

Artificial Intelligence in Drug Discovery: Applications and Techniques

Jianyuan Deng,^{*,†} Zhibo Yang,[‡] Iwao Ojima,[¶] Dimitris Samaras,[‡] and Fusheng Wang^{†,‡}

[†]*Department of Biomedical Informatics, Stony Brook University, Stony Brook, NY, USA*

[‡]*Department of Computer Science, Stony Brook University, Stony Brook, NY, USA*

[¶]*Department of Chemistry, Stony Brook University, Stony Brook, NY, USA*

E-mail: jianyuan.deng@stonybrook.edu

Abstract

Artificial intelligence (AI) has been transforming the practice of drug discovery in the past decade. Various AI techniques have been used in many drug discovery applications, such as virtual screening and drug design. In this survey, we first give an overview on drug discovery and discuss related applications, which can be reduced to two major tasks, i.e., molecular property prediction and molecule generation. We then present common data resources, molecule representations and benchmark platforms. As a major part of the survey, AI techniques are dissected into model architectures and learning paradigms. To reflect the technical development of AI in drug discovery over the years, the surveyed works are organized chronologically. We expect that this survey provides a comprehensive review on AI in drug discovery. We also provide a GitHub repository with a collection of papers (and codes, if applicable) as a learning resource, which is regularly updated.

Introduction

Drug discovery is well known as an expensive, time-consuming process, with low success rates. On average, developing a new drug costs 2.6 billion US dollars¹ and can take more than 10 years. Moreover, the success rate of launching a drug to market from Phase I clinical trial is daunting, less than 10%.² In the past decade, the practice of drug discovery has been undergoing radical transformations driven by the rapid development in artificial intelligence (AI).³⁻⁷ Popular applications of AI in drug discovery include virtual screening,⁸ *de novo* drug design,⁹ retrosynthesis and reaction prediction,¹⁰ and *de novo* protein design,¹¹ among others, which can be reduced to two categories, i.e., predictive and generative tasks. To power these AI applications, a wide range of AI techniques are involved, with model architectures evolving from traditional machine learning models to deep neural networks, such as convolutional neural networks, recurrent neural networks, graph neural networks and transformers, etc. Learning paradigms also shift from supervised learning to self-supervised learning and reinforcement learning.

In this survey, we focus on the applications and techniques of AI-driven discovery on small-molecule drugs. Biologics (e.g. antibodies, vaccines) are not covered. We first provide an overview of key applications in drug discovery and point out a collection of previously published perspectives, reviews, and surveys. We then introduce common data resources and representations of small molecules. We also discuss existing benchmark platforms for both molecular property prediction and molecule generation. With knowledge on data and representations, the related techniques, including model architectures and learning paradigms, will be elaborated. Finally, we discuss existing challenges and highlight some future directions. By assembling a Github repository¹ with the surveyed papers (and codes, if applicable), we expect this survey not only provides a comprehensive overview of AI in drug discovery but also serves as a learning resource for researchers interested in the interdisciplinary field.

¹https://github.com/dengjianyuan/Survey_AI_Drug_Discovery

Drug Discovery Overview

In this section, we first go over the definitions of key concepts in drug discovery, mainly on the screening and design of small molecules. Note that AI-powered drug repositioning, which repurposes existing drugs or drug combinations for new indications, is not included. Besides, target identification, exploiting -omics data for its druggability, is also out of the scope and thus not discussed. Rather, we refer the readers to previous publications on drug repositioning^{12,13} and -omics data for target identification.¹⁴

*Drug discovery*¹⁵ is a project motivated by the situation when there are no drugs for a disease or when existing drugs have limited efficacy and/or severe toxicity. At the earliest stage, an underlying hypothesis needs to be developed that activation or inhibition of a target (e.g., an enzyme, a receptor, an ion channel, etc) results in therapeutic effects for the disease, which involves target identification and target validation. For the selected target, intensive assays will be performed to find the hits and subsequently the leads (i.e., drug candidates), which involves *hit discovery*, *hit-to-lead phase* and *lead optimization*. The drug candidates then enter preclinical studies and clinical trials. If successful, the drug candidate can be launched into market as a medical product to treat the disease.

To accelerate the small-molecule drug discovery, *high-throughput screening (HTS)*^{16,17} has been proposed to increase the discovery efficiency since the 1980s, which is a hit-finding approach underpinned by development in automation and the availability of large chemical libraries. A prominent outcome of HTS is the large-scale *structure-activity relationship (SAR)* datasets, which contribute to the chemical databases such as PubChem¹⁸ and ZINC.¹⁹ Various computational techniques have been developed to search the chemical libraries for potentially active molecules to be tested in subsequent *in vitro* and *in vivo* assays,²⁰ which is also known as *virtual screening (VS)*. In other words, VS is to identify active molecules using computational approaches, based on knowledge about the target (*structure-based VS*) or known active ligands (*ligand-based VS*) to increase the odds of identifying active molecules.²¹ For the concept of active molecules, as mentioned above, activation or inhibition of a target

is the underlying hypothesis to treat a disease, which corresponds to two major classes of drugs with regard to the *mechanism of action (MoA)*, i.e., *agonists* and *antagonists*.²² An agonist is a molecule which activates the target to exert a biologic response as its endogenous ligand. On the contrary, an antagonist is a molecule which binds to the target to block the response. Based on more specific effects and mechanisms, agonists can further be classified as partial agonists, inverse agonists, biased agonists. Antagonists include competitive and non-competitive antagonists. To quantify the activity, various assays have been developed to measure *affinity* (or *potency*) and *efficacy*. Affinity is the fraction or extent to which a molecule binds to a target at a given concentration whereas potency is the necessary amount of a molecule to produce an effect of a given magnitude, inversely proportional to the affinity. Efficacy, on the other hand, describes the effect size, such as inhibition of an enzyme to 60%. Common activity measures are summarized in Table 1.

Table 1: Common Measures of Molecule Activity

Measures	Definition
Kd	Equilibrium dissociation constant
Km	Michaelis constant
Ki	Inhibition constant
IC50	Half maximal inhibitory concentration
EC50	Half maximal effective concentration

Nevertheless, sufficient activity is not the only criterion for an ideal drug candidate, which only makes it a *ligand*. Binding specificity is another concern.²³ Most often, a molecule can bind to multiple targets and unexpected side effects may arise due to binding promiscuity. Thus, high selectivity is another desired feature. In fact, drug candidates should also satisfy a combination of criteria,²⁴ with optimal physicochemical (*water solubility, acid-base dissociation constant, lipophilicity, permeability*), pharmacokinetic (*absorption, distribution, metabolism, excretion*), and pharmacodynamic (*activity, selectivity*) properties. Other properties considered during compound synthesis include the Synthetic Accessibility Score (SAS) and the Quantitative Estimation of Drug-likeness (QED). SAS is a heuristic score of how hard (10) or easy (1) it is to synthesize a given molecule based on a combination of the molecular

fragments’ contributions. QED is an estimate (0-1) on how likely a molecule is a viable drug candidate. At its core, drug discovery is a multi-objective optimization problem.²⁵ Usually, for each property of interest, a predictive model is built to map the molecular structure to the property value with either classification or regression, which is broadly referred to as *quantitative structure-activity relationship (QSAR)* modeling.²⁶ A more intriguing prospect from QSAR is that these models can be exploited inversely to reveal the structural features underlying the optimal properties to guide *drug design* from scratch, also known as *de novo* drug design.

Rather than merely screening existing chemical libraries,²⁷ drug design takes a step further to explore the vast *chemical space*, i.e., the space encompassing all possible small molecules²⁸ which has an estimated size around $10^{30} - 10^{60}$.²⁹ In drug design, there is a *design-make-test-analysis (DMTA)* cycle, which consists of iterative organic synthesis and property assays.³ To efficiently navigate the chemical space, quantitative drug design has been proposed since late 1970s.³⁰ Essentially, drug design epitomizes in two questions:³¹ 1) “Can molecular properties be deduced from molecular structures?” and 2) “Which structural features are relevant for certain molecular properties?” The former one also underlies the core assumption of VS and the latter is what QSAR tries to answer. In this sense, drug design can be viewed as an extension to VS, and involves both *molecular property prediction* and *molecule generation*, which are the major tasks in current AI-driven drug discovery.

Summary of Existing Reviews

Next, we briefly discuss existing reviews on AI-driven discovery by dividing them into three categories: 1) “General drug discovery review”, 2) “Drug discovery in the AI-era”, and 3) “Rethinking AI-driven drug discovery”.

General Drug Discovery Reviews

Many existing papers have covered the general aspects of drug discovery^{3,15} and related concepts, such as chemical space,²⁸ VS and HTS,^{21,32,33} optimal properties for drug candidates,²⁴ QSAR,^{26,34} target prediction,³⁵ and computer-aided drug design.²⁷ Besides, one prominent challenge in drug discovery is that molecular properties can be highly sensitive to minor structural changes. This is also known as the *activity cliffs (ACs)*, where pairs of structurally similar molecules exhibit significantly different activities.³⁶⁻³⁸ We strongly recommend the readers (especially those new to drug discovery) refer to these reviews for a better understanding on drug discovery and recognition of potential pitfalls.

Drug Discovery in the AI Era

AI has been widely applied in drug discovery. Since the early 2000s, machine learning models, such as random forest (RF), have been exploited for VS and QSAR.^{39,40} In 2012, AlexNet⁴¹ marked the advent of the deep learning era.⁴² Shortly after in the 2012 Merck Kaggle competition, deep neural networks (DNN) outperformed the standard RF model in predicting molecular activities.³⁹ More recently, the success of AI techniques in computer vision and natural language processing has shed more light into drug discovery^{4,6,43,44} and led to the burgeoning field of deep learning in chemistry.⁵ In 2019, potent inhibitors of discoidin domain receptor 1 (DDR1) were discovered in 21 days by researchers from Insilico Medicine.⁴⁵ In 2020, a novel antibiotic candidate against antibiotic-resistant bacteria, halicin, was identified by researchers from MIT.⁴⁶ Note that AI can be applied at different stages in drug discovery, from target identification and validation to drug response determination.⁶ Lead identification, the focus of this survey, involves two fundamental tasks, i.e., molecular property prediction and molecule generation. Molecular property prediction, at the core of VS, is to predict the property value of a molecule given its structure or learned representation,⁴⁷ which can be served for various purposes, such as drug-target interaction (DTI) prediction,³⁵ toxicity prediction⁴⁸ and drug-induced liver injury (DILI) prediction,⁴⁹ among

others. Molecule generation, underlying drug design, involves two levels of tasks: 1) realistic molecule generation, i.e., generating molecules within constraints imposed by the chemical rules, and 2) goal-directed molecule generation, i.e., generating chemically valid molecules with desired properties.^{50,51}

Rethinking AI-driven Drug Discovery

Despite the promise of AI in drug discovery, pitfalls still exist, which have been widely discussed.^{8,9,31,52–55} As opined by Bender et al.,^{53,54} “a method cannot save an unsuitable representation which cannot remedy irrelevant data for an ill thought-through question”. To circumvent potential hypes and unrealistic expectations thereof, there is indeed a necessity to take into consideration the hypotheses, the data, the representations, the models, the learning paradigms and moreover, these components as a whole, for any AI-driven application in drug discovery.

Structure of the Survey

Given the necessity of a clear understanding on both drug discovery applications and AI techniques, this survey starts from general aspects in drug discovery and then moves on to AI-driven drug discovery, covering data resources, molecule representations, model architectures, and learning paradigms. The organization of this survey is depicted in Fig 1.

Notably, current literature on the AI techniques is often fragmented. More often than not, the rationale behind the choice of a technique is simply because it has not been previously investigated.⁵¹ To gain more insights into the strengths and weaknesses of these AI techniques, we focus on the model architectures and learning paradigms. We also try to present the surveyed works chronologically so as to reflect the technical development over the years. Finally, we highlight existing challenges and future directions (see Section “Discussion”).

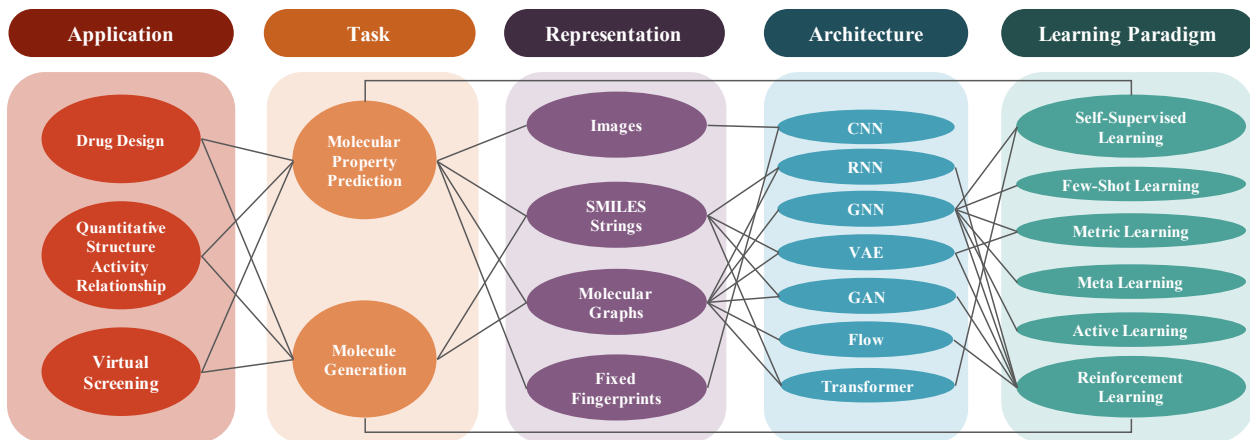


Figure 1: Applications and Techniques of AI in Drug Discovery. The applications of AI in small-molecule drug discovery include virtual screening, quantitative structure-activity relationship and drug design, which can be reduced to two major tasks: molecular property prediction and molecule generation. Small molecules can be represented by fixed fingerprints, molecular graphs, simplified molecular input entry system (SMILES) strings, and images. Various model architectures have been applied on each representation format, including convolutional neural networks (CNN), recurrent neural networks (RNN), graph neural networks (GNN), variational autoencoders (VAE), generative adversarial networks (GAN), normalizing flow models and transformers. Still, challenges exist for the low-data molecular property prediction and goal-directed molecule generation. To tackle these challenges, different learning paradigms have been proposed, such as self-supervised learning for the pretraining-finetuning practice and reinforcement learning for navigating the chemical space search. Other paradigms surveyed here also include few-shot learning, metric learning, meta learning and active learning.

Data, Representation and Benchmark Platforms

In this section, we first discuss the publicly available data resources. Then, we discuss how small molecules can be represented in machine-readable formats. Lastly, we summarize current benchmark platforms for both molecular property prediction and molecule generation.

Public Data Resources

With the improvements in HTS and related assays, data on molecular activity and related properties are ever increasing, which contribute to various public data resources. These resources typically provide information on molecular structures, molecular properties and target information,⁵⁶ as discussed below.

*PubChem*⁵⁷ was launched by the National Institutes of Health in 2004. With a collection of chemical information from 750 data sources, PubChem is the largest chemical database. As of August 2020, PubChem contains 111 million unique chemical structures with 271 million activity data points from 1.2 million biological assays experiments. PubChem provides direct download as well as web interfaces for online queries. Notably, PubChem is non-curated⁵⁶ and the bioactivity datasets from PubChem can be highly imbalanced.⁵⁸ Researchers may curate the data on their own. For example, Chithrananda et al⁵⁹ recently released a curated dataset of 77 million SMILES strings from PubChem. *ChEMBL*,⁶⁰ maintained by the European Molecular Biology Laboratory, is another large-scale chemical database. For example, in ChEMBL22 (version 22), there are more than 1.6 million distinct chemical structures with over 14 million activity values. Moreover, ChEMBL is manually curated in a comprehensive manner.⁵⁶ ChEMBL provides downloads in a variety of formats (e.g., Oracle, MySQL or PostgreSQL database) and also allows web application program interface (API) for data retrieval in XML or JSON format.⁶¹ Notably, based on ChEMBL, Mayr et al⁶² extracted a large-scale benchmark dataset for target prediction. ZINC,¹⁹ developed by the Irwin and Shoichet Laboratories in UCSF, contains a suite of molecules, annotated ligands and targets as well as the purchasability for over 120 million “drug-like” compounds. ZINC supports direct download from the website and also provides an API for retrieving data. Notably, some subsets of the ZINC database are more commonly used, such as the ZINC-250k⁶³ and the ZINC Clean Leads collections.⁶⁴

In addition to the aforementioned large-scale databases, there are also other data repositories,⁶⁵ such as PDBbind, BindingDB, DUD, DUD-E, MUV, STITCH, GLL&GDD, NRLiST BDB, KEGG, among others. Besides databases mainly derived from preclinical studies, there are also public data resources for the marketed drugs and their effects in human subjects, such as the datasets for adverse drug reactions (ADR) (e.g., DrugBank, SIDER, OFFSIDES and TWO-SIDES) and the datasets for DILI (e.g., DILIRank).⁶⁶

Small Molecule Representations

Molecules are often depicted as Kekulé diagrams with bonds and atoms (Fig 2A). Over the years, machine-readable representations have been developed to enable rapid computation, querying and storage of the molecules.⁶⁷ Molecules can be represented by the fixed molecular descriptors, which are further categorized by its dimensionality.⁵⁶ Specifically, there are 0D descriptors for molecules, such as molecular weight (MW), atom number, and atom-type count. 0D descriptors can be directly derived from the empirical formula and barely provide information on how atoms are connected. For example, the empirical formula of alanine, $C_3H_7NO_2$, can also represent lactamide.⁶⁷ To highlight different functional groups, descriptors incorporating more structural information have been proposed, such as fingerprints (Fig 2B). Fingerprints are binary vectors with each dimension in the vector indicating the presence or absence of a particular substructure. Among them, there are 1D descriptors to represent the substituent atoms, chemical bonds, structural fragments, and functional groups. There are also 2D descriptors to represent the atom connectivity and molecular topology, such as 1) Keyed fingerprints - molecular access system (MACCS) keys, 2) Path-based fingerprints - DayLight fingerprints and 3) Circular fingerprints - extended connectivity fingerprints (ECFPs) based on the Morgan algorithm.⁶⁸ Furthermore, 3D descriptors have also been developed to encode 3D-structural information like the steric properties, surface area, volume and binding site properties, among others.

Molecular descriptors have greatly boosted the application of computational methods, including machine learning models, in drug discovery.^{69,70} Nevertheless, these descriptors are fixed and not learnable towards improving model performance. With the advent of the AI era, various deep learning models have paved the way for end-to-end (E2E) predictions, where molecules can be embedded into a continuous latent space without hand-crafted rules. Among them, two major representation formats are molecular graphs and the simplified molecular input entry system (SMILES) strings.⁶⁷

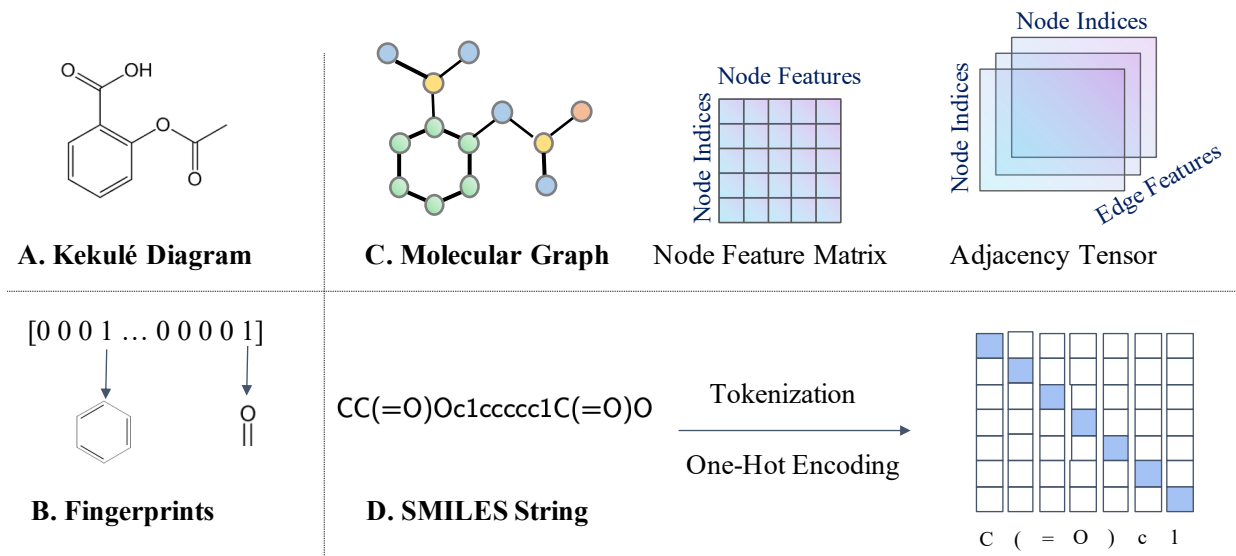


Figure 2: Illustration of Small Molecule Representations.

Molecular Graphs

The idea of graph representation is intuitive, where atoms are typically mapped to nodes and bonds to edges. Formally, a graph is defined as $G = (V, E)$, a set of nodes (atoms) V and a set of edges (bonds) E , where $(v_i, v_j) \in E$ indicates a bond between atoms v_i and v_j .⁶⁷ The attributes of atoms are represented by a node feature matrix X and each node v can be represented by an initial vector x_v and a hidden vector $h_v \in R^D$. Similarly, the attributes of bonds can also be represented by an edge feature matrix. Note that both node and edge feature matrices do not directly encode connections. Instead, an adjacency matrix A keeps track of the pairwise connection status. The element of A , a_{ij} , if equals to 1, means that there is a bond connecting node v_i and v_j ; otherwise, if a_{ij} equals to 0, these two nodes are not connected by a bond. Usually, the edge feature matrix and the adjacency matrix are combined to form an adjacency tensor (Fig 2C). Common node and edge features^{71,72} are summarized in Table 2.

One advantage of the graph representations is that they carry more structural information. Besides, molecular graphs as well as the subgraphs can be directly mapped to a chemical (sub-)structure and thus are highly interpretable.⁷³ One drawback of the graph

Table 2: Common Node and Edge Features in Molecular Graphs

Type	Feature	Notes
Node	Atom type	Element type
Node	Formal charge	Assigned charges
Node	Implicit Hs	Number of bonded hydrogens
Node	Chirality	R or S configuration
Node	Hybridization	Orbital hybridization: sp^x , $sp^x d^y$
Node	Aromaticity	Aromatic atom or not
Edge	Bond type	Single, double, triple or aromatic
Edge	Conjugated	Conjugated or not
Edge	Stereoisomers	cis (Z) or trans (E)

representation, however, is that these matrices require a large amount of disk space for storage and significant memory during computation, which may slow down the efficiency during molecule generation.⁶⁷

SMILES Strings

To accommodate the storage and computation efficiency, molecules are also commonly represented by the SMILES strings.⁷⁴ In SMILES, an atom is represented by the atomic symbols; for two-character symbols, the second letter will be represented in lower case. Elements in the organic subset, namely B, C, N, O, P, S, F, Cl, Br and I, can be written without brackets whereas for those not included, brackets should be applied with the attached hydrogens and formal charges written inside, such as [Fe2+]. The lower-case letters represent the atoms in aromatic rings; for instance, C is used for the normal carbon and c is used for the aromatic carbon. For bonds, there are single, double, triple and aromatic bonds, represented by the symbols -, =, # and :, respectively, where single bonds and aromatic bonds are usually omitted. For the branches in a molecule, they are denoted by enclosures in parentheses. To represent the cyclic structure, a single or aromatic bond is first broken down in the ring and then the bonds are numbered in any order with the ring-opening bonds by a digit following the atomic symbol at each ring. Notably one molecule may correspond to multiple SMILES strings.⁶⁷ To avoid conflicts, canonicalization methods⁷⁵ have been introduced to ensure only

one unique SMILES string is designated for the same molecule.

Usually the SMILES string are converted into one-hot vectors before fed into the machine learning models (Fig 2D).⁷⁶ Comparing to the graph representation, SMILES string is less computationally expensive. However, since SMILES strings do not directly encode the atomic connection, there can be a loss of the structural information.⁷⁷ Besides, due to the internal syntax of the SMILES (e.g., ring opening and closure, atom valency), using this linear notations for molecule generation is prone to generate invalid molecules.^{78,79}

Other Representations

Molecules can also be represented by more sophisticated 3D-atomic coordinates, commonly seen in structure-based VS or QSAR studies.⁸⁰⁻⁸³ Molecular topology, such as bond lengths, bond angles and torsional angles, can also be incorporated.⁸⁴ Some works have already exploited the 3D-representation to generate molecules.^{85,86} In addition to the raw 3D coordinates, well-established 3D properties, which capture the molecular conformation, can also be readily utilized for prediction tasks.⁸⁷ Furthermore, with the advances of computer vision, images of molecular structures (Fig 2A) emerge as another modality to represent molecules.⁸⁸⁻⁹²

Benchmark Platforms

To evaluate the performance of molecular property prediction and molecule generation, there are several benchmark platforms, which are discussed below.

As a major benchmark dataset platform for molecular property prediction, MoleculeNet was released by Wu et al in 2018,⁹³ which includes a set of datasets along with an open-source DeepChem package.⁹⁴ The benchmark datasets cover four categories: 1) Quantum mechanics (QM7, QM7b, QM8, QM9), 2) Physical chemistry (ESOL, FreeSolv, Lipophilicity), 3) Biophysics (PCBA, MUV, HIV, PDDBind, BACE), and 4) Physiology (BBBP, Tox21, ToxCast, SIDER, ClinTox), involving single task or multi tasks. Notably, for molecular property

prediction, datasets can be highly imbalanced. Thus, when choosing evaluation metrics (Table 3), positive rates should be considered. For instance, AUPRC is favored over AUROC in case of a low positive rate (e.g. less than 2%).⁹³ With regard to the dataset splitting, in addition to the common random split, MoleculeNet also provides other splitting ways, namely, scaffold split, stratified split and time split for different datasets. In other words, for each dataset, the recommended split way varies. For example, for the BACE dataset, since it is for a single target, the scaffold splitting is more suitable, whereas for the PDBBind dataset, since the data collection is over a long period, time splitting is recommended to better reflect the actual drug discovery effort over the years.

One limitation of MoleculeNet, however, is that it does not provide explicit training, validation and test folds for the datasets.⁹⁵ To improve reproducibility, the ChemBench package from MolMapNet⁹⁶ was released recently. MolMapNet also expands the MoleculeNet by adding pharmacokinetics-related datasets, such as PubChem CYP inhibition and liver microsomal clearance data. In addition to the benchmark datasets, Chemprop,⁷¹ for benchmarking the learned molecular representations, was proposed in 2019, which systematically compared the fixed molecular descriptors (e.g. ECFPs) and the learned molecular representations for molecular property prediction. In Chemprop, models were benchmarked extensively on 19 public and 16 proprietary industrial datasets. As a side note, Chemprop is related to the discovery of halicin.⁴⁶

As for benchmarking the molecule generation models, Olivecrona et al⁹⁷ developed REINVENT in 2017, which is a sequence-based generative model utilizing SMILES strings. REINVENT can be used to execute a range of tasks, such as generating analogues to a query structure and generating ligands for a given target. In 2020, Blaschke et al⁹⁸ proposed the updated version, REINVENT 2.0, as a production-ready tool for drug design. For benchmarking molecule generation utilizing molecular graphs, Mercado et al⁷² proposed GraphINVENT in 2020. To standardize the assessment for molecule generation, an evaluation framework GuacaMol⁹⁹ was proposed in 2019, which set a suite of tasks for distribution learning and

goal-directed design. More specifically, the generative model is examined on whether it can reproduce the property distribution of training sets (usually for VS purpose¹⁰⁰), and find the optimal molecule (for multi-objective optimization). A more recent evaluation platform MOSES was released by Polykovskiy et al in 2020,⁶⁴ which compiles a list of metrics (Table 3²⁾ for detecting common issues in generative models, such as overfitting and mode collapse.

Table 3: Commonly Used Evaluation Metrics

Application	Task	Metric	Purpose
Virtual screening	Molecular property prediction	Recall@k	Retrieval
Virtual screening	Molecular property prediction	Precision@k	Retrieval
Virtual screening	Molecular property prediction	AP@k	Retrieval
QSAR	Molecular property prediction	Accuracy	Classification
QSAR	Molecular property prediction	Recall	Classification
QSAR	Molecular property prediction	Precision	Classification
QSAR	Molecular property prediction	AUROC	Classification
QSAR	Molecular property prediction	AUPRC	Classification
QSAR	Molecular property prediction	MAE	Regression
QSAR	Molecular property prediction	RMSE	Regression
Drug design	Molecule generation	Validity	Distribution learning
Drug design	Molecule generation	Unique@k	Distribution learning
Drug design	Molecule generation	Novelty	Distribution learning
Drug design	Molecule generation	Diversity	Distribution learning
Drug design	Molecule generation	FCD	Distribution learning
Drug design	Molecule generation	KL divergence	Distribution learning
Drug design	Molecule generation	Scaffold similarity	Goal-directed design
Drug design	Molecule generation	Rediscovery	Goal-directed design

For instance, validity measures how well a model explicitly captures the chemical rules, such as valency; uniqueness and diversity examine whether the generative model collapses to producing only a limited set of molecules; novelty indicates whether the model is overfitted to just memorize the training examples. Furthermore, Fréchet ChemNet Distance (FCD) is a measure of how close the distributions of the generated set are to the distribution of molecules in the training set. A low FCD value corresponds to similar molecule distributions. Kullback-

²QSAR: quantitative structure-activity relationship; Recall@k: recall among top k molecules; Precision@k: precision among top k molecules; AP@k: average precision among top k molecules; AUROC: area under the receiver-operating characteristic curve; AUPRC: area under the precision-recall curve; MAE: mean absolute error; RMSE: rooted mean square error; Unique@k: uniqueness of the first k valid (generated) molecules; FCD: Fréchet ChemNet Distance; KL divergence: Kullback-Leibler divergence.

Leibler (KL) divergence measures the difference between two probability distributions. When the KL divergence value is small, the generated molecules can well approximate the targeted property in the training set. For goal-directed design, it relies on a formalism where molecules are scored individually based on each pre-defined criterion, such as containing a specific substructure, having certain physicochemical properties or exhibiting similarity or dissimilarity to certain molecules. Consequently, similarity and rediscovery are usually used for evaluation purpose. Specifically, rediscovery assesses if the generative model is able to rediscover a given molecule and similarity evaluates whether the model can generate molecules similar or dissimilar to a given molecule.

Model Architectures

Prior to the “deep learning” era, traditional machine learning models were widely used in VS.⁴⁰ Pertinent tasks include predictions of drug likeliness,^{101,102} physicochemical properties,^{103,104} pharmacokinetic parameters^{105–108} and pharmacodynamic properties.^{109,110} There are score-based classification models, support vector machines (SVM)¹¹¹ and K nearest neighbors (KNN) and probability-based classification models, random forest (RF),¹¹² naive bayes (NB), and logistic regression (LR). Despite the success of traditional machine learning models, deep neural networks (DNNs) have outperformed them in a variety of tasks.^{62,113}

Convolutional Neural Networks

Convolutional neural networks (CNNs) are mainly used in computer vision to process pixels of data in images.¹¹⁴ In CNNs, there are convolution layers and pooling (i.e. subsampling) layers (Fig 3). On top of these convolution layers and pooling layers, a vector representation is learned by concatenating the feature maps for a final prediction. CNNs share parameters across the filters, which largely reduces the number of parameters to be learned, thereby decreasing memory consumption and increasing computation speed.

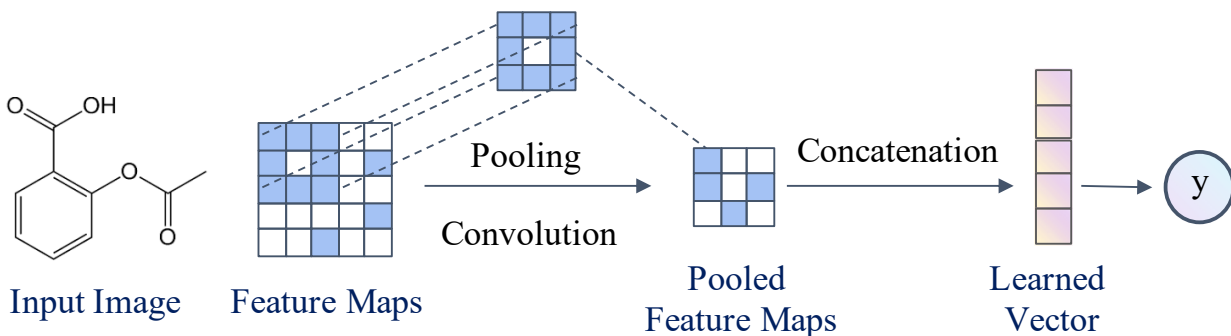


Figure 3: Illustration of Convolutional Neural Networks.

In drug discovery, CNNs can be applied to elucidate the bioactivity profiles based on microscopy images.^{115,116} Moreover, CNNs are also used for molecular property prediction.^{117,118} In 2015, Duvenaud et al¹¹⁸ applied CNNs on circular fingerprints, a refinement of the ECFPs,¹¹⁹ to create a differentiable fingerprint, which is among the first efforts using data-driven representation learning for molecular property prediction, instead of fixed chemical descriptors. This work has greatly motivated learning molecular representations.

In addition to fingerprints, CNNs can also effectively extract features directly from the images of molecular structure. For instance, Chemception¹²⁰ is trained on the 2D-structural images to predict free energy of solvation and inhibition of HIV replication. Later, Fernández et al⁸⁸ developed Toxic Colors, a framework for toxicity classification with the images as input. Cortes-Ciriano et al⁹⁰ further extended existing CNNs architectures (e.g. AlexNet,⁴¹ DenseNet-201,¹²¹ ResNet152¹²² and VGG-19¹²³) to Kekulé structure images for molecular property prediction, also known as KekuleScope. The experimental results of KekuleScope showed that CNNs on images as input can achieve comparable performance to RF and DNNs on ECFPs. Meyer et al⁸⁹ also predicted the MeSH-therapeutic-use classes based on compound images, which outperformed previous predictions based on transcriptomic data. More recently, Rifaioğlu et al⁹¹ proposed a large-scale DTI prediction system, DEEPScreen. Indeed, molecular property prediction with images as input are closely related to the progress in computer vision, which also prompts automatic extraction of chemical structures from literature and patents.^{92,124} The chemical structure recognition model can be further inte-

grated with models in natural language processing, such as DECIMER¹²⁵ and DECIMER 1.0,⁹² which are able to translate the bitmap images of a molecule into a SMILES string, as an image captioning task.¹²⁶

Recurrent Neural Networks

Recurrent neural networks (RNNs) are mainly used for processing sequential data.¹¹⁴ RNNs allow the connection among neurons in the same hidden layer to form a directed cycle (Fig 4A), thereby enabling the use of sequential input, such as language modeling¹²⁷ and music generation.¹²⁸ If unfolded in time steps, RNNs can be seen as a very deep feed-forward networks where all layers sharing the same weights. However, the long-term dependency of RNNs makes it difficult to learn the parameters due to the gradient explosion or vanishing problem.¹¹⁴ As a result, long short-term memory (LSTM)¹²⁹ and gated recurrent unit (GRU),¹³⁰ two variants of the vanilla RNN, have been developed to augment the network with a memory module. Different from CNNs’ operation on images, RNNs mainly take the SMILES strings as input for molecular property prediction and molecule generation. As discussed in Section “Small Molecule Representation”, the characters in a SMILES string are firstly converted into one-hot vectors (Fig 2) and then sequentially fed into RNNs, with a hidden vector to be updated at each step. For molecular property prediction, RNNs generate a final output after all steps are taken. For example, SMILES2Vec¹³¹ uses RNNs to learn features from SMILES and predicts a wide range of chemical properties. Mayr et al⁶² proposed SmilesLSTM to perform DTI prediction, which outperformed traditional machine learning models.

RNNs can also be applied for molecule generation, similar to language models for text generation.^{97,100,132} More specifically, RNNs generate output at each step in an auto-regressive manner (Fig 4B), where the output is dependent on the input from previous steps. Based on the input from the current step and prior steps, RNNs output a probability distribution over all possible tokens, from which a token is sampled as the output of the current step and will

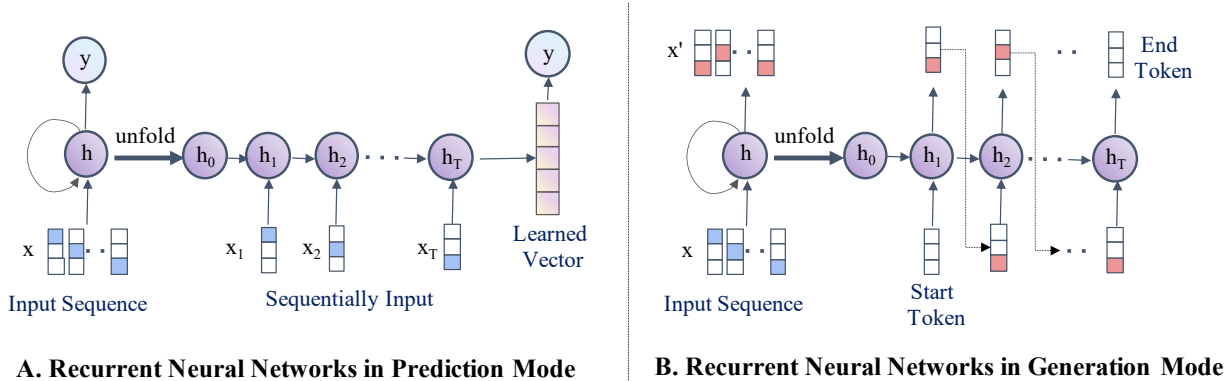


Figure 4: Illustration of Recurrent Neural Networks.

be used to predict the next token. However, due to the syntax of the “SMILES language” such as the ring opening-closure and the matched brackets, regular RNNs, including LSTM and GRU, cannot capture the algorithmic patterns of the sequence well owing to their inability to count.⁷⁹ As a result, the generated SMILES strings very often violate the chemical rules and become invalid. Thus, a memory-augmented version, Stack-RNN,^{79,133} was developed to alleviate the validity issue for SMILES-based molecule generation. Another solution for this problem is to adopt bidirectional RNNs, such as the bidirectional LSTM.^{134,135}

In addition to the SMILES strings, RNNs can also be applied on molecular graphs for generation purpose.^{136–139} For example, You et al¹³⁶ proposed GraphRNN to generate molecular graphs in an autoregressive manner, decomposing it as a process into generating a sequence of node and edge formations conditioned on the graph structure generated so far. Nonetheless, generating molecular graphs with RNNs requires a full trajectory of the graph generation, which tends to forget the states of initial generation steps quickly. Later You et al proposed GCPN and¹⁴⁰ designed the graph generation procedure as a Markov Decision Process (MDP), which only needs the intermediate state to generate the graph. Notably, RNNs can also be components of more complicated generative models, such as variational autoencoders^{78,141} and generative adversarial networks.^{142,143}

Graph Neural Networks

CNNs and RNNs are usually applied on data represented in the Euclidean space. In recent years, graph neural networks (GNNs) are gaining popularity to model data represented in graphs with a set of nodes and edges.¹⁴⁴ GNNs can handle node-level (e.g., node classification), edge-level (e.g., link prediction) and graph-level (e.g., graph regression) tasks, with neighborhood aggregation, pooling and readout operations. Small molecules, when represented as molecular graphs (Fig 2C), are naturally appealing to the application of GNNs for both molecular property prediction and molecule generation tasks (see Section “Molecular Graphs”).

Two major types of GNNs are convolutional GNNs (ConvGNNs) and recurrent GNNs.¹⁴⁴ In the recurrent GNNs, node representation is learned via some recurrent neural architectures, such as the graph gated neural network (GGNN).¹⁴⁵ On the other hand, ConvGNNs generalize the convolution operation from grid data to graph data and can stack multiple graph convolutional layers to extract high-level node representations. ConvGNNs play a central role in building up many other complex GNNs, which can be further categorized into two subtypes: 1) Spectral-based: ChebNet,¹⁴⁶ graph convolutional network (GraphConv)¹⁴⁷ and 2) Spatial-based: message passing neural networks (MPNN),¹⁴⁸ GraphSAGE,¹⁴⁹ graph attention network (GAT),¹⁵⁰ graph isomorphism network (GIN).¹⁵¹

In drug discovery, GNNs are often exploited for molecular property prediction (Fig 5A). For instance, Kearnes et al¹⁵² developed Weave to perform graph convolutions on molecular graphs for representation learning, where the graph convolutions, nonetheless, did not outperform the fingerprint-based models back in 2016. Later, Gilmer et al¹⁴⁸ proposed MPNN as a unified framework for quantum chemical properties prediction. MPNN has two phases in the forward pass, namely, message passing and readout. During message passing, for each atom, feature vectors from its neighbors are propagated into a message vector wherein the hidden vector for the atom is updated by the message vector. A readout function is used to aggregate the feature vectors into a graph feature vector, which is then passed to a fully con-

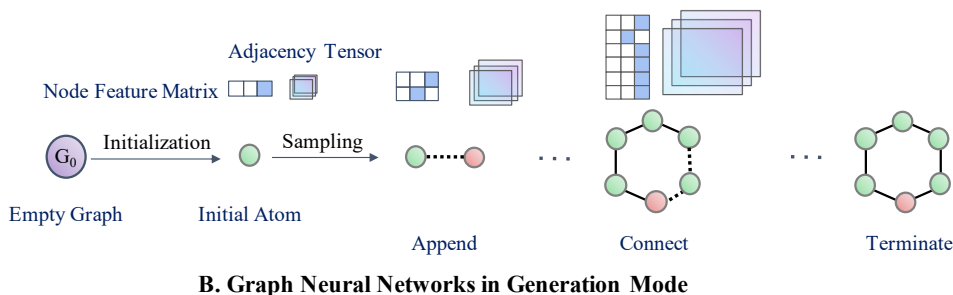
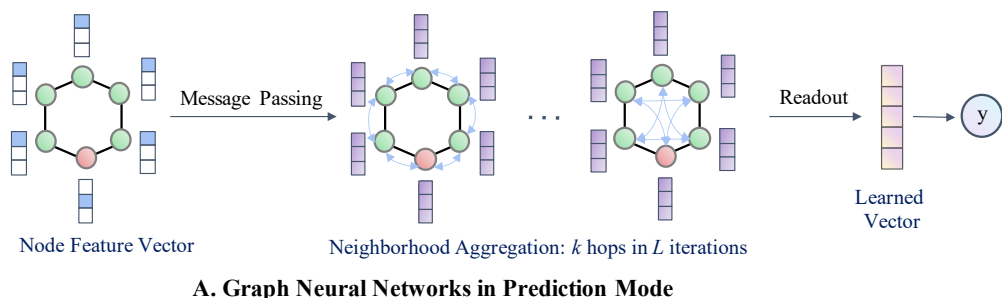


Figure 5: Illustrations of Graph Neural Networks.

nected layer for downstream predictions. Yang et al⁷¹ then expanded MPNN into directed MPNN (D-MPNN), which uses messages associated with directed edges (bonds) instead of nodes (atoms) used in MPNN, thereby preventing repeated message passing from the same node. Notably, Yang et al⁷¹ also introduced a practice to concatenate the 200 global molecular features calculated by RDKit¹⁵³ with the learned features by D-MPNN for downstream predictions, also adopted in later works.⁶³ Xiong et al⁷⁷ integrated graph attention mechanism into GNNs and developed Attentive FP, which is able capture topologically adjacent atoms’ interactions for improved molecular property prediction. More recently, Withnall et al¹⁵⁴ also made augmentations to the MPNN and proposed attention MPNN (AMPNN) and edge memory neural network (EMNN) for physicochemical property prediction. So far, GNNs have been widely applied for molecular property prediction. More examples include as SchNet,¹⁵⁵ PotentialNet,¹⁵⁶ and DimeNet,¹⁵⁷ among others.^{62,158–165} Moreover, subgraphs can directly map to molecular substructures, which also improves interpretability.^{77,166–168}

Partly encouraged by the superior performance of GNNs for molecular property prediction, GNNs are also exploited for molecule generation (Fig 5B). As mentioned above, RNNs

can be used to generate molecular graphs, which, nevertheless, needs to store a full trajectory for the graph generation process and tends to forget initial states.¹⁴⁰ In 2018, Li et al¹³⁸ developed a conditional graph generator, MolMP, which does not involve atom-level recurrent units. MolMP models the graph generation as a MDP problem, where the action to grow graph only depends on its current state. There are three actions in total - append, connect and terminate, the sampling process of which is parameterized by a neural network. Experimental results show that MolMP outperforms SMILES-based molecule generation in a variety of evaluation metrics, especially the validity. GNNs-based molecule generation can be used in common drug design applications such as designing molecules with certain scaffolds, presumably due to more straightforward mapping to chemical substructure with the graph representation. Furthermore, since molecule generation is usually driven by certain desired properties, reinforcement learning (see Section “Learning Paradigms”), therefore, is often integrated with GNNs for goal-directed drug design. Examples include GCPN,¹⁴⁰ MolDQN,¹⁶⁹ DeepGraphMolGen,¹⁷⁰ and MNCE-RL.¹⁷¹ For more practical issues on GNNs for molecule generation, such as generation schemes (single-shot vs iterative) and computation, we refer the readers to the guide by Mercado et al.⁵¹

Variational Autoencoders

Variational autoencoders (VAEs), a class of powerful probabilistic generative models, were first introduced by Kingma et al¹⁷² in 2013. VAEs, consisting of an encoder E and a decoder D . The encoder maps high-dimensional data into a low-dimensional, continuous latent space (Fig 6). Compared to common autoencoders, the latent space is regularized to be organized, ideally, through the KL divergence. In addition to reconstruction, VAEs approximate a probability distribution, which can be sampled for generation purpose. Thus, given input x , the parameters of VAEs are optimized by minimizing the reconstruction loss and the KL divergence:¹⁷³

$$\|x - D(E(x))\|^2 + KL(N(\mu_x, \sigma_x), N(0, 1)), \quad (1)$$

which is equivalent to maximize the evidence lower bound (ELBO). In Equation 1, $N(0, 1)$ denotes the unit normal distribution; μ_x and σ_x are learnable parameters, representing mean and variance of a Gaussian distribution.

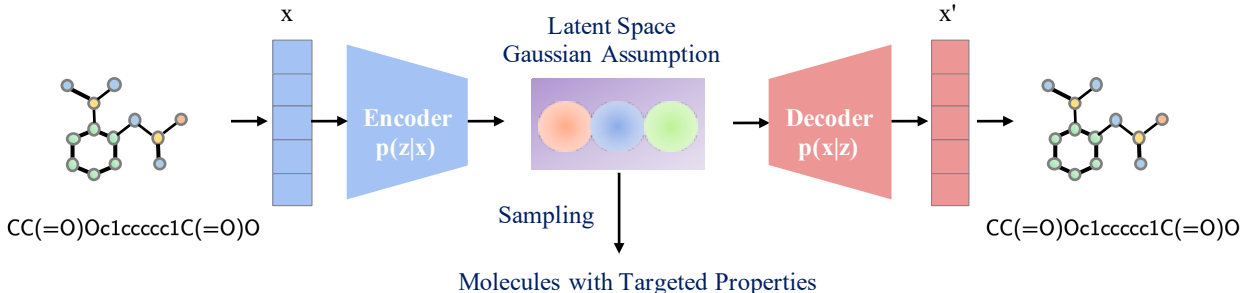


Figure 6: Illustration of Variational Autoencoders.

VAEs can be applied on SMILES strings for molecule generation. For example, Gómez-Bombarelli et al⁷⁸ developed a VAE model for automatic molecule design, where a pair of deep neural networks (i.e. an encoder and a decoder) is trained as an autoencoder to convert the input SMILES strings into a continuous vector representation. To train the autoencoder, a reconstruction loss is adopted in attempt to reproduce the original SMILES string. However, the ultimate goal is not to merely reconstruct the input. Rather, the autoencoder aims to learn a compact representation for the molecules. Thus, a constraint is applied in the autoencoder by jointly training a physical property regression model to organize the VAE’s latent space subjected to the property value, which can be used to sample molecules towards the desired property value. Partly due to the syntax of SMILES, the latent space learned by the autoencoder can be sparse and contain large “dead areas”, which correspond to invalid molecules. Therefore, VAEs with a focus on the syntax for valid molecule generation are proposed later, such as GrammarVAE¹⁷⁴ and syntax-directed VAE.¹⁷⁵ Other related works also include semi-supervised VAE (SSVAE) for continuous output,¹⁷⁶ conditional VAE (CVAE),¹⁷⁷ constrained graph VAE (CGVAE),¹⁷⁸ NeVAE,¹⁷⁹ GTM VAE¹⁴¹ and CogMol.¹⁸⁰ A variant of the VAE is the adversarial autoencoder (AAE),¹⁸¹ which replaces the KL divergence with an adversarial objective. More specifically, the Gaussian distribution assumption

as a prior on the latent representations for KL-divergence computation is replaced by other priors, i.e., an additional discriminator is added to force the encoder generates latent representations in a specific distribution (e.g. a uniform distribution). AAEs can also be used for molecule generation,^{182–184} which can improve reconstruction and the validity of generated molecules.

VAEs can also be applied on the graph representations for generation purpose. In 2018, Simonovsky et al¹⁸⁵ proposed a VAE framework (GraphVAE) to generate molecular graphs. Their main idea is to output a probabilistic fully-connected graph and use a standard graph matching algorithm to align it to the ground truth. Jin et al¹⁸⁶ developed the junction tree VAE (JT-VAE). In JT-VAE, a molecular graph is first mapped into a junction tree via a tree decomposition algorithm and the junction tree then undergoes the VAE’s encoding-decoding process. The learned latent space of the junction tree can be used to search for substructures, which then assemble into molecules with specific properties. A prominent merit of JT-VAE is that the validity of all generated molecules can be guaranteed. Ma et al¹⁸⁷ also proposed a regularization framework for VAEs (Regularized VAE) that regularize the output distribution of the decoder, thereby improving the validity. Later, Kajino et al¹⁸⁸ developed molecular hypergraph grammar VAE (MHG-VAE), where a molecular graph is described as a hypergraph and the grammar VAE¹⁷⁴ is trained by inputting the grammar for sequence production of the hypergraph. In 2019, Kwon et al¹⁸⁹ developed a non-autoregressive graph VAE and incorporated three additional learning objectives into the model, namely, approximate graph matching, reinforcement learning, and auxiliary property prediction, which is able to generate valid and diverse molecular graphs with various constraints. Moreover, graph-based VAEs can also embrace a strategy for drug design with the ability to retain a particular scaffold (i.e., substructure), such as the ScaffoldVAE.¹⁹⁰

One drawback of VAEs, nonetheless, is that the set of substructures by partitioning molecules can be quite large. Consequently, the iterative prediction of which substructure to add can be inaccurate, especially for infrequent substructures. To address this challenge, Fu

et al¹⁹¹ proposed a novel strategy, CORE, by combining scaffolding tree generation and adversarial training. Besides, the computational cost increasing with the number of nodes in a graph is another major challenge here, limiting the application to larger molecules. In 2020, Kwon et al¹⁹² proposed a compressed graph representation to alleviate computational complexity while maintaining the validity and diversity of generated molecules. More recently, Jin et al¹⁹³ developed hierarchical graph VAE (HierVAE) which can employ larger and more flexible graph motifs as building blocks for molecules. More specifically, the encoder produces a multi-resolution representation for each molecule in a fine-to-coarse fashion, from atoms to connected motifs while the autoregressive coarse-to-fine decoder adds one motif at a time. Notably, HierVAE can even be used to generate polymers.

Generative Adversarial Networks

Generative Adversarial Networks (GANs), developed by Goodfellow et al¹⁹⁴ in 2014, have made remarkable achievements in generating realistic synthetic samples. GANs consist of a generative model G , and a discriminative model D (Fig 7). The generator aims to generate new data points from a random distribution whereas the discriminator aims to classify whether the generated samples are from the training data distribution or from the generator. GANs can be trained by the min-max loss, which alternatively optimizes the generator and the discriminator using a min-max objective:

$$\min_G \max_D \mathcal{L}(G, D) = \mathbb{E}_{x \sim p_x} [\log(D(x))] + \mathbb{E}_{z \sim p_z} [\log(1 - D(G(z)))], \quad (2)$$

where p_x and p_z denote the distribution of the real data x and the noise prior z .

GANs can be applied to SMILES strings for molecule generation. In 2017, Guimaraes et al¹⁴² developed objective-reinforced GANs (ORGAN), built upon SeqGAN,¹⁹⁵ to generate molecules in SMILES strings while also optimizing several domain-specific metrics. The generator is based on LSTM, which is modeled as a stochastic policy in a reinforcement learning

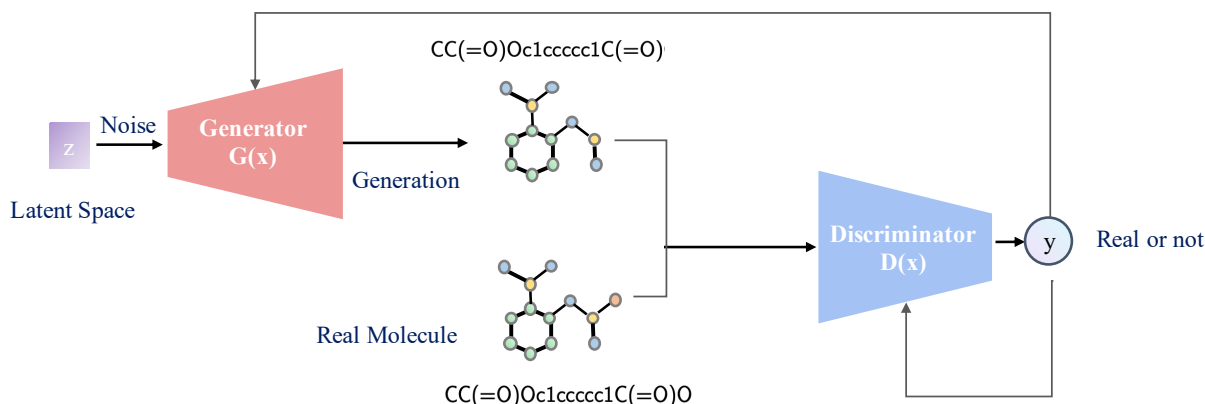


Figure 7: Illustration of Generative Adversarial Networks.

setting (more details in Section “Learning Paradigms”), whereas the Wasserstein loss is used to train the discriminator (a CNN model). Experimental results showed that the generated molecules exhibit drug-like structures as well as improvement in the evaluation metrics. Shortly after, an objective-reinforced GANs for inverse-design chemistry (ORGANIC)¹⁴³ was developed based upon ORGAN. ORGANIC can generate molecules with biased distribution towards certain attributes for both drug discovery and material design. In 2018, Putin et al¹⁹⁶ presented a reinforced adversarial neural computer (RANC) framework, which also combines GANs and RL. The generator of the RANC framework is a differentiable neural computer (DNC) with an explicit memory bank, instead of the LSTM model in ORGANIC and ORGAN. This is because the generation of discrete data using RNNs, particularly, LSTM with maximum likelihood estimation, can suffer from the so-called “exposure bias”, i.e., missing salient features of the data. RANC outperforms ORGANIC, as measured by several metrics: number of unique structures, passing medicinal chemistry filters (MCFs), Muegge criteria and high QED scores. RANC is able to generate molecules that match the distributions of the key chemical features/descriptors (e.g., MW, logP) and lengths of the SMILES strings from the training set.

GANs can also be applied on molecular graphs for molecule generation. In 2018, De Cao et al¹⁹⁷ developed MolGAN, an implicit, likelihood-free generative model for small molecular graph generation, which circumvents the expensive graph matching procedures.¹⁸⁵ Moreover,

they adapted the GANs to enable direct operation on molecular graphs. RL is also integrated to encourage the generation of molecules towards desired properties. Experimental results on the QM9 dataset showed that MolGAN is able to generate nearly 100% valid molecules, which outperforms ORGAN in validity. One drawback of MolGAN is its susceptibility to mode collapse, i.e., repeated samples being generated multiple times, leading to low uniqueness.¹⁹⁸

Normalizing Flow Models

In addition to the RNNs, VAEs and GANs, another major class of generative models is the normalizing flow models.¹⁹⁹ Representative works include the Non-linear Independent Component Estimation model (NICE),²⁰⁰ the Real-valued Non-Volume Preserving model (RealNVP),²⁰¹ and the Glow model, among others. In NICE, Dinh et al²⁰⁰ introduced tractable calculation for reversible transformations, which are also known as the affine coupling layers underlying the flow models. The basic idea of flow models is to learn an invertible mapping between complex distributions and simple prior distributions (Fig 8). By exploiting exact and tractable likelihood estimation for training, flow models enable efficient one-shot inference and 100% reconstruction of the training data.⁵¹

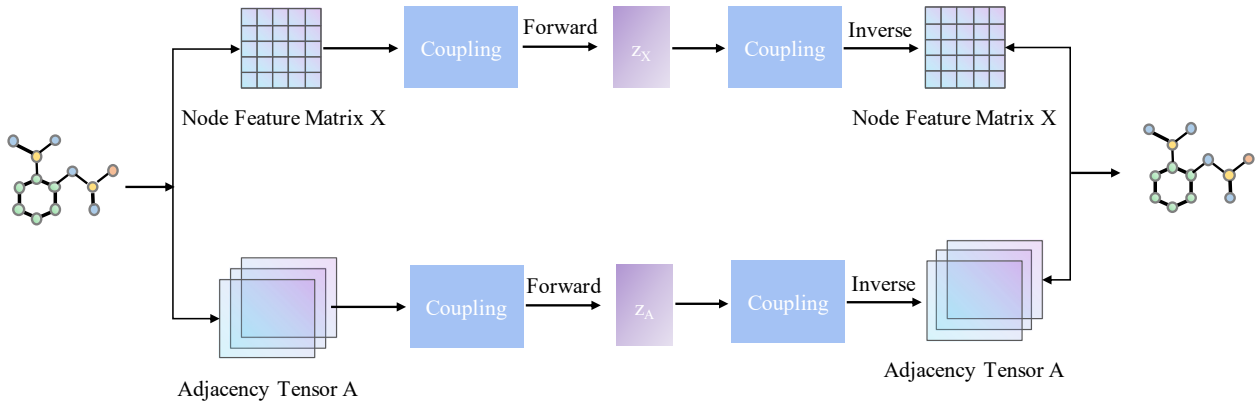


Figure 8: Illustrations of Flow Models.

In drug discovery, flow models have been applied to generate molecules, mainly on molecular graphs. In 2019, Madhawa et al²⁰² developed GraphNVP, the first flow model for molecular graph generation. In GraphNVP, the graph generation is decomposed into two steps,

i.e., generation of an adjacency tensor and generation of node attributes, which yields the exact likelihood maximization on the graph with two reversible flows. GraphNVP is able to generate valid molecules with minimal duplicates. The learned latent space can be further exploited to generate molecules with desired properties. Honda et al²⁰³ also developed an invertible flow model for molecular graph generation based on residual flows, also known as graph residual flow (GRF), which enables more flexible and complex non-linear mappings than the traditional coupling flows. Experimental results showed that GRF can achieve comparable performance with GraphNVP, while having much less parameters to learn. Notably, GraphNVP²⁰² and GRF²⁰³ generate molecular graphs in a single-shot manner,⁵¹ which may lead to low validity, nevertheless. Consequently, a sequential iterative graph generation manner is proposed for flow models. For example, GraphAF,²⁰⁴ an autoregressive flow-based model, adopts an iterative sampling process to leverage chemical domain knowledge, such as valency checking in each step. With the integration of chemical rules, GraphAF is able to generate molecules of 100% validity. Moreover, its training process is significantly faster than GCPN.¹⁴⁰ GraphAF can be further finetuned with RL, which achieves better performance on molecular property optimization compared to JT-VAE¹⁸⁶ and GCPN.¹⁴⁰ MoFlow, later developed by Zang et al,²⁰⁵ applies a validity correction to the generated graph, which not only enables efficient molecular graph generation in a single-shot manner, but also guarantees the chemical validity. The continuous latent space learned via encoding the molecular graphs can be further used to generate novel and optimized molecules during the decoding process towards desired properties. More recently, Luo et al²⁰⁶ developed GraphDF, which, on the contrary, aims to learn a discrete latent representation with the flow models and capture the original discrete distribution of the discrete graph structures without adding real-valued noise. For molecule generation, GraphDF sequentially samples the discrete latent variables and maps them to new nodes and edges via invertible transforms. The discrete transforms can circumvent the cost of computation while also achieving state-of-the-art performance in random molecule generation, property optimization and constrained optimization tasks.

For the flow-based generative models, the most prominent feature is that they are able to exactly reconstruct all the input data without duplicates due to the precise likelihood maximization, which can be an important complement for molecule generation. In particular, when the molecular property is highly sensitive to minor structural changes, i.e., activity cliffs,^{36,37} a replacement of a specific atom (node) might be needed. In other words, flow models can offer more precise modifications on existing molecular structures.

Transformers

RNNs have been widely applied to handle sequential input. However, RNNs can suffer from the gradients explosion or the vanishing problem.¹¹⁴ In 2017, a seminal work “Attention is all you need” proposed a novel transformer architecture, built with the self-attention mechanism.²⁰⁷ Transformers have now become the *de facto* standard in powerful language models, such as GPT,²⁰⁸ BERT,²⁰⁹ GPT-2,²¹⁰ RoBERTa,²¹¹ and GPT-3,²¹² and even in advanced computer vision models, such as DETR²¹³ and Vision Transformer.²¹⁴ Unlike RNNs, transformers forfeit recurrent connections. By adopting positional embedding, transformers are even better at dealing with long sequences.²⁰⁷

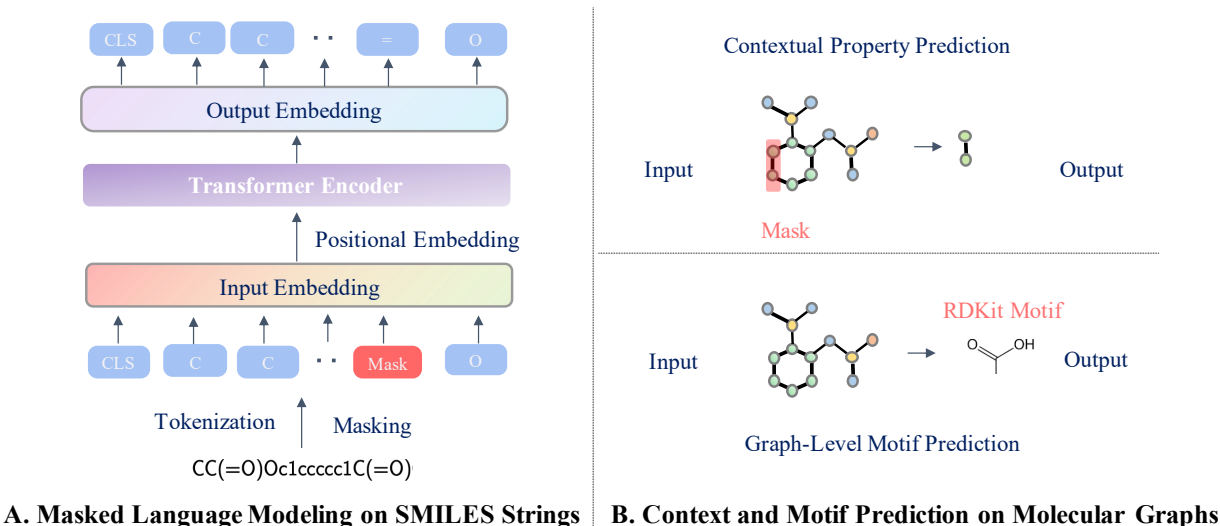


Figure 9: Illustrations of Self-Supervised Learning with Transformers.

Not unexpectedly, transformers are being actively applied in drug discovery. Notably,

transformers enable effective self-supervised pretraining, such as masked language modeling (Fig 9A). In 2019, Wang et al²¹⁵ developed SMILES-BERT, which consists of several transformer encoder layers, to improve molecular property prediction. SMILES-BERT is first pretrained on a large-scale corpus of SMILES strings via a SMILES recovery task and then fine-tuned on the downstream prediction tasks. Later, Honda et al²¹⁶ proposed to learn molecular representations through pretraining a sequence-to-sequence language model, which is termed as the SMILES Transformer. Chithrananda et al⁵⁹ also developed ChemBERTa, built upon the RoBERTa model²¹¹ for molecular property prediction. More recently, Fabian et al⁹⁵ applied the architecture of BERT²⁰⁹ to learn molecular representations, also referred to as MolBERT. When pretrained with masked language modeling and other tasks, MolBERT achieves improved performance for molecular property prediction compared to the fixed fingerprints. Moreover, transformers can also be applied on molecular graphs, especially considering that the transformer encoder can be viewed as a GAT variant.¹⁵⁰ For example, Rong et al⁶³ developed a novel framework, GROVER, to learn graph representations with the message passing transformer. By designing self-supervised contextual property prediction and graph-level motif prediction tasks (Fig 9B), GROVER is pretrained on 10 million unlabeled molecules and achieves state-of-the-art performance on 11 benchmark datasets.

In addition to molecular property prediction, transformers can also be exploited for molecule generation, such as MoleculeChef,²¹⁷ which can generate the reactants for a given product, similar to machine translation. More recently, transformers are also exploited for protein-specific molecule generation,²¹⁸ where the input is the amino acid sequence of the target protein and the output are ligands in the SMILES representation.

Learning Paradigms

Drug discovery, despite the light shed by AI, still faces major challenges. For molecular property prediction, labeled data points are at the core of machine learning models. Nonetheless,

in real-world settings, generating labeled data points in wet lab can be very expensive. Consequently, the datasets for model training are usually limited in size, exhibit high sparsity, and can be heavily biased and noisy, which is also termed as the low-data drug discovery problem.^{158,165} For molecule generation, although existing generative models, such as VAEs, can be used to generate molecules towards desired properties, the mechanism by mapping from the points in the latent space to real molecules which are most proximal can limit the exploration of the chemical space, leading to low novelty and diversity.⁷⁸ To address these challenges, various learning paradigms have been proposed. In this survey, we mainly focus on self-supervised learning and reinforcement learning to address molecular property prediction and molecule generation, respectively. Other learning paradigms are also discussed.

Self-Supervised Learning

The performance of deep neural networks, especially supervised learning, hinges on a large labeled dataset. Nevertheless, supervised learning is meeting its bottleneck due to its heavy reliance on expensive manually-labeled data.²¹⁹ In real-world problems such as molecular property prediction, the labeled data is often limited, sparse and biased, which leads to low generalizability of models. Self-supervised learning is promising paradigm and has achieved state-of-the-art performance in learning with limited labels, as adopted in the aforementioned language models, for instance, BERT.²⁰⁹ Notably, self-supervised learning should be distinguished from unsupervised learning. Unsupervised learning focuses on detecting patterns in data without labels, such as clustering, whereas self-supervised learning aims to recover the data. More specifically, it can be classified into two main types, i.e., generative and contrastive self-supervised learning.

For the generative self-supervised learning, a canonical task is the masked language modeling, as proposed in BERT,²⁰⁹ where the model is trained to predict the masked tokens, thereby recovering the original input. Model parameterization is usually implemented by optimizing the cross-entropy loss between the output and the masked tokens in the in-

put. Representative works of self-supervised learning in drug discovery are discussed in Section “Transformers” (e.g., MolBERT⁹⁵ and GROVER⁶³). Notably, self-supervised pretraining can avoid the negative transfer caused by supervised pretraining - transfer of knowledge from pretraining harms model generalization, as shown by Hu et al.¹⁶³ Besides, contrastive learning is another type of self-supervised learning. More specifically, contrastive learning aims to learn latent representations through contrasting data pairs (positive vs negative), where the positive and negative examples are constructed by augmenting the unlabeled samples in a self-supervised manner. Recently, contrastive learning has been employed to address the low-data drug discovery problem. For instance, Wang et al.²²⁰ proposed molecular contrastive learning of representations (MolCLR) on molecular graphs for molecular property prediction. Three molecular graph augmentation ways are used, i.e., atom masking, bond deletion, and subgraph removal. Through a contrastive loss, MolCLR learns molecular representations by contrasting positive vs negative molecules, where molecular graph pairs augmented from the same molecule are treated as the positive and the others denoted as negative. Experimental results show that MolCLR can effectively transfer the learned representations to downstream tasks and achieve state-of-the-art performance in molecular property prediction.

In addition to self-supervised learning, other learning paradigms have also been exploited to address the low-data drug discovery challenge. For example, *meta learning*²²¹ aims to learn a learner to be adapted to new tasks. In a study by Nguyen et al.,¹⁶⁵ the meta-learning initializations outperform multi-task pretraining baselines on 16 out of 20 in-distribution tasks and all out-of-distribution tasks. A member of the meta-learning family is *few-shot learning*,²²² the core idea of which is to generalize with a few examples. For instance, Altae Tran et al.¹⁵⁸ proposed a one-shot learning framework for activity classification, which lowers the amount of data required for predictions. Intuitively, through learning a distance metric, molecules can be embedded into the latent space in a more organized way. Thus, when new molecules come, their embeddings can be compared to the existing labeled molecules for

more accurate prediction. Closely to this idea, another paradigm is *metric learning*,^{223,224} which mainly deals with data with mixed distribution due to the activity cliffs.³⁸ Metric learning has been widely applied in computer vision,^{224,225} especially in situations where the similarity or distance must be computed for clustering or nearest neighbor classification purpose. At its core, a distance metric (e.g., cosine distance) is to be learned, based on which the learned latent representations of the input data can be separated according to their labels. In a recent work by Na et al,²²⁶ a generalized deep metric learning (GeDML) framework is proposed, which alleviates the structure-property mismatch problem through better separating molecules in the latent space. The representations learned via metric learning are also conducive for goal-directed molecule generation, i.e., search in the chemical space. For example, Koge et al²²⁷ proposed a molecular embedding framework by combining VAEs and metric learning. The idea is to make the molecules’ embedding in the latent space consistent with their properties, thereby enabling efficient search during molecule generation.

Reinforcement Learning

With improved performance on molecular property prediction, another challenge still poses for molecule generation, i.e., how to design molecules with the desired properties? As mentioned in Section “Model Architectures”, VAEs and flow models can be used to generate molecules with preferred properties by sampling from a learned latent space. However, the latent space can be highly dimensional and the objective functions defined in the latent space is usually non-convex, making it difficult to optimize the properties of generated molecules.¹⁶⁹ Consequently, reinforcement learning (RL) is often used as the alternative to navigate the chemical space, which mainly deals with how an agent should take actions in a certain state so as to maximize a reward or return.²²⁸ RL algorithms can be classified into 1) value-based (e.g., Q-learning), 2) policy-based (e.g., policy gradient) and 3) hybrid (e.g., actor-critic).²²⁹

In drug discovery, the DMTA cycle (see Section “Drug Discovery Overview”) underlying drug design,³ i.e., goal-directed molecule generation, can be potentially automated

by RL through connecting the generative model (i.e., the agent for molecule generation) and the predictive model (i.e., assigning rewards based on the predicted property values). For example, Zhou et al¹⁶⁹ adopted value-based, double Q-learning (DQN)²³⁰ to optimize the generated molecules. Other related works mainly adopt the policy-gradient algorithm, REINFORCE,²³¹ as an estimator of the gradient, such as ORGAN,¹⁴² REINVENT,⁹⁷ and ORGANIC.¹⁴³ Another policy-gradient algorithm, proximal policy optimization (PPO),²³² is also gaining popularity recently, which is improved from the trust region policy optimisation (TRPO).²³³ TRPO employs a trust region so that optimizations are restricted to a region where the approximation of the true cost function holds, thereby preventing policies updated too wildly and lowering the chance of a catastrophically “bad” update.²²⁹ However, TRPO requires the calculation of second-order gradients, being computationally expensive. PPO, on the contrary, only requires first-order gradients and can retain the performance of TRPO, exhibiting low sample complexity. Studies adopting PPO for *de novo* drug design include the work by Neil et al,¹³² GCPN,¹⁴⁰ DeepGraphMolGen¹⁷⁰ and MNCE-RL.¹⁷¹

Nevertheless, policy-gradient algorithms usually exhibit high variance since the gradient estimation can be noisy.²²⁹ To reduce the variance, an improved class of algorithms is the hybrid actor-critic method, which combines policy-gradient methods with learned value functions. For example, off-policy deterministic policy gradient (DPG) extends the standard policy gradients for stochastic policies to deterministic policies, which only integrates over the state space instead of both state and action spaces, thus requiring fewer samples in problems with large action spaces. Later, deep deterministic policy gradient (DDPG) utilizes neural networks on high-dimensional space is introduced, as adopted in MolGAN.¹⁹⁷ Another technique for variance minimization to accelerate convergence is to subtract the estimated reward from the true reward, which separates the policy training from value estimation, also known as the advantage actor-critic (A2C) algorithm adopted by Neil et al.¹³²

With RL training, chemical libraries shifted towards desired properties are expected to be generated. However, drug design is a multi-objective optimization problem.^{9,234} In order

to prioritize molecules based on the pre-defined goals, non-dominated sorting or pair-based comparisons are usually exploited to find solutions with the Pareto optimality.²³⁵⁻²³⁷ Another issue with RL is the trade-off between exploration and exploitation. As illustrated by Zhou et al,¹⁶⁹ this trade-off is a dilemma underlying by uncertainty. Due to the lack of a complete knowledge of the rewards for all the states, if constantly choosing the best action known to produce the highest reward (exploitation), the model will never learn anything about the rewards of the other states; on the other hand, if always choosing a random action (exploration), the model will not receive sufficient reward. A potential solution for the exploration-exploitation trade-off is *active learning*, which is a paradigm where the model can query an expert or any other information sources in an active manner during learning.^{31,238}

Discussions

There has been a surge of AI in drug discovery over the past decade, which is still gaining popularity. Nonetheless, there are still challenges to be addressed. Despite the prosperity of deep learning models, it should be emphasized that data is at the core of developing and evaluating the models.^{239,240} To make the models (either predictive or generative) more useful, data must be in sufficient amount and should maintain high quality. However, a fact to our dismay is that although existing chemical libraries have a large amount of molecules, the number of data points for each specific assay can be very scarce.²³⁹ Sometimes, even the quality of benchmark datasets is questionable with regard to the representative power for real-world drug discovery imposed by the vast chemical space.⁵⁵ Datasets in drug discovery can be highly imbalanced.⁵⁸ Thus, when evaluating the models, there is need to obtain appropriate datasets and also consider data balancing methods as well as proper evaluation metrics (e.g., AUPRC vs AUROC).⁹

Besides, for the DMTA cycle (see Section “Drug Discovery Overview”) in drug design, it should always be driven by a need or certain hypotheses.⁹ Even equipped with perfect

predictive and generative models, a question still remains, i.e., what are the hypotheses for designing a drug candidate? In other words, what are the desired properties underlying an ideal drug candidate? To generate the insights into drug design, real-world data (e.g., electronic health records (EHR) and marketed drug databases)²⁴¹ is receiving substantial attention for understanding the effectiveness and side effects of different therapeutics. Recently, we mined a large-scale EHR database for the innate properties underlying opioid analgesics with reduced overdose effects.²⁴² We also mined the DrugBank database to identify key pharmacological components (i.e., carriers, transporters, enzymes and targets) underlying drug-drug interactions (DDIs).²⁴³ These patterns emerging from real-world data (RWD) allows hypotheses generation and can calibrate drug design insights.²⁴¹

Another challenge is that deep learning, despite its superior performance, still leaves the model elusive for human interpretation. An ongoing need is, therefore, to develop explainable models with high interpretability. More specifically, there are four aspects to cover:³¹ 1) Transparency, which is knowing how the system reaches a particular answer; 2) Justification, which is elucidating why the answer provided by the model is acceptable; 3) Informativeness, which is providing new information to human decision makers; and 4) Uncertainty estimation, which is quantifying how reliable a prediction is. An ideal state is that AI can allow scientists to hone their knowledge and beliefs on the investigated process. For more details on explainable AI in drug discovery, we refer the readers to the review by Jimenez-Luna et al.³¹

In addition to the scientific challenges, technical concerns remain. One unignorable reality is that, even for the state-of-the-art representation learning on molecular graphs, fixed fingerprints can still outperform GNN-derived representations for molecular property prediction.²⁴⁴ In fact, ECFPs are a component of some GNN models.^{63,71} Besides, there is a lack of a unified protocol for AI-driven drug discovery studies. For example, different benchmark datasets, different split folds and evaluation metrics are used across the studies for molecular property prediction, let alone the varying hyper-parameters tuning, training and evaluation

procedure.²⁴⁴ For molecule generation, Walters et al²³⁹ already proposed a few guidelines to evaluate the novelty of AI-discovered molecules. Likewise, protocols for molecular property prediction are also needed.

Overall, there are many promising opportunities as well as significant challenges when applying AI in drug discovery. In order to launch successful applications, we need to understand the basic concepts and consider the task, the data, the molecule representation, the model architecture and the learning paradigm as a whole. In this survey, we have covered multiple aspects centered around AI-driven drug discovery. We envision that with these aspects well understood, more meaningful contributions will be made to substantially transform this field.

Author contributions statement

J.D. and Z.Y. conceived the manuscript and the GitHub repository. J.D. drafted the manuscript and built the GitHub repository. All authors made critical revisions and review on the manuscript.

Acknowledgement

This project is partially funded by a Stony Brook University OVPR Seed Grant. The neural networks templates are from Visuals by dair.ai (<https://github.com/dair-ai/ml-visuals>).

Supporting Information Available

https://github.com/dengjianyuan/Survey_AI_Drug_Discovery

References

- (1) Mullard, A. New drugs cost US \$2.6 billion to develop. *Nat. Rev. Drug Discov.* **2014**, *13*, 877.
- (2) Dowden, H.; Munro, J. Trends in clinical success rates and therapeutic focus. *Nat. Rev. Drug Discov.* **2019**, *18*, 495–497.
- (3) Schneider, G. Automating drug discovery. *Nat. Rev. Drug Discov.* **2018**, *17*, 97.
- (4) Chen, H.; Engkvist, O.; Wang, Y.; Olivecrona, M.; Blaschke, T. The rise of deep learning in drug discovery. *Drug Discov. Today* **2018**, *23*, 1241–1250.
- (5) Mater, A. C.; Coote, M. L. Deep learning in chemistry. *J. Chem. Inf. Model* **2019**, *59*, 2545–2559.
- (6) Vamathevan, J.; Clark, D.; Czodrowski, P.; Dunham, I.; Ferran, E.; Lee, G.; Li, B.; Madabhushi, A.; Shah, P.; Spitzer, M., et al. Applications of machine learning in drug discovery and development. *Nat. Rev. Drug Discov.* **2019**, *18*, 463–477.
- (7) Paul, D.; Sanap, G.; Shenoy, S.; Kalyane, D.; Kalia, K.; Tekade, R. K. Artificial intelligence in drug discovery and development. *Drug Discovery Today* **2021**, *26*, 80.
- (8) Stumpfe, D.; Bajorath, J. Current trends, overlooked issues, and unmet challenges in virtual screening. *J. Chem. Inf. Model* **2020**, *60*, 4112–4115.
- (9) Schneider, P.; Walters, W. P.; Plowright, A. T.; Sieroka, N.; Listgarten, J.; Goodnow, R. A.; Fisher, J.; Jansen, J. M.; Duca, J. S.; Rush, T. S., et al. Rethinking drug design in the artificial intelligence era. *Nat. Rev. Drug Discov.* **2020**, *19*, 353–364.
- (10) Boström, J.; Brown, D. G.; Young, R. J.; Keserü, G. M. Expanding the medicinal chemistry synthetic toolbox. *Nat. Rev. Drug Discov.* **2018**, *17*, 709–727.

- (11) Strokach, A.; Becerra, D.; Corbi-Verge, C.; Perez-Riba, A.; Kim, P. M. Fast and flexible protein design using deep graph neural networks. *Cell Syst.* **2020**, *11*, 402–411.
- (12) Pushpakom, S.; Iorio, F.; Eyers, P. A.; Escott, K. J.; Hopper, S.; Wells, A.; Doig, A.; Guilliams, T.; Latimer, J.; McNamee, C., et al. Drug repurposing: progress, challenges and recommendations. *Nat. Rev. Drug Discov.* **2019**, *18*, 41–58.
- (13) Tsigelny, I. F. Artificial intelligence in drug combination therapy. *Brief. Bioinformatics* **2019**, *20*, 1434–1448.
- (14) Paananen, J.; Fortino, V. An omics perspective on drug target discovery platforms. *Brief. Bioinformatics* **2020**, *21*, 1937–1953.
- (15) Hughes, J. P.; Rees, S.; Kalindjian, S. B.; Philpott, K. L. Principles of early drug discovery. *Br. J. Pharmacol.* **2011**, *162*, 1239–1249.
- (16) Pereira, D.; Williams, J. Origin and evolution of high throughput screening. *Br. J. Pharmacol.* **2007**, *152*, 53–61.
- (17) Bender, A.; Bojanic, D.; Davies, J. W.; Crisman, T. J.; Mikhailov, D.; Scheiber, J.; Jenkins, J. L.; Deng, Z.; Hill, W. A. G.; Popov, M., et al. Which aspects of HTS are empirically correlated with downstream success? *Curr Opin Drug Discov Devel* **2008**, *11*, 327.
- (18) Wang, Y.; Bryant, S. H.; Cheng, T.; Wang, J.; Gindulyte, A.; Shoemaker, B. A.; Thiessen, P. A.; He, S.; Zhang, J. Pubchem bioassay: 2017 update. *Nucleic Acids Res.* **2017**, *45*, D955–D963.
- (19) Sterling, T.; Irwin, J. J. ZINC 15–ligand discovery for everyone. *J. Chem. Inf. Model* **2015**, *55*, 2324–2337.

- (20) Kim, S. Getting the most out of PubChem for virtual screening. *Expert Opin Drug Discov* **2016**, *11*, 843–855.
- (21) Scior, T.; Bender, A.; Tresadern, G.; Medina-Franco, J. L.; Martínez-Mayorga, K.; Langer, T.; Cuanalo-Contreras, K.; Agrafiotis, D. K. Recognizing pitfalls in virtual screening: a critical review. *J. Chem. Inf. Model* **2012**, *52*, 867–881.
- (22) Salahudeen, M. S.; Nishtala, P. S. An overview of pharmacodynamic modelling, ligand-binding approach and its application in clinical practice. *Saudi Pharm J* **2017**, *25*, 165–175.
- (23) Hu, Y.; Bajorath, J. Compound promiscuity: what can we learn from current data? *Drug Discov. Today* **2013**, *18*, 644–650.
- (24) Yusof, I.; Shah, F.; Hashimoto, T.; Segall, M. D.; Greene, N. Finding the rules for successful drug optimisation. *Drug Discov. Today* **2014**, *19*, 680–687.
- (25) Nicolaou, C. A.; Brown, N. Multi-objective optimization methods in drug design. *Drug Discov. Today: Technologies* **2013**, *10*, e427–e435.
- (26) Muratov, E. N.; Bajorath, J.; Sheridan, R. P.; Tetko, I. V.; Filimonov, D.; Poroikov, V.; Oprea, T. I.; Baskin, I. I.; Varnek, A.; Roitberg, A., et al. QSAR without borders. *Chem. Soc. Rev.* **2020**, *49*, 3525–3564.
- (27) Schneider, G.; Fechner, U. Computer-based de novo design of drug-like molecules. *Nat. Rev. Drug Discov.* **2005**, *4*, 649–663.
- (28) Dobson, C. M. Chemical space and biology. 2004.
- (29) Sliwoski, G.; Kothiwale, S.; Meiler, J.; Lowe, E. W. Computational methods in drug discovery. *Pharmacol. Rev.* **2014**, *66*, 334–395.
- (30) Van Drie, J. H. Computer-aided drug design: the next 20 years. *J. Comput. Aided Mol. Des.* **2007**, *21*, 591–601.

- (31) Jiménez-Luna, J.; Grisoni, F.; Schneider, G. Drug discovery with explainable artificial intelligence. *Nat. Mach. Intell.* **2020**, *2*, 573–584.
- (32) Bajorath, J. Integration of virtual and high-throughput screening. *Nat. Rev. Drug Discov.* **2002**, *1*, 882–894.
- (33) Schneider, G. Virtual screening: an endless staircase? *Nat. Rev. Drug Discov.* **2010**, *9*, 273–276.
- (34) Polishchuk, P. Interpretation of quantitative structure–activity relationship models: past, present, and future. *J. Chem. Inf. Model* **2017**, *57*, 2618–2639.
- (35) Sydow, D.; Burggraaff, L.; Szengel, A.; van Vlijmen, H. W.; IJzerman, A. P.; van Westen, G. J.; Volkamer, A. Advances and challenges in computational target prediction. *J. Chem. Inf. Model* **2019**, *59*, 1728–1742.
- (36) Maggiora, G. On outliers and activity cliffs—why QSAR often disappoints. *J Chem Inf Model* **2006**, *46*, 1535–1535.
- (37) Stumpfe, D.; Hu, Y.; Dimova, D.; Bajorath, J. Recent progress in understanding activity cliffs and their utility in medicinal chemistry: miniperspective. *J. Med. Chem.* **2014**, *57*, 18–28.
- (38) Bajorath, J. Duality of activity cliffs in drug discovery. *Expert Opin Drug Discov* **2019**, *14*, 517–520.
- (39) Ma, J.; Sheridan, R. P.; Liaw, A.; Dahl, G. E.; Svetnik, V. Deep neural nets as a method for quantitative structure–activity relationships. *J. Chem. Inf. Model* **2015**, *55*, 263–274.
- (40) Lavecchia, A. Machine-learning approaches in drug discovery: methods and applications. *Drug Discov. Today* **2015**, *20*, 318–331.

- (41) Krizhevsky, A.; Sutskever, I.; Hinton, G. E. Imagenet classification with deep convolutional neural networks. *Advances in Neural Information Processing Systems* **2012**, *25*, 1097–1105.
- (42) Alom, M. Z.; Taha, T. M.; Yakopcic, C.; Westberg, S.; Sidike, P.; Nasrin, M. S.; Van Esesn, B. C.; Awwal, A. A. S.; Asari, V. K. The history began from alexnet: A comprehensive survey on deep learning approaches. *arXiv preprint arXiv:1803.01164* **2018**,
- (43) Öztürk, H.; Özgür, A.; Schwaller, P.; Laino, T.; Ozkirimli, E. Exploring chemical space using natural language processing methodologies for drug discovery. *Drug Discov. Today* **2020**, *25*, 689–705.
- (44) Jiménez-Luna, J.; Grisoni, F.; Weskamp, N.; Schneider, G. Artificial intelligence in drug discovery: Recent advances and future perspectives. *Expert Opin Drug Discov* **2021**, 1–11.
- (45) Zhavoronkov, A.; Ivanenkov, Y. A.; Aliper, A.; Veselov, M. S.; Aladinskiy, V. A.; Aladinskaya, A. V.; Terentiev, V. A.; Polykovskiy, D. A.; Kuznetsov, M. D.; Asadulaev, A., et al. Deep learning enables rapid identification of potent DDR1 kinase inhibitors. *Nat. Biotechnol.* **2019**, *37*, 1038–1040.
- (46) Stokes, J. M.; Yang, K.; Swanson, K.; Jin, W.; Cubillos-Ruiz, A.; Donghia, N. M.; MacNair, C. R.; French, S.; Carfrae, L. A.; Bloom-Ackermann, Z., et al. A deep learning approach to antibiotic discovery. *Cell* **2020**, *180*, 688–702.
- (47) Chuang, K. V.; Gunsalus, L. M.; Keiser, M. J. Learning Molecular Representations for Medicinal Chemistry: Miniperspective. *J. Med. Chem.* **2020**, *63*, 8705–8722.
- (48) Mayr, A.; Klambauer, G.; Unterthiner, T.; Hochreiter, S. DeepTox: toxicity prediction using deep learning. *Frontiers in Environmental Science* **2016**, *3*, 80.

- (49) Andrade, R. J.; Chalasani, N.; Björnsson, E. S.; Suzuki, A.; Kullak-Ublick, G. A.; Watkins, P. B.; Devarbhavi, H.; Merz, M.; Lucena, M. I.; Kaplowitz, N., et al. Drug-induced liver injury. *Nat. Rev. Dis. Primers* **2019**, *5*, 1–22.
- (50) Elton, D. C.; Boukouvalas, Z.; Fuge, M. D.; Chung, P. W. Deep learning for molecular design—a review of the state of the art. *Mol. Syst. Des. Eng.* **2019**, *4*, 828–849.
- (51) Mercado, R.; Rastemo, T.; Lindelöf, E.; Klambauer, G.; Engkvist, O.; Chen, H.; Bjerrum, E. J. Practical Notes on Building Molecular Graph Generative Models. *Applied AI Letters* **2020**,
- (52) Schaduengrat, N.; Lampa, S.; Simeon, S.; Gleeson, M. P.; Spjuth, O.; Nantase-namat, C. Towards reproducible computational drug discovery. *J. Cheminformatics* **2020**, *12*, 9.
- (53) Bender, A.; Cortes-Ciriano, I. Artificial intelligence in drug discovery: what is realistic, what are illusions? Part 1: Ways to make an impact, and why we are not there yet. *Drug Discov. Today* **2020**,
- (54) Bender, A.; Cortes-Ciriano, I. Artificial intelligence in drug discovery: what is realistic, what are illusions? Part 2: a discussion of chemical and biological data used for AI in drug discovery. *Drug Discov. Today* **2021**,
- (55) Walters, W. P.; Barzilay, R. Critical assessment of AI in drug discovery. *Expert Opin Drug Discov* **2021**, 1–11.
- (56) Rifaioglu, A. S.; Atas, H.; Martin, M. J.; Cetin-Atalay, R.; Atalay, V.; Doğan, T. Recent applications of deep learning and machine intelligence on in silico drug discovery: methods, tools and databases. *Brief. Bioinformatics* **2019**, *20*, 1878–1912.
- (57) Kim, S.; Chen, J.; Cheng, T.; Gindulyte, A.; He, J.; He, S.; Li, Q.; Shoemaker, B. A.;

- Thiessen, P. A.; Yu, B., et al. PubChem in 2021: new data content and improved web interfaces. *Nucleic Acids Res.* **2021**, *49*, D1388–D1395.
- (58) Korkmaz, S. Deep learning-based imbalanced data classification for drug discovery. *J. Chem. Inf. Model* **2020**, *60*, 4180–4190.
- (59) Chithrananda, S.; Grand, G.; Ramsundar, B. ChemBERTa: Large-Scale Self-Supervised Pretraining for Molecular Property Prediction. *arXiv preprint arXiv:2010.09885* **2020**,
- (60) Gaulton, A.; Hersey, A.; Nowotka, M.; Bento, A. P.; Chambers, J.; Mendez, D.; Mutowo, P.; Atkinson, F.; Bellis, L. J.; Cibrián-Uhalte, E., et al. The ChEMBL database in 2017. *Nucleic Acids Res.* **2017**, *45*, D945–D954.
- (61) Davies, M.; Nowotka, M.; Papadatos, G.; Dedman, N.; Gaulton, A.; Atkinson, F.; Bellis, L.; Overington, J. P. ChEMBL web services: streamlining access to drug discovery data and utilities. *Nucleic Acids Res.* **2015**, *43*, W612–W620.
- (62) Mayr, A.; Klambauer, G.; Unterthiner, T.; Steijaert, M.; Wegner, J. K.; Ceulemans, H.; Clevert, D.-A.; Hochreiter, S. Large-scale comparison of machine learning methods for drug target prediction on ChEMBL. *Chem. Sci* **2018**, *9*, 5441–5451.
- (63) Rong, Y.; Bian, Y.; Xu, T.; Xie, W.; Wei, Y.; Huang, W.; Huang, J. Grover: Self-supervised message passing transformer on large-scale molecular data. *arXiv preprint arXiv:2007.02835* **2020**,
- (64) Polykovskiy, D.; Zhebrak, A.; Sanchez-Lengeling, B.; Golovanov, S.; Tatanov, O.; Belyaev, S.; Kurbanov, R.; Artamonov, A.; Aladinskiy, V.; Veselov, M., et al. Molecular sets (MOSES): a benchmarking platform for molecular generation models. *Front. Pharmacol.* **2020**, *11*.

- (65) Lagarde, N.; Zagury, J.-F.; Montes, M. Benchmarking data sets for the evaluation of virtual ligand screening methods: review and perspectives. *J. Chem. Inf. Model* **2015**, *55*, 1297–1307.
- (66) Chen, M.; Suzuki, A.; Thakkar, S.; Yu, K.; Hu, C.; Tong, W. DILIrank: the largest reference drug list ranked by the risk for developing drug-induced liver injury in humans. *Drug Discov Today* **2016**, *21*, 648–653.
- (67) David, L.; Thakkar, A.; Mercado, R.; Engkvist, O. Molecular representations in AI-driven drug discovery: a review and practical guide. *J. Cheminformatics* **2020**, *12*, 1–22.
- (68) Morgan, H. L. The generation of a unique machine description for chemical structures—a technique developed at chemical abstracts service. *J. Chem. Doc* **1965**, *5*, 107–113.
- (69) Subramanian, G.; Ramsundar, B.; Pande, V.; Denny, R. A. Computational modeling of β -secretase 1 (BACE-1) inhibitors using ligand based approaches. *J. Chem. Inf. Model* **2016**, *56*, 1936–1949.
- (70) Zang, Q.; Mansouri, K.; Williams, A. J.; Judson, R. S.; Allen, D. G.; Casey, W. M.; Kleinstreuer, N. C. In silico prediction of physicochemical properties of environmental chemicals using molecular fingerprints and machine learning. *J. Chem. Inf. Model* **2017**, *57*, 36–49.
- (71) Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M., et al. Analyzing learned molecular representations for property prediction. *J. Chem. Inf. Model* **2019**, *59*, 3370–3388.
- (72) Mercado, R.; Rastemo, T.; Lindelöf, E.; Klambauer, G.; Engkvist, O.; Chen, H.; Bjerrum, E. J. Graph Networks for Molecular Design. *Mach. Learn.: Sci. Technol.* **2020**,

- (73) Jin, W.; Barzilay, R.; Jaakkola, T. Multi-objective molecule generation using interpretable substructures. *International Conference on Machine Learning*. 2020; pp 4849–4859.
- (74) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J Chem Inform Comput Sci* **1988**, *28*, 31–36.
- (75) Weininger, D.; Weininger, A.; Weininger, J. L. SMILES. 2. Algorithm for generation of unique SMILES notation. *J Chem Inform Comput Sci* **1989**, *29*, 97–101.
- (76) Bian, Y.; Xie, X.-Q. Generative chemistry: drug discovery with deep learning generative models. *J. Mol. Model.* **2021**, *27*, 1–18.
- (77) Xiong, Z.; Wang, D.; Liu, X.; Zhong, F.; Wan, X.; Li, X.; Li, Z.; Luo, X.; Chen, K.; Jiang, H., et al. Pushing the boundaries of molecular representation for drug discovery with the graph attention mechanism. *J. Med. Chem.* **2019**, *63*, 8749–8760.
- (78) Gómez-Bombarelli, R.; Wei, J. N.; Duvenaud, D.; Hernández-Lobato, J. M.; Sánchez-Lengeling, B.; Sheberla, D.; Aguilera-Iparraguirre, J.; Hirzel, T. D.; Adams, R. P.; Aspuru-Guzik, A. Automatic chemical design using a data-driven continuous representation of molecules. *ACS Cent. Sci.* **2018**, *4*, 268–276.
- (79) Popova, M.; Isayev, O.; Tropsha, A. Deep reinforcement learning for de novo drug design. *Sci. Adv.* **2018**, *4*, eaap7885.
- (80) Ragoza, M.; Hochuli, J.; Idrobo, E.; Sunseri, J.; Koes, D. R. Protein–ligand scoring with convolutional neural networks. *J. Chem. Inf. Model* **2017**, *57*, 942–957.
- (81) Jiménez, J.; Skalic, M.; Martinez-Rosell, G.; De Fabritiis, G. K deep: protein–ligand absolute binding affinity prediction via 3d-convolutional neural networks. *J. Chem. Inf. Model* **2018**, *58*, 287–296.

- (82) Lim, J.; Ryu, S.; Park, K.; Choe, Y. J.; Ham, J.; Kim, W. Y. Predicting drug–target interaction using a novel graph neural network with 3D structure-embedded graph representation. *J. Chem. Inf. Model* **2019**, *59*, 3981–3988.
- (83) Hernandez, M.; Liang Gan, G.; Linvill, K.; Dukatz, C.; Feng, J.; Bhisetti, G. A quantum-inspired method for three-dimensional ligand-based virtual screening. *J. Chem. Inf. Model* **2019**, *59*, 4475–4485.
- (84) Wu, K.; Wei, G.-W. Quantitative toxicity prediction using topology based multitask deep neural networks. *J. Chem. Inf. Model* **2018**, *58*, 520–531.
- (85) Skalic, M.; Jiménez, J.; Sabbadin, D.; De Fabritiis, G. Shape-based generative modeling for de novo drug design. *J. Chem. Inf. Model* **2019**, *59*, 1205–1214.
- (86) Simm, G.; Pinsler, R.; Hernández-Lobato, J. M. Reinforcement learning for molecular design guided by quantum mechanics. International Conference on Machine Learning. 2020; pp 8959–8969.
- (87) Hemmerich, J.; Asilar, E.; Ecker, G. F. COVER: conformational oversampling as data augmentation for molecules. *J. Cheminformatics* **2020**, *12*, 1–12.
- (88) Fernandez, M.; Ban, F.; Woo, G.; Hsing, M.; Yamazaki, T.; LeBlanc, E.; Rennie, P. S.; Welch, W. J.; Cherkasov, A. Toxic colors: the use of deep learning for predicting toxicity of compounds merely from their graphic images. *J. Chem. Inf. Model* **2018**, *58*, 1533–1543.
- (89) Meyer, J. G.; Liu, S.; Miller, I. J.; Coon, J. J.; Gitter, A. Learning drug functions from chemical structures with convolutional neural networks and random forests. *J. Chem. Inf. Model* **2019**, *59*, 4438–4449.
- (90) Cortés-Ciriano, I.; Bender, A. KekuleScope: prediction of cancer cell line sensitivity

- and compound potency using convolutional neural networks trained on compound images. *J. Cheminformatics* **2019**, *11*, 1–16.
- (91) Rifaioğlu, A. S.; Nalbat, E.; Atalay, V.; Martin, M. J.; Cetin-Atalay, R.; Doğan, T. DEEPScreen: high performance drug–target interaction prediction with convolutional neural networks using 2-D structural compound representations. *Chem. Sci* **2020**, *11*, 2531–2557.
- (92) Rajan, K.; Brinkhaus, H. O.; Sorokina, M.; Zielesny, A.; Steinbeck, C. DECIMER-Segmentation: Automated extraction of chemical structure depictions from scientific literature. *J. Cheminformatics* **2021**, *13*, 1–9.
- (93) Wu, Z.; Ramsundar, B.; Feinberg, E. N.; Gomes, J.; Geniesse, C.; Pappu, A. S.; Leswing, K.; Pande, V. MoleculeNet: a benchmark for molecular machine learning. *Chem. Sci* **2018**, *9*, 513–530.
- (94) Ramsundar, B.; Eastman, P.; Walters, P.; Pande, V.; Leswing, K.; Wu, Z. *Deep Learning for the Life Sciences*; O'Reilly Media, 2019; <https://www.amazon.com/Deep-Learning-Life-Sciences-Microscopy/dp/1492039837>.
- (95) Fabian, B.; Edlich, T.; Gaspar, H.; Segler, M.; Meyers, J.; Fiscato, M.; Ahmed, M. Molecular representation learning with language models and domain-relevant auxiliary tasks. *arXiv preprint arXiv:2011.13230* **2020**,
- (96) Shen, W. X.; Zeng, X.; Zhu, F.; li Wang, Y.; Qin, C.; Tan, Y.; Jiang, Y. Y.; Chen, Y. Z. Out-of-the-box deep learning prediction of pharmaceutical properties by broadly learned knowledge-based molecular representations. *Nat. Mach. Intell.* **2021**, *3*, 334–343.
- (97) Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H. Molecular de-novo design through deep reinforcement learning. *J. Cheminformatics* **2017**, *9*, 1–14.

- (98) Blaschke, T.; Arús-Pous, J.; Chen, H.; Margreitter, C.; Tyrchan, C.; Engkvist, O.; Papadopoulos, K.; Patronov, A. REINVENT 2.0: An AI Tool for De Novo Drug Design. *J. Chem. Inf. Model* **2020**,
- (99) Brown, N.; Fiscato, M.; Segler, M. H.; Vaucher, A. C. GuacaMol: benchmarking models for de novo molecular design. *J. Chem. Inf. Model* **2019**, *59*, 1096–1108.
- (100) Segler, M. H.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating focused molecule libraries for drug discovery with recurrent neural networks. *ACS Cent. Sci.* **2018**, *4*, 120–131.
- (101) Walters, W. P.; Murcko, M. A. Prediction of ‘drug-likeness’. *Adv. Drug Deliv. Rev.* **2002**, *54*, 255–271.
- (102) Schneider, N.; Jäckels, C.; Andres, C.; Hutter, M. C. Gradual in silico filtering for druglike substances. *J. Chem. Inf. Model* **2008**, *48*, 613–628.
- (103) Palmer, D. S.; O’Boyle, N. M.; Glen, R. C.; Mitchell, J. B. Random forest models to predict aqueous solubility. *J. Chem. Inf. Model* **2007**, *47*, 150–158.
- (104) Schroeter, T.; Schwaighofer, A.; Mika, S.; Ter Laak, A.; Suelzle, D.; Ganzer, U.; Heinrich, N.; Müller, K.-R. Machine learning models for lipophilicity and their domain of applicability. *Mol. Pharm.* **2007**, *4*, 524–538.
- (105) Hou, T.; Wang, J.; Li, Y. ADME evaluation in drug discovery. 8. The prediction of human intestinal absorption by a support vector machine. *J. Chem. Inf. Model* **2007**, *47*, 2408–2415.
- (106) Tian, S.; Li, Y.; Wang, J.; Zhang, J.; Hou, T. ADME evaluation in drug discovery. 9. Prediction of oral bioavailability in humans based on molecular properties and structural fingerprints. *Mol. Pharm.* **2011**, *8*, 841–851.

- (107) Sakiyama, Y.; Yuki, H.; Moriya, T.; Hattori, K.; Suzuki, M.; Shimada, K.; Honma, T. Predicting human liver microsomal stability with machine learning techniques. *J. Mol. Graph. Model.* **2008**, *26*, 907–915.
- (108) Vasanthanathan, P.; Taboureau, O.; Oostenbrink, C.; Vermeulen, N. P.; Olsen, L.; Jørgensen, F. S. Classification of cytochrome P450 1A2 inhibitors and noninhibitors by machine learning techniques. *Drug Metab. Dispos* **2009**, *37*, 658–664.
- (109) Riddick, G.; Song, H.; Ahn, S.; Walling, J.; Borges-Rivera, D.; Zhang, W.; Fine, H. A. Predicting in vitro drug sensitivity using Random Forests. *Bioinformatics* **2011**, *27*, 220–224.
- (110) Zhao, C.; Zhang, H.; Zhang, X.; Liu, M.; Hu, Z.; Fan, B. Application of support vector machine (SVM) for prediction toxic activity of different data sets. *Toxicology* **2006**, *217*, 105–119.
- (111) Heikamp, K.; Bajorath, J. Support vector machines for drug discovery. *Expert Opin Drug Discov* **2014**, *9*, 93–104.
- (112) Svetnik, V.; Liaw, A.; Tong, C.; Culberson, J. C.; Sheridan, R. P.; Feuston, B. P. Random forest: a classification and regression tool for compound classification and QSAR modeling. *J Chem Inform Comput Sci* **2003**, *43*, 1947–1958.
- (113) Dahl, G. Deep learning how I did it: Merck 1st place interview. *Online article available from <http://blog.kaggle.com/2012/11/01/deep-learning-how-i-did-it-merck-1st-place-interview>* **2012**,
- (114) LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. *Nature* **2015**, *521*, 436–444.
- (115) Simm, J.; Klambauer, G.; Arany, A.; Steijaert, M.; Wegner, J. K.; Gustin, E.; Chupakhin, V.; Chong, Y. T.; Vialard, J.; Buijnsters, P., et al. Repurposing high-

- throughput image assays enables biological activity prediction for drug discovery. *Cell Chem. Biol.* **2018**, *25*, 611–618.
- (116) Hofmarcher, M.; Rumetshofer, E.; Clevert, D.-A.; Hochreiter, S.; Klambauer, G. Accurate prediction of biological assays with high-throughput microscopy images and convolutional networks. *J. Chem. Inf. Model* **2019**, *59*, 1163–1171.
- (117) Ramsundar, B.; Kearnes, S.; Riley, P.; Webster, D.; Konerding, D.; Pande, V. Massively multitask networks for drug discovery. *arXiv preprint arXiv:1502.02072* **2015**,
- (118) Duvenaud, D.; Maclaurin, D.; Aguilera-Iparraguirre, J.; Gómez-Bombarelli, R.; Hirzel, T.; Aspuru-Guzik, A.; Adams, R. P. Convolutional networks on graphs for learning molecular fingerprints. *arXiv preprint arXiv:1509.09292* **2015**,
- (119) Glen, R. C.; Bender, A.; Arnby, C. H.; Carlsson, L.; Boyer, S.; Smith, J. Circular fingerprints: flexible molecular descriptors with applications from physical chemistry to ADME. *IDrugs* **2006**, *9*, 199.
- (120) Goh, G. B.; Siegel, C.; Vishnu, A.; Hodas, N. O.; Baker, N. Chemception: a deep neural network with minimal chemistry knowledge matches the performance of expert-developed QSAR/QSPR models. *arXiv preprint arXiv:1706.06689* **2017**,
- (121) Huang, G.; Liu, Z.; Van Der Maaten, L.; Weinberger, K. Q. Densely connected convolutional networks. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2017; pp 4700–4708.
- (122) He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2016; pp 770–778.
- (123) Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* **2014**,

- (124) Staker, J.; Marshall, K.; Abel, R.; McQuaw, C. M. Molecular structure extraction from documents using deep learning. *J. Chem. Inf. Model* **2019**, *59*, 1017–1029.
- (125) Rajan, K.; Zielesny, A.; Steinbeck, C. DECIMER: towards deep learning for chemical image recognition. *J. Cheminformatics* **2020**, *12*, 1–9.
- (126) Hossain, M. Z.; Sohel, F.; Shiratuddin, M. F.; Laga, H. A comprehensive survey of deep learning for image captioning. *ACM Computing Surveys (CSUR)* **2019**, *51*, 1–36.
- (127) Mikolov, T.; Kombrink, S.; Burget, L.; Černocký, J.; Khudanpur, S. Extensions of recurrent neural network language model. 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). 2011; pp 5528–5531.
- (128) Boulanger-Lewandowski, N.; Bengio, Y.; Vincent, P. Modeling temporal dependencies in high-dimensional sequences: Application to polyphonic music generation and transcription. *arXiv preprint arXiv:1206.6392* **2012**,
- (129) Hochreiter, S.; Schmidhuber, J. Long short-term memory. *Neural Comput.* **1997**, *9*, 1735–1780.
- (130) Chung, J.; Gulcehre, C.; Cho, K.; Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. *arXiv preprint arXiv:1412.3555* **2014**,
- (131) Goh, G. B.; Hodas, N. O.; Siegel, C.; Vishnu, A. Smiles2vec: An interpretable general-purpose deep neural network for predicting chemical properties. *arXiv preprint arXiv:1712.02034* **2017**,
- (132) Neil, D.; Segler, M.; Guasch, L.; Ahmed, M.; Plumbley, D.; Sellwood, M.; Brown, N. Exploring deep recurrent models with reinforcement learning for molecule design. Proceedings of The International Conference on Learning Representations. 2018.
- (133) Joulin, A.; Mikolov, T. Inferring algorithmic patterns with stack-augmented recurrent nets. *arXiv preprint arXiv:1503.01007* **2015**,

- (134) Ståhl, N.; Falkman, G.; Karlsson, A.; Mathiason, G.; Bostrom, J. Deep reinforcement learning for multiparameter optimization in de novo drug design. *J. Chem. Inf. Model* **2019**, *59*, 3166–3176.
- (135) Zheng, S.; Yan, X.; Yang, Y.; Xu, J. Identifying structure–property relationships through SMILES syntax analysis with self-attention mechanism. *J. Chem. Inf. Model* **2019**, *59*, 914–923.
- (136) You, J.; Ying, R.; Ren, X.; Hamilton, W.; Leskovec, J. Graphrnn: Generating realistic graphs with deep auto-regressive models. International Conference on Machine Learning. 2018; pp 5708–5717.
- (137) Li, Y.; Vinyals, O.; Dyer, C.; Pascanu, R.; Battaglia, P. Learning deep generative models of graphs. *arXiv preprint arXiv:1803.03324* **2018**,
- (138) Li, Y.; Zhang, L.; Liu, Z. Multi-objective de novo drug design with conditional graph generative model. *J. Cheminformatics* **2018**, *10*, 1–24.
- (139) Popova, M.; Shvets, M.; Oliva, J.; Isayev, O. MolecularRNN: Generating realistic molecular graphs with optimized properties. *arXiv preprint arXiv:1905.13372* **2019**,
- (140) You, J.; Liu, B.; Ying, R.; Pande, V.; Leskovec, J. Graph convolutional policy network for goal-directed molecular graph generation. *arXiv preprint arXiv:1806.02473* **2018**,
- (141) Sattarov, B.; Baskin, I. I.; Horvath, D.; Marcou, G.; Bjerrum, E. J.; Varnek, A. De novo molecular design by combining deep autoencoder recurrent neural networks with generative topographic mapping. *J. Chem. Inf. Model* **2019**, *59*, 1182–1196.
- (142) Guimaraes, G. L.; Sanchez-Lengeling, B.; Outeiral, C.; Farias, P. L. C.; Aspuru-Guzik, A. Objective-reinforced generative adversarial networks (ORGAN) for sequence generation models. *arXiv preprint arXiv:1705.10843* **2017**,

- (143) Sanchez-Lengeling, B.; Outeiral, C.; Guimaraes, G. L.; Aspuru-Guzik, A. Optimizing distributions over molecular space. An objective-reinforced generative adversarial network for inverse-design chemistry (ORGANIC). *ChemRxiv* **2017**, 2017.
- (144) Wu, Z.; Pan, S.; Chen, F.; Long, G.; Zhang, C.; Philip, S. Y. A comprehensive survey on graph neural networks. *IEEE Trans Neural Netw Learn* **2020**,
- (145) Li, Y.; Tarlow, D.; Brockschmidt, M.; Zemel, R. Gated graph sequence neural networks. *arXiv preprint arXiv:1511.05493* **2015**,
- (146) Defferrard, M.; Bresson, X.; Vandergheynst, P. Convolutional neural networks on graphs with fast localized spectral filtering. *arXiv preprint arXiv:1606.09375* **2016**,
- (147) Kipf, T. N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv preprint arXiv:1609.02907* **2016**,
- (148) Gilmer, J.; Schoenholz, S. S.; Riley, P. F.; Vinyals, O.; Dahl, G. E. Neural message passing for quantum chemistry. International Conference on Machine Learning. 2017; pp 1263–1272.
- (149) Hamilton, W. L.; Ying, R.; Leskovec, J. Inductive representation learning on large graphs. *arXiv preprint arXiv:1706.02216* **2017**,
- (150) Veličković, P.; Cucurull, G.; Casanova, A.; Romero, A.; Lio, P.; Bengio, Y. Graph attention networks. *arXiv preprint arXiv:1710.10903* **2017**,
- (151) Xu, K.; Hu, W.; Leskovec, J.; Jegelka, S. How powerful are graph neural networks? *arXiv preprint arXiv:1810.00826* **2018**,
- (152) Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P. Molecular graph convolutions: moving beyond fingerprints. *J. Comput. Aided Mol. Des.* **2016**, 30, 595–608.
- (153) Landrum, G. RDKit: Open-Source Cheminformatics Software. *RDKit* **2016**,

- (154) Withnall, M.; Lindelöf, E.; Engkvist, O.; Chen, H. Building attention and edge message passing neural networks for bioactivity and physical–chemical property prediction. *J. Cheminformatics* **2020**, *12*, 1–18.
- (155) Schütt, K. T.; Kindermans, P.-J.; Sauceda, H. E.; Chmiela, S.; Tkatchenko, A.; Müller, K.-R. Schnet: A continuous-filter convolutional neural network for modeling quantum interactions. *arXiv preprint arXiv:1706.08566* **2017**,
- (156) Feinberg, E. N.; Sur, D.; Wu, Z.; Husic, B. E.; Mai, H.; Li, Y.; Sun, S.; Yang, J.; Ramsundar, B.; Pande, V. S. PotentialNet for molecular property prediction. *ACS Cent. Sci.* **2018**, *4*, 1520–1530.
- (157) Klicpera, J.; Groß, J.; Günnemann, S. Directional message passing for molecular graphs. *arXiv preprint arXiv:2003.03123* **2020**,
- (158) Altae-Tran, H.; Ramsundar, B.; Pappu, A. S.; Pande, V. Low data drug discovery with one-shot learning. *ACS Cent. Sci.* **2017**, *3*, 283–293.
- (159) Liu, S.; Demirel, M. F.; Liang, Y. N-gram graph: Simple unsupervised representation for graphs, with applications to molecules. *arXiv preprint arXiv:1806.09206* **2018**,
- (160) Lu, C.; Liu, Q.; Wang, C.; Huang, Z.; Lin, P.; He, L. Molecular property prediction: A multilevel quantum interactions modeling perspective. Proceedings of the AAAI Conference on Artificial Intelligence. 2019; pp 1052–1060.
- (161) Cai, C.; Guo, P.; Zhou, Y.; Zhou, J.; Wang, Q.; Zhang, F.; Fang, J.; Cheng, F. Deep learning-based prediction of drug-induced cardiotoxicity. *J. Chem. Inf. Model* **2019**, *59*, 1073–1084.
- (162) Wang, X.; Li, Z.; Jiang, M.; Wang, S.; Zhang, S.; Wei, Z. Molecule property prediction based on spatial graph embedding. *J. Chem. Inf. Model* **2019**, *59*, 3817–3828.

- (163) Hu, W.; Liu, B.; Gomes, J.; Zitnik, M.; Liang, P.; Pande, V.; Leskovec, J. Strategies for pre-training graph neural networks. *arXiv preprint arXiv:1905.12265* **2019**,
- (164) Hao, Z.; Lu, C.; Huang, Z.; Wang, H.; Hu, Z.; Liu, Q.; Chen, E.; Lee, C. ASGN: An active semi-supervised graph neural network for molecular property prediction. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020; pp 731–752.
- (165) Nguyen, C. Q.; Kretasoulas, C.; Branson, K. M. Meta-Learning Initializations for Low-Resource Drug Discovery. *arXiv preprint arXiv:2003.05996* **2020**,
- (166) Li, X.; Yan, X.; Gu, Q.; Zhou, H.; Wu, D.; Xu, J. DeepChemStable: Chemical stability prediction with an attention-based graph convolution network. *J. Chem. Inf. Model* **2019**, *59*, 1044–1049.
- (167) Tang, B.; Kramer, S. T.; Fang, M.; Qiu, Y.; Wu, Z.; Xu, D. A self-attention based message passing neural network for predicting molecular lipophilicity and aqueous solubility. *J. Cheminformatics* **2020**, *12*, 1–9.
- (168) Pathak, Y.; Laghuvarapu, S.; Mehta, S.; Priyakumar, U. D. Chemically interpretable graph interaction network for prediction of pharmacokinetic properties of drug-like molecules. Proceedings of the AAAI Conference on Artificial Intelligence. 2020; pp 873–880.
- (169) Zhou, Z.; Kearnes, S.; Li, L.; Zare, R. N.; Riley, P. Optimization of molecules via deep reinforcement learning. *Scientific reports* **2019**, *9*, 1–10.
- (170) Khemchandani, Y.; O’Hagan, S.; Samanta, S.; Swainston, N.; Roberts, T. J.; Bollegala, D.; Kell, D. B. DeepGraphMolGen, a multi-objective, computational strategy for generating molecules with desirable properties: a graph convolution and reinforcement learning approach. *J. Cheminformatics* **2020**, *12*, 1–17.

- (171) Xu, C.; Liu, Q.; Huang, M.; Jiang, T. Reinforced Molecular Optimization with Neighborhood-Controlled Grammars. *arXiv preprint arXiv:2011.07225* **2020**,
- (172) Kingma, D. P.; Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114* **2013**,
- (173) Kingma, D. P.; Welling, M. An introduction to variational autoencoders. *arXiv preprint arXiv:1906.02691* **2019**,
- (174) Kusner, M. J.; Paige, B.; Hernández-Lobato, J. M. Grammar variational autoencoder. International Conference on Machine Learning. 2017; pp 1945–1954.
- (175) Dai, H.; Tian, Y.; Dai, B.; Skiena, S.; Song, L. Syntax-directed variational autoencoder for structured data. *arXiv preprint arXiv:1802.08786* **2018**,
- (176) Kang, S.; Cho, K. Conditional molecular design with deep generative models. *J. Chem. Inf. Model* **2018**, *59*, 43–52.
- (177) Lim, J.; Ryu, S.; Kim, J. W.; Kim, W. Y. Molecular generative model based on conditional variational autoencoder for de novo molecular design. *J. Cheminformatics* **2018**, *10*, 1–9.
- (178) Liu, Q.; Allamanis, M.; Brockschmidt, M.; Gaunt, A. L. Constrained graph variational autoencoders for molecule design. *arXiv preprint arXiv:1805.09076* **2018**,
- (179) Samanta, B.; De, A.; Jana, G.; Gómez, V.; Chattaraj, P. K.; Ganguly, N.; Gomez-Rodriguez, M. Nevae: A deep generative model for molecular graphs. *J Mach Learn Res* **2020**,
- (180) Chenthamarakshan, V.; Das, P.; Padhi, I.; Strobelt, H.; Lim, K. W.; Hoover, B.; Hoffman, S. C.; Mojsilovic, A. Target-specific and selective drug design for covid-19 using deep generative models. *arXiv preprint arXiv:2004.01215* **2020**,

- (181) Makhzani, A.; Shlens, J.; Jaitly, N.; Goodfellow, I.; Frey, B. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644* **2015**,
- (182) Kadurin, A.; Nikolenko, S.; Khrabrov, K.; Aliper, A.; Zhavoronkov, A. druGAN: an advanced generative adversarial autoencoder model for de novo generation of new molecules with desired molecular properties in silico. *Mol. Pharm.* **2017**, *14*, 3098–3104.
- (183) Blaschke, T.; Olivecrona, M.; Engkvist, O.; Bajorath, J.; Chen, H. Application of generative autoencoder in de novo molecular design. *Mol. Inform.* **2018**, *37*, 1700123.
- (184) Polykovskiy, D.; Zhebrak, A.; Vetrov, D.; Ivanenkov, Y.; Aladinskiy, V.; Mamoshina, P.; Bozdaganyan, M.; Aliper, A.; Zhavoronkov, A.; Kadurin, A. Entangled conditional adversarial autoencoder for de novo drug discovery. *Mol. Pharm.* **2018**, *15*, 4398–4405.
- (185) Simonovsky, M.; Komodakis, N. Graphvae: Towards generation of small graphs using variational autoencoders. International Conference on Artificial Neural Networks. 2018; pp 412–422.
- (186) Jin, W.; Barzilay, R.; Jaakkola, T. Junction tree variational autoencoder for molecular graph generation. International Conference on Machine Learning. 2018; pp 2323–2332.
- (187) Ma, T.; Chen, J.; Xiao, C. Constrained generation of semantically valid graphs via regularizing variational autoencoders. *arXiv preprint arXiv:1809.02630* **2018**,
- (188) Kajino, H. Molecular hypergraph grammar with its application to molecular optimization. International Conference on Machine Learning. 2019; pp 3183–3191.
- (189) Kwon, Y.; Yoo, J.; Choi, Y.-S.; Son, W.-J.; Lee, D.; Kang, S. Efficient learning of non-autoregressive graph variational autoencoders for molecular graph generation. *J. Cheminformatics* **2019**, *11*, 1–10.

- (190) Lim, J.; Hwang, S.-Y.; Kim, S.; Moon, S.; Kim, W. Y. Scaffold-based molecular design using graph generative model. *arXiv preprint arXiv:1905.13639* **2019**,
- (191) Fu, T.; Xiao, C.; Sun, J. Core: Automatic molecule optimization using copy & refine strategy. Proceedings of the AAAI Conference on Artificial Intelligence. 2020; pp 638–645.
- (192) Kwon, Y.; Lee, D.; Choi, Y.-S.; Shin, K.; Kang, S. Compressed graph representation for scalable molecular graph generation. *J. Cheminformatics* **2020**, *12*, 1–8.
- (193) Jin, W.; Barzilay, R.; Jaakkola, T. Hierarchical generation of molecular graphs using structural motifs. International Conference on Machine Learning. 2020; pp 4839–4848.
- (194) Goodfellow, I. J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative adversarial networks. *arXiv preprint arXiv:1406.2661* **2014**,
- (195) Yu, L.; Zhang, W.; Wang, J.; Yu, Y. Seqgan: Sequence generative adversarial nets with policy gradient. Proceedings of the AAAI Conference on Artificial Intelligence. 2017.
- (196) Putin, E.; Asadulaev, A.; Ivanenkov, Y.; Aladinskiy, V.; Sanchez-Lengeling, B.; Aspuru-Guzik, A.; Zhavoronkov, A. Reinforced adversarial neural computer for de novo molecular design. *J. Chem. Inf. Model* **2018**, *58*, 1194–1204.
- (197) De Cao, N.; Kipf, T. MolGAN: An implicit generative model for small molecular graphs. *arXiv preprint arXiv:1805.11973* **2018**,
- (198) Salimans, T.; Goodfellow, I.; Zaremba, W.; Cheung, V.; Radford, A.; Chen, X. Improved techniques for training gans. *arXiv preprint arXiv:1606.03498* **2016**,
- (199) Kobyzev, I.; Prince, S.; Brubaker, M. Normalizing flows: An introduction and review of current methods. *IEEE PAMI* **2020**,

- (200) Dinh, L.; Krueger, D.; Bengio, Y. Nice: Non-linear independent components estimation. *arXiv preprint arXiv:1410.8516* **2014**,
- (201) Dinh, L.; Sohl-Dickstein, J.; Bengio, S. Density estimation using real nvp. *arXiv preprint arXiv:1605.08803* **2016**,
- (202) Madhawa, K.; Ishiguro, K.; Nakago, K.; Abe, M. Graphnvp: An invertible flow model for generating molecular graphs. *arXiv preprint arXiv:1905.11600* **2019**,
- (203) Honda, S.; Akita, H.; Ishiguro, K.; Nakanishi, T.; Oono, K. Graph residual flow for molecular graph generation. *arXiv preprint arXiv:1909.13521* **2019**,
- (204) Shi, C.; Xu, M.; Zhu, Z.; Zhang, W.; Zhang, M.; Tang, J. Graphaf: a flow-based autoregressive model for molecular graph generation. *arXiv preprint arXiv:2001.09382* **2020**,
- (205) Zang, C.; Wang, F. MoFlow: an invertible flow model for generating molecular graphs. Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining. 2020; pp 617–626.
- (206) Luo, Y.; Yan, K.; Ji, S. GraphDF: A Discrete Flow Model for Molecular Graph Generation. *arXiv preprint arXiv:2102.01189* **2021**,
- (207) Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; Polosukhin, I. Attention is all you need. *arXiv preprint arXiv:1706.03762* **2017**,
- (208) Radford, A.; Narasimhan, K.; Salimans, T.; Sutskever, I. Improving language understanding by generative pre-training. *OpenAI* **2018**,
- (209) Devlin, J.; Chang, M.-W.; Lee, K.; Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* **2018**,

- (210) Radford, A.; Wu, J.; Child, R.; Luan, D.; Amodei, D.; Sutskever, I. Language models are unsupervised multitask learners. *OpenAI blog* **2019**, *1*, 9.
- (211) Liu, Y.; Ott, M.; Goyal, N.; Du, J.; Joshi, M.; Chen, D.; Levy, O.; Lewis, M.; Zettlemoyer, L.; Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692* **2019**,
- (212) Brown, T. B.; Mann, B.; Ryder, N.; Subbiah, M.; Kaplan, J.; Dhariwal, P.; Neelakantan, A.; Shyam, P.; Sastry, G.; Askell, A., et al. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165* **2020**,
- (213) Carion, N.; Massa, F.; Synnaeve, G.; Usunier, N.; Kirillov, A.; Zagoruyko, S. End-to-end object detection with transformers. European Conference on Computer Vision. 2020; pp 213–229.
- (214) Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929* **2020**,
- (215) Wang, S.; Guo, Y.; Wang, Y.; Sun, H.; Huang, J. SMILES-BERT: large scale unsupervised pre-training for molecular property prediction. Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics. 2019; pp 429–436.
- (216) Honda, S.; Shi, S.; Ueda, H. R. SMILES transformer: pre-trained molecular fingerprint for low data drug discovery. *arXiv preprint arXiv:1911.04738* **2019**,
- (217) Bradshaw, J.; Paige, B.; Kusner, M. J.; Segler, M. H.; Hernández-Lobato, J. M. A model to search for synthesizable molecules. *arXiv preprint arXiv:1906.05221* **2019**,

- (218) Grechishnikova, D. Transformer neural network for protein-specific de novo drug generation as a machine translation problem. *Scientific reports* **2021**, *11*, 1–13.
- (219) Liu, X.; Zhang, F.; Hou, Z.; Wang, Z.; Mian, L.; Zhang, J.; Tang, J. Self-supervised learning: Generative or contrastive. *arXiv preprint arXiv:2006.08218* **2020**, *1*.
- (220) Wang, Y.; Wang, J.; Cao, Z.; Farimani, A. B. MolCLR: Molecular contrastive learning of representations via graph neural networks. *arXiv preprint arXiv:2102.10056* **2021**,
- (221) Vanschoren, J. Meta-learning: A survey. *arXiv preprint arXiv:1810.03548* **2018**,
- (222) Wang, Y.; Yao, Q.; Kwok, J. T.; Ni, L. M. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys (CSUR)* **2020**, *53*, 1–34.
- (223) Kulis, B., et al. Metric learning: A survey. *Found. Trends Mach. Learn.* **2012**, *5*, 287–364.
- (224) Yang, Z.; Bastan, M.; Zhu, X.; Gray, D.; Samaras, D. Hierarchical Proxy-based Loss for Deep Metric Learning. *arXiv preprint arXiv:2103.13538* **2021**,
- (225) Movshovitz-Attias, Y.; Toshev, A.; Leung, T. K.; Ioffe, S.; Singh, S. No fuss distance metric learning using proxies. Proceedings of the IEEE International Conference on Computer Vision. 2017; pp 360–368.
- (226) Na, G. S.; Chang, H.; Kim, H. W. Machine-guided representation for accurate graph-based molecular machine learning. *Phys. Chem. Chem. Phys.* **2020**, *22*, 18526–18535.
- (227) Koge, D.; Ono, N.; Huang, M.; Altaf-Ul-Amin, M.; Kanaya, S. Embedding of Molecular Structure Using Molecular Hypergraph Variational Autoencoder with Metric Learning. *Mol. Inform.* **2020**,
- (228) Sutton, R. S.; Barto, A. G. *Reinforcement learning: An introduction*; MIT press, 2018.

- (229) Arulkumaran, K.; Deisenroth, M. P.; Brundage, M.; Bharath, A. A. A brief survey of deep reinforcement learning. *arXiv preprint arXiv:1708.05866* **2017**,
- (230) Van Hasselt, H.; Guez, A.; Silver, D. Deep reinforcement learning with double q-learning. *Proceedings of the AAAI Conference on Artificial Intelligence*. 2016.
- (231) Williams, R. J. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning* **1992**, *8*, 229–256.
- (232) Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* **2017**,
- (233) Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M.; Moritz, P. Trust region policy optimization. *International Conference on Machine Learning*. 2015; pp 1889–1897.
- (234) Deng, J.; Yang, Z.; Li, Y.; Samaras, D.; Wang, F. Towards Better Opioid Antagonists Using Deep Reinforcement Learning. *arXiv preprint arXiv:2004.04768* **2020**,
- (235) Yasonik, J. Multiobjective de novo drug design with recurrent neural networks and nondominated sorting. *J. Cheminformatics* **2020**, *12*, 1–9.
- (236) Domenico, A.; Nicola, G.; Daniela, T.; Fulvio, C.; Nicola, A.; Orazio, N. De novo drug design of targeted chemical libraries based on artificial intelligence and pair-based multiobjective optimization. *J. Chem. Inf. Model* **2020**, *60*, 4582–4593.
- (237) Liu, X.; Ye, K.; Van Vlijmen, H.; Emmerich, M.; IJzerman, A. P.; van Westen, G. DrugEx v2: De Novo Design of Drug Molecule by Pareto-based Multi-Objective Reinforcement Learning in Polypharmacology. *ChemRxiv.14474127.v2* **2021**,
- (238) Reker, D.; Schneider, G. Active-learning strategies in computer-assisted drug discovery. *Drug Discov. Today* **2015**, *20*, 458–465.
- (239) Walters, W. P.; Murcko, M. Assessing the impact of generative AI on medicinal chemistry. *Nat. Biotechnol.* **2020**, *38*, 143–145.

- (240) Sambasivan, N.; Kapania, S.; Highfill, H.; Akrong, D.; Paritosh, P.; Aroyo, L. M. “Everyone wants to do the model work, not the data work”: Data Cascades in High-Stakes AI. proceedings of the 2021 CHI Conference on Human Factors in Computing Systems. 2021; pp 1–15.
- (241) Singh, G.; Schulthess, D.; Hughes, N.; Vannieuwenhuyse, B.; Kalra, D. Real world big data for clinical research and drug development. *Drug Discov. Today* **2018**, *23*, 652–660.
- (242) Deng, J.; Hou, W.; Dong, X.; Hajagos, J.; Saltz, M.; Saltz, J.; Wang, F. A Large-Scale Observational Study on the Temporal Trends and Risk Factors of Opioid Overdose: Real-World Evidence for Better Opioids. *Drugs-Real World Outcomes* **2021**, 1–14.
- (243) Deng, J.; Wang, F. An Informatics-based Approach to Identify Key Pharmacological Components in Drug-Drug Interactions. *AMIA Jt Summits Transl Sci Proc* **2020**, *2020*, 142.
- (244) Jiang, D.; Wu, Z.; Hsieh, C.-Y.; Chen, G.; Liao, B.; Wang, Z.; Shen, C.; Cao, D.; Wu, J.; Hou, T. Could graph neural networks learn better molecular representation for drug discovery? A comparison study of descriptor-based and graph-based models. *J. Cheminformatics* **2021**, *13*, 1–23.

Graphical TOC Entry

Some journals require a graphical entry for the Table of Contents. This should be laid out “print ready” so that the sizing of the text is correct.

Inside the tocentry environment, the font used is Helvetica 8 pt, as required by *Journal of the American Chemical Society*.

The surrounding frame is 9 cm by 3.5 cm, which is the maximum permitted for *Journal of the American Chemical Society* graphical table of content entries. The box will not resize if the content is too big: instead it will overflow the edge of the box.

This box and the associated title will always be printed on a separate page at the end of the document.