Reviews • KEYNOTE REVIEW

Teaser This paper highlights the progress and challenges in **de novo** *drug design using graph neural network technology.*

ELSEVIER

Graph neural networks for automated *de novo* drug design

Jiacheng Xiong[1,2], **Zhaoping Xiong**[1,3], **Kaixian Chen**[1,2], **Hualiang Jiang**[1,2] **and Mingyue Zheng**[1,2]

[1] Drug Discovery and Design Center, State Key Laboratory of Drug Research, Shanghai Institute of Materia Medica, Chinese Academy of Sciences, 555 Zuchongzhi Road, Shanghai 201203, China
[2] University of Chinese Academy of Sciences, No. 19A Yuquan Road, Beijing 100049, China
[3] Shanghai Institute for Advanced Immunochemical Studies, and School of Life Science and Technology, ShanghaiTech University, Shanghai 200031, China
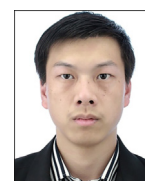
The goal of *de novo* drug design is to create novel chemical entities with desired biological activities and pharmacokinetics (PK) properties. Over recent years, with the development of artificial intelligence (AI) technologies, data-driven methods have rapidly gained in popularity in this field. Among them, graph neural networks (GNNs), a type of neural network directly operating on the graph structure data, have received extensive attention. In this review, we introduce the applications of GNNs in *de novo* drug design from three aspects: molecule scoring, molecule generation and optimization, and synthesis planning. Furthermore, we also discuss the current challenges and future directions of GNNs in *de novo* drug design.

Jiacheng Xiong was awarded a BSc by the China Pharmaceutical University. He is currently a postgraduate student in the Shanghai Institute of Materia Medica (SIMM), Chinese Academy of Sciences, His research interests mainly focus on the application of graph neural networks in drug design.

Hualiang Jiang was awarded a PhD by SIMM, Chinese Academy of Sciences, in 1995. Dr Jiang's research interests are in computational chemistry/biology and drug design. He has been engaged in the establishment of an innovative drug research platform by integrating target discovery and drug design methods and technologies.

Mingyue Zheng was awarded a PhD by SIMM, Chinese Academy of Sciences, in 2006. He is currently a professor in the State Key Laboratory of Drug Research at SIMM. His main research interests are in artificial intelligence approaches for rational drug design and discovery. He has been engaged in machine-learning based methodology development around the discovery and structural optimization of lead compounds, the assessment of drug ADME/T properties, as well as the application of these methods to practical drug design and discovery.

Introduction

The discovery and development of new drugs is a long, costly, and risky process that now takes ~10–15 years with an average cost of >US$2.5 billion [1]. Within this process, one-third of the time and money are spent during the early drug discovery stages [2]. Therefore, it is attractive for the pharmaceutical industry to develop automated and effective techniques to quickly discover diverse and plausible candidate molecules in vast and discrete chemical spaces. This *de novo* drug design method can be generally divided into three tasks: (i) molecule generation; (ii) molecule scoring; and (iii) molecule optimization [3]. Furthermore, how to synthesize the designed molecule is also a crucial issue that must be taken into account during *de novo* drug design. Therefore, we also incorporate synthesis planning in the workflow of automated *de novo* drug design in this review.

Early efforts have been devoted to produce and evaluate new molecules, such as atom-based elongation [4,5], fragment-based combination [6], and molecular docking [7]. In recent years, with the accumulation of biomedical data and progress in computing power, AI, an emerging

Corresponding authors: Jiang, H. (hljiang@simm.ac.cn), Zheng, M. (myzheng@simm.ac.cn)

computer technology, has developed rapidly in the field of drug design and shows tremendous potential for practical application [8,9]. The discovery of promising drug-like compounds using AI was ranked as one of the top-ten technological breakthroughs of 2020 by *MIT Technology Review* [10]. AI refers to a computer system that can learn from experience, adjust to new inputs, and perform human-like tasks. AI systems powered by machine learning (ML) can leverage massive amounts of data to uncover new insights and patterns, without dependency on any theoretical or practical knowledge discovered by humans [11]. Thus, it is suitable to deal with problems encountered during drug design, where there is rich data available but less well-defined rules. In particular, deep learning, as a branch of ML, shows exceptionally fascinating prospects in drug design because of its powerful capability to learn complicated relationships between features and outcomes from large-scale data [12]. Among various deep-learning models, GNNs [13] have attracted increasing attention from the field of drug molecules design, because the resulting graph is the most intuitive and concise way to represent molecules [14,15].

Chemical graph theory describes a molecule as an undirected graph in which nodes and edges correspond to atoms and chemical bonds, respectively [16]. The neural networks operating on such molecular graphs can all be regarded as a kind of generalized GNN. According to different mission objectives, those GNNs for molecule learning can be further summarized into four categories, which have different roles in *de novo* drug design. The first category aims to predict the labels of the whole graphs, which can be used in the prediction of various molecular properties. The second category aims to predict the labels of nodes or edges, which can be used in the prediction of reaction and retrosynthesis. The third category

aims to learning the implicit representation of the graphs, which can be used in the one-step generation and optimization of molecules, whereas the fourth category aims to learn the transforming rules of the graph, which can be used in the iterative generation and optimization of molecules as well as the prediction of reaction and retrosynthesis. There have been many excellent reviews on the application of AI technology in drug discovery [17–19]. Here, we specifically focus on the application of GNNs in *de novo* drug design and briefly introduce the recent progress and outlook of this field (Fig. 1).

Molecule scoring

The scoring of a molecule is mainly based on the prediction of its drug-likeness and pharmacological activity, two essential elements of a successful drug. The methods of scoring molecules have been developed for a long time and have significant roles in virtual drug screening. However, during *de novo* drug design, it can be applied to not only screen some chemical entities with desired properties, but also restrict generators to sample in a specific region of chemical space.

Ligand-based scoring

Ligand-based scoring does not need any receptor information. It is based on the acquaintance of the quantitative relationship between the molecular structure and molecular properties, and can be utilized in evaluating both drug-likeness [20] and pharmacological activity [21].

A pioneering study by Duvenaud et al. introduced GNNs in this field for the first time [22]. They proposed a graph convolution model that could directly operate on graphs to encode the mole-
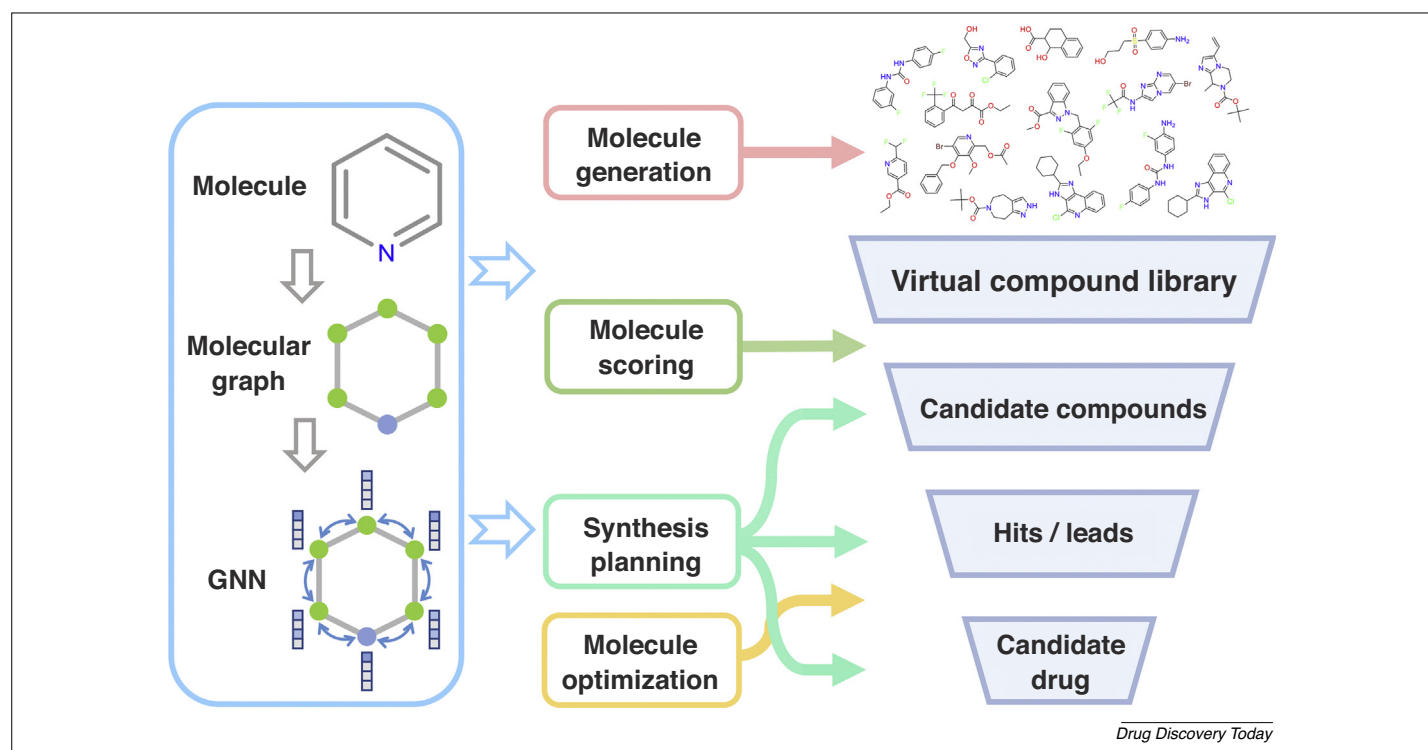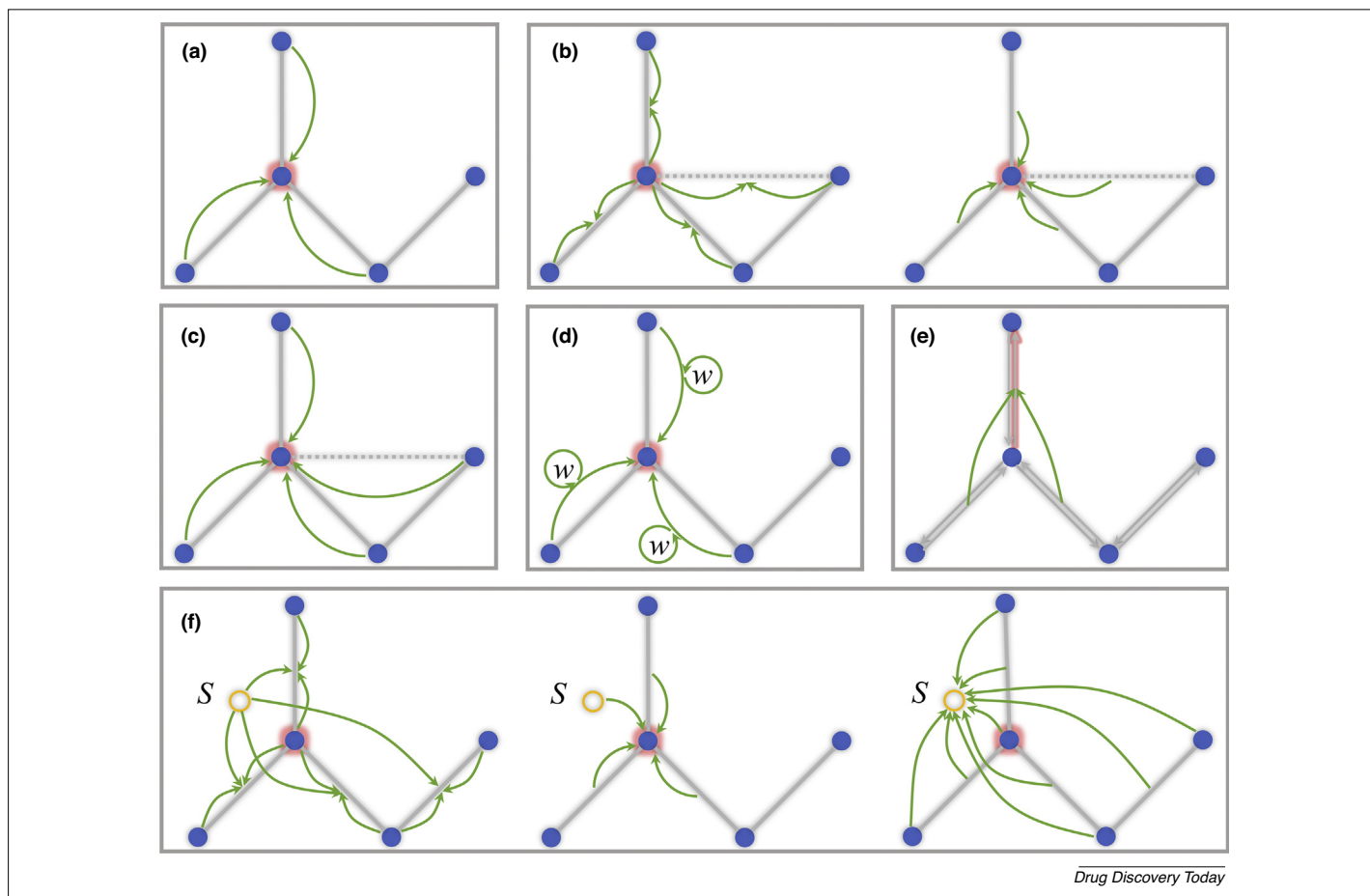

FIGURE 1

The applications of graph neural networks (GNNs) in all stages of automated *de novo* drug design.

FIGURE 2

The message-passing process of graph neural networks, where *w* denotes the attention weight and *S* denotes the virtual supernode. **(a)** Models of Duvenaud et al. and Coley et al.; **(b)** Weave; **(c)** enn-s2s; **(d)** Attention FP; **(e)** D-MPNN; and **(f)** MEGNet.

cules. Consistent with the circular fingerprints, the features of atoms in this architecture are updated according to their first order neighbors through the same local filter (Fig. 2a). However, through summing the representations calculated by all distinct graphs and replacing the hash function used in traditional circular fingerprints with a single-layer neural network, this architecture can generate differentiable fingerprints able to be learned via backpropagation. It was demonstrated that these neural graph fingerprints obtained better results than Morgan fingerprints in the task of predicting solubility.

Kearnes et al. [23] reported another molecular graph convolution method, which comprises 'weave modules', as an extension to Duvenaud's method. Except for atoms, the weave module also introduces updatable hidden states for all edges in the graph. Moreover, when updating features of an atom, the weave module combines information from all other atoms and their corresponding pairs rather than only neighbor atoms, which enables it to transmit information between distant atoms at the expense of increased computational complexity (Fig. 2b). Although the weave model does not outperform all fingerprint-based methods, it presents a practical strategy to add edge information explicitly into graph convolution neural networks.

Gilmer et al. [24] summarized the commonalities between several existing models for graph-structured data and reformulated them into a general framework called Message Passing Neural Networks (MPNNs). MPNNs consider graph convolutions operations as a message-passing process where information of a node is passed to another, the features of which then will be updated. The authors also proposed enn-s2s, a novel variation within this framework (Fig. 2c), which obtained state-of-the-art results on predicting quantum mechanical properties of molecules at that time.

In the above three studies, only simple structural attributes are taken as input featurization, which can be directly obtained from molecular structural formula, such as atom type, bond type, and graph distances. Coley et al. [25] introduced an alternative graph convolution method that preserved spatial and other molecular information beyond connectivity. A set of molecule-level attributes is first calculated using an atomic contribution method, and these atom-level contributions and other structural attributes are together integrated into the initial representation of atoms. These representations of atoms are updated with the message passed from their neighboring atoms in the graph convolution layer and then aggregated into the representation of molecules in the pooling layer (Fig. 2a). The tests on several benchmark tasks, including aqueous solubility, octanol solubility, and melting point prediction, suggested that model performances were significantly enhanced by the addition of molecule-level spatial information.

For learning graph-level representation, a virtual supernode, sometimes referred to as a global state, can be introduced in the

molecular graph [26–28]. To take an example of MEGNet [28], the attributes of a global state connected with all nodes and edges along with the attributes of atoms and edges together constitute the representation of the initial graph. Then, the initial graph representation is updated, in the first two steps, the information from the global state and the other part is applied to update the attributes of bonds and atoms. Afterward, global state attributes are updated according to the information from bonds, atoms, and previous global state attributes (Fig. 2f). Remarkably, incorporating state variables, such as temperature, pressure, and entropy, as global state inputs, Chen et al. developed a combined free energy model that could predict internal energy of molecules at 100 and 200 K even though these data points were not included in the training data.

Although numerous deep learning models for predicting molecular properties have been proposed, and achieved superior results on many standard data sets, the black-box nature of neural networks hampered their practical applications, because it is not easy to judge whether those models have learned the expected knowledge or hidden variables of training data [29]. Improved interpretability of the model, to some extent, can solve this knotty problem. Toward this end, Xiong et al. [30] reported Attentive FP, a new GNN architecture with a graph attention mechanism. Attentive FP calculates a series of attention weights between the neighbor node pairs to represent the degree of interaction between them according to their features. For a certain atom, the messages passed from its neighbor atoms have to be multiplied by such attention weights between them, and then can be used to update its representation (Fig. 2d). Furthermore, the attention mechanisms at molecule levels are also implemented by introducing a super virtual node that connects all the atoms of the molecule. Through these designs, Attentive FP can not only achieve state-of-the-art predictive performances on a variety of data sets, but, more significantly, also interpret what it learns from those data sets. For example, when predicting the number of aromatic atoms in a molecule, Attention FP can accurately mark the position of aromatic atoms according to the weights of attention. Beyond that, Tang et al. [31] introduced a self-attention mechanism into their GNN, where the attention weight of an atom is calculated according to its hidden state. It can be used to visualize the relationship between the substructure contributions to the target property of a molecule.

The message passing between nodes through the undirected edge allows tottering walks, which are likely to introduce noise into the graph representation. To overcome this problem, Yang et al. [32] introduced a graph convolutional called D-MPNN, which can be viewed as a variant of the generic MPNN architecture. The most significant difference between them is the nature of the messages sent during the message-passing phase. D-MPNN uses messages associated with directed bonds rather than with atoms (Fig. 2e). At the end of the message-passing phase, the incoming bond features are summed and taken as the representation of corresponding atoms that will be further summed and taken as the representation of the entire molecule in the readout phase. Also, fixed molecular descriptors are combined into the representation of the molecule. An interesting case of D-MPNN in the molecular property prediction was the discovery of new antibiotics, as represented by halicin [33].

Although antibiotic resistance is a growing threat to public health, the discovery of new antibiotics is becoming increasingly difficult because of the many limitations of the traditional antibiotic discovery method [34]. Trained on a collection of 2335 molecules, D-MPNN was applied to identify potential lead compounds with activity against *Escherichia coli* in several discrete chemical libraries, comprising $>10^7$ molecules. Through this approach, halicin, a potent broad-spectrum antibiotic structurally divergent from conventional antibiotics, was discovered. Moreover, eight additional antibacterial compounds were also discovered from another large compound library comprising >100 million molecules. The *in silico* screen of the 100 million molecules was completed within 4 days, which highlighted the high reasoning speed of deep learning models and possibilities for its further application for screening a larger scale virtual compound library.

The excellent performance of GNNs in predicting the properties of small molecules has attracted the attention of pharmaceutical companies. Recently, Merck and Stanford jointly conducted a study where they trained ML models on 31 chemical data sets from Merck describing results of various absorption, distribution, metabolism, excretion, and toxicity (ADMET) assays and compared results of random forests with those of their GNN model [35]. Their study showed the performance of their GNN model on those data sets was significantly better than a random forest model based on the atom pair molecular fingerprint. Moreover, they tested their models on external data collected from the literature and subsequent data generated after model training. The results showed that their GNN model had better generalization abilities than the random forest model. They also found that multitask style training could further boost the performance of their GNN model.

Receptor-based scoring

Receptor-based scoring is designed to predict the drug–target interactions (DTIs) based on information about the receptor protein. At present, there are three main forms of receptor protein input information in a deep neural network (DNN): (i) the protein sequence (Fig. 3a); (ii) the structures of the protein pocket (Fig. 3b); and (iii) the structures of protein–ligand complexes (Fig. 3c).

Gao et al. [36] used amino acid sequences and gene ontology annotations as receptor protein input information and entered them into two different Long Short-Term Memory networks. The outputs of the two networks were summed and regarded as protein representations. The molecular graph was projected to a dense vector with GNN that had the same length as the protein representation vector. The representations of the protein and molecule were fed separately into two three-layer fully connected network networks. Then, the inner product of the outputs activated with a sigmoid function was taken as the probability that an interaction existed between the input protein and drug. Remarkably, the attention mechanism was incorporated in the integration of the representations of proteins and ligands, and the model displayed how it learned the DTIs. This is the first end-to-end deep learning work that performs interpretable predictions of DTIs directly from low-level representations. Another similar model was proposed by Tsubaki et al. [37], which also represents protein sequences and molecular graphs of ligands with independent vectors and integrates them within a DNN to predict their interactions. The
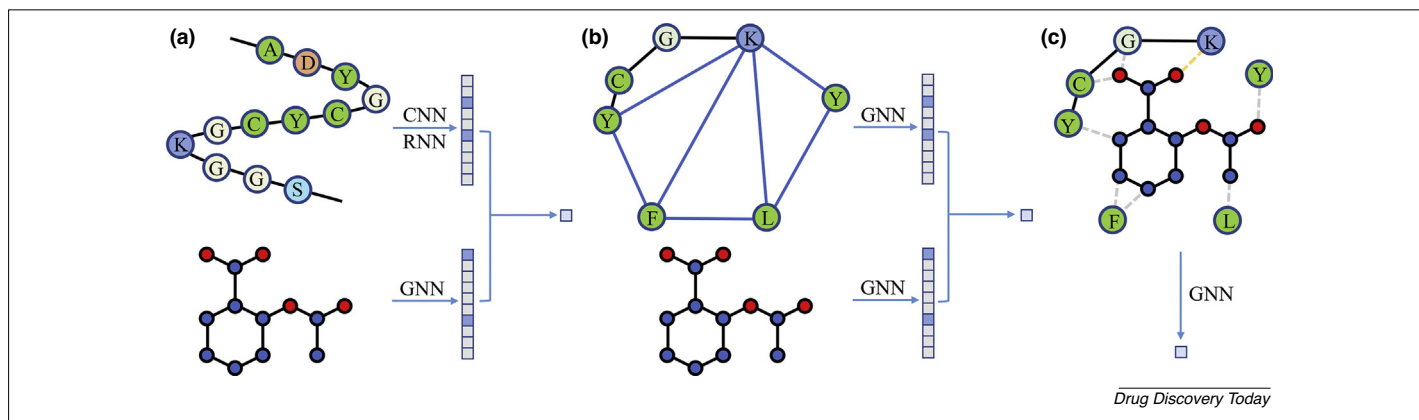
FIGURE 3

The three forms of protein information input in deep neural networks. **(a)** The protein sequence; **(b)** structures of the protein pocket; and **(c)** the structures of protein–ligand complex. Abbreviations: CNN, convolutional neural network; GNN, graph neural network; RNN, recurrent neural network.

difference is that Tsubaki et al. applied a convolutional neural network (CNN) to learn the representations of protein sequences rather than a recurrent neural network (RNN).

The models of Gao et al. and Tsubaki et al. demonstrated the ability of neural attention mechanisms to capture noncovalent interactions between compounds and proteins in a few examples. However, systemic and comprehensive assessments of this learning capacity have been lacking. Recently, Li et al. [27] compiled a benchmark data set containing labels of noncovalent interactions for >10 000 compound–protein complexes and conducted a systematic analysis based on this benchmark data set to assess the interpretability of neural attention mechanisms in existing models. The results showed that current attention-based models trained only using binding affinity labels had difficulty in accurately capturing the noncovalent interactions between compounds and proteins. The authors then developed a multiobjective neural network, called MONN, to simultaneously learn both pairwise noncovalent interactions and binding affinities between proteins and compounds. Comparison tests showed that MONN could achieve accurate prediction in both tasks, and its performance was better than other state-of-the-art ML methods.

Given that only a part of the protein structure, usually referred to as the ligand-binding pocket, interacts directly with ligand molecules, the methods focusing on intermolecular interactions between ligand molecules and receptor pockets might show higher generalization ability than those that try to extract certain patterns present in molecules or sequences. Feinberg et al. [38] proposed a staged gated GNN including both intramolecular and intermolecular interactions. The workflow of this model has three steps: (i) Intramolecular information propagation only based on covalent bonds; (ii) intramolecular and intermolecular information propagation based on both covalent bonds and spatial distance; and (iii) node aggregation for graph embedding. This was the first graph-based deep learning model specifically designed for the presentation of protein–ligand complexes and achieved state-of-the-art performance for protein–ligand binding affinity. Lim et al. [39] proposed an alternative approach to predict DTIs according to protein–ligand binding poses. There are two different message-passing processes in their model: (i) information propagation between atoms within ligand based on their covalent bonds;

and (ii) information propagation between ligand atoms and receptor atoms based on their spatial distance. These two message-passing processes are carried out simultaneously, and the subtraction between their output is further used to update the features of atoms on the graph. Apart from DTI prediction, the model was also applied for binding pose prediction. It improved the area under the receiver operating characteristic curve and area under the precision-recall curve by ~0.11 and 0.26 compared with the results of docking.

The approaches based on protein–ligand binding poses require protein–ligand complexes as input. Therefore, for those without experimental complex structure information, the accuracy of these models is also dependent on the accuracy of methods for establishing protein structure and deriving putative ligand-binding poses. To overcome those problems, Torng et al. [40] proposed a graph-convolutional neural network taking the structure of protein pockets and ligands as input for DTI prediction. First, each receptor pocket is represented as a graph where functional atoms of the corresponding residue that are within 6 Å of the bound ligand are regarded as nodes, and any pair of nodes within 7 Å of each other is connected with an edge. The initial representation of each residue is generated with the FEATURE program [41]. Then, a graph autoencoder is trained to learn general protein pocket features on a set of 965 representative protein pockets. A ligand Graph-CNN and a pocket Graph-CNN initialized with learned weights from the graph autoencoder are used to extract features from the ligand graphs and pocket graphs in parallel. Finally, the features of the pocket and molecular are concatenated to generate a joint target–ligand feature that is then fed to a fully connected layer to predict the interactions between pockets and ligands. The model of Torng et al. achieved better or comparable performance compared with 3DCNN ligand-scoring, AutoDock Vina, RF-Score, and NNScore on common virtual screening benchmark data sets.

GNN can automatically learn the representations of molecules to predict their various properties. However, the learned representations are still based on the input chemical structure. Hence, problems in previous quantitative structure–activity relationship research, such as the prediction of nonadditive effects and activity cliffs, might still be challenging for those GNN models.

In addition, as a data-driven approach, another obstacle to achieving its full potential is the lack of high-quality training data sets. In 2017, Wu et al. [42] introduced MoleculeNet, a large-scale benchmark for molecular ML. To this day, the 17 public data sets used in their work have always been the widely used benchmark data sets for molecular ML. However, many of them are in fact too small for deep learning models, especially those data sets for physical–chemical properties. The smallest data set for physical chemistry properties, the FreeSolv data set, contains only 643 molecules, whereas the biggest, the Lipophilicity data set, contains just 4200 molecules. Besides, the biases in data sets are also a problem that needs to be addressed. Some widely used data sets for DTI prediction, including DUD-E, MUV, and PDBbind, were found to have serious chemical bias [43,44]. Given these biases, AI-based models for DTIs prediction do not in fact need to learn the protein–ligand interaction information from protein–ligand complexes and can achieve similar performance even trained on data sets containing only ligand or protein structures [45,46]. Thus, there is an urgent need to design sufficiently large and unbiased data sets to train and evaluate AI-based molecular-scoring models. Large

pharmaceutical companies often accumulate huge amounts of data, especially negative data that are difficult to obtain from the literature but are indispensable for ML models. Hence, increasing cooperation between AI researchers and pharmaceutical companies would be beneficial. However, in most cases, pharmaceutical companies are reluctant to share data directly with other researchers for confidentiality reasons. Federated learning could be a solution in such cases [47].

Molecule generation and optimization

The generation and optimization of molecules is at the heart of automated *de novo* drug design. The approaches to generative modeling are usually divided into two categories: nonautoregressive manners (Fig. 4a) and autoregressive manners (Fig. 4b). Nonautoregressive generative models build a graph by generating its edge-feature matrix and node-feature matrix simultaneously. The nonautoregressive generative models include variational autoencoders (VAE) [15,48], generative adversarial networks (GANs) [49,50], and reversible flow models [51]. Autoregressive generative models build a graph by iteratively refining its
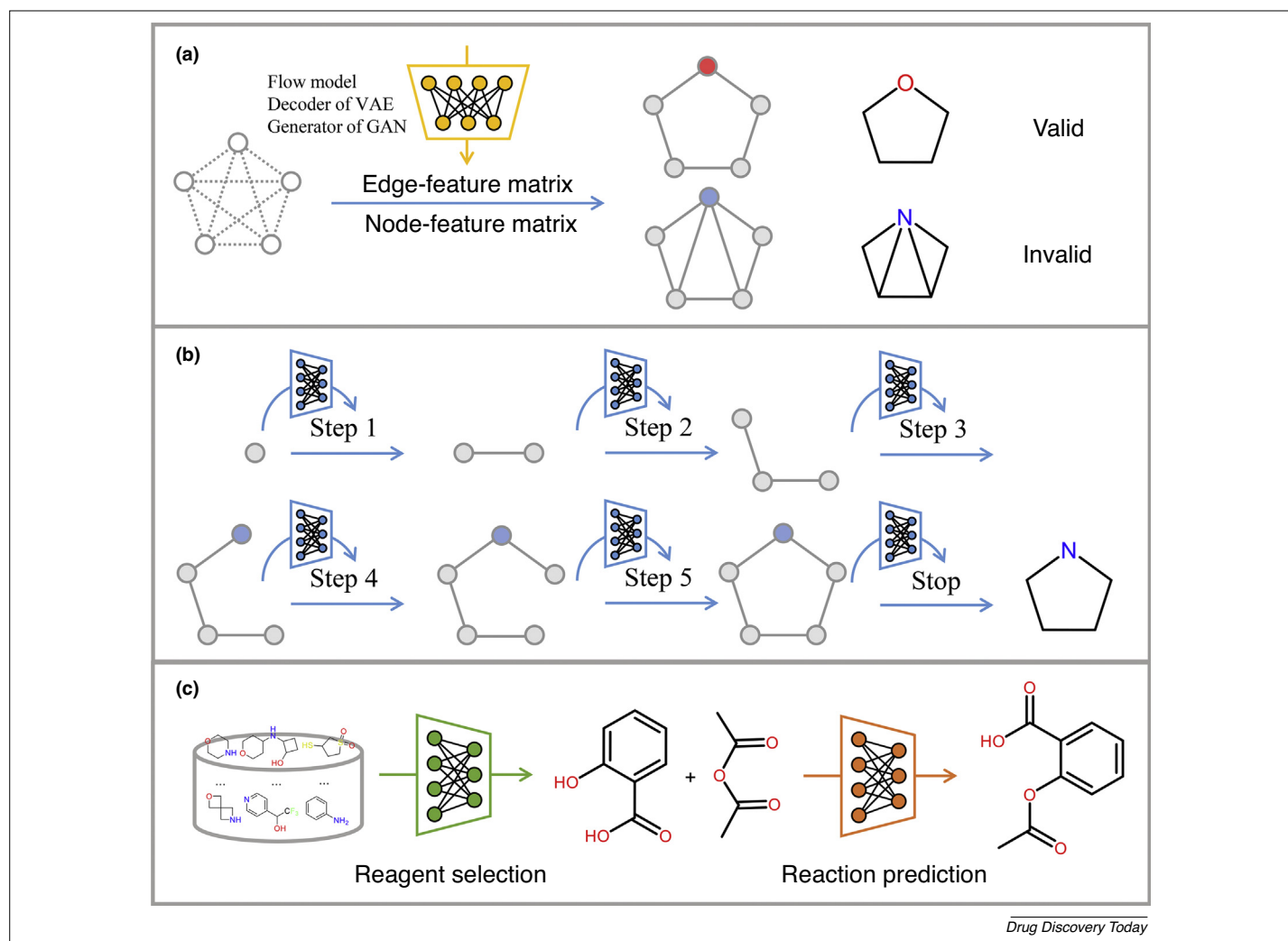

FIGURE 4

Schematic diagram of three approaches for generating molecular graphs. **(a)** Nonautoregressive; **(b)** autoregressive; and **(c)** virtual chemical reaction. Abbreviations: GAN, generative adversarial network; VAE, variational autoencoders.

intermediate structure. A typical representative of the autoregressive generation model is RNN [52]. These different types of generative architecture can also be used in combination, such as autoregressive VAE [14,53,54] and autoregressive flow models [55]. Besides, as a special kind of graph structure data, the molecules can also be generated by virtual chemical reactions [56] (Fig. 4c). Currently, the performance of the deep generative model is often evaluated with the following metrics: (i) validity, the percentage of generated graphs corresponding to valid molecules; (ii) novelty, the percentage of generated valid molecules not present in the training set; and (iii) uniqueness, the percentage of unique valid molecules out of all generated molecules. Over the past few years, numerous graph-based molecular generative models have been reported. Here, we introduce several representative works from each class of model.

VAE-based generative models

GraphVAE [48], reported by Simonovsky et al., is a variational autoencoder for generating the entire molecule simultaneously. A molecular graph is characterized by its adjacency matrix, edge attribute tensor, and node attribute matrix. An edge-conditioned graph convolution network is used as an encoder to embed the graph into continuous representations. The decoder is used to output a probabilistic fully connected graph with the predefined number of nodes. Through this design, the model can sidestep the nondifferentiable problems occurring in the iterative construction of discrete structures. The reconstruction loss of molecular graphs is calculated by approximate graph matching for aligning generated graphs and ground truth. The model was trained by minimizing the loss of reconstruction as well as the Kullback–Leibler divergence [57] between latent space distribution and standard normal distribution. This model generates target molecular graphs from the set of all graphs with the predefined number of nodes. Given that the number of all possible graphs will increase exponentially with the linearly increases in the predefined number of nodes, the model is not effective at generating graphs of large molecules. This model achieved a 55.7% validity ratio in the QM9 data set containing molecules no more than nine heavy atoms in size, but only achieved a validity score of 13.5% for the ZINC data set, containing about ~250 000 drug-like organic molecules of up to 38 heavy atoms.

Kwon et al. [15] extended and improved the nonautoregressive graph VAE to generate molecular graphs. Apart from approximate graph matching, they also trained the VAE by incorporating two additional learning objectives: reinforcement learning for the validity of molecules, and an auxiliary task for property prediction. The reward network for reinforcement learning takes a probabilistic graph as input and reports a reward value according to the chemical validity of the input probabilistic graph. VAE is trained to generate a probabilistic graph toward maximizing the output of the reward network. With such a reinforcement learning scheme, the proposed model exhibited a validity score of >90% for both the QM9 and ZINC data sets. Moreover, by incorporating auxiliary property prediction into the VAE learning, the model is able to generate probabilistic graphs that correspond to desired properties.

GAN-based generative models

The first generative adversarial network for molecular graph generation was MolGAN [50], which comprises three main components: (i) a generator; (ii) a discriminator; and (iii) and a reward network. The generator is a simple multilayer perceptron that creates the node feature matrixes and adjacency tensors of new molecular graphs using the input vector sampled from a standard normal distribution. It learns to generate molecules resembling the true molecular graphs in the training data set. The discriminator is a graph convolutional neural network that learns to distinguish molecular graphs produced by the generator. The generator and discriminator are trained iteratively to compete with, and improve, each other until the models reach a balance (the so-called 'Nash equilibrium'). The reward network is another graph convolutional neural network utilized to encourage the generation of molecules with particular properties. MolGAN can generate fixed-size molecular graphs simultaneously with both higher validity and novelty than previous VAE-based graph generative models, while not requiring a permutation-dependent likelihood function obtained from the time-consuming graph-matching process. A central limitation of MolGAN is that it sufferers considerably from the 'mode collapse' issue (i.e., when the generator finds some especially plausible outputs, it can only produce those outputs and miss other modes in training data, which leads to insufficient diversity of generated samples). MolGAN only showed a 10.4% uniqueness ratio on the QM9 data sets.

Invertible flow-based generative models

GraphNVP [51], introduced by Madhawa et al., is the first invertible normalizing flow-based molecular graph generation model. This model was constructed by a sequence of invertible transformations and aims to learn the distribution of vector representations of actual molecular graphs. The training criterion of this model is simply the negative log-likelihood over the training data set. The generation of a molecular graph is decomposed into two steps: generation of (i) an adjacency tensor and (ii) node attributes. Given the invertibility of this model, molecular graph generation can be facilely implemented by inversely executing the computational process of the training phase. Most surprisingly, this model generates molecular graphs with an almost 100% uniqueness ratio and 100% reconstruction accuracy. What is more, the learned latent space of this model displays excellent smoothness, which is crucial for decorating molecules, creating a more desirable novel molecule with slight modification by perturbing the latent representation of the original molecular graph. However, as a one-shot generative model, the molecules generated by GraphNVP also suffer from a low validity ratio (42.6% on the ZINC data set).

Autoregressive generative models

Instead of generating the whole molecular graph in one step, autoregression-based methods dynamically generate the nodes and edges based on existing subgraph structures. Taking MolMP designed by Li et al. as an example [58], this model formulates graph generation as a Markov decision process. A molecular graph is built by iteratively modifying its intermediate structure by three graph transitions: (i) append, adding a new atom to an intermediate graph and connecting it to an existing atom with a new bond; (ii) connect, connecting an existing atom to the new atom with typed bonds; and (iii) terminate, ending the generation process. The probability of sampling each transition is parametrized with the GNN. Another autoregression-based generative model, named

MolRNN, was also proposed by Li et al. This model adds a single-molecule level recurrent unit to MolMP, so that the model samples of each graph transition depend on not only the current state of the graph, but also the previous states. Experiments show that the molecule-level recurrent unit in MolRNN provides significant improvements to the model performance. These models have been effectively applied to solve several drug design problems, including the generation of compounds containing a given scaffold, compounds with specific drug-likeness and synthetic accessibility requirements, as well as dual inhibitors against JNK3 and GSK-3β.

Autoregressive flow-based generative models

Shi et al. [55] introduced GraphAF, a flow-based autoregressive model for graph generation. They defined the adding order of nodes and edges of a molecule in training data according to the breadth-first search order. Then, the whole generative process of the molecular graph was decomposed into multiple independent one-step graph transformation processes able to be trained in parallel. In the generation phase, GraphAF creates a molecule through an iterative sampling process, which allows leveraging chemical domain knowledge for valency checking. GraphAF was able to achieve a validity rate as high as 68% on the ZINC data set even without chemical knowledge rules, and a 100% validity ratio with chemical rules.

Autoregressive VAE-based generative models

Jin et al. [14] made valuable contributions to the development of VAE-based generative models, proposing a junction tree variational autoencoder (JT-VAE) for molecular graph generation. Deviating from previous studies that created molecules atom by atom, their approach exploits subgraph components as building blocks of molecules so as not to generate chemically invalid intermediaries. The whole molecular graph is first decomposed into a cycle-free junction tree, where each node in the tree represents a substructure in the molecule. The original molecular graph is encoded into its latent embeddings by a standard graph message-passing network, whereas its associated junction tree is encoded via its latent embeddings with two-phase messages passing. In the first bottom-up phase, messages are initiated from the leaf nodes and propagated iteratively toward the root. In the second top-down phase, messages are propagated from the root to all the leaf nodes. Those messages are updated through a gated recurrent unit. The latent representations are then decoded back into a molecular graph in two stages. First, the tree decoder constructs the entire tree in a top-down fashion by generating one node at a time. Then, the junction tree is converted into a valid molecule by sampling subgraphs one by one following the order in which the tree itself was decoded. Compared with nonautoregressive graph VAE, JT-VAE shows superior performance in the validity of generated molecules, achieving a 100% validity ratio with the ZINC data set.

To further improve the capability of their model in molecular optimization, Jin et al. [53] proposed another variational junction tree encoder-decoder similar to JT-VAE and combined it with adversarial training. This model was trained on a data set of paired molecules to learn a multimodal mapping between one molecular graph and another with better properties. It showed a significant performance boost compared with JT-VAE on multiple molecular optimization tasks, including drug likeness, activity against dopamine receptors, and penalized LogP.

Fu et al. [54] proposed a new strategy for molecular optimization called Copy & Refine. They argued that growing the new molecular structure by iteratively predicting the best substructure to add was often inaccurate because of the large set of available substructures, especially for adding rare substructures. Moreover, given that the substructures picked to build a refined molecule usually exist in the input molecule, at each generating step, their model first considers whether to copy a substructure from the input molecule (Copy) or sample a novel substructure from the entire space of substructures (Refine). With such a mechanism, Copy & Refine can significantly outperform several recent molecule optimization baselines in various measures, especially in terms of modifying rare substructures.

Virtual reaction-based generative models

Although there has been a flood of new deep learning methods developed for molecule generation and optimization, there are very few molecules produced by those models to be synthesized and used for further research. It appears ironic that the number of generative models is more than the number of molecules created by them in the real world. The most important reason for is that most generative models ignore the synthesizability of molecules [59]. With the aim of producing molecules with higher synthesizability, Bradshaw et al. [56] proposed a generative model mimicking the real-world process of designing new molecules. Their model first generates a bag of initial reactants and then uses a reaction predictor model to transform this bag of molecules into a product molecule. They chose Molecular Transformer developed by Schwaller et al. as a reaction predictor model and designed a VAE, named MOLECULE CHEF, to produce the reactant bag. In the encoder, the reactant molecules are embedded into a continuous space using Gated Graph Neural Networks. The molecule embeddings in the multiset are summed to form one order-invariant embedding for the whole multiset. The encoder is an RNN that samples input from the latent space and output multiset of reactants in sequence. To better mimic the process of selecting reactant molecules from an easily obtainable set, the output of the decoder is restricted to pick the molecules from a fixed set of reactant molecules. Thus, their generative model generates not only molecules, but also corresponding synthesis routes using available reactants simultaneously.

Shoichet et al. [60] conducted a study where 170 virtual million molecules generated from 70 000 building blocks were docked against AmpC β-lactamase and the D4 dopamine receptor. The results showed hit rates against AmpC β-lactamase and D4 dopamine receptor of 11% and 24%, respectively, which was higher than with traditional virtual screening. Their study demonstrated the advantages of using ultralarge libraries to discover desirable hit compounds, leading to the opinion that 'Bigger is better in virtual drug screens' [61]. Thus, producing a massive number of molecules by deep generating models has important practical meaning for the virtual screening of drugs. At present, the methods to generate novel molecules conforming to chemical rules are already relatively mature. Many generative models are able to achieve almost 100% validity and novelty ratios against the ZINC data set. However, for *de novo* drug design, it is not enough to only design

Reviews • KEYNOTE REVIEW

molecules that are unique and do not violate the rules of valence. The generated molecules should have suitable properties, which need to be proved by experiments. The results of experiments can also be fed back to the generative model and eventually form a vicious circle. However, the difficulties in molecular synthesis limit the implementation of this process. At present, screening the compounds in stock with virtual screening or high-throughput screening is still the mainstream approach in drug discovery. Improving the synthesizability of generative molecules should be one of the most important goals in the next phase of research on generative models. We consider that the combination of generative models and synthesis planning models will be a valuable study direction. In addition, the new evaluation metrics of generative models are also worth exploring. Currently, the commonly used metrics to evaluate deep generation models are only validity, novelty, and uniqueness. Admittedly, those three metrics can intuitively reflect the efficiency of models in exploring chemical space and some potential problems of models such as 'mode collapse'. However, they cannot reflect the quality of the molecules generated by the model. Recently, Bush et al. proposed a 'Turing test', where algorithmically generated molecules were compared with those generated by medicinal chemists [62], which was a positive attempt to evaluate the molecules created by generative models.

Synthesis planning

The goal of computer-aided synthesis planning is to help chemists decide how to synthesize small-molecule compounds. Two crucial issues in this field are the prediction of chemical reaction outcomes and the planning of retrosynthesis routes [63]. Early work is mainly based on reaction rules [64,65] or quantum chemical calculations [66]. The reaction rules are manually encoded by human experts or automatically generated from reaction databases. The principal limitation of such rule-based methods is that they are unable to predict novel chemistry reactions with no available templates. As for the methods based on quantum chemical calculation, the enormous expenditure of time and money hinder it from large-scale use. Hence, data-driven approaches have attracted increasing research interest in recent years [67,68].

Prediction of general chemical reactions

The pioneering work of Jin et al. and Coley et al. [69] applied GNNs to the prediction of chemical reaction products. The workflow can be divided into three steps. First, a Weisfeiler–Lehman Network is trained to predict the probability of each edge change in a molecular graph during a chemical reaction. The top K atom pairs with the highest predicted possibility of edge change are selected as the reaction centers. Then, valid candidate product molecules are generated by enumerating the possible bond changes in reaction centers subject to several constraints, such as the maximum number of neighbors of an atom. Finally, a Weisfeiler–Lehman Difference Network is trained to rank candidate products. Their model achieved the top accuracy of 79.6% in the USPTO data set, 10% higher than that of the top-performing template-based approach, while operating orders of magnitude faster. In this model, the process of ranking candidate products is treated as an completely isolated task depending only on the structure of candidate molecules. Thus, it neglects the fact that outcomes generated by

combinations of those more probable bond changes are more likely to be the real outcome. In another improved version [70], the authors calculated the initial ranking of candidate reaction outcomes based on the predicted probability of bond change and trained a Weisfeiler–Lehman Difference Network to refine these preliminary rankings. The reagents were excluded from the prediction of reactivity centers. With such improvements, the accuracy of this model was 5.3% higher than that of the previous version.

Qian et al. [71] reported a graph-based method for predicting chemical reaction products that achieved an accuracy of >90% against the USPTO data set. Consistent with the approach of Jin et al. and Coley et al., the model of Qian et al. also starts from estimation of reaction centers utilizing a GNN. The main difference is that their model estimates the likelihood of changes in not only covalent bonds, but also in hydrogen counts and formal charges. Then, the most probable product is inferred from such predicted probability via Integer Linear Programming (ILP) rather than from an additional ranking network, which shows that a ranking network might not be necessary for determining the candidate products. Some chemistry constraints, such as the octet rule, can be translated into the restraint conditions of ILP, which allows the chemists to interact with the model by setting constraints according to their chemistry insights.

In addition to the aforementioned supervised learning method, reinforcement-learning algorithms have also achieved great performance in this field. A Graph Transformation Policy Network [72], introduced by Do et al., is a model that combines GNNs and reinforcement learning, achieving an accuracy of 83.20% against the USPTO data set. It treats the molecular graph transformation from reactants to products as an iterative process, where a new molecule is generated by changing one bond in the input molecule and the new molecule is then taken as the input for the next iteration. This iterative process runs until the generated molecule reaches the termination criterion. This model has three main components: a GNN, a node pair prediction network, and a policy network. The GNN is responsible for calculating the atom representation, the node pair prediction network is responsible for computing the most possible reaction atom pairs, and the policy network is responsible for finding the optimal sequence of bond changes that transforms the reactants into products. Combined with the prediction of the reaction center and inference of product into a unified neural network, this model can be trained in an end-to-end manner. In addition, the stepwise generation of product molecules enables the model to display intermediate molecules, which dramatically increases its interpretability.

Prediction of specific types of chemical reaction

For organic reactions involving proton abstraction and formation of carbanion intermediates, such as alkylations, Michael additions, or aldol condensations, the acidity of C-H groups within organic molecules is a crucial factor deciding the reaction site. Given that GNN had been implemented effectively to predict all types of molecule-wide properties, Roszak et al. attempted to use it to predict the acidity of C-H groups, a type of atomic property [73]. Equipped with four graph convolutional layers, their model can perform accurate pK_a prediction with a mean absolute error (MAE) of 2.2 pK_a units, significantly lower than the state-of-the-art

commercial solutions, Schrödinger's commercial Epik package, the MAE of which is 3.3 pK_a units. On the issue of reaction prediction, their model can correctly identify the reaction site of C−H acid with 90.5% accuracy on a data set containing 12 873 pK_a-controlled reactions collected from the 'high-impact chemical journals', whereas the WLDN5 reported by Coley et al. achieved an accuracy of only ∼50% against the same data set.

Prediction of retrosynthesis

Retrosynthesis prediction can be regarded as an inverse process of reaction prediction. However, it is more challenging because a molecule can have many synthetic routes, whereas there is only one specific result for a chemical reaction under a given reaction condition. Retrosynthesis prediction is also valuable because retrosynthetic analysis is the most common strategy for chemists to solve organic synthesis problems.

Dai et al. [74] proposed a template-based method for retrosynthesis prediction. They embedded reactants, products, and templates within a GNN and described the reasoning process of retrosynthesis prediction with a two-step probabilistic graphical model. The Conditional Graph Logic Network (GLN) is responsible for determining which chemical reaction templates to apply in retrosynthesis prediction according to the joint probability that the subgraph pattern from the reaction template is matched against the product and the set of reactants. The parameters of GLN are estimated by the maximum likelihood method. GLN shares the interpretability of template-based methods while taking advantage of the scalability and expressiveness of neural networks to learn when such rules apply. GLN achieved 63.2% accuracy against the standard USPTO-50K data set, improving the previously reported best performance of 55.3% [75] by a large margin.

Combining GNN and Transformer [76], Yan et al. proposed a retrosynthesis prediction model named RetroXpert [77]. In this model, the retrosynthesis prediction is decomposed into two subtasks: (i) bond disconnection prediction; and (ii) reactant prediction. In the first step, a GNN taking the molecule graph as the input is trained to predict the bond disconnection within the graph, which is similar to the previous method in forward synthesis prediction. In the second step, the split synthon graphs are generated according to the disconnection prediction results, converted to canonical SMILES, and concatenated with the canonical SMILES of product. Then such sequence data are taken as the input of the Transformer to predict the SMILES of reagents. Retro-Xpert reached 70.4% accuracy against the USPTO-50K data set, achieving a significant improvement of 7.1% over the GLN of Dai et al. Shi et al. [78] and Somnath et al. [79] proposed two other template-free models that also divided retrosynthesis prediction into two subtasks, bond disconnection prediction and reactant prediction. The main difference between their models and the RetroXpert lies in the process of transforming the synthons to reactants. Shi et al. transformed the synthons to reactants through an autoregressive graph generative model, whereas Somnath et al. completed synthons into valid reactants by adding specific leaving groups. Those added leaving groups were selected from a vocabulary containing 170 leaving groups by a classification model.

In just the past 3 years, the accuracy of reverse reaction prediction on the standard USPTO-50K data set has dramatically increased from 37.3% to 70.4%, benefiting from the application of GNNs (Table 1). Despite such rapid progress, there is still a long way to go for the study on retrosynthesis prediction. Unlike the field of chemical reaction product prediction, where some existing deep learning models are powerful enough to match human experts, the current state-of-the-art prediction models of retrosynthesis are still some way from solving practical problems independently. These deep learning models are mainly developed for single-step retrosynthesis prediction, which is much simpler than planning a complete synthesis route. Several strategies, such as Monte Carlo tree search [67], depth-first proof-number search [80], reinforcement learning [81], hypergraph exploration [82], and multiple beam-like searches [83], have been used to extend single-step retrosynthetic models to full route planning. However, the possible synthetic routes of a compound are numerous and it is

TABLE 1

Summary of models for reaction and retrosynthesis prediction that have been tested correspondingly on the USPTO and USPTO-50K data sets

Class	Model name	Sequence based	Graph based	Template based	Top accuracy (%)	Year	Refs
Reaction prediction	WLDN		√		79.6	2017	[69]
	S2S	√			80.3	2018	[87]
	GTPN		√		82.4	2019	[72]
	WLDN5		√		85.6	2019	[70]
	Schwaller et al	√			90.4	2019	[88]
	Qian et al.		√		>90	2020	[71]
Retrosynthesis prediction	Seq2Seq	√			37.4	2017	[89]
	Retrosim			√	52.9	2017	[90]
	Karpov et al.	√			42.7	2019	[91]
	Lee et al.	√			43.8	2019	[92]
	SCROP	√			59.0	2019	[93]
	GLN		√	√	64.2	2019	[74]
	RetroXpert	√	√		70.4	2020	[77]
	Lin et al.	√			54.6	2020	[94]
	GET-LT1	√	√		57.4	2020	[95]
	G2Gs		√		61.0	2020	[78]
	GRAPHRETRO		√		67.8	2020	[79]

difficult to judge whether a certain inverse synthetic route is reasonable. Hence, we are lacking reliable standards to evaluate and compare the effectiveness of those strategies. In the future, effort should be made to push the research of retrosynthesis prediction from a single-step reaction to planning the entire synthesis route.

Concluding remarks and prospects

To design new drugs from scratch automatically is the dream of all drug researchers. In recent years, with the assistance of deep learning technologies represented by GNNs, remarkable progress has been made in many aspects related to this target. However, there are also some obvious problems that need to be addressed. For example, there are serious biases in some benchmark data sets, which may cause the performances of deep learning models on corresponding tasks to be grossly overestimated; the molecules produced by the generative model are difficult to synthesize and, thus, their properties lack experimental verification; and studies on retrosynthesis prediction remain mainly confined to single-step retrosynthesis prediction. Furthermore, as the most popular deep learning models for dealing with chemical molecule data, GNNs also have distinct problems such as 'oversmoothing'. When stacking too many multiple layers, the features of different nodes in a graph become indistinguishable, which seriously damages the performance of models. Given this limitation, most current GNNs are no deeper than four layers. However, with the increase in model depth, the representation ability of the model will also increase. The effectiveness of ultradeep neural networks has been proved in the field of computer vision. The current state-of the art CNNs are usually >100 layers [84,85]. Recently, Li et al. [86] built a 56-layer GNN improving the state-of-the-art in the task of point cloud semantic segmentation by 3.7%. Their work demonstrated the possibility of stacking GNNs as deep as CNNs and the advantages of ultra-deep GNNs. However, a key strategy they used to build such a deep GNN was dynamically changing edges in the graph, which is not suitable for graphs with fixed edges, such as molecules. Thus, methods to build deeper GNNs for molecule learning need to be further explored.

Conflict of interest

The authors declare that they have no conflict of interest.

Acknowledgments

References

1 DiMasi, J.A. et al. (2016) Innovation in the pharmaceutical industry: new estimates of R&D costs. J. Health. Econ. 47, 20–33

2 Paul, S.M. et al. (2010) How to improve R&D productivity: the pharmaceutical industry's grand challenge. Nat. Rev. Drug Discov. 9, 203–214

3 Schneider, G. and Fechner, U. (2005) Computer-based de novo design of drug-like molecules. Nat. Rev. Drug Discov. 4, 649–663

4 Nishibata, Y. and Itai, A. (1991) Automatic creation of drug candidate structures based on receptor structure. Starting point for artificial lead generation. Tetrahedron 47, 8985–8990

5 Nishibata, Y. and Itai, A. (1993) Confirmation of usefulness of a structure construction program based on three-dimensional receptor structure for rational lead generation. J. Med. Chem. 36, 2921–2928

6 Verlinde, C.L. et al. (1992) In search of new lead compounds for trypanosomiasis drug design: a protein structure-based linked-fragment approach. J. Comput. Aided Mol. Des. 6, 131–147

7 Pagadala, N.S. et al. (2017) Software for molecular docking: a review. Biophys. Rev. 9, 91–102

8 Zhavoronkov, A. et al. (2019) Deep learning enables rapid identification of potent DDR1 kinase inhibitors. Nat. Biotechnol. 37, 1038–1040

9 Yang, Y. et al. (2020) Discovery of highly potent, selective, and orally efficacious P300/CBP histone acetyltransferases inhibitors. J. Med. Chem. 63, 1337–1360

10 MIT Technology Review, www.technologyreview.com/10-breakthrough-technologies/2020/ [accessed 10.2.21].

11 Ekins, S. et al. (2019) Exploiting machine learning for end-to-end drug discovery and development. Nat. Mater. 18, 435–441

12 Chen, H. et al. (2018) The rise of deep learning in drug discovery. Drug Discov. Today 23, 1241–1250

13 Kipf, T.N. and Welling, M. (2016) Semi-supervised classification with graph convolutional networks. arXiv 2016, 1609.02907

14 Jin, W. et al. (2018) Junction tree variational autoencoder for molecular graph generation. Proc. Machine Learn. Res. 80, 2323–2332

15 Kwon, Y. et al. (2019) Efficient learning of non-autoregressive graph variational autoencoders for molecular graph generation. J. Cheminf. 11, 70

16 García-Domenech, R. et al. (2008) Some new trends in chemical graph theory. Chem. Rev. 108, 1127–1169

17 Mak, K.-K. and Pichika, M.R. (2019) Artificial intelligence in drug development: present status and future prospects. Drug Discov. Today 24, 773–780

18 Öztürk, H. et al. (2020) Exploring chemical space using natural language processing methodologies for drug discovery. Drug Discov. Today 25, 689–705

19 Chan, H.S. et al. (2019) Advancing drug discovery via artificial intelligence. Trends Pharmacol. Sci. 40, 592–604

20 Tian, S. et al. (2015) The application of in silico drug-likeness predictions in pharmaceutical research. Adv. Drug Deliv. Rev. 86, 2–10

21 Zoete, V. et al. (2016) SwissSimilarity: a web tool for low to ultra high throughput ligand-based virtual screening. J. Chem. Inf. Model. 56, 1399–1404

22 Duvenaud, D.K. et al. (2015) Convolutional networks on graphs for learning molecular fingerprints. In NIPS'15: Proceedings of the 28th International Conference on Neural Information Processing Systems (Cortes, C., ed.), pp. 2224–2232, MIT Press

23 Kearnes, S. et al. (2016) Molecular graph convolutions: moving beyond fingerprints. J. Comput. Aided Mol. Des. 30, 595–608

24 Gilmer, J. et al. (2017) Neural message passing for quantum chemistry. Proc. Machine Learn. Res. 70, 1263–1272

25 Coley, C.W. et al. (2017) Convolutional embedding of attributed molecular graphs for physical property prediction. J. Chem. Inf. Model. 57, 1757–1772

26 Ishiguro, K. et al. (2019) Graph Warp Module: an auxiliary module for boosting the power of graph neural networks. arXiv 2019, 1902.01020

27 Li, S. et al. (2020) MONN: a multi-objective neural network for predicting compound-protein interactions and affinities. Cell Syst. 10, 308–322

28 Chen, C. et al. (2019) Graph networks as a universal machine learning framework for molecules and crystals. Chem. Mat. 31, 3564–3572

29 Guidotti, R. et al. (2018) A survey of methods for explaining black box models. ACM Comput. Surv. 51, 1–42

30 Xiong, Z. et al. (2019) Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. J. Med. Chem. 63, 8749–8760

31 Tang, B. et al. (2020) A self-attention based message passing neural network for predicting molecular lipophilicity and aqueous solubility. J. Cheminf. 12, 1–9

32 Yang, K. et al. (2019) Analysing learned molecular representations for property prediction. J. Chem. Inf. Model. 59, 3370–3388

33 Stokes, J.M. et al. (2020) A deep learning approach to antibiotic discovery. Cell 180, 688–702

34 Fernandes, P. (2015) The global challenge of new classes of antibacterial agents: an industry perspective. Curr. Opin. Pharmacol. 24, 7–11

35 Feinberg, E.N. et al. (2020) Improvement in ADMET prediction with multitask deep featurization. J Med Chem 63, 8835–8848

36 Gao, K.Y. *et al.* (2018) Interpretable drug target prediction using deep neural representation. *In Proceedings of the 27th International Joint Conference on Artificial Intelligence*, ICJAI. pp. 3371–3377

37 Tsubaki, M. *et al.* (2019) Compound–protein interaction prediction with end-to-end learning of neural networks for graphs and sequences. *Bioinformatics* 35, 309–318

38 Feinberg, E.N. *et al.* (2018) PotentialNet for molecular property prediction. *ACS Cent. Sci.* 4, 1520–1530

39 Lim, J. *et al.* (2019) Predicting drug–target interaction using a novel graph neural network with 3D structure-embedded graph representation. *J. Chem. Inf. Model.* 59, 3981–3988

40 Torng, W. and Altman, R.B. (2019) Graph convolutional neural networks for predicting drug-target interactions. *J. Chem. Inf. Model.* 59, 4131–4149

41 Bagley, S.C. and Altman, R.B. (2008) Characterizing the microenvironment surrounding protein sites. *Protein Sci.* 4, 622–635

42 Wu, Z. *et al.* (2018) MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci.* 9, 513–530

43 Sieg, J. *et al.* (2019) In need of bias control: evaluating chemical data for machine learning in structure-based virtual screening. *J. Chem. Inf. Model.* 59, 947–961

44 Tran-Nguyen, V.-K. *et al.* (2020) LIT-PCBA: an unbiased data set for machine learning and virtual screening. *J. Chem. Inf. Model.* 60, 4263–4273

45 Yang, J. *et al.* (2020) Predicting or pretending: artificial intelligence for protein–ligand interactions lack of sufficiently large and unbiased datasets. *Front. Pharmacol.* 11, 69

46 Chen, L. *et al.* (2019) Hidden bias in the DUD-E dataset leads to misleading performance of deep learning in structure-based virtual screening. *PLoS ONE* 14, e0220113

47 Li, T. *et al.* (2020) Federated learning: challenges, methods, and future directions. *IEEE Signal Proc Mag* 37, 50–60

48 Simonovsky, M. and Komodakis, N. (2018) GraphVAE: towards generation of small graphs using variational autoencoders. *Lect. Notes Comp. Sci.* 11139, 412–422

49 Maziarka, . *et al.* (2020) Mol-CycleGAN: a generative model for molecular optimization. *J. Cheminf.* 12, 1–18

50 De Cao, N. and Kipf, T. (2018) MolGAN: an implicit generative model for small molecular graphs. *arXiv* 2018, 1805.11973

51 Madhawa, K. *et al.* (2019) GraphNVP: an invertible flow model for generating molecular graphs. *arXiv* 2019, 1905.11600

52 Popova, M. *et al.* (2019) MolecularRNN: generating realistic molecular graphs with optimized properties. *arXiv* 2019, 1905.13372

53 Jin, W. *et al.* (2018) Learning multimodal graph-to-graph translation for molecular optimization. *arXiv* 1812.01070

54 Fu, T. *et al.* (2020) Core: automatic molecule optimization using copy & refine strategy. In *Proceedings of the 35th AAAI Conference on Artificial Intelligence*, AAAI. pp. 638–645

55 Shi, C. *et al.* (2020) GraphAF: a flow-based autoregressive model for molecular graph generation. *arXiv* 2020, 2001.09382

56 Bradshaw, J. *et al.* (2019) A model to search for synthesizable molecules. *Adv. Neural Inf. Process. Syst.* 33, 7937–7949

57 Kullback, S. and Leibler, R.A. (1951) On information and sufficiency. *Ann. Math. Stat.* 22, 79–86

58 Li, Y. *et al.* (2018) Multi-objective de novo drug design with conditional graph generative model. *J. Cheminf.* 10, 33

59 Gao, W. and Coley, C.W. (2020) The synthesizability of molecules proposed by generative models. *J. Chem. Inf. Model.* 60, 5714–5723

60 Lyu, J. *et al.* (2019) Ultra-large library docking for discovering new chemotypes. *Nature* 566, 224–229

61 Gloriam, D.E. (2019) Bigger is better in virtual drug screens. *Nature* 566, 193–194

62 Bush, J.T. *et al.* (2020) A Turing test for molecular generators. *J. Med. Chem.* 63, 11964–11971

63 Coley, C.W. *et al.* (2018) Machine learning in computer-aided synthesis planning. *Acc. Chem. Res.* 51, 1281–1289

64 Corey, E.J. *et al.* (1985) Computer-assisted analysis in organic synthesis. *Science* 228, 408–418

65 Szymkuć, S. *et al.* (2016) Computer-assisted synthetic planning: the end of the beginning. *Angew. Chem. Int. Ed.* 55, 5904–5937

66 Herges, R. and Hoock, C. (1992) Reaction planning: computer-aided discovery of a novel elimination reaction. *Science* 255, 711–713

67 Segler, M.H. *et al.* (2018) Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* 555, 604–610

68 Coley, C.W. *et al.* (2019) A robotic platform for flow synthesis of organic compounds informed by AI planning. *Science* 365, eaax1566

69 Jin, W. *et al.* (2017) Predicting organic reaction outcomes with Weisfeiler-Lehman network. In *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems* (von Luxberg, U. and Guyon, I., eds), pp. 2607–2616, Curran Associates Inc

70 Coley, C.W. *et al.* (2019) A graph-convolutional neural network model for the prediction of chemical reactivity. *Chem. Sci.* 10, 370–377

71 Qian, W.W. *et al.* (2020) Integrating deep neural networks and symbolic inference for organic reactivity prediction. *ChemRxiv* 2020, 11659563

72 Do, K. *et al.* (2019) Graph transformation policy network for chemical reaction prediction. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining* (Teredesai, A. and Kumar, V., eds), In pp. 750–760, Association for Computing Machinery

73 Roszak, R. *et al.* (2019) Rapid and accurate prediction of pKa values of C–H acids using graph convolutional neural networks. *J. Am. Chem. Soc.* 141, 17142–17149

74 Dai, H. *et al.* (2019) Retrosynthesis prediction with conditional graph logic network. *Adv. Neural Inf. Process. Syst.* 33, 8872–8882

75 Marwin *et al.* (2017) Neural-symbolic machine learning for retrosynthesis and reaction prediction. *Chemistry* 23, 5966–5971

76 Vaswani, A. *et al.* (2017) Attention is all you need. In *NIPS'17: Proceedings of the 31st International Conference on Neural Information Processing Systems* (von Luxberg, U. and Guyon, I., eds), pp. 5998–6008, Curran Associates Inc

77 Yan, C. *et al.* (2020) RetroXpert: Decompose retrosynthesis prediction like a chemist. *ChemRxiv* 2020, 11869692.v5

78 Shi, C. *et al.* (2020) A graph to graphs framework for retrosynthesis prediction. *arXiv* 2003.12725

79 Somnath, V.R. *et al.* (2020) Learning graph models for template-free retrosynthesis. *arXiv* 1609.02907

80 Kishimoto, A. *et al.* (2019) Depth-first proof-number search with heuristic edge cost and application to chemical synthesis planning. *Adv. Neural Inf. Process. Syst.* 32, 7226–7236

81 Schreck, J.S. *et al.* (2019) Learning retrosynthetic planning through simulated experience. *ACS Cent. Sci.* 5, 970–981

82 Schwaller, P. *et al.* (2020) Predicting retrosynthetic pathways using transformer-based models and a hyper-graph exploration strategy. *Chem. Sci.* 11, 3316–3325

83 Mikulak-Klucznik, B. *et al.* (2020) Computational planning of the synthesis of complex natural products. *Nature* 588, 83–88

84 He, K. *et al.* (2016) Deep residual learning for image recognition. *In Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition*, IEEE. pp. 770–778

85 Hu, J. *et al.* (2018) Squeeze-and-excitation networks. *In Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, IEEE. pp. 7132–7141

86 Li, G. *et al.* (2019) DeepGCNs: can GNNs go as deep as CNNs? In *Proceedings of IEEE/CVF International Conference on Computer Vision*, IEEE. pp. 9267–9276

87 Schwaller, P. *et al.* (2018) 'Found in Translation': predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *Chem. Sci.* 9, 6091–6098

88 Schwaller, P. *et al.* (2019) Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction. *ACS Cent. Sci.* 5, 1572–1583

89 Liu, B. *et al.* (2017) Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS Cent. Sci.* 3, 1103–1113

90 Coley, C.W. *et al.* (2017) Computer-assisted retrosynthesis based on molecular similarity. *ACS Cent. Sci.* 3, 1237–1245

91 Karpov, P. *et al.* (2019) A transformer model for retrosynthesis. *Lecture Notes Comp. Sci.* 11731, 817–830

92 Lee, A. *et al.* (2019) Molecular transformer unifies reaction prediction and retrosynthesis across pharma chemical space. *Chem. Commun.* 55, 12152–12155

93 Zheng, S. *et al.* (2019) Predicting retrosynthetic reactions using self-corrected transformer neural networks. *J. Chem. Inf. Model.* 60, 47–55

94 Lin, K. *et al.* (2020) Automatic retrosynthetic route planning using template-free models. *Chem. Sci.* 11, 3355–3364

95 Mao, K. *et al.* (2020) Molecular graph enhanced transformer for retrosynthesis prediction. *bioRxiv* . http://dx.doi.org/10.1101/2020.03.05.979773 [published online March 6, 2020]