improved the performance compared to a single self-attention calculation.

24.2.2.2.3 Graph-convolutional neural network

Molecules can be represented in different ways, including one-line representations (e.g., SMILES), two-dimensional drawings, and three-dimensional geometries showing the spatial arrangement of atoms. However, under the hood of these methods is the molecular graph, which is how a computer reads the molecular structure. A molecular graph $G = (V, E)$ is defined by a set of atoms or nodes V and edges E representing the bonds between them. In addition to bonds and atoms, the graph can also encode molecular features at its nodes and edges, such as bonds and atoms types, aromaticity, chirality, if the atom is part of a ring, and many others. Thus a molecular graph can represent molecules concisely to input implicit molecular features into ML models.
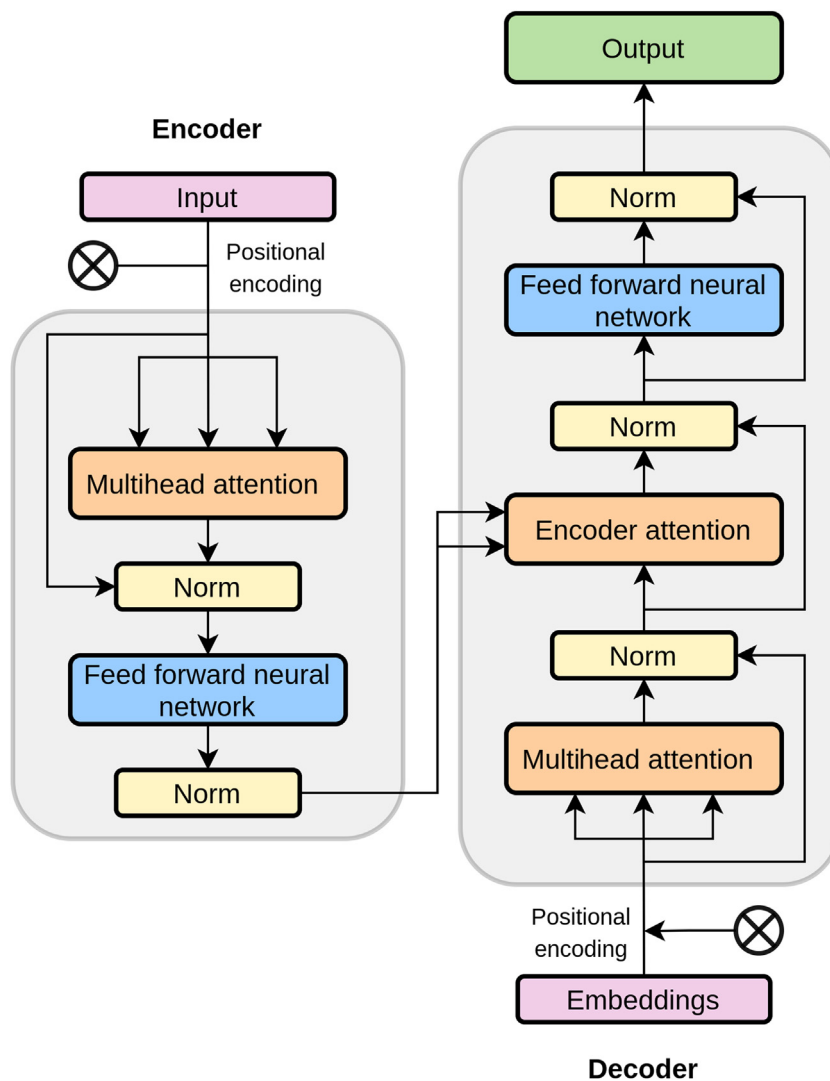
FIGURE 24.4   The transformer architecture shows the encoder and decoder. The encoder consists of multihead attention that allows the model to focus on important parts of the import. The decoder also contains attention layers and processes the context vector output by the encoder. The final layer of the decoder is a fully connected linear layer and is responsible for the output of the model.

Neural network architectures working directly on molecular graphs evolved with time, starting with a publication by Duvenaud et al. (2015) describing a neural-based molecular fingerprint. Further contributions, such as Weave modules and "supernodes" introduced important concepts to train graph-based neural networks. The main concept of the graph neural network is that molecular features can be learned directly from the molecular

structure or graph, instead of using explicit descriptors. The learning process using a graph convolutional neural network (GCNN) has three properties designed to deal with molecular inputs: (1) the order an atom is represented on the molecular graph does not change the output of a layer (order invariance); (2) any permutation applied to individual atoms is also applied to the pairs this atom makes with other atoms in the molecule; (3) atom pairs can be described as **ab** or **ba** (pair invariance) (Kearnes et al., 2016).

Gilmer et al. published a seminal work that generalized the concept of neural networks for self-supervised learning on molecular graphs (Gilmer et al., 2017). The main idea of this work is that the GCNNs on literature at the time could be reformulated into a common framework called message passing neural networks (MPNNs). In the MPNN framework, the neural network receives input graphs consisting of nodes features x_v, edges features e_{vw}, and an adjacency matrix A with vector-valued entries to indicate different bonds in the molecule and the spatial distance for every atom pair. The node or atomic features consisted of very simple descriptors, including atomic number, atom types, acceptor/donor of electrons, aromaticity, hybridization, and the number of hydrogen atoms.

The forward pass of training in the MPNN framework consists of two phases, a message-passing phase, and a readout phase. During the message passing phase, the hidden states at each node are updated according to the functions:

$$m_v^{t+1} = \sum_{w \in N(v)} M_t(h_v^t, h_w^t, e_{vw})$$

$$h_v^{t+1} = U_t(h_v^t, m_v^{t+1})$$

where, M_t and U_t are the message passing and update functions, respectively, h_v^t is the hidden state at time step t for node v and e_{vw} is the edge feature vector between nodes v and w. In this phase, each node "messages" its neighbors and aggregates information to update its own context. After a few iterations of messaging, a readout phase computes a graph-level (or molecule-level) feature vector using a Readout function R on the learned features of all nodes:

$$y = R(\{h_v^T | v \in G\})$$

where the readout function R is

$$R = \sum_{v \in V} \sigma\left(i(h_v^T, h_v^0)\right) \odot \left(j(h_w^T)\right)$$

where i and j are neural networks and $\odot$ denote element-wise multiplication. The commutative property of the element-wise multiplication on the readout function is essential for the network to deal with graph isomorphism and be invariant to the permutation of atoms or nodes on the graph. In the final phase, the learned graph-feature vector is passed to linear layers for prediction tasks.

The authors demonstrated how previous work could be cast into the MPNN framework, showing the respective message passing and update functions for each model, thus aggregating much of the knowledge about molecular graph neural networks into a more elegant form that could be explored in future research (Gilmer et al., 2017).

24.2.2.2.4 Knowledge graphs

Despite not being a neural network, knowledge graphs are gaining momentum in AI, especially when combined with ML and DL models. A knowledge graph represents knowledge using a graph structure, where each node is an entity or object and the edges between them represent what kind of relationship is present (Bullock et al., n.d.; Kejriwal, n.d.; Paulheim & Cimiano, 2016). For example, a knowledge graph of the human proteome could include all druggable proteins and drugs as nodes and the type of bioactivity (e.g., inhibitor, substrate, antagonist, agonist, etc.) as edges linking drugs to targets.

In the current COVID-19 pandemic, different groups (see Section 24.3) are using knowledge graphs to extract hidden relations between approved drugs and SARS-CoV-2 and host proteins that could be targeted. For instance, biomedical information from databases such as PubMed, PubChem, and ChEMBL can be mined to identify molecules already tested against relevant targets. In this scenario, neural networks can be trained using the information from a knowledge graph to make predictions for new drugs and targets, accelerating the discovery of new treatments (Ge et al., n.d.; Richardson et al., 2020; Zeng et al., 2020).

24.3 Selected drug repurposing strategies

Several computational approaches have been explored to quickly respond to the COVID-19 pandemic and the necessity to find possible treatments. A selection of cases employing AI methods is listed in Table 24.1.

Ge and coworkers developed a data-driven drug repositioning by combining seven networks with information about drug—target and protein—protein interactions, molecule similarity, and sequence similarity of human and viral sources to predict drugs targeting SARS-CoV-2 (Ge et al., n.d.; Richardson et al., 2020; Zeng et al., 2020). The authors used well-known publicly available databases, such as DrugBank (Wishart et al., 2006), ChEMBL (Bento et al., 2014), BindingDB (Gilson et al., 2016), and UniProt. The final knowledge graph was assembled by merging the nodes and edges of the individual networks, where each node represented drugs or targets, while the edges between them described the identified interactions, including similarity (i.e., molecular and primary sequence) and drug-target or target-target interactions. Thus by aggregating the information between nodes the final graph could be used to select an initial list of potential drug candidates that could be used to treat COVID-19.

A graph convolutional network (GCN) was used to learn and extract the hidden information from the knowledge graph, allowing the authors to access novel drug-target and target-target interactions and find molecules that could be repurposed to SARS-CoV-2 (Ge et al., n.d.; Richardson et al., 2020; Zeng et al., 2020). GCN's are powerful neural networks that can access the rich information within the nodes of a graph (Torng & Altman 2019; Zhang et al., 2019) by allowing the nodes to exchange information (or messages), which are updated every iteration of the algorithm.

Given a graph, the nodes can exchange messages m^t during training of a GCNN:

$$m_v^t = \sum_{r \in R} \sum_{u \in N_r(v), \ e = (u,v,r) \in E}$$

TABLE 24.1 Potential repurposable drugs against Coronavirus disease-2019.

Repurposed drug candidates	Previous use	Method	Predicted targets	Experimental validation
Mefuparib hydrochloride (CVL218)	Cancer	Knowledge-graph, docking in vitro and in vivo assays	The nucleocapsid (N protein)	Mefuparib achieved an EC_{50} of 5.12 μM
Clomiphene, Toremifene and Bazedoxifen, Indomethacin, Dexamethasone, Melatonin, Chloroquine, and Multiple targets of human and Coronaviruses	Antiviral activities against	Hydroxychloroquine Coronaviruses have been described for selective estrogen modulators (toremifene) and antiparasitic (chloroquine and	Multiple	Knowledge-graph, docking in vitro and in vivo assays hydroxychloroquine). In addition, antiinflammatory agents may help alleviate the inflammation caused by SARS-CoV-2 infection.
Atazanavir and Efavirens	HIV	Molecule transformer drug-target interaction (MT-DTI)	SARS-CoV-2 3C-like protease (Mpro)	Atazanavir can be found in lungs after administration and have shown inhibitory activity against Mpro [2]
Baricitinib	Rheumatoid arthritis	Knowledge-graph	Human AP2-associated protein kinase 1 (AAK1)	A pilot trial indicated that baricitinib improved clinical parameters (e.g., cough and dyspnea) in a small group of patients[3].
Temsirolimus, Sirolimus, Bremelanotide, Everolimus, Deutetrabenazine, Latamoxef, Cefazolin, Cefoxitin, Bacampicillin, Pheneticillin, Enoxacin, Linezolid and Colchiceine	Multiple	The proprietary DL model for drug-target interaction (Ligand Design™ by Cyclica)	SARS-CoV-2 3C-like protease (Mpro) and Spike protein. TMPRSS2 and Cathepsin B	Sirolimus is currently under clinical trials against COVID-19 (NCT04461340, NCT04341675, NCT04371640, NCT04482712, and NCT04374903). Colchicine is under clinical trial for the treatment of moderate symptomatic COVID-19 patients (NCT04527562).

where $a_{u,v,r}$ represent the weights v, u, and r associated with edge $e = (v, u, r)$. $A_{u,v,r}$ is the aggregation function to gather the messages arriving at a node and is given by

$$A(u, v, r) = \frac{a(u, v, r)}{\sum_u a(u, v, r)}$$

The hidden state h^t for node v at time step t is given by

$$h_v^t = \frac{relu(W^t concat(h_v^{t-1}, m_v^t) + h_v^{t-1} + b^t)}{\left| relu(W^t concat(h_v^{t-1}, m_v^t) + h_v^{t-1} + b^t) \right|_2}$$

The final knowledge graph was mined to gather an initial list of drugs, which was further refined by extracting drugs that could be related with SARS-CoV-2 from 20 million texts from PubMed using a DL method called Biomedical Entity Relation Extraction and manual inspection. Briefly, the authors encoded pairs of drugs and SARS-CoV-2 in a bag of sentences $S(e_1, e_2)$, where each entry corresponded to sentences containing drug e_1 and target e_2. This bag of sentences was then used as input to a DL model containing self-attention modules and gated recurrent units (GRU) to generate a sentence representation h_s that could be used to calculate the contribution of each sentence to the relation prediction:

$$\beta = softmax\ (W \odot h_s)$$

where a softmax function is used to scale β to the range [0, 1], with values closer to 1 indicating more relevant weights. Finally, a binary classifier was used to predict the relation r_{e_1, e_2} drugs and targets:

$$r_{e_1, e_2} = classifier\ (\beta \odot h)$$

The final list of drug candidates was further refined using the connectivity map approach based on transcriptome analysis of 10 SARS-CoV patients. To validate the text extraction approach, the authors conducted a retrospective study on SARS-CoV and Middle East respiratory syndrome-Coronavirus (MERS-CoV), which are two Coronaviruses that are similar to SARS-CoV-2. Interestingly, drugs previously described as active against these viruses were among the top of the list of predicted results, demonstrating that the knowledge graph method can predict drugs with some degree of activity against Coronaviruses.

The in silico process returned the poly-ADP-ribose polymerase 1 (PARP1) inhibitor, Mefuparib hydrochloride (CVL218), currently in Phase I clinical trials, as a potential drug for repurposing against SARS-CoV-2. Inhibition assays showed that CVL218 inhibited SARS-CoV-2 replication by 35.16% at 3 μm, which was higher than the inhibition exhibited by Arbidol (21.73% at 3 μm), one of the standard treatments for COVID-19 in China. In addition, the authors showed that the antiviral activity of CVL218 was dose-dependent, with no obvious cytopathic effects on treated cells. Time-of-addition assays further demonstrated that CVL218 displayed potent inhibitory activity during the course of the assay, with a mild inhibition at entry and significantly inhibiting replication postentry.

In vivo studies showed that CVL218 significantly inhibited the production of IL-6 induced by the CpG oligodeoxynucleotide 1826 (CPG-ODN1826) in peripheral blood mononuclear cells (PMBC), indicating it might be an alternative to treat proinflammatory action by SARS-CoV-2 infection. CVL218 also showed favorable pharmacokinetics and toxicity profiles in rats and monkey models. Taken together, these findings show how combining AI with wet-lab tests can accelerate drug discovery, especially in critical scenarios such as the COVID-19 pandemic.

Zeng and coworkers developed CoV-KGE, a network-based approach to identify potentially repurposable drugs based on the interactions between their human targets and

Coronavirus-associated proteins (Zeng et al., 2020). CoV-KGE is an extensive knowledge graph of 15 million edges including 39 types of interactions (e.g., activation, inhibition, blockage, etc.) connecting drugs, genes/proteins, and diseases. Using this approach, the authors identified 41 drugs that could be tested for the treatment of COVID-19 in clinical trials.

To develop CoV-KGE, the authors gathered drug−gene interactions, gene−gene interactions, drug−disease associations, and gene−disease information from the Global Network of Biomedical Relationships (GNBR). In addition, drug−drug interactions, mechanism of action, pharmacodynamics, side effects, drug anatomical therapeutic chemical (ATC) codes, and toxicity information for 3481 FDA-approved and clinically investigational drugs from DruBank that overlapped with GNBR were also included. Finally, experimentally Coronaviruses−gene relationships and known genes associated with other Coronaviruses are included in the knowledge graph.

A DL model called RotatE was used to learn and extract the embeddings of the knowledge graph, which contained information about interactions between the nodes, such as how a drug interacts with a given gene/protein from Coronaviruses. Briefly, the knowledge graph can be described as triplets (h, r, t), where the head (h) and tail entities (t) interact via some type of relation (r). The RotatE model defines a relation type as a rotation of the head to the tail in a complex vector space. More specifically, given a triplet (h, r, t), the tail can be described as:

$$t = r \odot h$$

where h, r, $t \in C^k$ are the embeddings of the knowledge graph. During training, the loss function is optimized to minimize the distance d_r between positive (existing) relations, while maximizing the distance between negative (nonexisting) relations.

$$d_r(h, t) = |h \odot r - t|$$

$$L = -log\sigma(\gamma - d_r(h, t) - \sum_{i=1}^{n} p(h_i, t, r_i)$$

where σ is the sigmoid function, γ is a margin parameter and h_i, r_i, t_i is the ith negative triplet.

On a retrospective validation using COVID-19 clinical trials data (https://covid19-trials.com/), CoV-KGE displayed high performance, being able to identify 14 types of drugs annotated by their ATC code. For instance, Toremifene, Indomethacin, and Niclosamide were identified among the top drugs, which is consistent with their previously described activity on other Coronaviruses. In total, 41 repurposable drug candidates were selected based on the CoV-KGE predictions and availability of clinical evidence against SARS-CoV-2. The most interesting drug types included selective estrogen receptor modulators (clomiphene, toremifene, and bazedoxifene), antiinflammatory agents (indomethacin, dexamethasone, and melatonin). The antiparasitic chloroquine and hydroxychloroquine were also among the top predicted drugs, but due to their low efficacy in clinical trials and potentially serious side effects (e.g., prolongation of QT), the authors highlight that further experimental results are warranted to explain the discrepancy between their antiviral activity and low efficacy.

Overall, CoV-KGE is a promising A. I approach to accelerate the discovery of known drugs that can be used in critical moments, such as the COVID-19 pandemic. However, the authors highlight some potential limitations, such as noise from different experimental settings in the training data, the lack of pharmacokinetics data, and dose-dependent profiles on SARS-CoV-2.

Beck et al. (2020) used a transformer-based approach to predict the binding affinity of 3410 commercially available antivirals against different protein targets from SARS-CoV-2, including 3CLpro, spike protein (S), RNA-dependent RNA-polymerase (RdRP), helicase, endoRNAse, 3′-to-5′-exonuclease and 2′-O-ribose methyltransferase. The Transformer is a DL architecture that uses a self-attention mechanism to prioritize specific parts of a SMILES sequence during inference; the model can focus on relevant atoms or fragments from a molecule. Therefore the premise of the approach is analogous to understanding texts in different languages, for instance by learning the semantic relationships of words to execute a task, such as predicting the most probable word in a text or the sentiment expressed by it (Lecun et al., 2015; Young et al., 2018). Attention has been extremely successful in NLP, allowing models to achieve state-of-the-art results in language translation, Q&A tasks, sequence classification, and other language-based tasks.

Their model, called Molecular transformer−drug−target interaction (MT-DTI) was pretrained on approximately 1 billion SMILES strings and the FASTA sequences of target proteins, which bypass the need for three-dimensional structures (e.g., X-ray) of protein-target complexes (Shin et al., n.d.) By using a pretrained model, the authors explored the transfer learning capabilities of MT-DTI, allowing the model to adapt previously learned features to make predictions for new tasks (Cai et al., 2020; Tan et al., 2018), which is especially important when not enough data is available, which was the case for SARS-CoV-2 inhibitors at the time.

Therefore the task consisted of training the model to understand the chemical sequences of small compounds and protein targets to predict the binding affinity. The authors found that atazanavir, remdesivir, and efavirenz were potential inhibitors of M^{pro}, while atazanavir also yielded nanomolar predicted binding affinity for RdRP, helicase, endoRNAse, 3′-to-5′-exonuclease, and 2′-O-ribose methyltransferase (Beck et al., 2020). It is important to highlight that although these drugs were predicted as nanomolar inhibitors, experimental confirmation is essential to validate the computational analysis. In the case of atazanavir, there is experimental evidence of weak Mpro inhibition (Fintelman-Rodrigues et al., 2020; Mahdi et al., 2020). Remdesivir has been shown to inhibit SARS-CoV-2 replication in vitro, despite its mechanism of action not being elucidated (Choy et al., 2020; Wang et al., 2020).

The UK-based company BenevolentAI (https://www.benevolent.com/) used a knowledge graph containing different biomedical sources to explore new strategies for SARS-CoV-2 (Richardson et al., 2020). Among the drugs identified by mining the hidden information within the graph, there were 378 inhibitors of the P2-associated protein kinase 1 (AAK1), a regulator of viral endocytosis in AT2 alveolar epithelial cells. Inhibitors of AAK1 could then potentially block viral entry into alveolar cells. However, the authors argued that only one of these drugs, baricitinib, a Janus kinase inhibitor, had an acceptable safety profile. In addition, the low therapeutic dosing of baricitinib (2mg or 4mg daily) makes it a promising drug for repurposing (Richardson et al., 2020). Pilot studies in COVID-19 patients showed that baricitinib in combination with other antivirals had

no significant adverse effects and improved clinical parameters (Cantini et al., 2020; Kalil et al., 2021). Baricitinib is currently under clinical trials (NCT04393051, NCT04421027, and NCT04399798) for patients with moderate to serious COVID-19.

Gupta and Zhou adopted a hybrid approach by combining classical and neural network-based molecular dynamics (MD) force fields to prioritize FDA-approved drugs against SARS-CoV-2 Mpro, which consists of an interesting strategy to explore Newtonian mechanics and quantum properties of the system (Gupta & Zhou, 2020). The authors used a structure-based pipeline consisting of an initial docking of 1615 FDA-approved drugs against Mpro, followed by MD simulations to select the most promising compounds. After molecular docking, 62 promising drugs were selected, followed by 100 ns MD simulations using the CHARMM36 force field which filtered the list to 26 drugs. In the next step, 5 ns MD was carried out using a neural network-based force field to model the interactions between each drug and the protein-solvent system.

The neural network called ANI-2x (Accurate NeurAl-2x) was trained on small molecules their density functional theory (DFT) energies. The local atomic environment of each molecule was defined using Behler and Parrinello symmetry functions (Behler, 2011):

$$G_i^x = (G_1, G_2, G_3, G_m)$$

where G_i represents radial and angular degrees of freedom of the ith atom, which can be used to calculate the potential energy surface of the molecule. The G_i functions are transformations of the Cartesian coordinates to account for movements that do not change the energy (e.g., translation and rotation). If Cartesian coordinates were used, the neural network output and consequently the energy of the system would not be invariant to translation or rotation of the molecules. In summary, the ANI-2x neural network receives as input local information about each atom of the molecule to predict the total energy of the system. Extensive validation of ANI-2x showed similar accuracy to DFT calculations. In addition, ANI-2x predictions were 10^6 times faster than DFT, allowing it to be used in MD simulations along with classical force fields.

The hybrid ANI/MM approach estimated the total potential energy of the system as the sum of drug- and target-based potentials and the interaction between these terms:

$$U_r = U_{ANI}(r_{ANI}) + U_{MM}(r_{MM}) + U_{ANI/MM} \, r_{ANI}(r_{MM})$$

where the $U_{ANI/MM} \, r_{ANI}(r_{MM})$ represents the nonbonded interactions between the drugs and the system, including Coulombic and Lennard-Jones interactions.

In the last step of the protocol, the drug list was filtered based on the free energy of binding estimation using MM/PBSA simulations. In total, 9 drugs, including dihydroergotamine, midostaurin, ziprasidone, etoposide, apixaban, fluorescein, tadalafil, rolapitant, and palbociclib were identified as potential inhibitors of M^{pro}. Among the selected drugs, tadalafil ($K_d = 52.2 \, \mu M$), midostaurin ($K_d = 43.5 \, \mu M$), and dihydroergotamine ($K_d = 107.6$) had measured bioactivity against Mpro. In addition, the authors did not find Mpro inhibition data for a random list of 62 drugs filtered off by the docking step, indicating that the computational protocol was able to find meaningful biochemical features to make bioactivity predictions.

Polypharmacology is another paradigm in drug discovery that explores the interactions of drugs with multiple targets. In this scenario, Redka and coworkers explored the polypharmacology of approved drugs against SARS-CoV-2 and host targets to prioritize

treatments for COVID-19 (Redka et al., 2020). The dataset consisted of three sources of information: (1) small molecules with annotated clinical safety data, (2) a panel of 15 host proteins associated with SARS-CoV and MERS-CoV infection, and (3) homology models of SARS-CoV-2 Mpro and spike protein (at the time of writing no experimental structures were available at the Protein DataBank). To predict potential drug−target interactions, a proprietary DL model called Ligand Design was used, which supports both structure-based and de novo designs.

The Ligand Design approach works by iteratively selecting molecules (children) from the initial population (parents) to optimize an objective function. In their study, the authors used an objective function called MatchMaker, which is a DL model trained with structural and experimental data of the whole human proteome to predict drug-target interactions. In total, the authors screened 10,224 drugs over more than 8700 protein targets of interest. This collection of drugs and targets was labeled PolypharmDB.

The top approved drugs predicted hits for SARS-CoV-2 Mpro and spike protein consisted mainly of mTOR-signaling pathway modulators and antibiotics (Table 24.1). An extended list of 338 hits showed a miscellaneous collection of classes, including calcium channel blockers (e.g., diltiazem), serotonin receptor antagonists (e.g., sumatriptan) and HIV protease inhibitors (e.g., nelfinavir and saquinavir), ACE inhibitors (e.g., lisinopril) and histamine antagonists (e.g., cimetidine).

Among host targets included in PolypharmDB, transmembrane protease serine 2 (TMPRSS2) and cathepsin B were described in the literature as essential for SARS-CoV-2 infection. Different drug classes were predicted to inhibit TMPRSS2, cathepsin B, and dual inhibition of these targets, including monoamine oxidase inhibitors (e.g., Phenelzine, Selegiline, and Isocarboxazid), cholinesterase inhibitors (rivastigmine), a nonsteroidal aromatase inhibitor (anastrozole), and an antihistamine drug (antazoline).

Overall, the collection of molecules selected from PolypharmDB is interesting because it spans a variety of chemical classes. However, the reported results should be used cautiously. According to the authors, the prospective performance of the MatchMaker and Ligand Design approach was not validated at the time, which adds a degree of uncertainty to the results for new targets. In addition, the use of homology models for SARS-CoV-2 targets introduces another bottleneck to prediction confidence. It is worth mentioning the authors did submit the experimental structures of Mpro and spike proteins to the same protocol after publishing the manuscript and found 88.7% overlap in predictions.

24.4 Future perspectives and challenges

In this chapter, we summarized AI approaches used to repurpose drugs against SARS-CoV-2. The AI literature is in constant change, with new models, datasets, and strategies being published every week. Remarkable progress was made since the beginning of the COVID-19 pandemic, where different research groups developed methods to find promising drugs in the chemical space haystack. However, the true applicability of the suggested molecules is yet to be investigated. Even if a molecule is predicted to be active; the real success story will come after clinical trials, and we are yet to see AI-tailored drugs being approved.

Another possible step forward is the prediction of new pandemics. We might still be far away from predicting the exact date a pandemic will start, but data such as social media posts, social and geographic events, medical records, and previous pandemics data could be used to alert authorities about possible disease outbreaks. Using AI to predict epidemics is still in their infancy, but some companies have already shown that it's a promising technology, such as BlueDot alert about SARS-CoV-2 seven days before the global alert by WHO (Niiler, 2020), and Metabiota predictions that Japan, Thailand, Taiwan, and South Korea would be at risk (Allam et al., 2020).

AI could also prove useful in the prediction of new viral targets and their three-dimensional structures. AI have already achieved great success as exemplified by AlphaFold outperforming current methods for protein structure prediction (Jumper et al., 2021). Expanding on this success story, some companies are already investing in the prediction of structures for whole proteomes and how to use this knowledge for bioactivity prediction. An example of this kind of approach is the Human3DProteome platform by the Wales-based company Moleculomics (https://human3dproteome.com/private/). When the next pandemic strikes, we might be able to access important targets and focus on drug discovery strategies to gain insight on optimization strategies to deliver more potent and safer drugs.

24.5 Conclusions

AI is reshaping the drug discovery process. The amount of chemical and biological information in databases allows researchers to use ML and DL algorithms to extract previously hidden relationships between drugs and biological targets. In this chapter, different AI methods were introduced but it is important to highlight that there is no privileged methodology. The choice between, for example, LSTM-based models and graph models, depends on a number of variables, including the type of data, computational power, and user expertise. We recommend the reader experiment with different methods to understand how they work and carefully design their projects to select the most appropriate to solve a specific task.

As shown in the selected examples, drugs with very different indications (e.g., estrogen modulators, kinase inhibitors, and cancer) are being predicted as potential treatments of COVID-19 patients, as coadjutants, or with potential antiviral activity against SARS-CoV-2. In addition, some predicted drugs have associated experimental evidence against other Coronaviruses, indicating that AI is not simply finding random noise but learning meaningful patterns from data.

The ability of AI methods to find patterns from raw data are outstanding. However, the real validation of these methods is yet to come, in the form of a repurposed drug for COVID-19.

References

Abadi, M., Barham, Chen, J., Chen, Z., Davis, A., Dean, J., et al. (2016). A system for large-scale machine learning. In Proceedings of the *twelfth USENIX symposium on operating systems design and implementation (OSDI)* 16 (pp. 265–283).
Allam, Z., Dey, G., & Jones, D. (2020). Artificial intelligence (AI) provided early detection of the Coronavirus (COVID-19) in China and will influence future urban health policy internationally. *Artificial Intelligence*, 1(2), 156–165. Available from https://doi.org/10.3390/ai1020009.

Alshemali, B., & Kalita, J. (2020). Improving the reliability of deep neural networks in NLP: A review. *Knowledge-Based Systems*, *191*, 105210. Available from https://doi.org/10.1016/j.knosys.2019.105210.

Ashtawy, H. M., & Mahapatra, N. R. (2018). Boosted neural networks scoring functions for accurate ligand docking and ranking. *Journal of Bioinformatics and Computational Biology*, *16*(2). Available from https://doi.org/10.1142/S021972001850004X.

Beck, B. R., Shin, B., Choi, Y., Park, S., & Kang, K. (2020). Predicting commercially available antiviral drugs that may act on the novel coronavirus (SARS-CoV-2) through a drug-target interaction deep learning model. *Computational and Structural Biotechnology Journal*, *18*, 784—790. Available from https://doi.org/10.1016/j.csbj.2020.03.025.

Behler, J. (2011). Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *The Journal of Chemical Physics*, *134*(7), 074106. Available from https://doi.org/10.1063/1.3553717.

Bento, A. P., Gaulton, A., Hersey, A., Bellis, L. J., Chambers, J., Davies, M., Krüger, F. A., Light, Y., Mak, L., McGlinchey, S., Nowotka, M., Papadatos, G., Santos, R., & Overington, J. P. (2014). The ChEMBL bioactivity database: An update. *Nucleic Acids Research*, *42*(1), D1083—D1090. Available from https://doi.org/10.1093/nar/gkt1031.

Bjerrum, E. J. (2017). *SMILES enumeration as data augmentation for neural network modeling of molecules*. arXiv.

Blaschke, T., Arús-Pous, J., Chen, H., Margreitter, C., Tyrchan, C., Engkvist, O., et al. (n.d.). *REINVENT 2.0—an AI tool for de novo drug* (Vol. 2020).

Boobier, S., Hose, D. R. J., Blacker, A. J., & Nguyen, B. N. (2020). Machine learning with physicochemical relationships: Solubility prediction in organic solvents and water. *Nature Communications*, *11*(1). Available from https://doi.org/10.1038/s41467-020-19594-z.

Bottou, L., Curtis, F. E., & Nocedal, J. (2018). Optimization methods for large-scale machine learning. *SIAM Review*, *60*(2), 223—311. Available from https://doi.org/10.1137/16M1080173.

Bullock, J., Luccioni, A., Pham, K., Lam, C., & Luengo-Oroz, M. (n.d.). *Mapping the landscape of artificial intelligence applications against COVID-19, 2020* (pp. 1—14).

Burki, T. (2020). A new paradigm for drug development. *Lancet Digital Health, 2, 226.*

Butler, K. T., Davies, D. W., Cartwright, H., Isayev, O., & Walsh, A. (2018). Machine learning for molecular and materials science. *Nature*, *559*(7715), 547—555. Available from https://doi.org/10.1038/s41586-018-0337-2.

Cai, C., Wang, S., Xu, Y., Zhang, W., Tang, K., Ouyang, Q., Lai, L., & Pei, J. (2020). Transfer learning for drug discovery. *Journal of Medicinal Chemistry*, *63*(16), 8683—8694. Available from https://doi.org/10.1021/acs.jmedchem.9b02147.

Cantini, F., Niccoli, L., Matarrese, D., Nicastri, E., Stobbione, P., & Goletti, D. (2020). Baricitinib therapy in COVID-19: A pilot study on safety and clinical impact. *Journal of Infection*, *81*(2), 318—356. Available from https://doi.org/10.1016/j.jinf.2020.04.017.

Chen, H., Engkvist, O., Wang, Y., Olivecrona, M., & Blaschke, T. (2018). The rise of deep learning in drug discovery. *Drug Discovery Today*, *23*(6), 1241—1250. Available from https://doi.org/10.1016/j.drudis.2018.01.039.

Cheng, J., Dong, & Lapata, M. (2016). *Long short-term memory-networks for machine reading*.

Cho, K., Merrienboer, G., Bahdanau, D., Bougares, F., Schwenk, H., et al. (2014). *learning phrase representations using RNN encoder-decoder for statistical machine translation*.

Chollet, F., et al. (2015). Keras [Internet]. *GitHub*; Available from: https://keras.io.

Choy, K. T., Wong, A. Y. L., Kaewpreedee, P., Sia, S. F., Chen, D., Hui, K. P. Y., Chu, D. K. W., Chan, M. C. W., Cheung, P. P. H., Huang, X., Peiris, M., & Yen, H. L. (2020). Remdesivir, lopinavir, emetine, and homoharringtonine inhibit SARS-CoV-2 replication in vitro. *Antiviral Research*, *178*. Available from https://doi.org/10.1016/j.antiviral.2020.104786.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems*, 303—314. Available from https://doi.org/10.1007/bf02551274.

Duvenaud, D., Maclaurin, D., Aguilera-Iparraguirre, J., Gómez-Bombarelli, R., Hirzel, T., Aspuru-Guzik, A., & Adams, R. P. (2015). Convolutional networks on graphs for learning molecular fingerprints. *Advances in neural information processing systems* (Vols. 2015-, pp. 2224—2232). Neural information processing systems foundation.

Eraslan, G., Avsec, Ž., Gagneur, J., & Theis, F. J. (2019). Deep learning: New computational modelling techniques for genomics. *Nature Reviews. Genetics*, *20*(7), 389—403. Available from https://doi.org/10.1038/s41576-019-0122-6.

Esteva, A., Robicquet, A., Ramsundar, B., Kuleshov, V., DePristo, M., Chou, K., Cui, C., Corrado, G., Thrun, S., & Dean, J. (2019). A guide to deep learning in healthcare. *Nature Medicine*, *25*(1), 24—29. Available from https://doi.org/10.1038/s41591-018-0316-z.

Feinberg, E. N., Sur, D., Wu, Z., Husic, B. E., Mai, H., Li, Y., Sun, S., Yang, J., Ramsundar, B., & Pande, V. S. (2018). PotentialNet for molecular property prediction. *ACS Central Science*, 4(11), 1520−1530. Available from https://doi.org/10.1021/acscentsci.8b00507.

Fintelman-Rodrigues, N., Sacramento, C. Q., Lima, C. R., da Silva, F. S., Ferreira, A. C., Mattos, M., de Freitas, C. S., Soares, V. C., da Silva Gomes Dias, S., Temerozo, J. R., Miranda, M. D., Matos, A. R., Bozza, F. A., Carels, N., Alves, C. R., Siqueira, M. M., Bozza, P. T., & Souza, T. M. L. (2020). Atazanavir, alone or in combination with ritonavir, inhibits SARS-CoV-2 replication and proinflammatory cytokine production. *Antimicrobial Agents and Chemotherapy*, 64(10). Available from https://doi.org/10.1128/AAC.00825-20.

Gardiner, L. J., Carrieri, A. P., Wilshaw, J., Checkley, S., Pyzer-Knapp, E. O., & Krishna, R. (2020). Using human in vitro transcriptome analysis to build trustworthy machine learning models for prediction of animal drug toxicity. *Scientific Reports*, 10(1). Available from https://doi.org/10.1038/s41598-020-66481-0.

Gawehn, E., Hiss, J. A., & Schneider, G. (2016). Deep learning in drug discovery. *Molecular Informatics*, 35(1), 3−14. Available from https://doi.org/10.1002/minf.201501008.

Ge, Y., Tian, T., Huang, S., Wan, F., Li, J., Li, S., et al. (n.d.). *A data-driven drug repositioning framework discovered a potential therapeutic agent targeting COVID-19*.

Gilmer, J., Schoenholz, S., Riley, P., Vinyals, O., & Dahl, G. E. (2017). *Neural message passing for quantum chemistry*. arXiv.

Gilson, M. K., Liu, T., Baitaluk, M., Nicola, G., Hwang, L., & Chong, J. (2016). BindingDB in 2015: A public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic Acids Research*, 44(1), D1045−D1053. Available from https://doi.org/10.1093/nar/gkv1072.

Goh, G. B., Hodas, N. O., & Vishnu, A. (2017). Deep learning for computational chemistry. *Journal of Computational Chemistry*, 38(16), 1291−1307. Available from https://doi.org/10.1002/jcc.24764.

Guedes, I. A., de Magalhães, C. S., & Dardenne, L. E. (2014). Receptor-ligand molecular docking. *Biophysical Reviews*, 6(1), 75−87. Available from https://doi.org/10.1007/s12551-013-0130-2.

Gupta, A., & Zhou, H. X. (2020). Profiling SARS-CoV-2 main protease (MPRO) binding to repurposed drugs using molecular dynamics simulations in classical and neural network-trained force fields. *ACS Combinatorial Science*, 22(12), 826−832. Available from https://doi.org/10.1021/acscombsci.0c00140.

Gupta, A., Müller, A. T., Huisman, B. J. H., Fuchs, J. A., Schneider, P., & Schneider, G. (2018). Generative recurrent networks for de novo drug design. *Molecular Informatics*, 37(1). Available from https://doi.org/10.1002/minf.201700111.

He, F., Liu, T., & Tao, D. (n.d.). Why ResNet works? Residuals generalize. *IEEE Transactions on Neural Networks and Learning Systems*, 2020, 1−14.

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, 9(8), 1735−1780. Available from https://doi.org/10.1162/neco.1997.9.8.1735.

Hornik, K. (1991). Approximation capabilities of multilayer feedforward networks. *Neural Networks*, 4(2), 251−257. Available from https://doi.org/10.1016/0893-6080(91)90009-T.

Howard, J., & Ruder, S. (2018). Universal language model fine-tuning for text classification. In *ACL 2018 - Proceedings of the conference fifty sixth annual meeting of the association for computational linguistics, (long papers)* (Vol. 1, pp. 328−339). Association for Computational Linguistics (ACL). Available from https://doi.org/10.18653/v1/p18-1031.

Jiménez-Luna, J., Grisoni, F., Weskamp, N., & Schneider, G. (2021). Artificial intelligence in drug discovery: Recent advances and future perspectives. *Expert Opinion on Drug Discovery*, 1−11. Available from https://doi.org/10.1080/17460441.2021.1909567.

Jumper, J., Evans, R., Pritzel, A., Green, T., Figurnov, M., Ronneberger, O., Tunyasuvunakool, K., Bates, R., Žídek, A., Potapenko, A., Bridgland, A., Meyer, C., Kohl, S. A. A., Ballard, A. J., Cowie, A., Romera-Paredes, B., Nikolov, S., Jain, R., Adler, J., & Hassabis, D. (2021). Highly accurate protein structure prediction with AlphaFold. *Nature*. Available from https://doi.org/10.1038/s41586-021-03819-2.

Kalil, A. C., Patterson, T. F., Mehta, A. K., Tomashek, K. M., Wolfe, C. R., Ghazaryan, V., Marconi, V. C., Ruiz-Palacios, G. M., Hsieh, L., Kline, S., Tapson, V., Iovine, N. M., Jain, M. K., Sweeney, D. A., El Sahly, H. M., Branche, A. R., Pineda, J. R., Lye, D. C., & Sandkovsky, U. (2021). Baricitinib plus remdesivir for hospitalized adults with Covid-19. *New England Journal of Medicine*, 384(9), 795−807. Available from https://doi.org/10.1056/NEJMoa2031994.

Kearnes, S., McCloskey, K., Berndl, M., Pande, V., & Riley, P. (2016). Molecular graph convolutions: Moving beyond fingerprints. *Journal of Computer-Aided Molecular Design*, 30(8), 595−608. Available from https://doi.org/10.1007/s10822-016-9938-8.

Howard, J., Fastai, & GitHub. (2018). Fastai. https://github.com/fastai. Accessed 10 Feb 2020.

Kejriwal, M. (n.d.). *Knowledge graphs and COVID-19: Opportunities, challenges, and implementation. Special issue 1-COVID* (Vol. 19).

Kim, S., Thiessen, P. A., Bolton, E. E., Chen, J., Fu, G., Gindulyte, A., Han, L., He, J., He, S., Shoemaker, B. A., Wang, J., Yu, B., Zhang, J., & Bryant, S. H. (2016). PubChem substance and compound databases. *Nucleic Acids Research*, 44(1), D1202−D1213. Available from https://doi.org/10.1093/nar/gkv951.

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, 521(7553), 436−444. Available from https://doi.org/10.1038/nature14539.

Lee, K., & Kim, D. (2019). In-silico molecular binding prediction for human drug targets using deep neural multi-task learning. *Genes*, 10(11). Available from https://doi.org/10.3390/genes10110906.

Li, X., Li, Z., Wu, X., Xiong, Z., Yang, T., Fu, Z., Liu, X., Tan, X., Zhong, F., Wan, X., Wang, D., Ding, X., Yang, R., Hou, H., Li, C., Liu, H., Chen, K., Jiang, H., & Zheng, M. (2020). Deep learning enhancing kinome-wide poly-pharmacology profiling: Model construction and experiment validation. *Journal of Medicinal Chemistry*, 63(16), 8723−8737. Available from https://doi.org/10.1021/acs.jmedchem.9b00855.

Liang, Z., Powell, A., Ersoy, I., Poostchi, M., Silamut, K., Palaniappan, K., Guo, P., Hossain, M.A., Sameer, A., Maude, R.J., Huang, J.X., Jaeger, S., & Thoma, G. (2017). CNN-based image analysis for malaria diagnosis. In Proceedings IEEE international conference on bioinformatics and biomedicine, BIBM 2016 (pp. 493−496). Institute of Electrical and Electronics Engineers Inc. Available from https://doi.org/10.1109/BIBM.2016.7822567.

Lin, Z., Feng, M., Santos, C., Yu, M., Xiang, B., Zhou, B., et al. (2017). *A structured self-attentive sentence embedding*.

Lipton, Z., Berkowitz, J., & Elkan, C. (2015). *A critical review of recurrent neural networks for sequence learning*.

Lo, Y. C., Rensi, S. E., Torng, W., & Altman, R. B. (2018). Machine learning in chemoinformatics and drug discovery. *Drug Discovery Today*, 23(8), 1538−1546. Available from https://doi.org/10.1016/j.drudis.2018.05.010.

Mahdi, M., Mótyán, J. A., Szojka, Z. I., Golda, M., Miczi, M., & Tőzsér, J. (2020). Analysis of the efficacy of HIV protease inhibitors against SARS-CoV-2′s main protease. *Virology Journal*, 17(1). Available from https://doi.org/10.1186/s12985-020-01457-0.

Mater, A. C., & Coote, M. L. (2019). Deep learning in chemistry. *Journal of Chemical Information and Modeling*, 2545−2559. Available from https://doi.org/10.1021/acs.jcim.9b00266.

Mayr, A., Klambauer, G., Unterthiner, T., Hochreiter, S., & DeepTox. (2016). Toxicity prediction using deep learning. *Frontiers in Environmental Science and Engineering China*, 3.

Moret, M., Friedrich, L., Grisoni, F., Merk, D., & Schneider, G. (2020). Generative molecular design in low data regimes. *Nature Machine Intelligence*, 2(3), 171−180. Available from https://doi.org/10.1038/s42256-020-0160-y.

Niiler, E. (2020). *An AI epidemiologist sent the first warnings of the Wuhan virus*. <https://www.wired.com/story/ai-epidemiologist-wuhan-public-health-warnings/>.

Olivecrona, M., Blaschke, T., Engkvist, O., & Chen, H. (2017). Molecular de-novo design through deep reinforcement learning. *Journal of Cheminformatics*, 9(1). Available from https://doi.org/10.1186/s13321-017-0235-x.

Paszke, A., Gross, Massa, F., Lerer, A., Bradbury, J., Chanan, et al. (2019). *An imperative style, high-performance deep learning library*. arXiv.

Paszke, A., Gross, S., Chintala, S., Chanan, G., DeVito, Y. E., et al. (2017). *Automatic differentiation in PyTorch*.

Paulheim, H., & Cimiano, P. (2016). Knowledge graph refinement: A survey of approaches and evaluation methods. *Semantic Web*, 489−508. Available from https://doi.org/10.3233/SW-160218.

Redka, D. S., MacKinnon, S. S., Landon, M., Windemuth, A., Kurji, N., & Shahani, V. (2020). PolypharmDB, a deep learning-based resource, quickly identifies repurposed drug candidates for COVID-19. *ChemRxiv*. Available from https://doi.org/10.26434/chemrxiv.12071271.v1.

Richardson, P., Griffin, I., Tucker, C., Smith, D., Oechsle, O., Phelan, A., & Stebbing, J. (2020). Baricitinib as potential treatment for 2019-nCoV acute respiratory disease. *The Lancet*, 395(10223), e30−e31. Available from https://doi.org/10.1016/S0140-6736(20)30304-4.

Robinson, M. C., Glen, R. C., & Lee, A. A. (2020). Validating the validation: Reanalyzing a large-scale comparison of deep learning and machine learning models for bioactivity prediction. *Journal of Computer-Aided Molecular Design*, 34(7), 717−730. Available from https://doi.org/10.1007/s10822-019-00274-0.

Rosenblatt, F. (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, 65(6), 386−408. Available from https://doi.org/10.1037/h0042519.

Rumelhart, D., Durbin, R., Golden, R., Chauvin, Y., & Backpropagation. (1995). The basic theory. *Backpropagation: Theory, Architectures and Applications*, 1−34.

Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1986). Learning representations by back-propagating errors. *Nature, 323*(6088), 533−536. Available from https://doi.org/10.1038/323533a0.

Samuel, A. L. (1959). Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 210−229. Available from https://doi.org/10.1147/rd.33.0210.

Santana, M. V. S., & Silva-Jr, F. P. (2021). De novo design and bioactivity prediction of SARS-CoV-2 main protease inhibitors using recurrent neural network-based transfer learning. *BMC Chemistry, 15*(1). Available from https://doi.org/10.1186/s13065-021-00737-2.

Shen, C., Ding, J., Wang, Z., Cao, D., Ding, X., & Hou, T. (2020). From machine learning to deep learning: Advances in scoring functions for protein−ligand docking. *Wiley Interdisciplinary Reviews: Computational Molecular Science, 10*(1). Available from https://doi.org/10.1002/wcms.1429.

Shin, B., Park, S., Kang, K., & Ho, J. C. (n.d.). *Self-attention based molecule representation for predicting drug-target interaction.* ArXiv, 2019, 1−18.

Tan, C., Sun, F., Kong, T., Zhang, W., Yang, C., & Liu, C. (2018). A survey on deep transfer learning. In Lecture notes in computer science (including subseries lecture notes in artificial intelligence and lecture notes in bioinformatics) (Vol. 11141, pp. 270−279). Springer Verlag. Available from https://doi.org/10.1007/978-3-030-01424-7_27.

Torng, W., & Altman, R. B. (2019). Graph convolutional neural networks for predicting drug-target interactions. *Journal of Chemical Information and Modeling, 59*(10), 4131−4149.

Valletta, J. J., Torney, C., Kings, M., Thornton, A., & Madden, J. (2017). Applications of machine learning in animal behaviour studies. *Animal Behaviour, 124*, 203−220. Available from https://doi.org/10.1016/j.anbehav.2016.12.005.

Vaswani, S., Parmar., Uszkoreit, J., Jones, G., et al. (2017). Attention is all you need. *Advances in neural information processing systems, 30*, 5998−6008.

Vouldodimos, A., Doulamis, N., Doulamis, A., & Protopapadakis, E. (2018). Deep learning for computer vision: A brief review. *Computational Intelligence and Neuroscience*, 1−13. Available from https://doi.org/10.1155/2018/7068349.

Walters, W. P., & Murcko, M. (2020). Assessing the impact of generative AI on medicinal chemistry. *Nature Biotechnology, 38*(2), 143−145. Available from https://doi.org/10.1038/s41587-020-0418-2.

Wang, M., Cao, R., Zhang, L., Yang, X., Liu, J., Xu, M., Shi, Z., Hu, Z., Zhong, W., & Xiao, G. (2020). Remdesivir and chloroquine effectively inhibit the recently emerged novel coronavirus (2019-nCoV) in vitro. *Cell Research, 30*(3), 269−271. Available from https://doi.org/10.1038/s41422-020-0282-0.

Wishart, D. S., Knox, C., Guo, A. C., Shrivastava, S., Hassanali, M., Stothard, P., Chang, Z., & Woolsey, J. (2006). DrugBank: A comprehensive resource for in silico drug discovery and exploration. *Nucleic Acids Research, 34*, D668−D672. Available from https://doi.org/10.1093/nar/gkj067.

Yosinski, J., Clune, J., Nguyen, Fuchs, T., & Lipson, H. (2015). *Understanding neural networks through deep visualization.*

Young, T., Hazarika, D., Poria, S., & Cambria, E. (2018). Recent trends in deep learning based natural language processing [Review Article]. *IEEE Computational Intelligence Magazine, 13*(3), 55−75. Available from https://doi.org/10.1109/MCI.2018.2840738.

Yu, Y., Si, X., Hu, C., & Zhang, J. (2019). A review of recurrent neural networks: LSTM cells and network architectures. *Neural Computation, 31*(7), 1235−1270. Available from https://doi.org/10.1162/neco_a_01199.

Zeng, X., Song, X., Ma, T., Pan, X., Zhou, Y., Hou, Y., Zhang, Z., Li, K., Karypis, G., & Cheng, F. (2020). Repurpose open data to discover therapeutics for COVID-19 using deep learning. *Journal of Proteome Research, 19*(11), 4624−4636. Available from https://doi.org/10.1021/acs.jproteome.0c00316.

Zhang, L., Tan, J., Han, D., & Zhu, H. (2017). From machine learning to deep learning: Progress in machine intelligence for rational drug discovery. *Drug Discovery Today, 22*(11), 1680−1685. Available from https://doi.org/10.1016/j.drudis.2017.08.010.

Zhang, S., Tong, H., Xu, J., & Maciejewski, R. (2019). Graph convolutional networks: a comprehensive review. *Computational Social Networks, 6*(1), 1−23.

Zhou, Y., Wang, F., Tang, J., Nussinov, R., & Cheng, F. (2020). Artificial intelligence in COVID-19 drug repurposing. *The Lancet Digital Health, 2*(12), e667−e676. Available from https://doi.org/10.1016/S2589-7500(20)30192-8.

Zou, J., Huss, M., Abid, A., Mohammadi, P., Torkamani, A., & Telenti, A. (2019). A primer on deep learning in genomics. *Nature Genetics, 51*(1), 12−18. Available from https://doi.org/10.1038/s41588-018-0295-5.