Briefings in Bioinformatics, 21(3), 2020, 919–935

doi: 10.1093/bib/bbz042
Advance Access Publication Date: 3 June 2019
Review article

Graph convolutional networks for computational drug development and discovery

Mengying Sun, Sendong Zhao, Coryandar Gilvary, Olivier Elemento, Jiayu Zhou and Fei Wang

Corresponding author: Fei Wang Department of Healthcare Policy and Research, Weill Cornell Medicine, New York, NY 10065, USA. Tel. 646-962-9405; E-mail: few2001@med.cornell.edu, feiwang03@gmail.com

Abstract

Despite the fact that deep learning has achieved remarkable success in various domains over the past decade, its application in molecular informatics and drug discovery is still limited. Recent advances in adapting deep architectures to structured data have opened a new paradigm for pharmaceutical research. In this survey, we provide a systematic review on the emerging field of graph convolutional networks and their applications in drug discovery and molecular informatics. Typically we are interested in why and how graph convolution networks can help in drug-related tasks. We elaborate the existing applications through four perspectives: molecular property and activity prediction, interaction prediction, synthesis prediction and *de novo* drug design. We briefly introduce the theoretical foundations behind graph convolutional networks and illustrate various architectures based on different formulations. Then we summarize the representative applications in drug-related problems. We also discuss the current challenges and future possibilities of applying graph convolutional networks to drug discovery.

Key words: graph convolution network; computational drug development.

Introduction

Drug development is an expensive and time-consuming process where thousands of chemical compounds are being tested and experiments are being conducted in order to find out drugs that are safe and effective. The general process of drug development involves steps as shown in Figure 1. Modern drug development aims to speed up the intermediate steps and thus reduces cost by leveraging machine learning tools on drug development, typically at drug discovery and preclinical research stages. In short, molecular compounds are filtered through a progressive series of tests which determine their properties, effectiveness and toxicity for later stages. Machine leaning is increasingly being used to better predict molecular properties in early stages can significantly reduce the load of later processes (e.g. clinical trials), saving tons of resources as well as time.

Currently, applications of machine learning to developing drugs include but are not limited to the following: bioactivity or physico-chemical prediction, by using the widely adopted

Mengying Sun, MS, is a PhD student in Department of Computer Science and Engineering, Michigan State University, East Lansing, MI USA.
Sendong Zhao, PhD, is a postdoctoral associate in Division of Health Informatics, Department of Healthcare Policy and Research at Weill Cornell Medicine at Cornell University, New York, NY, USA.
Coryandar Gilvary, BS, is a PhD student in the Institute for Computational Biomedicine and the Tri-I Program in Computational Biology & Medicine at Weill Cornell Medicine at Cornell University, New York, NY, USA.
Olivier Elemento, PhD, is the Director of the Englander Institute for Precision Medicine (EIPM); he is also an Associate Professor in the Department of Physiology and Biophysics, at Weill Cornell Medicine at Cornell University, New York, NY, USA.
Jiayu Zhou, PhD, is an Assistant Professor in Department of Computer Science and Engineering, Michigan State University, East Lansing, MI USA.
Fei Wang, PhD, is an Associate Professor in the Division of Health Informatics, Department of Healthcare Policy and Research at Weill Cornell Medicine at Cornell University, New York, NY, USA.
Submitted: 01 January 2019; **Received (in revised form):** 01 March 2019

Figure 1. The drug development process.

quantitative structure–activity (property) relationship (QSAR/QSPR) models; predicting interactions for drug–protein and drug–drug pairs; *de novo* molecular design which produces molecular structures with desirable pharmacological properties; synthesis prediction, predicting products for synthesis reactions. Due to the fact that traditional machine learning methods can only handle fixed-size inputs, most early era drug discovery have used feature engineering, that is generating and using problem-specific molecular descriptors. Generally, a set of problem-specific molecular descriptors are used as features in those tasks. Commonly used descriptors include the following: (1) molecular fingerprints, encoding the structure of a molecule by a series of binary digits that represent the presence of particular substructures; (2) descriptors derived from quantum/physical chemistry and differential topology, processed by statisticians and chemiformaticians; and (3) Simplified Molecular-Input Line-Entry System (SMILES) strings, which uniquely characterize a molecule's structure and represent it into a line notation. Given predefined predictors (i.e. input variables), classification or prediction model is then built and learned via machine learning algorithms.

Recent years have witnessed an increasing amount of large chemical databases being available for pharmaceutical research. Therefore, new attempts have risen in using deep neural networks for application to drug development. Deep learning [1] has achieved remarkable success and been widely adopted over the past decade for learning tasks in domains such as natural language processing [2] and computer vision [3]. The strength of deep learning lies in its ability to learn complex relationships between input features and output decisions from large scale data. Its application in drug discovery and molecular informatics is still in its infancy but has already revealed great potential. Several commonly used deep architectures have been employed in drug-related works [4–7] and achieved substantial improvements compared to traditional machine learning methods. However, limitations remain for deep models due to the following reasons. First, most current deep models are still based on handcrafted features or predefined descriptors, preventing structural information to be learned directly from raw inputs. Second, the existing architectures are not well suited for structured data like molecules. The structural information inside is neither considered nor sufficiently used in the process of feature extraction for these architectures. Therefore, better suited architectures are critical for further improving the potentials of deep learning in drug discovery.

Structured data, e.g. images, have been successfully taken care of by convolutional neural networks (CNNs), a special architecture of deep neural networks. CNN reveals the state-of-art performance in image-related tasks since it can automatically extract task relevant features from raw images via a convolution operator [8]. For such drugs and small molecules which are composed of atoms and chemical bonds, we have different type structures, i.e. graph, for them, in which each atom is a node and each chemical bond is an edge. A straightforward attempt would be adapting the convolution process similarly for molecular graphs. However, unlike images, graphs have irregular shapes and sizes; there is no spatial order imposed on the nodes, whose neighbors are also position dependent. The traditional convolution on regular grid-like structures thus cannot be applied on graphs directly. In fact, a wide range of structural data in the real world are often formed as graphs rather than images, meaning that developing methods that deals with irregular structures are of great importance and urgently demanded.

Efforts have been made throughout the literature towards the generalization of convolution operator on non-Euclidean structured data, resulting in the so-called graph convolutional networks (GCN). GCNs have been established as the state of the art approach for drug-related tasks in the way that (1) it extracts features by considering the structure of the data and (2) it enables automatic feature extraction from raw inputs instead of from handcrafted features which might miss important information caused by the bias from domain experts. Current emerging GCNs follow two main streams. One can be summarized as spatial GCN, which formulates convolution directly in the spatial domain by summing up all feature vectors of all neighboring nodes in the graph; the other is known as spectral GCN, which defines convolution in the graph spectral domain, following spectral graph theory [9]. Recent works [10, 11] have also shown that spectral convolution can be characterized as a special case of spatial convolution. However, due to different theoretical basis, we still treat them as separate convolution operations in the following sections. Having convolution defined in both domain, generative GCN utilizes the convolution process to encode hidden representations and generate molecule graphs.

In this survey, we specifically focus on the recent advance in GCN and its applications on drug discovery, instead of presenting in the context of general deep learning as in previous surveys [12–14]. Therefore, our review is focused on drug-related applications including recent ones, aiming to help readers to gain insights into the current progress of newly developed deep architectures on drug discovery. We summarize the most relevant papers in Table 1. Moreover, we also provide a summary for all data sources that have been used in those studies and are available for public access in later sections.

The remaining content is organized as follows: we provide theoretical support of graph convolution in Principles of Graph Convolution section and detailed architectures of GCNs and their applications in Application of GCN on drug discovery section . Information for public data source is summarized in Databases for drug discovery and molecular bioinformatics section . Discussion section discusses the challenges and possibilities beyond the current approaches.

Principles of graph convolution

There are many important real problems dealing the data form of graphs or networks, like social networks, knowledge graphs, protein-interaction networks, molecular graphs, etc. However, applying deep learning to the these graph data is non-trivial because of the unique characteristics of graphs [15]. Recently, great attention has been paid to the generalization of neural network models to such structured graph data [15, 16]. In the past couple of years, a number of papers re-visited this problem of generalizing neural networks to work on arbitrarily structured graphs [17–22], some of them achieving very promising results in domains that have previously been dominated by, e.g.

Table 1. Summary of papers surveyed in this review

Author	Dataset	Architecture	Highlight
		Molecular property and activity	
Niepert *et al.* [27]	NCI	Spatial	Proposed a general approach to extract locally connected regions from graphs which is analogue to traditional CNN and applied to molecular activity classification task
Duvenaud *et al.* [28]	HCEP	Spatial	Proposed graph convolution to extract neural fingerprints and tested on molecular property prediction tasks
Kearnes *et al.* [29]	PCBA, MUV, Tox21	Spatial	Proposed a weave module network that considering both atom and bond features in the convolution operator and applied to molecular activity classification tasks
Schutt *et al.* [30]	QM	Spatial	Proposed deep tensor network on predicting molecular chemical properties
Gilmer *et al.* [10]	QM9	Spatial	Proposed a unified framework called message passing network and explored extensions to model molecular properties
Schutt *et al.* [31]	QM9, MD17, ISO17	Spatial	Proposed a continuous graph convolution operator based on radial basis functions and applied to predict molecular energies and atomic forces
Li *et al.* [32]	Tox21, ToxCast, MUV, PCBA, HIV, FreeSolv	Spatial	Proposed a graph level representation for molecular property prediction
Liu *et al.* [33]	ADME	Spatial	Application of spatial graph convolution on molecular ADME property prediction
Altae-Tran *et al.* [34]	Tox21, SIDER, MUV	Spatial	Application of spatial graph convolution for limited data on molecule ADME property prediction by leveraging few shot learning
Li *et al.* [35]	ESOL, NCI, ADME, Tox21, ClinTox, SIDER, Toxcast	Spectral	Proposed adaptive graph convolution framework in which the graphs are updated through iterations and applied to molecular property prediction
Shang *et al.* [36]	Tox21, HIV, FreeSolv, Lipophilicity	Spectral	Incoporate edge attention into spectral graph convolution framework and applied to molecular property prediction
Ryu *et al.* [37]	ZINC, QM9, HCEP	Spectral	Incoporate edge attention into spectral graph convolution framework and applied to molecular property prediction
Pham *et al.* [38]	PCBA	Spatial	Combine GCN and memory network to predict molecular activity
		Interaction prediction	
Gao *et al.* [39]	BindingDB	Spatial	Utilized graph convolution networks to obtain molecule representations from raw molecules and used for drug-target
		interaction prediction	
Zitnik *et al.* [40]	STRING, STITCH, SIDER, OFFSIDES	Spatial	Utilized graph convolution networks to obtain molecule representations in a multi-model network consist of drugs and proteins and used for DDI prediction
Asada *et al.* [41]	DrugBank	Spatial	Enhancing DDI extraction from texts by using molecular structure information
Ma *et al.* [42]	TWOSIDES, SIDER, OFFSIDES, CPI, TTD	Spectral	Integrate different data sources and predict DDI via spectral graph convolution approaches
		Synthesis prediction	
Jin *et al.* [43]	USPTO	Spatial	Utilized graph convolution to identify reaction outcomes in chemical reactions
		***De novo* drug design**	
Simonovsky *et al.* [44]	QM9, ZINC	VAE	Incorporate graph convolution into variational auto-encoder to generate small molecule graphs
Li *et al.* [45]	ChEMBL	Spatial	Utilized GCNs to express probabilistic dependencies among a graph's nodes and edges for generating molecule graphs
Li *et al.* [46]	ChEMBL	Spatial	Utilized GCNs to generate larger molecules based on sequential generation
Jin *et al.* [47]	ZINC	Spatial	Incorporate graph convolution and junction tree into variational auto-encoder to generate more valid molecule graphs based on substructures

kernel-based methods, graph-based regularization techniques and others. In the following subsections, we present the representation of the graph and two ways of graph convolution, i.e. spatial convolution and spectral convolution. The spatial convolution GCN is a differentiable message-passing schema which operates on local graph neighborhoods to arbitrary graphs. It is more popular than spectral convolution for graphs like social netwoks, knowledge graphs and molecular graphs. The idea of spectral convolution GCN is to use the spectral theory to achieve the convolution operation on the topological graph. It is usually applied for dealing with the data, like images and videos.

Graph definition

Consider a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$, where $\mathcal{V}, \mathcal{E}$ denote the set of vertices and edges with $|\mathcal{V}| = n$, $\mathbf{A}$ denotes the weighted adjacency matrix

$$\mathbf{A}_{ij} = \begin{cases} a_{ij}, & \text{if } e_{ij} \in \mathcal{E} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

$a_{ij} = 1$ if the graph is unweighted and $\mathbf{A}$ is symmetric if the edges are undirected. A graph signal $\mathbf{x} \in \mathbb{R}^n$ is a real-valued vector defined on all vertices. For a certain vertex, $\mathbf{h} \in \mathbb{R}^d$ denotes attributes associated with the vertex (graph signal $\mathbf{x}$ is a feature map for all vertices while vertex attributes $\mathbf{h}$ is a feature vector for one vertex). The graph Laplacian matrix $\mathbf{L}$ is defined as

$$\mathbf{L} = \mathbf{D} - \mathbf{A}, \tag{2}$$

where $\mathbf{D}$ is a diagonal matrix whose diagonal elements $\mathbf{D}_{ii} = \sum_{j=1}^{n} \mathbf{A}_{ij}$ denotes the degree of corresponding vertices.

Spatial convolution

Among the early attempts of generalizing discriminative embeddings for structured data, Dai *et al.* [23] proposed *structure2vec*, a latent variable model for embedding graph structured data, using approximate inference algorithms in graphic models. The authors showed that the solution to the inference algorithms implies a propagation equation where the representation of a node is a function of neighborhood marginals (nodes) and messages from neighbors. Later on, a large portion of GCNs are built on top of this concept with extensive modifications, known as spatial convolution.

Spatial convolution aims to construct convolution directly in the vertex domain. The key idea follows that the representation of a certain node is updated by aggregating information from its neighboring nodes (recursively). It coincides with the Weisfeiler-Lehman algorithm [24], commonly used to test whether two graphs are isomorphism, in which the node labels are augmented repeatedly by the sorted set of labels of the neighbor nodes. The underlying mechanism for such propagation is to first regard neighborhood information as graph substructures and then model such substructures through differentiable functions by recursively projecting different (or the same) substructures into different (or the same) feature spaces [25]. Information that flows between neighbors and the center node is also referred as *Messages* [10]. The way by which messages are passed towards the center node produces different propagation rules that characterize the network architecture.

The propagation rules can be summarized into two phases—message passing phase and readout phase as shown in Equation (3) and (4) as described in [10].

$$\mathbf{m}_i^{l+1} = \sum_{j \in \mathcal{N}_i} \mathcal{M}(\mathbf{h}_i^l, \mathbf{h}_j^l, \mathbf{h}_{ij}^l) \tag{3}$$

$$\mathbf{h}_i^{l+1} = \mathcal{U}(\mathbf{h}_i^l, \mathbf{m}_i^{l+1}), \tag{4}$$

where $\mathbf{h}_i$ and $\mathbf{h}_j$ denote node representations and $\mathbf{h}_{ij}$ refers to edge features for node pair i and j. $\mathcal{N}_i$ is the set of neighbors for node i. $\mathcal{M}(\cdot)$ is a function that transforms neighbor information into hidden representation that will be passed to the center node. After neighborhood information is aggregated, the representation of center node i is updated through another function $\mathcal{U}(\cdot)$. Variants can be made based on the choice of function $\mathcal{M}(\cdot)$ and $\mathcal{U}(\cdot)$ which we will address in detail in later sections.

Spectral convolution

The spectral convolution theorem [26] pursues Fourier-like basis for signals defined on graph, typically focus on the properties of graph Laplacian matrix $\mathbf{L}$.

Fourier transform

The classic Fourier transform decomposes a signal in time domain into its frequency domain, as a linear combination of a set of basis functions. The weights for the basis functions can be obtained by the following equation:

$$F(\omega) = \mathcal{F}[f(t)] = \int f(t) e^{-i\omega t} dt, \tag{5}$$

where ω is the frequency associated with basis function $e^{-i\omega t}$ and $f(t)$ is a signal in the time domain. Fourier transform enables convolution between two signals to be achieved by simple product in the frequency domain followed by a projection back to the time domain.

By defining convolutions as linear operators that diagonalize in the Fourier basis that are represented by the eigenvectors of the Laplacian operator, we arrive at the graph Fourier transform. Typically, consider the eigen decomposition of Laplacian $\mathbf{L} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$, as $\mathbf{L}$ is a real symmetric positive semidefinite matrix, it has a complete set of orthonormal eigenvectors $\{u_l\}_{l=0}^n \in \mathbb{R}^n$, known as the graph Fourier modes, and their associated ordered real nonnegative eigenvalues $\{\lambda_l\}_{l=0}^n$, identified as the frequencies of the graph.

Graph Fourier transform

The graph Fourier transform of a signal $\mathbf{x} \in \mathbb{R}^n$ is thus defined as

$$\hat{\mathbf{x}} = \mathbf{U}^T \mathbf{x}, \tag{6}$$

and the convolution of two signals follows:

$$\mathbf{x} * \mathbf{h} = \mathbf{U}((\mathbf{U}^T \mathbf{x}) \odot (\mathbf{U}^T \mathbf{h})), \tag{7}$$

where $\odot$ denotes element-wise product. Having convolution defined for two graph signals, it naturally follows that a signal

$\mathbf{x}$ is filtered by $\mathbf{g}_\theta$ as

$$\tilde{\mathbf{x}} = g_\theta(\mathbf{L})x = \mathbf{U}g_\theta(\Lambda)\mathbf{U}^T\mathbf{x}. \quad (8)$$

The first generation of graph convolution is defined as above and proposed in [17]. However, there are several limitations that prevent it from practical use: (1) the filter is not localized, (2) the learning complexity is $\mathcal{O}(n^2)$ due to matrix-vector multiplicationand (3) number of parameters depend on input size. To confront the limitations, [21] introduce a polynomial filter, reducing the number of parameters to the order of the polynomial

$$\tilde{\mathbf{x}} = \mathbf{U}\mathbf{g}_\theta(\Lambda)\mathbf{U}^T\mathbf{x} \approx \mathbf{U}\begin{pmatrix} \sum_{j=0}^{K}\theta_j\lambda_1^j & & \\ & \ddots & \\ & & \sum_{j=0}^{K}\theta_j\lambda_n^j \end{pmatrix}\mathbf{U}^T\mathbf{x}. \quad (9)$$

The resulting filter is localized to neighbors that are K hops away and is independent on the input size. Moreover, to avoid explicit computation of polynomials, a recursive formulation was proposed using Chebyshev approximation, resulting in the following equation:

$$\mathbf{g}_\theta(L) = \sum_{k=0}^{K}\theta_k\mathbf{T}_k(\tilde{\mathbf{L}}), \quad (10)$$

where $\mathbf{T}_k(x) = 2x\mathbf{T}_{k-1}(x) - \mathbf{T}_{k-2}(x)$ can be applied to recursively calculate polynomials. Later on in [22], the authors considered 1st order neighborhood and simplified the formulation as

$$\mathbf{g}_\theta(\mathbf{L}) = \theta\tilde{\mathbf{A}}, \quad (11)$$

where $\tilde{\mathbf{A}} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{I} + \mathbf{A})\mathbf{D}^{-\frac{1}{2}}$ denotes the normalized adjacency matrix with self-loop added on each node.

Application of GCN on drug discovery

In this section, we review the previous studies on main applications related to computational drug development and discovery. The open source codes for different tasks of computational drug development and discovery via GCNs are presented in Table 2.

Quantitative Structure Activity/Property Relationship Prediction

QSAR (QSPR) predicts the relationship between biological activities (chemical properties) and molecular descriptors. Machine learning methods for this problem have been widely explored in the literature [50–52]. For most drug-related computational methods, a fundamental problem is about what types of input representations to deal with. Handcrafted features cannot fully encode the structure information of a molecular graph. In addition, those predefined inputs are not data/task driven, resulting in less predictive power. GCNs are thus introduced to confront the aforementioned limitations.

Biological property and activity

Duvenaud *et al.* [28] first proposed a method to generate differentiable and data-driven fingerprints using neural networks. In this study, the hash function in regular circular fingerprints, which encodes the substructure of each atom in a molecule, was replaced by a smooth function. Here, circular fingerprints are designed to encode which substructures are present in a molecule in a way that is invariant to atom-relabeling [53]. Therefore, a real-valued vector was learned to represent a molecule instead of a binary vector and is called neural fingerprints. The final representation of a molecule was obtained by aggregating representations of all atoms, going through a softmax layer that enabled interpretation of learned features. The encoding procedure was convolutional in the sense that neighborhood information was aggregated to update the center atom, and the same local filter was applied for atoms with the same neighborhood size (ranged from 1 to 5) and its neighboring atoms. The authors evaluated the generated fingerprints for several drug properties including solubility, drug efficacy and organic photovoltaic efficiency with neural fingerprints outperformed traditional circular fingerprints. Moreover, the representations are interpretable such that features strongly associated with predictions are more activated by certain fragments in the molecule structure. However, one deficiency lies in the fact that training both fingerprints and the prediction model takes more time compared to models built on top of precalculated fingerprints, especially for large datasets.

In addition to node features or labels, edge information is also important and can also be encoded in graph convolution. Kearnes *et al.* [29] proposed a graph convolution framework

Table 2. Open source codes for different tasks of computational drug development and discovery via GCNs.

Author	Application	Code
Li *et al.* [32]	Drug discovery	github.com/microljy/graph_level_drug_discovery
Shang *et al.* [36]	Drug property prediction	github.com/Luckick/EAGCN
Ryu *et al.* [37]	Drug property prediction	github.com/seongokryu/augmented-gcn
Gao *et al.* [39]	DTI	github.com/IBM/InterpretableDTIP
Feng *et al.* [48]	DTI	github.com/simonfqy/PADME
Zitnik *et al.* [40]	DDI	github.com/marinkaz/decagon
Ma *et al.* [42]	DDI	github.com/matenure/mvGAE
Fout *et al.*	PPI	github.com/fouticus/pipgcn
Jin *et al.*	Synthesis prediction	github.com/wengong-jin/nips17-rexgen
Simonovsky and Komodakis [44]	*De novo* molecular design	github.com/tkipf/gae
Li *et al.* [45]	*De novo* molecular design	github.com/shllln/GenerativeGraph
Li *et al.* [46]	*De novo* molecular design	github.com/kevinid/molecule_generator
Jin *et al.* [47]	*De novo* molecular design	github.com/wengong-jin/icml18-jtnn
Olivecrona *et al.* [49]	*De novo* molecular design	github.com/MarcusOlivecrona/REINVENT

to learn molecular representations for data-driven tasks considering both node and edge features. Specifically, each layer contained atom representations and pair-wise (edge) representations. Propagation took place for all the relational modules across different layers: atom to atom (A → A), atom to pair (A → P), pair to atom (P → A) and pair to pair (P → P), forming a *Weave* module. Each layer followed such weave module architecture while at the last convolution layer, only atom representations were used for down-streaming tasks. Transition across the same representations (A → A, P → P) was achieved by neural nets. For transition across different representations (A → P, P → A), an additional order-invariant aggregation operation was used after feature transformation. The authors evaluated their methods for biological activities from 259 datasets, which are composed of PCBA [54], the 'maximum unbiased validation' datasets constructed by Rohrer and Baumann [55], the enhanced directory of useful decoys [56] and the training set for the Tox21 challenge [57], in a multi-task setting where activities were predicted simultaneously. The inputs were molecular graphs with atom features as well as atom–pair features. The proposed method was then compared to baseline methods using Morgan fingerprints generated with RDKit. The proposed WeaveNet did not consistently outperform the state-of-arts, but provided an approach that incorporated edge features in addition to node features.

A similar application can be found in Liu *et al.* [33] where the authors developed an absorption, distribution, metabolism, and excretion (ADME) property prediction system using GCN. The convolution operator was similar to weave modules. For each atom, the neighborhood information was first transformed through a fully connected layer and then aggregated and reduced using different reduce operator. The representation for a center atom was then formed by concatenating results from all reduce operator including maximum, summation and average. The reduced representation was then combined with atom input feature. Evaluation was also made on multi-task scheme for five selected ADME endpoints: human microsomal clearance, human CYP450 inhinition, aqueous equilibrium solubility, pregnane X receptor induction and bioavailability, with improvement achieved by the proposed Chemi-Net compared to rule-based prediction models.

To obtain a molecule representation, most previous studies sum or average atom level representations but ignore the graph structure of the molecule. Li *et al.* [32] introduced an alternative approach to generate graph-level representation using GCN. The authors achieved their goal by introducing a dummy super node which was assumed to be connected to all nodes in the graph by a directed edge. The node level propagation follows conventional spatial graph convolution. For the dummy super node, the representation was updated using all the nodes in the graph and itself, with which an additional weight matrix was associated. Authors evaluated proposed method on both biological activity classification and molecular property prediction.

The previous frameworks were all built on spatial graph convolution, while spectral graph convolution was less employed in QSAR (QSPR) tasks due to the following reason. Molecules are composed of different atoms with varying sizes, thus end up in different topologies. The spectrum derived in spectral GCN is subject to one particular graph such that it cannot be transferred to other graphs. However, there are still studies on utilizing spectral GCN on drug-related tasks that bypass the constraints.

The early spectral GCN fixed graph structure without training, therefore incapable of learning from topological structure. Li *et al.* [35] constructed graph convolution that accepts flexible graph inputs and learned additional topology information for each input graph. The graph adjacency matrix was updated through a parametric distance metric where the weights for adjusting distance measure were learned during training procedure. Then Laplacian was updated based on a residual scheme, where at each iteration a small portion of the learned Laplacian was added to the original Laplacian matrix. The propagation followed conventional spectral graph convolution. Such adaptive setting allowed inputs to have unique graph Laplacian so that each compound had its unique convolution filter. The authors verified the proposed method through multi-task prediction on several molecular datasets and showed that additional topology learned was helpful to improving the prediction accuracy.

Addition to node features, edge attributes could also lead to different graph representations even for the same molecule. To jointly learn edge weights and node features using spectral graph convolution had not been explored in previous literature. Therefore, Shang *et al.* [36] proposed edge attention-based graph convolution network to deal with multi-relational graphs. In a multi-relational graph, each edge feature (binary or categorical) was considered as a relation. Each relation contained a dictionary with values to be learned for each relation category. The dictionary was shared across all graphs, therefore the proposed method was insensitive to input sizes. The convolution followed one-hop spectral convolution, with representations of different relations concatenated or weighted averaged before the last (prediction) layer. The authors evaluated both classification and regression tasks on four different datasets. Although the proposed method works for various input size due to dictionary setting, it was only feasible when edge attributes were binary or categorical variables.

The same atoms usually have different molecular properties according to their local chemical environments. However, previous studies treat all atoms and bonds with equal importance without respect to their chemical environments. To address this issue, Ryu *et al.* [37] introduced attention mechanism to distinguish atoms in different environments and to extract structural information determining molecular properties. Given a center node, an attention weight was multiplied for each neighbor before aggregation. The attention weight was determined by the representation of center node and neighbor node through a coupling matrix. The propagation followed one-hop spectral convolution. The network contained six convolutional layers and three fully connected layers and was verified for molecular property prediction tasks on three datasets. In addition, the authors included comprehensive interpretable results to compare proposed methods and general GCNs. For example, PC analysis for atom feature vectors under k-means clustering; comparison between molecules under different properties for two methods, showing that attention mechanism resulted in more reasonable and interpretable results.

Quantum mechanical property

Besides biological properties, another crucial point of speeding up drug discovery is to accurately simulate molecular dynamics. Quantum mechanical (QM) simulations, where millions of molecules are scanned to determine their energetics, is essential to predicting the efficacy of the molecules. However, traditional QM simulation methods, e.g. Density functional theory (DFT) method, scale with $\mathcal{O}(N^3)$ of the system size, are so expensive that either they are restricted for small systems or other approximation methods with less accuracy are adopted. Therefore, sev-

eral studies have been focused on developing deep architectures for fast screening quantum properties of molecules using GCNs.

Schutt *et al.* [30] proposed a deep tensor network for predicting molecular total energy. The atom representations were refined by a sequence of interaction passes, where the interaction was defined as a function of element-wise product of projected neighbor representation and projected distance between them. The energy contribution of each atom was then computed based on refined representation using two fully connected layers. The final molecular energy was obtained by summing energies from all atoms. Later on, Gilmer *et al.* [10] further reformulated various previous works [21, 22, 28–30] into a common framework called message-passing neural network as described in Spatial Convolution section. The authors proposed enn-s2s, an extended variant of message-passing networks to extract features from molecular graphs. The proposed framework generated neighborhood messages using both bond types and interatomic distances, followed by a set2set model [58] for inserting messages into center atoms. The authors evaluated their method on QM9 dataset that consists of various types of energies associated with molecules and other chemical properties and showed superior performance.

Due to discretization of atom distances, the filter learned in [10] was also discrete, incapable of capturing the gradual positional changes of the atoms and ended up with discrete energy predictions. Schutt *et al.* [31] proposed another graph convolution with continuous filters that mapped a position (distance) to a corresponding filter value. The architecture consisted of a sequence of atom-wise layers and interaction layers. Atom-wise layers recombined feature maps to new representations for each atom with weights shared across atoms while interaction layers updated atomic representations based on atom distances. A residual connection was employed in interaction layer such that the residual was obtained via a convolution layer that used radial basis function as continuous filter generator. The resulted residual was then directly added to atom representation as new updates. The authors demonstrated the strength of proposed method by predicting molecular energies and atomic forces on three different datasets.

Incorporate GCN with other learning architecture

GCN was originally inspired by traditional convolutional networks and thus there has been studies on generalizing GCN that is analogue to traditional CNN. Niepert *et al.* [27] proposed such framework in their work. The central challenge was to define a receptive filed as in traditional CNN but for arbitrary graphs. The authors achieved this by first selecting a fixed-length sequence of nodes from the graph using graph labeling procedures. Nodes from two different graphs were assigned to similar positions if their structure roles in the graphs were similar. Given selected node sequence, neighborhood was then assembled for each node by breadth-first search. After that, the receptive filed for a node was constructed by normalizing the assembled neighborhood. The normalization procedure aimed to find a labeling such that the expected distance between two graphs in the vector space and in the graph space was minimized. Each vertex attribute corresponded to an input channel. Based on generated receptive fields, arbitrary functions or architectures could be used for down streaming tasks. For instance, the authors used two convolutional layers followed by a dense layer and a softmax layer for several molecular bioactivity classification tasks.

There was also recent work on leveraging memory network for modeling molecules, by incorporating graph convolution. In [38], Pham *et al.* proposed graph memory networks for modeling molecules. The memory network consisted of a controller and external memory where memory cells encoded the representations for each node and controller iteratively read from and wrote to the memory. The representation of a graph was a weighted sum of all memory cells using attention mechanism. The representation for each memory cell was updated following spatial graph convolution paradigm. The controller was implemented using skip connections. Due to that the number of molecules in molecular activity prediction was usually limited, in order to render the overfitting brought by limited data, the author used multi-task scheme to evaluate their method. There were nine Bioassay activity tests in total and each task had one constant vector as input query to denote different tasks.

Deep architectures require large amount of training data in order to achieve significant improvements in predictive power and it is common that some tasks may contain insufficient data to make meaningful predictions. By combining one-shot learning with GCNs, Altae-Tran *et al.* [34] demonstrated that the learning of suitable distance metrics upon small molecules can be significantly improved by incorporating graph CNNs. The objective was to utilize information in the training tasks to construct strong classifiers for the testing tasks, using the similarity of learned representations between molecules in different task groups, i.e. the label of a query molecule was the weighted sum of labels of the support molecules based on their distance. In the proposed method, molecular representations were obtained by graph convolutional layers. Embeddings for generating task driven similarity measures were realized by an iterative long short term memory (LSTM) in a residual net fashion. At each iteration, a small portion was added to the current representation through an attentive LSTM framework. After obtaining similarity measure, the final prediction of the molecule label could be derived immediately. The authors evaluated their method on several well-known molecular property datasets. Moreover, by transferring a trained model to another dataset, the authors showed that one-shot models have limited power in generalizing to other unrelated systems.

Interaction prediction

Modern pharmacotherapy seeking compounds for disease treatment relies on several types of interactions: (1) interactions between ligands (small molecules that can be either pre-market compounds or existing drugs) and proteins (targets), (2) interaction between protein and protein for accurately locating interacting interfaces in pathway-regulation approaches and (3) drug–drug interactions(DDIs) for detecting potential adverse reactions and finding new uses of existing drugs. Therefore, interaction prediction is also crucial in drug development. However, the identification of such interactions is difficult due to costly experimental assays and rareness of complex interactions among drugs and proteins in small clinic settings. Computational methods enable large scale testing of potential interactions within a relatively short time upon the availability of large drug databases nowadays. Traditional computational methods aggregate drug and target features or similarity measures to identify interactions using standard machine learning algorithms [59–62]. The emergence of GCNs enables learnable representations for molecules, thus provide a new

paradigm for detecting interactions among drugs and targets using deep learning frameworks.

Ligand–protein (drug–target) interaction

Prediction of interaction between ligands (pre-market compounds or existing drugs) and proteins (targets) is a fundamental problem in drug discovery. However, several challenges remain due to the following reasons. First, the chemical space of synthesizable ligands is intractable, thus making the prediction still an open problem. Second, traditional methods often view interaction predication as a binary classification problem and are unable to handle cold-target problems [63, 64], i.e. target proteins never appear in the training set, which is very common in practice. Third, most traditional methods, though effective, lacks biological interpretations while it is important in bioinformatics domain.

Therefore, in order to address the limitations, Feng *et al.* [48] proposed a framework based on deep neural networks, to predict real-valued interaction strength between compounds and proteins instead of binary class labels. Its superiority relies on the learning of the molecular structure of each drug with GCNs, making the representations of drugs have richer chemical information encoded inside rather than taking each drug just as a node in Knowledge Graphs (KGs). Encoding molecular structures for drugs makes it possible to calculate the real-valued interaction strength between drugs and proteins. This study took both molecular structure and protein information as inputs, so it is capable of solving cold-target (and cold-drug) problems. Lau and Dror [65] developed several novel graph convolutional methods for crystallographic data, showing that the latent features learned through graph convolution are effective in other protein–ligand downstream regression/classification applications.

Gao *et al.* [39] proposed an end-to-end deep framework that was interpretable to predict drug–target interaction. The representation of a protein which consisted of a sequence of amino acids was learned though an LSTMrecurrent neural network while the neural fingerprint [28] of a drug molecule was obtained via graph convolution layers. A two-way attention mechanism was introduced to keep track of the likelihood of drug atoms interacting with each amino acid component, thus enables interpretability. The attention weights were then used to aggregate atoms into a molecule representation (drug) and amino acids into a protein (target) representation. Finally the attention-based representations were fed into a classifier to make a prediction. Besides amino acid sequence, gene ontology information was also used to derive protein embeddings. The learned interactions could be traced back to atom level and amino acid level to see which part of the drug and target contributes most to the corresponding interaction.

It is obvious that the performance of these above interaction predictions depends on the quality of these knowledge graphs, i.e. the validity of the existing edges in these KGs. However, many real-world knowledge graphs tend to contain relationships from multiple sources of varying quality. For example, drug–target interactions extracted from unstructured text (such as medical literature) are less reliable than manually curated ones. Therefore, an approach can make the noisier cheaper data effectively leveraged for more accurate predictions is necessary. To this end, Neil *et al.* [66] introduced an attention mechanism to GCNs by adding an attention parameter for the network to learn how much to trust an edge during training to alleviate the impact of noisy edges. It not only improves the performance on clean datasets, but also favorably accommodates noise in KGs.

Protein–protein interaction

Though not directly related to drug design, protein–target interaction provides target binding site information for developing drugs which regulate protein pathways. Such interactions can also be predicted using graph convolution networks. A successful application was revealed in [67]. Given two proteins, each was fed into a two-layer graph convolution network with embedding learned for each atom. Pair-wise representations were then obtained by concatenation of atom embeddings, followed by a fully connected layer to classify whether two atoms from two proteins reacted with each other. The features used in the experiments were handcrafted, including sequence-based features and features computed from the structure. The application was thus aimed at extracting information from existing features by considering the protein structure at propagation step.

Drug–drug interaction

DDI occurs when two drugs are co-administrated and the effect of one drug is influenced by the other. DDI prediction not only helps preventing adverse reactions but also helps finding new use of drugs (e.g. beneficial DDI gives guidance on drug combinations and can be seen as a new drug during treatment), providing additional information regarding the process of drug development. Therefore, we include works on DDI prediction in this section in spite of its indirect relation to narrow-sense drug development. Zitnik *et al.* [40] proposed a framework based on GCNs to identify DDI further on the polypharmacy side effects level, i.e. DDI with different types, providing additional guidance on drug combinational treatments. The proposed framework, *Decagon*, was characterized as multi-relational link prediction in multi-modal networks. It contained a graph convolutional encoder and a tensor factorization decoder. The encoder considered a graph with two types of entities: drug and protein; and three type of interactions: protein–protein interaction (PPI), drug–protein interaction (DTI) and DDI. Each side effect was represented as a different type of edge. The encoder followed spatial convolution convention with different weights assigned to nodes and neighbors with different relation types. Each layer considered the 1st order neighborhood and further neighbors are involved by stacking layers. The encoder mapped each node to an embedding, while the decoder aimed to reconstruct edge labels from learned representations. In the decoding stage, a relation score was calculated for each node pair using tensor factorization method. The entire model was optimized by minimizing the cross-entropy loss. The authors compared their methods with tensor factorization models and deep learning-based approaches such as DeepWalk [68] and achieved significant improvements.

In order to learn a more comprehensive and accurate drug similarity from heterogeneous data sources, Ma *et al.* [42] proposed a multi-view drug similarity integration framework using graph auto-encoders with attention mechanism. Each view was represented as a similarity matrix obtained by features from a data source. Different views were aggregated through an attention mechanism where a learnable attention weight was associated with each view. The attention weights were diagonalized to reduce computational complexity. Then the fused similarity kernel was fed into auto-encoder framework to extract informative representations. The propagation follows spectral graph convolution as in [22]. The authors evaluated the proposed methods on DDI identification tasks for both DDI occurrence and DDI with different relation types.

Structural information can be also combined with other data types to enhance the prediction performance. Asada *et al.* [41] proposed a framework for DDI extraction from texts. Given a text corpus with drug mentions, the goal was to classify drug–drug pairs into different types of interactions (mechanism, effect, advice, int and no interaction). Traditional methods used CNN to extract features to predict drug–drug relations [69]. The authors incorporated drug structure information into the framework by first pre-training a GCN for DDI identification (binary). After that, drug embeddings from the fixed GCN were concatenated with text embeddings from CNN, fed into a fully connected layer to predict interaction types. The drug structure was obtained from DrugBank and text documents were from DDIExtraction 2013 shared task [70]. The results showed that structure information was useful in extracting DDI from texts.

Synthesis prediction

Predicting organic reaction outcomes is a fundamental step in designing reaction sequences that produce specific target molecules. It involves two steps: candidate generation and filtering. The state-of-art solution for candidate generation is based on reaction templates, which specify a molecular subgraph pattern that it can be applied to and the corresponding graph transformation. The templates are either handcrafted or generated from the reaction database [71–73], which suffers from coverage and efficiency issues. Also, the matching procedure is expensive, making current approaches only feasible for small datasets and limited reaction types. In [43], Jin *et al.* proposed a template-free approach for reaction prediction leveraging graph neural networks. The inputs were predefined atom and bond features. Utilizing graph convolution, the network learned to identify a reaction center without using templates, by predicting reactivity score for each atom pair in the reactant molecules. The forward propagation follows spatial graph convolution. Top scored atom pairs were used to generate candidates products. A 2nd network was then built for ranking the candidates to find true reaction outcome. The authors proposed two models for both candidate generation and ranking stage. The proportion of reactions where true product was found in the model-generated candidate set was used as evaluation metric for reaction center identification. Coverage precision was used for candidate ranking. Results showed that the proposed method outperformed template-based approaches by a large margin.

In the domain of chemistry, retrosynthesis is the standard method for designing the production of chemical compounds. The principle is that, going backwards mentally, the compound is broken down into ever smaller components until the basic components have been obtained. This analysis provides the 'cooking recipe,' which is then used for working forwards in the laboratory to produce the target molecule, proceeding from the starting materials. Although easy in theory, the process presents difficulties in practice. Just like in chess, in every step or move you've got variety of possibilities to choose from. In chemistry, however, there are orders of magnitude more possible moves than in chess, and the problem is much more complex. Computer-aided retrosynthesis would be a highly valuable tool; however, past approaches were slow and provided results of unsatisfactory quality. To this end, Segler *et al.* [74] proposed a new computer-assisted synthesis planning (CASP) method based on deep learning which employs Monte Carlo Tree Search to efficiently discover retrosynthetic routes is now focused on putting the discovery into action. Compared to the traditional CASP method, this new method draws heavily on deep neural networks and the idea of reinforcement learning, and is an important improvement on traditional CASP methods. Compared to GCNs-based methods from computer science community, the design of this new method proposed by Segler *et al.* borrowed a lot of chemistry knowledge and reflected deep understanding of chemical reactions.

De novo molecular design

The ultimate goal of drug development is to discover new chemical structures with desired pharmacological properties. However, drug design in reality is difficult and expensive due to virtually infinite search space [75]. De novo molecular design thus aims to leverage computational methods to automate the molecular generation procedure. Early studies utilized rule-based methods to reduce search space and generate molecules [76, 77]. Generation models in deep learning enables effective generation of molecules based on SMILES strings [49, 78]. However, SMILES and fingerprints are too simple to deliver the topological information of molecular structures and leads to arelatively low learning accuracy. Molecular graphs intuitively and concisely express molecules with 2D topological information. Hence, they are widely adopted in chemical education as well as chemical informatics. Indeed, there have been efforts to develop DL models based on molecular graphs. GCNs, as an extension of the CNN, introduce new representations for molecules and thus enable direct realization of generating molecule graphs instead of pipeline implementation generating intermediate representations, shedding new light on molecule generation. The GCN benefits from the advantage of the CNN architecture; it performs with a high accuracy but a relatively low computational cost by utilizing fewer parameters compared to a fully connected multi-layer perceptron model. It can also identify important atom features that determine molecular properties by analyzing relations between neighboring atoms. The design of information propagation between neighboring atoms in molecular graph is a simple effective way to encode the structure information in molecules [28].

Simonovsky *et al.* [44] proposed variational autoencoders for generating graphs for small molecules. The encoder was defined by a variational posterior and the decoder by a generative distribution, each contained learnable parameters. The input of the encoder were graph adjacency matrix, edge feature tensor and node features. The authors used edge conditional convolutions as encoder. The decoder output a probabilistic fully connected graph on predefined number of nodes, from which discrete samples could be drawn. The model was trained by minimizing the upper bound on negative log-likelihood. Reconstruction ability of the autoencoder was facilitated by approximate graph matching for aligning generated graph and ground truth. The authors compared GCN-based VAE with traditional character-based generator [79] and grammar-based generator [80] and demonstrated proposed method of generating more chemically valid molecules. However, the proposed model was useful only for generating small graph due to its predefined node number always equal or larger than the actual molecule size; moreover, the output was a dense representation, making the parameters and matching complexity grow quickly.

Instead of generating the whole graph at once, a probabilistic approach which sequentially generates atoms and expands the graph was explored in Li *et al.* [45]. The generation process could be regarded as a sequence of decisions on whether add a node or edge and pick one node to connect with the new

node. The decisions were made based on probabilities depicted by a GCN. The network was learned by maximizing the expected joint log-likelihood of distribution over graphs and ordering of its nodes and edges. The generative model could be used to achieve conditional generation. Typical inputs were used to condition the generation process. The authors evaluated the proposed method on generating graphs of specific topologies and graphs of molecules. Results showed the proposed method generated more valid graphs compared to models with LSTM architectures. However, scalability was still a challenge for the proposed method since large graphs required more propagation steps to ensure information flow and training such graph models was more difficult than training LSTM networks. A similar framework has been adopted in Li *et al.* [46]. At each step, a graph transition (append, connect, terminate) was sampled and performed. The probability for sampling each transition was parametrized using GCN. The transition mapping was denoted as the decoding scheme. The authors explored two decoding policies, one was parametrized as a Markov process, the other used molecule level recurrent unit to increase the scalability of the model.

A major drawback of generating molecular graphs in an atom-wise fashion was the production of low-quality intermediates. Jin *et al.* [47] proposed a two-stage molecule graph-generating approach by exploiting the substructures as valid components, therefore significantly improved the quality of generated molecules. The method first generated a junction tree structure to represent the subgraph components, which were used as building blocks. In the second phase, the subgraphs were assembled together into a molecular graph. The graph was encoded by a standard GCN. The final graph representation was obtained by averaging all node representations. The tree structure was encoded using a message passing network where message was constructed through a gated recurrent unit. The final tree representation was encoded from the tree root. In the decoding procedure, a tree was first decoded, then the graph was decoded from the junction tree by enumerating and ranking subgraphs. The authors evaluated the proposed methods in three ways: (1) molecule reconstruction and validity: reconstructed input molecules from their latent representations, and decoded valid molecules when sampling from prior distribution; (2) Bayesian optimization: tested how the model could produce novel molecules with desired properties; and (3) constrained molecule optimization: modified given molecules to improve specified properties, while constraining the degree of deviation from the original molecules. The proposed method was compared to state-of-art SMILES-based VAEs [80, 81] and showed remarkable improvement.

Databases for drug discovery and molecular bioinformatics

So far, we have presented applications of GCNs on drug discovery. In this section, we provide a summary in Table 3 for the databases that used by the surveyed papers. We divide them into groups corresponding to the drug applications. We also include several databases that are integrated from multiple sources and available for open access.

Molecular property and activity

Biological property and activity

PubChem [54] is a large public database of chemical molecules and their activities against biological assays. It consists of three primary channels: PubChem BioAssay (PCBA), PubChem Compound and PubChem Substance. Substance includes chemical compounds and their information reported from all the contributors. Compound is derived from Substance and consists of normalized representations of the chemical structures in Substance. BioAssay contains bioactivity results from 1.25 million high-throughput screening programs. PubChem compound ids are commonly used across different chemical databases for the reference of the same compounds.

The Maximum Unbiased Validation (MUV) [55] dataset is a subset of PCBA, generated using refined nearest neighbor analysis and is unbiased with regard to analogue bias and artificial enrichment. It contains 17 challenging tasks for around 90 000 compounds and is specifically designed for validation of virtual screening techniques. The positives examples in these datasets are selected to be structurally distinct from one another.

ChEMBL [82] is a database of bioactive molecules with drug-like properties. It contains binding, functional and ADMET (absorption, distribution, metabolism, excretion and toxicity) information of the molecular compounds which is manually derived from primary published literature followed by further standardization. The database provides 5.4 million bioactivity measurements for more than 1 million compounds and 5200 protein targets.

ZINC [83, 84] contains a curated collection of commercially available chemical compounds prepared especially for virtual screening. It provides information associated with the molecules from more than 20 resources such as chemical structures, bioactivities and target information. It contains over 200 million compounds in ready-to-dock, 3D formats.

NCI [85] is a database of chemical compounds that are screened for activities against different cancer cell lines. It includes biological test data and chemical structures for around 250K molecules. The HIV dataset, produced by the Drug Therapeutics Program [86] using AIDS antiviral screen, is also part of the NCI database, where compounds were checked for evidence of anti-HIV activity. Screening results were evaluated and placed into three categories: confirmed active, confirmed inactive and confirmed moderately active. The dataset contains screening results for 43850 compounds together with the structure information.

Tox21 [57], ToxCast [87] and ClinTox [63] are datasets containing molecular toxicity information. Tox21, known as the Toxicology in the 21st Century Program, is a collaboration between federal agencies aiming to develop innovative test methods to better predict how substances may affect humans and the environment. It contains qualitative toxicity measurements on 12 biological targets, including nuclear receptors and stress response pathways. ToxCast is a dataset provided by Environmental Protection Agency for the developing of efficient methods to prioritize, screen and evaluate chemicals. It contains toxicology data for 1800 chemicals from a broad range of sources using high-throughput screening methods and computational toxicology approaches. ClinTox is part of the MoleculeNet [63] benchmark data that includes drug compounds that failed clinical trials for toxicity reasons and those approved by the FDA. It contains two classification tasks for 1491 drug compounds.

FreeSolv [88] is a database of experimental and calculated hydration free energies for small molecules in water. The values are derived from alchemical free energy calculations using molecular dynamics simulations. It currently contains molecular property data for 643 molecules together with their chemical structures. ESOL [89] is another dataset containing aequeous solubility data for thousands of low molecular weight compounds.

Table 3. Database used in papers in this survey

Category	Database	Owner	Link	Release	Entities	Note
Molecular property and activity	CEPDB [90]	Harvard University	http://www.molecularspace.org/explore/	2011	Molecular properties	The Clean Energy Project Database
	PCBA [54]	National Institutes of Health (NIH)	https://pubchem.ncbi.nlm.nih.gov/	2004	Molecular bioactivities	PCBA Database
	ChEMBL [82]	European Molecular Biology Laboratory (EMBL)	https://www.ebi.ac.uk/chembl/	2009	Protein targets and bioactivities	European Molecular Biology Laboratory Database
	ZINC [83, 84]	University of California, San Francisco (UCSF)	http://zinc15.docking.org/	2004	Compounds	
	NCI [85]	National Cancer Institute	https://cactus.nci.nih.gov/download/nci/	1999	Molecular bioactivities	
	Tox21 [57]	US Environmental Protection Agency (EPA), NIH	https://ntp.niehs.nih.gov/results/tox21/index.html	2015	Toxicity measurements	Toxicology in the 21st Century
	ToxCast [87]	US Environmental Protection Agency (EPA)	https://www.epa.gov/chemical-research/toxicity-forecasting	2016	Toxicity measurements	Toxicity Forecasting
	FreeSolv [88]	University of California, Irvine (UCI)	https://escholarship.org/uc/item/6sd403pz	2013	Molecule energy	
	MUV [55]	National Institutes of Health (NIH)	http://www.pharmchem.tu-bs.de/lehre/baumann/MUV.html	2009	Molecular bioactivities	MUV
	DUD-E [56]	University of California, San Francisco (UCSF)	http://dude.docking.org/	2012	Compound affinities	Directory of useful decoys, Enhanced
	HIV [86]	National Cancer Institute	https://wiki.nci.nih.gov/display/NCIDTP-data/AIDS+Antiviral+Screen+Data	2004	Compound activities	
	ADME [121]	University of California San Diego (UCSD)	http://modem.ucsd.edu/adme/databases/databases.htm	2010	Molecular properties	Absorption, Distribution, Metabolism, Excretion
	ESOL [89]	Jealott's Hill International Research Centre	-	2004	Molecular properties	Estimated Solubility
DDI/PPI	SIDER [97]	European Molecular Biology Laboratory (EMBL)	http://sideeffects.embl.de/	2010	ADRs	Side Effect Resource
	STITCH [99] Interacting Chemicals	European Molecular Biology Laboratory (EMBL)	http://stitch.embl.de	2008	Protein-chemical interaction	Search Tool for
	OFFSIDES [98]	Columbia University	http://tatonettilab.org/resources/tatonetti-stm.html	2012	ADRs	Off-label Side Effects

Continued

Table 3. Continued

Category	Database	Owner	Link	Release	Entities	Note
	TWOSIDES [98]	Columbia University	http://tatonettilab.org/resources/tatonetti-stm.html	2012	polypharmacy side effects	Drug Drug Side Effect
	TTD [103–106]	National University of Singapore	http://bidd.nus.edu.sg/group/cjttd/	2002	Target, Drugs	Therapeutic Target Database
	DBD5 [107]	University of Massachusetts	https://zlab.umassmed.edu/zdock/benchmark.shtml	2015	Protein protein interaction	Docking Benchmark Database
	BindingDB [108]	University of California San Diego (UCSD)	https://www.bindingdb.org/bind/index.jsp	2002	Target, Drugs	
	DrugBank [101, 102]	University of Alberta	https://www.drugbank.ca/	2006	Drug, Target	
Synthesis Reactions	USPTO [109]	-	https://bitbucket.org/dan2097/patent-reaction-extraction	2014	Chemical reactions	Patent reaction extraction
	MoleculeNet [63]	Integrated from multiple sources	http://moleculenet.ai/datasets-1	2017	Multiple categories	
Integrated Database	Decagon [40]	Integrated from multiple sources	http://snap.stanford.edu/decagon/	2018	DDI, PPI	
	QM [30, 96]	Integrated from multiple sources	http://quantum-machine.org/datasets/	2013-2016	Molecules	

Quantum chemical propety

The Clean Energy Project Database (CEPDB) [90] is the database for Harvard Clean Energy Project, a virtual high-throughput screening initiative to identify promising new candidates for carbon-based solar cell materials. The project established an automated *in silico* framework to study the potential candidate structures for organic photovoltaics. The database provides information derived from DFT simulations, on 2.3 million candidate molecular motifs which include both known and virtual compounds.

Quantum Machine (QM) [30, 91–96] is a database that contains molecules and their quantum mechanical properties such as atom energies and forces. It includes data from multiple sources and aims to accelerate the development of a machine that can quickly and accurately simulate quantum-chemical systems from first principles. It has released QM7, QM8, QM9 and MD datasets since year 2013.

Interaction database

The Side Effect Resource (SIDER) [97] is a database containing information on marketed medicines and their recorded adverse drug reactions (ADRs). The information is extracted from public documents and package inserts. Currently, there are 1430 drugs and 5868 side effects (SEs), with 139 756 drug–SE pairs in the database. A dataset of drug indications is also provided to reduce false positives.

The Off-label Side Effect (OFFSIDES) [98] is a database of 438 801 off-label side effects for 1332 drugs and 10 097 adverse events. The off-label denotes side effects that are not listed on the FDA's official drug label. The information was collected using adverse event reporting systems which collect reports from patients, doctors and companies. Generated by the same lab, TWOSIDES [98] is a resource of polypharmacy side effects for pairs of drugs. It contains only side effects caused by the combination of drugs rather than by any single drug. Like OFFSIDES, the information was generated through adverse event reporting systems. The database contains 868 221 significant associations between 59 220 pairs of drugs and 1301 adverse events.

Search Tool for Interacting Chemicals (STITCH) [99] is a database that integrates data sources for 430 000 chemicals and more than 9 600 000 proteins into a single resource. It provides binding affinities between chemicals and different interaction targets, resulting in global networks for both chemical–chemical interactions and chemical–protein interactions. The protein space is shared with STRING [100], a protein–protein network database. All the interactions are associated with a confidence score that represent link strength between network entities.

DrugBank [101, 102] is a comprehensive database that contains detailed molecular information about drugs. There are two types of drugs in the database, FDA-approved small molecules and biotech drugs. It also provides information on targets, indications and pathways. The data fields for each drug are hyperlinked to other databases (PubChem, ChEBI, PDB, KEGG, etc.). The newest release contains 11 680 drug entries and 5129 non-redundant protein (i.e. drug target/enzyme/transporter/carrier) sequences that are linked to these drug entries.

Therapeutic Target Database (TTD) [103–106] contains information about known therapeutic proteins and nucleic acid targets described in the literature. Besides targets, it also provides information for corresponding drugs, target disease conditions and pathway information. The sequence and structure infor-

mation are also available via cross-links to other databases. Currently the database contains 3101 targets and 34 019 drugs.

Docking Benchmark Database (DBD5) [107] is a benchmark database that contains a collection of distinct protein docking test cases. The complexes are a subset selection of structures from the Protein Data Bank (PDB). It includes individually crystallized receptor and ligand PDBs and the co-crystallized complex PDBs for testing protein docking algorithms.

BindingDB [108] is a database of measured binding affinities on the interactions of drug targets proteins and small drug-like molecules. The database contains 1 454 892 binding data, with 7082 protein targets and 652 068 small molecules.

Synthesis database

USPTO [109] is a database that contains reaction information for chemicals reactants. The reactions are extracted from patent applications in United States Patent and Trademark Office (USPTO). A total number of 424 621 exact atom-mapped reactions were extracted and included in the database.

Integrated benchmark database

MoleculeNet [63] is a benchmark designed for testing machine learning methods of molecular properties. It is built upon multiple public databases and covers over 700 000 compounds tested on a range of different properties. The datasets are characterized into four categories: physiology, biophysics, physical chemistry and quantum mechanics. It contains a sub-collection of the aforementioned databases, including QM, Toxicity datasets (Tox21, ToxCast. ClinTox), bioactivity datasets (PCBA, MUV, HIV), bioproperty datasets (ESOL, FreeSolv) and interaction dataset (SIDER). It also contains other datasets such as BBBP, binary labels of blood–brain barrier penetration; Lipophilicity, experimental results of octanol/water distribution coefficient; BACE, binding results for a set of inhibitors of human β-secretase 1; and PDBbind, Binding affinities for bio-molecular complexes.

Decagon [40] provides preprocessed data of protein targets and drug molecules and their interactions. The network consists of DTI, PPI and DDI that were derived from multiple data sources. Typically, DTIs were extracted from STITCH database; DDIs were derived from TWOSIDES database; PPIs were integrated from human PPI network and STRING database. Drug side effects combined from SIDER and OFFSIDES are also included.

To improve the consistency in the evaluation of generative models for *de novo* molecular design, Brown *et al*. [110] introduced an evaluation framework, GuacaMol, based on a suite of standardized benchmarks. The benchmark tasks encompass measuring the fidelity of the models to reproduce the property distribution of the training sets, the ability to generate novel molecules, the exploration and exploitation of chemical space and a variety of single and multi-objective optimization tasks. The benchmarking framework is available as an open-source Python package.

Polykovskiy *et al*. [111] introduced MOSES for drug discovery (https://github.com/molecularsets/), which implements several popular molecular generation models and includes a set of metrics that evaluate the diversity and quality of generated molecules. MOSES is meant to standardize the research on molecular generation and facilitate the sharing and comparison of new models. Additionally, it provides a large-scale comparison of existing state-of-the-art models and elaborates on current challenges for generative models that might prove fertile ground for new research. This platform should allow for a fair and comprehensive comparison of new generative models.

Discussion

GCN can be characterized as imposing relational inductive biases on modeling structured data as summarized in a recent review paper [112]. The emerging of GCNs and their successful applications in domains such as molecular bioinformatics depict the power of combining deep learning, which assumes a minimum priori, and structured approaches, that impose strict constraints on the inputs and models. In a broader sense, GCN is applicable to any data structures that can be represented as graphs and thus of tremendous significance in various real world applications.

In the field of drug discovery, deep learning enables large scale prediction of chemical properties and activities within a relatively short time, automating and speeding up the process of drug discovery. The introduction of graph convolutional network provides more accurate predictions compared to traditional methods by intrinsically considering the molecular structures. Moreover, when combined with other mechanisms such as attentions, graph convolutional network generates biological interpretable results, for instance, in interaction predictions.

However, despite the recent success that graph convolutional networks have achieved, challenges remain in order to fully release the potentials of graph convolutional networks on drug discovery. Herein, we summarize the challenges as well as opportunities in the following subsections.

Database challenges and opportunities

Deep models require large amount of data in order to learn the complex relationships between inputs and targets. Although large databases are becoming available, insufficiency still exists due to following reasons. First, for certain molecular properties (e.g. solubility, toxicity), available data are either limited or spread as different small datasets. Besides collecting more data, a unified platform to better integrate different data sources is also essential for cross reference and acquiring more sufficient data. Second, existing databases curate mostly positive samples. For example, in interaction networks, if two entities interact, they are included into the database while information for non-interaction pairs are often neglected. This not only induces imbalance problems, but also leads to situation where control information are not available at all. Current computational methods either design new objectives or manually generate negative samples to confront the limitation whereas identifying negative samples is in fact difficult. Therefore, officially curated negative samples are important for more accurate predictions using machine learning methods. Third, more specified and detailed information can be added to the database. For example, even for interacted pairs, the interaction could be of different functionality. When a drug interacts with another drug, the effect could be either synergistic or antagonistic while in reality only antagonistic effects are recorded [113]. In fact, synergistic DDI effects are beneficial such that it can provide important guidance for drug combination in patient care [114].

Methodology challenges and opportunities

Molecular compounds, especially proteins, are 3D-shaped entities where the folding structure in 3D space greatly impact their

functionalities [115]. Current graph convolution mostly operates on flat 2D graphs where structure information in the third dimensional space is neglected. There has been some attempts on developing convolution operator on 3D structures [116–118] and extending graph convolutional networks to 3D structures is definitely a direction worth exploring.

On the other hand, high-order structures are less focused and explored on 2D graphs while it may actually provide additional information. For example, in a disease–protein network analysis [119], the authors found that disease pathways do not correspond to single well-connected components and higher-order network structures (motifs) provide additional information to disease pathway discovery. Monti *et al*. [120] investigated spectral graph convolution for motifs and validated on citation network data. Exploration of motif-based graph convolution and its application on drug discovery has not been well established yet and therefore is a promising future direction.

Existing graph convolution operates on regular graphs whereas for certain relations, a hyper graph can be formed. For example, different drugs may share the same ADRs, targets or indications which can be converted into a hyper graph. How to define appropriate convolution on hyper graphs in order to extract useful information has not been investigated in the literature.

Network design challenges and opportunities

Currently graph convolutional networks are commonly used in two scenarios. In the first case, each data point is represented as a graph where prediction happens at graph level, e.g. molecule property and activity prediction. In the second case, only one graph is presented and each data point denotes one node in the graph, e.g. drug–target interaction network. The 1st one aims to extract structure information for each entity while the 2nd one aims to propagate affinity information between entities.

One improvement that can be made is to combine two scenarios and construct an end-to-end framework, utilizing both low-level structure information and global network structure information. For the 2nd case, usually at most two entities are presented in the interaction network while in reality, drug discovery involves more than two entities and additional entities are helpful in providing additional information. For instance, in a larger network, the entities could be drugs, targets, diseases and even ADRs (relations). However, graph convolution for multi-modal networks with more than two modalities is not well considered in the literature. One possible reason for this is that although graph convolutional network is applicable to any data that can be represented as a graph, however, the graph representations are not always explicit for existing data. For example, molecules are naturally graphs while patient records are not although they can be carefully designed to form certain graphs. In the case of multi-model network for drug discovery, entities with three or more types are often more complex and therefore designing an appropriate graph in order to apply the convolution framework is critical.

Interpretability challenges and opportunities

Due to the complexity of deep neural network, it always suffers from critiques of lack of interpretability. In the domain of bioinformatics and health-related feild, however, interpretability is very important when assessing a computational model and for better understanding the underlying mechanisms. Therefore, designing subtle architectures that allow for interpreting or visualizing complex relationships is a challenge as well as opportunities for GCN applications in drug discovery. Previous studies [39, 67] successfully showed interaction complexes between drug and protein entities (DTI and PPI) using either attention mechanisms or node-pair scores. Other mechanisms are needed to further improve the explainability of learned models.

Key Points

- GCN is a class of computational techniques aiming at extracting features from general graphs through graph convolutions.
- GCN can be applied in computational drug development if we treat each drug molecule structure as a graph with the atoms as nodes and bonds as edges.
- GCN has been successfully applied in many drug development problems including QSAR/QSPR, drug–target/DDI prediction, and *denovo* drug molecule structure design.
- There are many publicly available drug-related databases for developing GCN-based approaches in various computational drug development applications.
- There are still challenges on GCN for computational drug development, including comprehensive data, optimal model design and model interpretability.

Funding

This work has been supported by National Science Foundation (NSF) IIS-1650723 (to F.W.), NSF IIS-1750326 (to F.W.), NSF IIS-1615597 (to J.Z.), NSF IIS-1749940 (to J.Z.), NSF IIS-1565596 (to J.Z.), Office of Naval Research (ONR) N00014-18-102585 (to F.W.) and ONR N00014-17-1-2265 (to J.Z.).

References

1. LeCun Y, Bengio Y, Hinton G. Deep learning. *Nature* 2015; **521**(7553):436.
2. Young T, Hazarika D, Poria S, Cambria E. Recent trends in deep learning based natural language processing. *IEEE Computational Intelligence Magazine* 2018;**13**(3):55–75.
3. Rawat W, Wang Z. Deep convolutional neural networks for image classification: a comprehensive review. *Neural Comput* 2017;**29**(9):2352–449.
4. Wang C, Liu J, Luo F, *et al*. Pairwise input neural network for target-ligand interaction prediction. In: *Bioinformatics and Biomedicine (BIBM), 2014 IEEE International Conference on* , pp. 67–70. IEEE.
5. Wallach I, Dzamba M, Heifets A. Atomnet: A deep convolutional neural network for bioactivity prediction in structure-based drug discovery. *arXiv preprint arXiv:1510.02855*, 2015.
6. Wang Y, Zeng J. Predicting drug–target interactions using restricted boltzmann machines. *Bioinformatics* 2013; **29**(13):i126–34.
7. Wan F, Zeng J. *Deep learning with feature embedding for compound–protein interaction prediction. bioRxiv*, 2016, 086033.
8. LeCun Y, *et al*. Generalization and network design strategies. *Connectionism Perspect*, 1989, 143–55.

9. Chung FRK. *Spectral Graph Theory*, Vol. 92. United States: American Mathematical Soc., 1997.
10. Gilmer J, Schoenholz SS, Riley PF, *et al*. Neural message passing for quantum chemistry. In: *Proceedings of the 34th International Conference on Machine Learning*, *2017*. Vol. 70, p. 1263–72. ICML, Sydney.
11. Monti F, Boscaini D, Masci J, *et al*. Geometric deep learning on graphs and manifolds using mixture model cnns. In: *Proc. CVPR*, Vol. 1. Hawaii: CVPR, 2017, 3.
12. Gawehn E, Hiss JA, Schneider G. Deep learning in drug discovery. *Mol Inform* 2016;**35**(1):3–14.
13. Jing Y, Bian Y, Hu Z, *et al*. Deep learning for drug design: an artificial intelligence paradigm for drug discovery in the big data era. *AAPS J* 2018;**20**(3):58.
14. Chen H, Engkvist O, Wang Y, *et al*. The rise of deep learning in drug discovery. *Drug Discov Today* 2018; **23**(6):1241–50.
15. Zhang Z, Cui P, Zhu W. Deep learning on graphs: a survey. *CoRR* 2018. abs/1812.04202.
16. Zhou J, Cui G, Zhang Z, *et al*. Graph neural networks: a review of methods and applications. *arXiv preprint arXiv:1812.08434*, 2018.
17. Bruna J, Zaremba W, Szlam A, LeCun Y. Spectral networks and locally connected networks on graphs. *arXiv preprint arXiv:1312.6203*, 2013.
18. Henaff M, Bruna J, LeCun Y. Deep convolutional networks on graph-structured data. *arXiv preprint arXiv:1506.05163*, 2015.
19. Duvenaud DK, Maclaurin D, Iparraguirre J, *et al*. Convolutional networks on graphs for learning molecular fingerprints. In: C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett (eds). *Advances in Neural Information Processing Systems*, Vol. 28. Spain: NIPS, 2015, 2224–32.
20. Li Y, Tarlow D, Brockschmidt M, Zemel R. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493*, 2015.
21. Defferrard M, Bresson X, Vandergheynst P. Convolutional neural networks on graphs with fast localized spectral filtering. In: *Advances in Neural Information Processing Systems*. Barcelona: NIPS, 2016, 3844–52.
22. Kipf TN, Welling M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv: 1609.02907*, 2016.
23. Dai H, Dai B, Song L. Discriminative embeddings of latent variable models for structured data. In: *International Conference on Machine Learning*, New York: ICML, 2016, 2702–11.
24. Weisfeiler B, Lehman AA. A reduction of a graph to a canonical form and an algebra arising during this reduction. *Nauchno-Technicheskaya Informatsia* 1968;**2**(9):12–6.
25. Xu K, Hu W, Leskovec J, Jegelka S. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826*, 2018.
26. Davies BE, Leydold J, Stadler PF. Discrete nodal domain theorems. *Linear Algebra and its Applications* 2001;**336**(1–3): 51–60 .
27. Niepert M, Ahmed M, Kutzkov K. Learning convolutional neural networks for graphs. In: *International Conference on Machine Learning*, New York: ICML, 2016, 2014–23.
28. Duvenaud DK, Maclaurin D, Iparraguirre J, *et al*. Convolutional networks on graphs for learning molecular fingerprints. In: *Advances in Neural Information Processing Systems*, 2015, 2224–32.
29. Kearnes S, McCloskey K, Berndl M, *et al*. Molecular graph convolutions: moving beyond fingerprints. *J Comput Aided Mol Des* 2016;**30**(8):595–608.
30. Schutt KT, Arbabzadah F, Chmiela S, *et al*. Quantum-chemical insights from deep tensor neural networks. *Nat Commun* 2017;**8**:13890.
31. Schutt K, Kindermans P-J, Felix HES, *et al*. Schnet: a continuous-filter convolutional neural network for modeling quantum interactions. In: *Advances in Neural Information Processing Systems*, Long Beach: NIPS, 2017, 992–1002.
32. Li J, Cai D, He X. Learning graph-level representation for drug discovery. *arXiv preprint arXiv:1709.03741*, 2017.
33. Liu K, Sun X, Jia L, *et al*. Chemi-net: a graph convolutional network for accurate drug property prediction. *arXiv preprint arXiv:1803.06236*, 2018.
34. Altae-Tran H, Ramsundar B, Pappu AS, *et al*. Low data drug discovery with one-shot learning. *ACS Cent Sci* 2017; **3**(4):283–93.
35. Li R, Wang S, Zhu F, Huang J. Adaptive graph convolutional neural networks. *Thirty-Second AAAI Conference on Artificial Intelligence 2018*. AAAI, New Orleans.
36. Shang C, Liu Q, Chen K-S, *et al*. Edge attention-based multi-relational graph convolutional networks. *arXiv preprint arXiv:1802.04944*, 2018.
37. Ryu S, Lim J, Kim WY. Deeply learning molecular structure–property relationships using graph attention neural network. *arXiv preprint arXiv:1805.10988*, 2018.
38. Pham T, Tran T, Venkatesh S. Graph memory networks for molecular activity prediction. In: *24th International Conference on Pattern Recognition (ICPR) 2018*, 639–44. IEEE, Beijing.
39. Gao KY, Fokoue A, Luo H, *et al*. Interpretable drug target prediction using deep neural representation. In: *IJCAI*. Stockholmsmassan: IJCAI, 2018, 3371–7.
40. Zitnik M, Agrawal M, Leskovec J. Modeling polypharmacy side effects with graph convolutional networks. *Bioinformatics* 2018;**34**(13):i457–i466.
41. Asada M, Miwa M, Sasaki Y. Enhancing drug–drug interaction extraction from texts by molecular structure information. *arXiv preprint arXiv:1805.05593*, 2018.
42. Ma T, Xiao C, Zhou J, Wang F. Drug similarity integration through attentive multi-view graph auto-encoders. *arXiv preprint arXiv:1804.10850*, 2018.
43. Jin W, Coley C, Barzilay R, *et al*. Predicting organic reaction outcomes with Weisfeiler-Lehman network. In: *Advances in Neural Information Processing Systems*. Long Beach: NIPS, 2017, 2604–13.
44. Simonovsky M, Komodakis N. Graphvae: towards generation of small graphs using variational autoencoders. *arXiv preprint* arXiv:*1802.03480*, 2018.
45. Li Y, Vinyals O, Dyer C, *et al*. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324*, 2018.
46. Li Y, Zhang L, Liu Z. Multi-objective de novo drug design with conditional graph generative model. *arXiv preprint arXiv:1801.07299*, 2018.
47. Jin W, Barzilay R, Jaakkola T. Junction tree variational autoencoder for molecular graph generation. *arXiv preprint arXiv:1802.04364*, 2018.
48. Feng Q, Dueva E, Cherkasov A, Ester M. Padme: a deep learning-based framework for drug–target interaction prediction. *arXiv preprint arXiv:1807.09741*, 2018.
49. Olivecrona M, Blaschke T, Engkvist O, *et al*. Molecular de-novo design through deep reinforcement learning. *J Chem* 2017;**9**(1):48.

50. Svetnik V, Liaw A, Tong C, *et al*. Forest random: a classification and regression tool for compound classification and qsar modeling. *J Chem Inf Comput Sci* 2003;**43**(6): 1947–58.
51. Chen B, Sheridan RP, Hornak V, *et al*. Comparison of random forest and pipeline pilot naive bayes in prospective qsar predictions. *J Chem Inf Model* 2012;**52**(3):792–803.
52. Tropsha A. Best practices for qsar model development, validation, and exploitation. *Mol Inform* 2010;**29**(6–7):476–88.
53. Glen RC, Bender A, Arnby CH, *et al*. Circular fingerprints: flexible molecular descriptors with applications from physical chemistry to adme. *IDrugs* 2006;**9**(3):199.
54. Wang Y, Xiao J, Suzek TO, *et al*. Pubchem's bioassay database. *Nucleic Acids Res* 2011;**40**(D1):D400–12.
55. Rohrer SG, Baumann K. Maximum unbiased validation (muv) data sets for virtual screening based on pubchem bioactivity data. *J Chem Inf Model* 2009;**49**(2):169–84.
56. Mysinger MM, Carchia M, Irwin JJ, *et al*. Directory of useful decoys, enhanced (dud-e): better ligands and decoys for better benchmarking. *J Med Chem* 2012;**55**(14): 6582–94.
57. Mayr A, Klambauer G, Unterthiner T, *et al*. Deeptox: toxicity prediction using deep learning. *Front Environ Sci* 2016; **3**(80).
58. Vinyals O, Bengio S, Kudlur M. Order matters: sequence to sequence for sets. *arXiv preprint arXiv:1511.06391*, 2015.
59. Cheng F, Zhao Z. Machine learning-based prediction of drug–drug interactions by integrating drug phenotypic, therapeutic, chemical, and genomic properties. *J Am Med Inform Assoc* 2014;**21**(e2):e278–86.
60. Zitnik M, Zupan B. Collective pairwise classification for multi-way analysis of disease and drug data. In: *Biocomputing 2016: Proceedings of the Pacific Symposium*. 2016, 81–92. World Scientific, Hawaii.
61. Huang L, Li F, Sheng J, *et al*. Drugcomboranker: drug combination discovery based on target network analysis. *Bioinformatics* 2014;**30**(12):i228–36.
62. Vilar S, Harpaz R, Uriarte E, *et al*. Drug–drug interaction through molecular structure similarity analysis. *J Am Med Inform Assoc* 2012;**19**(6):1066–74.
63. Wu Z, Ramsundar B, Feinberg EN, *et al*. Moleculenet: a benchmark for molecular machine learning. *Chem Sci* 2018; **9**(2):513–30.
64. Gomes J, Ramsundar B, Feinberg EN, Pande VS. Atomic convolutional networks for predicting protein–ligand binding affinity. *arXiv preprint arXiv:1703.10603*, 2017.
65. Lau T, Dror R. *Brendan: A Deep Convolutional Network for Representing Latent Features of Protein–Ligand Binding Poses*.
66. Neil D, Briody J, Lacoste A, *et al*. Interpretable graph convolutional neural networks for inference on noisy knowledge graphs. *arXiv preprint arXiv:1812.00279*, 2018.
67. Fout A, Byrd J, Shariat B, *et al*. Protein interface prediction using graph convolutional networks. In: *Advances in Neural Information Processing Systems*, Long Beach: NIPS, 2017, 6533–42.
68. Perozzi B, Al-Rfou R, Skiena S. Deepwalk: online learning of social representations. In: *Proceedings of the 20th ACM SIGKDD International Conference on Knowledge discovery and Data Mining*. New York: ACM, 2014, 701–10.
69. Zeng D, Liu K, Lai S, *et al*. Relation classification via convolutional deep neural network. In: *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers* 2014, 2335–44. ACM, Ireland.
70. Segura-Bedmar I, Martínez P, Zazo MH. Semeval-2013 task 9: extraction of drug–drug interactions from biomedical texts (ddiextraction 2013). In: *Second Joint Conference on Lexical and Computational Semantics (* SEM), Volume 2: Proceedings of the Seventh International Workshop on Semantic Evaluation (SemEval 2013)*, Vol. 2, 2013, 341–50. SemEval, Atlanta.
71. Chen JH, Baldi P. No electron left behind: a rule-based expert system to predict chemical reactions and reaction mechanisms. *J Chem Inf Model* 2009;**49**(9):2034–43.
72. Coley CW, Barzilay R, Jaakkola TS, *et al*. Prediction of organic reaction outcomes using machine learning. *ACS Cent Sci* 2017;**3**(5):434–43.
73. Wei JN, Duvenaud D, Aspuru-Guzik A. Neural networks for the prediction of organic chemistry reactions. *ACS Cent Sci* 2016;**2**(10):725–32.
74. Segler MHS, Preuss M, Waller MP. Planning chemical syntheses with deep neural networks and symbolic ai. *Nature* 2018;**555**(7698):604.
75. Schneider G, Fechner U. Computer-based de novo design of drug-like molecules. *Nat Rev Drug Discov* 2005;**4**(8): 649.
76. Böhm H-J. The computer program ludi: a new method for the de novo design of enzyme inhibitors. *J Comput Aided Mol Des* 1992;**6**(1):61–78.
77. Mauser H, Stahl M. Chemical fragment spaces for de novo design. *J Chem Inf Model* 2007;**47**(2):318–24.
78. Segler MHS, Kogej T, Tyrchan C, *et al*. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent Sci* 2017;**4**(1):120–31.
79. Gómez-Bombarelli R, Wei JN, Duvenaud D, *et al*. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent Sci* 2018;**4**(2): 268–76.
80. Kusner MJ, Paige B, Miguel Hernández-Lobato J. Grammar variational autoencoder. In: *Proceedings of the 34th International Conference on Machine Learning 2017*, Vol. 70, pp. 1945–54. JMLR.org, Sydney.
81. Dai H, Tian Y, Dai B, Skiena S, Song L. Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786*, 2018.
82. Gaulton A, Bellis LJ, Patricia Bento A, *et al*. Chembl: a large-scale bioactivity database for drug discovery. *Nucleic Acids Res* 2011;**40**(D1):D1100–7.
83. Irwin JJ, Sterling T, Mysinger MM, *et al*. Zinc: a free tool to discover chemistry for biology. *J Chem Inf Model* 2012; **52**(7):1757–68.
84. Sterling T, Irwin JJ. Zinc 15–ligand discovery for everyone. *J Chem Inf Model* 2015;**55**(11):2324–37.
85. National Cancer Institute. NCI open database compounds. https://cactus.nci.nih.gov/download/nci/, 1999. (29 July 2018, date last accessed).
86. Monga M, Sausville EA. Developmental therapeutics program at the NCI: molecular target and drug discovery process. *Leukemia* 2002;**16**(4):520.
87. Richard AM, Judson RS, Houck KA, *et al*. Toxcast chemical landscape: paving the road to 21st century toxicology. *Chem Res Toxicol* 2016;**29**(8):1225–51.
88. Mobley DL, Guthrie JP. Freesolv: a database of experimental and calculated hydration free energies, with input files. *J Comput Aided Mol Des* 2014;**28**(7):711–20.
89. Delaney JS. Esol: estimating aqueous solubility directly from molecular structure. *J Chem Inf Comput Sci* 2004;**44**(3): 1000–5.

90. Hachmann J, Olivares-Amaya R, Atahan-Evrenk S, *et al*. The Harvard clean energy project: large-scale computational screening and design of organic photovoltaics on the world community grid. *J Phys Chem Lett* 2011;**2**(17):2241–51.
91. Blum LC, Reymond J-L. 970 million druglike small molecules for virtual screening in the chemical universe database GDB-13. *J Am Chem Soc* 2009;**131**:8732.
92. Rupp M, Tkatchenko A, Müller K-R, *et al*. Fast and accurate modeling of molecular atomization energies with machine learning. *Phys Rev Lett* 2012;**108**:058301.
93. Ruddigkeit L, Van Deursen R, Blum LC, *et al*. Enumeration of 166 billion organic small molecules in the chemical universe database gdb-17. *J Chem Inf Model* 2012;**52**(11):2864–75.
94. Ramakrishnan R, Hartmann M, Tapavicza E, *et al*. Electronic spectra from tddft and machine learning in chemical space. *J Chem Phys* 2015;**143**(8):084111.
95. Ramakrishnan R, Dral PO, Rupp M, *et al*. Quantum chemistry structures and properties of 134 kilo molecules. *Sci Data* 2014;**1**(140022).
96. Chmiela S, Tkatchenko A, Sauceda HE, *et al*. Machine learning of accurate energy-conserving molecular force fields. *Sci Adv* 2017;**3**(5):e1603015.
97. Kuhn M, Letunic I, Jensen LJ, *et al*. The sider database of drugs and side effects. *Nucleic Acids Res* 2015;**44**(D1): D1075–9.
98. Tatonetti NP, Patrick PY, Daneshjou R, *et al*. Data-driven prediction of drug effects and interactions. *Sci Transl Med* 2012;**4**(125):125ra31–1.
99. Szklarczyk D, Santos A, von Mering C, *et al*. Stitch 5: augmenting protein–chemical interaction networks with tissue and affinity data. *Nucleic Acids Res* 2015;**44**(D1):D380–4.
100. Szklarczyk D, Franceschini A, Wyder S, *et al*. String v10: protein–protein interaction networks, integrated over the tree of life. *Nucleic Acids Res* 2014;**43**(D1):D447–52.
101. Wishart DS, Knox C, An CG, et al. Drugbank: a comprehensive resource for in silico drug discovery and exploration. *Nucleic Acids Res* 2006;**34**(suppl_1):D668–72.
102. Wishart DS, Feunang YD, Guo AC, *et al*. Drugbank 5.0: a major update to the drugbank database for. *Nucleic Acids Res* 2018;**46**(D1):D1074–D1082, 2017.
103. Chen X, Ji ZL, Zong Chen Y. Ttd: therapeutic target database. *Nucleic Acids Res* 2002;**30**(1):412–5.
104. Zhu F, Han BC, Kumar P, *et al*. Update of ttd: therapeutic target database. *Nucleic Acids Res* 2009;**38**(Suppl 1): D787–91.
105. Zhu F, Shi Z, Qin C, *et al*. Therapeutic target database update 2012: a resource for facilitating target-oriented drug discovery. *Nucleic Acids Res* 2011;**40**(D1):D1128–36.
106. Yang H, Qin C, Li YH, *et al*. Therapeutic target database update 2016: enriched resource for bench to clinical drug target and targeted pathway information. *Nucleic Acids Res* 2015;**44**(D1):D1069–74.
107. Vreven T, Moal IH, Vangone A, *et al*. Updates to the integrated protein–protein interaction benchmarks: docking benchmark version 5 and affinity benchmark version 2. *J Mol Biol* 2015;**427**(19):3031–41.
108. Gilson MK, Liu T, Baitaluk M, *et al*. Bindingdb in 2015: a public database for medicinal chemistry, computational chemistry and systems pharmacology. *Nucleic Acids Res* 2015;**44**(D1):D1045–53.
109. Lowe D. USPTO patent reaction extractor. https://bitbucket.org/dan2097/patent-reaction-extraction/overview, 2014. (29 July 2018, date last accessed).
110. Brown N, Fiscato M, Segler MHS, Vaucher AC. Guacamol: benchmarking models for de novo molecular design. *arXiv preprint arXiv:1811.09621*, 2018.
111. Polykovskiy D, Zhebrak A, Sanchez-Lengeling B, *et al*. Molecular sets (moses): a benchmarking platform for molecular generation models. *arXiv preprint arXiv:1811.12823*, 2018.
112. Battaglia PW, Hamrick JB, Bapst V, *et al*. Relational inductive biases, deep learning, and graph networks. *arXiv preprint* arXiv:1806.01261. 2018.
113. Ruifeng Liu MDM, Hameed A, Kumar K, *et al*. Data-driven prediction of adverse drug reactions induced by drug–drug interactions. *BMC Pharmacol Toxicol* 2017;**18**(1):44.
114. Li Y, Zhang P, Sun Z, *et al*. Data-driven prediction of beneficial drug combinations in spontaneous reporting systems. In: *AMIA Annual Symposium Proceedings 2016*, Vol. 2016, p. 808. American Medical Informatics Association, Washington, DC.
115. Dill KA, MacCallum JL. The protein-folding problem, 50 years on. *Science* 2012;**338**(6110):1042–6.
116. Kuzminykh D, Polykovskiy D, Kadurin A, *et al*. 3d molecular representations based on the wave transform for convolutional neural networks. *Mol Pharm* 2018;**15**(10): 4378–85.
117. Torng W, Altman RB. 3d deep convolutional neural networks for amino acid environment similarity analysis. *BMC Bioinformatics* 2017;**18**(1):302.
118. Verma N, Boyer E, Verbeek J. Feastnet: feature-steered graph convolutions for 3d shape analysis. In: *CVPR 2018-IEEE Conference on Computer Vision & Pattern Recognition*, 2018. CVPR, Salt Lake City.
119. Agrawal M, Zitnik M, Leskovec J, *et al*. Large-scale analysis of disease pathways in the human interactome. In: *Pacific Symposium on Biocomputing*, Vol. 23. Hawaii: World Scientific, 2018, 111–22.
120. Monti F, Otness K, Bronstein MM. Motifnet: a motif-based graph convolutional network for directed graphs. *arXiv preprint arXiv:1802.01572*, 2018.
121. Vugmeyster Y, Harrold J, Xu X. Absorption, distribution, metabolism, and excretion (adme) studies of biotherapeutics for autoimmune and inflammatory conditions. *AAPS J* 2012;**14**(4):714–27.