Deep Learning in Chemistry

*Adam C. Mater, Michelle L. Coote**

ARC Centre of Excellence for Electromaterials Science, Research School of Chemistry, Australian National University, Canberra, Australian Capital Territory 2601, Australia

ABSTRACT

Machine learning enables computers to address problems by learning from data. Deep learning is a type of machine learning that uses hierarchical recombination of features to extract pertinent information, and then learn the patterns represented in the data. Over the last eight years, its abilities have increasingly been applied to a wide variety of chemical challenges, from improving computational chemistry, to drug and materials design, and even synthesis planning. This review aims to explain the concepts of deep learning to chemists from any background and will follow this with an overview of the diverse applications demonstrated in the literature. We hope that this will empower the broader chemical community to engage with this burgeoning field and foster the growing movement of deep learning accelerated chemistry.

INTRODUCTION

Deep learning has emerged as a dominant force within machine learning over the last ten years through a series of demonstrations of its frequently superhuman predictive power[1-7]. These initial demonstrations have fostered a desire among researchers to harness its abilities to address challenges in a diverse range of areas. Chemistry stands as one of these areas, with a variety of immensely complex problems such as retrosynthesis, reaction optimization, and drug design. Historically, these have presented fierce opposition to computational approaches based on hand coded heuristics and rules, with these approaches being met with skepticism by chemists[8-11]. There are strong analogies between these problems and those which deep learning has come to dominate, such as computer vision and natural language processing[12]. As a result of this, chemistry has seen a steady increase in the deployment of these technologies, with many demonstrating significant improvements in predictive accuracy and ability to replicate human decision making[13-24].

With the prevalence deep learning is likely to achieve within chemistry, it is important that chemical researchers not familiar with the minutiae of deep learning become comfortable with how these techniques function. There have been a number of reviews covering subfields of deep learning in chemistry. Goh et al.'s[14] review serves as an excellent overview for theoretical chemists and has accessible explanations of the core deep learning concepts. While not strictly a review, Wu, Ramsundar et al.'s[13] paper on MoleculeNet provides an extensive summary of the available descriptors and datasets as well as model comparisons. In addition to this there are a number of broader reviews covering machine learning for drug design[25-26], synthesis planning[11], materials science[27], quantum mechanical calculations[28], and cheminformatics[29]. This paper seeks to adopt a central stance on deep learning in chemistry, explaining the core ideas in the broadest possible sense, without emphasis on mathematical detail, and with reference to chemical applications. This

understanding will then be used to provide a broad overview of the influences deep learning has so far had across applied and theoretical chemistry.

THE BIG PICTURE

Machine learning is an extremely broad sub-field of artificial intelligence that aims to solve the problem of computers learning from data. Representation learning is a subset of machine learning in which computational models learn internal representations of objects that inform the decisions or predictions that they make. Finally, deep learning is a subset of representation learning in which multiple layers of internal representations, initially of simple shapes such as edges, are combined to form increasingly complex objects, like faces[30]. Chemistry stands as an exemplar of this phenomenon, with the behavior of molecules determined not simply by atoms, but their immediate grouping into functional groups, followed by interactions between these groups at increasing ranges. Ostensibly, this makes chemistry an ideal candidate for these methods. Unfortunately, molecules also supply a set of challenging problems including sampling sufficiently diverse molecules and their accompanying conformational space, effectively representing molecules, and obtaining suitably large datasets.

Understanding how these problems are being addressed requires an introduction to the methods of deep learning. Machine learning, and thus deep learning, at its core contains three components: the data (and its associated representation), the model that will learn to interpret the data, and a prediction space from which we draw utility. The model in deep learning (as well as other methodologies) represents an optimization cycle of three sub-components: the learner, evaluation, and optimization. These ideas are summarized in Figure 1. Understanding chemical deep learning requires familiarity with each of these ideas and the unique challenges chemistry presents in each. The first section of this review seeks to disambiguate these topics, beginning with an exploration

of data and how molecules are represented. This leads into a discussion of three of the dominant model architectures in chemical deep learning. The prediction space will then be examined, to explain how chemical problems must be phrased in order to make them amenable to deep learning. This section will conclude with a brief overview of terms that are frequently referenced in the literature.

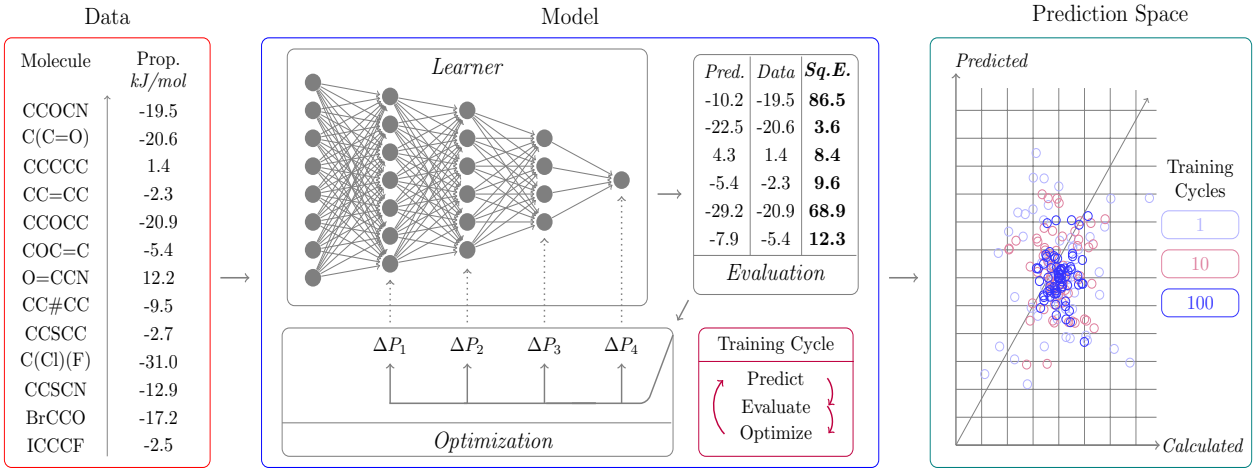

Figure 1 - **The Big Picture of Deep Learning**. The learner shown in this image is a deep feedforward network, however this same procedure applies to a plethora of learners. The ΔP term indicates the change to the parameters in each network layer after the input layer. The data in this image is fictitious and thus labelled simply as property.

The Data. Learning cannot happen without data, and in the case of supervised learning, this data must be labelled. These labels indicate the ground truth associated with the data point, such as associating a label of 'truck' with an image of a truck. In a chemical sense, the data can be a representation of a molecule with its free energy of solvation labelled or any other property. This creates one of the first big challenges of deep learning, how can enough data be obtained? The most dominant demonstrations of deep learning's potential are in fields where data is abundant, typically where millions, if not billions, of data points can be obtained through distributed

collection via social networks or even more broadly, the internet[1, 31]. In the case of science, the requisite volume of data only exists in certain applications. In chemistry, all levels of data are present, with extensive data available for successful reactions or ground state energies, a moderate amount of data for specific properties such as ionization energies, through to relatively small databases for properties such as free energies of solvation[32-34]. As a result of this need for data, chemical deep learning has formed a strong link with computational chemistry due to the latter's capacity to generate huge volumes of data significantly faster than it could be obtained in a laboratory[33, 35]. This presents challenges however, due to the poorer accuracy of these calculations relative to experimentally obtained results. Lab-derived datasets are available, and are the gold standard, but aside from reaction databases, the number of data points they contain is not usually on the same order of magnitude[36].

Additionally, effective assessment of deep learning models requires that the data undergoes subsequent splitting. Assessing a model on the data it was trained on leads to significant overfitting in which the model learns to reproduce that specific set of data but not the trends underlying it. To stop this 'memorization' of data, it is common to test the models on data that they have not yet seen. This is typically done by dividing the data into three separate sets: the training, validation, and test sets. The training set (typically 60-80% of the data) is given to the network in its entirety and its labels are used to adjust the network's parameters in supervised learning. The validation set (typically 10-20% of the data) is used to ensure that the model is not overfitting by providing a constant estimate of its performance on unseen examples. In addition to this, when training multiple models validation data is used to identify the best performing model. Finally, the third dataset, the test set, is used as the final performance evaluation of the chosen model on the remainder of the withheld data. In order to remove any bias in the partitioning of the data into these

sets, k-fold cross validation is used, in which the data partitioning process is randomized k times[37]. Any model is highly dependent on the way in which the data is represented. Due to this, deep learning has a strong interest in the long-standing cheminformatics problem of how best to **represent chemical structures for a computer**.

There are three key invariances that must be captured, two of which are intuitively captured by the human visual processing. Formally, these are:

- **Permutation invariance** – the representation must be unaltered by a change in the specified order of the atoms.
- **Translational invariance** – the representation must not be changed by a translation in space.
- **Rotational invariance** – the representation must be unchanged by a rotation operation.

Familiar examples of these variances are shown in Figure 2 below. An additional requirement for some models is a fixed size input. This is typically achieved by padding the representation with zeros for smaller molecules.

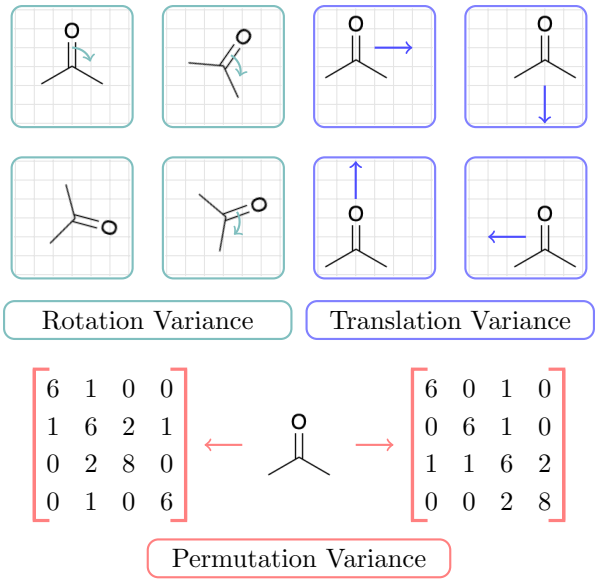

Figure 2: **Three key variances in common molecular descriptors that much be overcome for deep learning**. The top two invariance grids show acetone undergoing rotation and translation in a fixed reference grid. Permutation invariance shows two equivalent acetone representations as atom connectivity matrices introduced by Spialter[38]. The atom connectivity matrix has nuclear charges listed along the diagonal, with off diagonal elements representing bonds of associated bond order between the diagonally located atoms that they link. To facilitate the following discussion of model architectures, a brief exploration of the most widely used molecular representations is required.

A **molecular graph** is a set of vertices (atoms) that are connected by edges (bonds). This can be expressed in matrix form, with an example shown in Figure 2. Originally, deep learning models utilized extended connectivity fingerprints (ECFP). These involve assigning an integer identifier to each atom and updating it to include information from neighboring atoms by expanding a circular radius that analyzed the atoms contained within. Within this circle, the atoms were sorted to achieve permutation invariance and, by compressing spatial information into integer identifiers, the two spatial invariances were also satisfied. Each of these integer identifiers were passed

through a hashing function to produce a number, which, combined with modulo arithmetic, allowed a particular index within a fixed vector to be switched to a one[39]. This vector has a fixed size, achieves the three invariances, but contains only zeroes and ones and is thus referred to as a bit vector. This is the basic methodology that inspired the molecular graph-based models that will be described below. The idea of gathering information about an atom's local environment while preserving their invariances was retained, but critically, they encode the molecular information in a real valued vector allowing for significantly richer information to be embedded.

The **Simplified Molecular Input Line Entry System (SMILES)** is a classic cheminformatics representation that uses a set of ordered rules and specialized syntax to encode three dimensional chemical structures as strings of text[40-41]. An additional procedure can be applied on top of this to create permutation invariance, a process known as canonicalization. The other frequently used text-based identifier, the international chemical identifier (InChI), is not regularly used in deep learning due to multiple studies finding that its more complex and numeric formulations lead to deterioration in predictive performance[42-43]. A reaction variant of SMILES, which contains specialized grammar to describe chemical transformations is also frequently used in machine learning for models that operate on reaction datasets[44-45].

Graph inputs currently dominate due to their ability to extract higher-level features, and the increase in predictive performance that comes with this. It must also be noted that there are additional representations such as point clouds[46] and Coulomb matrices[47] that are also used. Finally, regardless of representation, molecules must be entered into datasets order to be transformed into a model input. To digitize the enormous number of structures in the literary corpus, deep learning has been used to automate the digitization of these structures[48].

The Model. In any given deep learning framework, the model is the component that transforms the data into a prediction, classification, or action. The model relies on an interplay between its learner, evaluation, and optimization. The learner contains a set of parameters which define how each input point is converted into an output. This prediction is then quantitatively compared to the desired output via an evaluation or cost function. Finally, optimization alters the parameters of the model to decrease the difference between the predicted and the desired output for each data point. This cycle of the model making predictions, which are then evaluated, and finally used to optimize the model's parameters is bundled into a single training cycle. These ideas are summarized visually in Figure 1.

Deep learning is named for to the computational depth of its learner, i.e. how many sequential layers of calculations are required. The learner is thus the defining feature of deep learning methods, with an intimate link being formed with the field of connectionism. Connectionism is focused on the development of artificial neural networks (ANNs) and their many variants. These learners are neurologically inspired systems of interconnected virtual neurons (an example network is shown as the learner in Figure 1). Due to their prominence in deep learning methods, the remainder of the model discussion will focus on variants of ANNs. A mathematical discussion is not the intent of this review, however much of this discussion is inspired by *Deep Learning* by Goodfellow et al.[30] which contains an extensive and rigorous treatment of deep learning methods.

Despite the enormous diversity in the learner architecture, the evaluation and optimization procedures are dominated by a few methods. In the case of neural networks, the evaluation step is typically a simple function that assesses the learner's performance across batches, or all, of the data; two common examples are the root mean squared deviation (RMSD) or the cross-entropy

cost function. The optimization typically employed for neural networks is the powerful backpropagation algorithm[49]. This method propagates the gradients backwards from the outputs through to the inputs, and using the information contained within these, alters the parameters of each non-input node in a manner that lowers to deviation between the predicted and true values[49]. To highlight what makes the learner networks so different, three of the dominant architectures will now be discussed.

A **deep neural network (DNN)** is the prototypical deep learning architecture. DNNs contains three separate layer types, input, hidden and output. Each layer is comprised of a set of neurons and in fully connected systems, each hidden layer neuron connects to all neurons in the previous and following layers. The 'wiring' of the network (how many layers there are and how they are connected), as well as what function each neuron performs is typically referred to as the network's topology, and the performance of the network is highly dependent on the chosen topology.

Each neuron in the input layer receives a single, real number from each data point and is thus represented as a fixed size vector. DNNs were frequently used with ECFP representations, in which a one indicates the presence of a particular substructural feature which may or may not correspond to a recognizable function group, and a zero its absence[39].

The neurons within the hidden and output layers have two types of trainable parameters. Every incoming connection has a scalar weight associated with it, that is expressed within a matrix, and then each neuron has its own scalar term called a bias, collected into a vector for each layer. The forward data pass is computed by multiplying the input vector with the weight matrix, to produce an output vector. The bias is then added to this output vector, and it is then passed through an activation function. This function is critical as it allows the network to model nonlinear phenomena. One of the simplest and most widely used activation functions is the rectified linear

unit (ReLU)[50], which simply maps any non-positive number to zero and returns any positive number unchanged. This vector now becomes the input for the next layer of the network and the process continues until the output layer is reached.

The output layer is typically either a single real number, indicating that the network is built for regression (i.e for predicting a property such as the enthalpy of combustion), or a vector that contains the likelihood of the input being classified as certain objects, and thus a classification network. In the case of classification tasks, the softmax activation function is commonly used; it converts a vector of real numbers into a probability distribution where the sum of all terms is one and all terms are between zero and one. This allows the network to produce a distribution over the classes, indicating which is most likely. The utilization of matrix operations allows these models to leverage graphical processing units (GPUs) to massively accelerate the computation[51]. A summary of this matrix multiplication process is given in Figure 3.

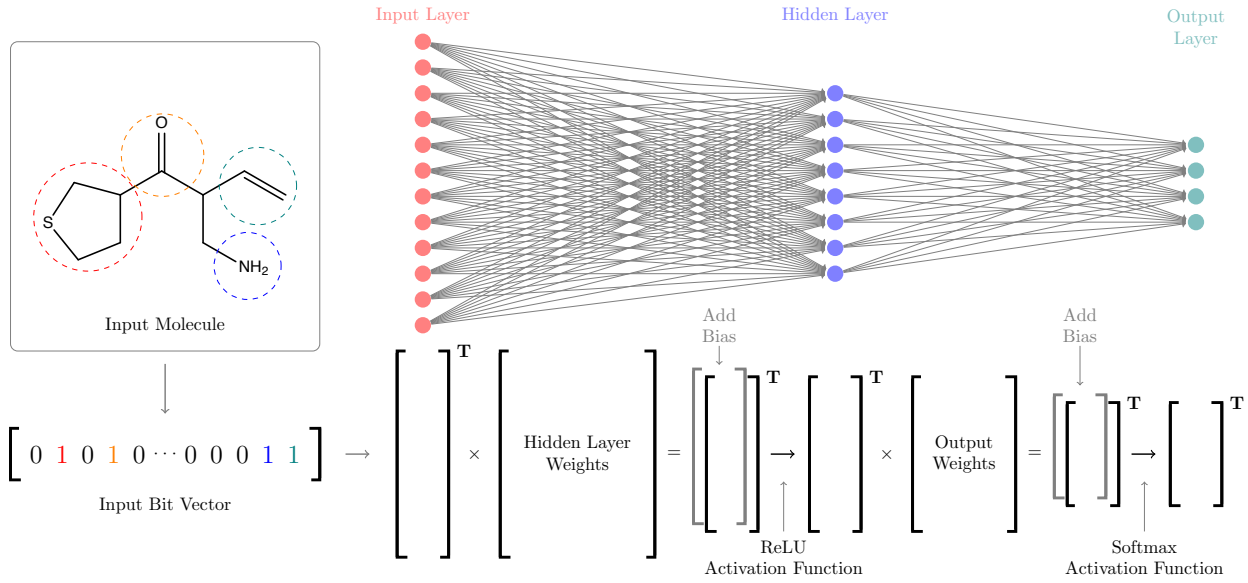

Figure 3: **Matrix view of a typical neural network forward pass:** The input molecule was chosen at random, and the bit vector is a simple structural representation that can roughly be viewed as ones indicating the presence of certain substructural feature, and zeros representing the

absence. The bold T's above the vectors indicate that the transpose is used in the multiplication in order to make the operation defined.

Learning in these networks involves the backpropagation algorithm, which applies the multivariate chain rule from calculus to efficiently calculate the gradients of each trainable parameter in the network, and then uses these to alter the parameters in a way that lowers the cost function. DNNs have been effective at addressing chemical problems. However, other deep learning architectures that evolved in two of AI's largest research areas, computer vision (CV) and natural language processing (NLP), have largely superseded them.

Graph convolutional neural networks (GCNN). Computer vision is the field of research that aims to use computers to see in a manner similar to humans. Convolutional neural networks (CNNs) are networks specialized for interacting with grid like data, such as a 2D image. As molecules are typically not represented as 2D grids, chemists have focused on a variant of this approach: graph convolutional neural networks (GCNN) on molecular graphs.

Molecular graphs confer key advantages: they bypass the conformational challenge of using 3D representations while maintaining invariance to rotation and translation due to their pairwise definition. A wide variety of molecular graph implementations have developed in recent years[18, 22-23, 52-55] and the MoleculeNet paper by Wu, Ramsundar et al.[13] offers a concise conceptual comparison of six major variants. To facilitate the following explanation, the framework of neural message passing networks put forth by Gilmer et al[53]. will be used.

Neural message passing networks are a chemically motivated system to understand and compare these GCNN systems. Fundamentally this approach utilizes a convolutional layer, simply a matrix of scalar weights, to exchange information between atoms or bonds within a molecule and produce

a fixed length, real-valued vector that embeds the molecular information. To begin, they generate or compute a feature vector for each atom within the molecule; this can contain information such as how many hydrogens are attached to the atom, its hybridization, whether or not it is aromatic or in a ring, etc. These feature vectors are then collected into a matrix. Additionally, they generate a graph topology matrix that specifies the connectivity of the graph, similar to Figure 2 although often without bond order or atomic number along the diagonal. In a forward convolutional pass, these three matrices are multiplied together. This allows information to be exchanged between the feature vectors of each atom with its immediate neighbors, in accordance with the connectivity specified by the topology matrix. This updates each atom's feature vector to include information about its local environment. This updated feature vector matrix is then passed through an activation function (i.e. ReLU) and can then be iteratively updated by using it as the feature matrix in another convolutional pass. This propagates information throughout the molecule. Finally, these atom feature vectors are either summed or concatenated to give a unique, learned representation of the molecule as a real valued vector (see Figure 4). Alternative approaches to generating this learned representation have been put forth, such as using traditional computer vision CNNs on 2D grid images[56] of molecules. However, molecular graphs remain the dominant paradigm.

The learned representation in vector form is referred to as a representation in latent space, and is then used as the input for a traditional fully connected DNN to finally make the classification or prediction. This process of learning its own molecular representation is the cause of it being in the broader class of representation learning methods. Backpropagation is once again used to train these networks by propagating gradients backwards and determining how to change the convolution matrix weights and the parameters in the DNN.

Recurrent neural networks (RNNs), introduced by Hopfield[57] in 1982, are specialized for dealing with sequences of arbitrary length. This makes them ideally suited to handling textual representation of chemical information, such as SMILES[40]. The critical difference is that in the previous architectures each data input is distinct, while in an RNN each input will influence the next one. An illustrative example is viewing any particular input, such as a SMILES string, as time series data. The presence of a carbon atom at one moment in time influences what the next character is likely to be. This is expressed in the architecture by feeding the output of the hidden layer for that carbon into the hidden layer of the next atom. As a more complex illustration, this process can be used to model reactions by utilizing the SMILES reaction strings to encode the information, and train the network to predict the product (See Figure 4). The feeding of one hidden state into the next gives the system a recursive relationship within the hidden layer, but it can be viewed as directional by 'unfolding' the network to form of an unfolded, acyclic network graph. By doing this, it maintains a history of all previous inputs, and they influence its prediction at a later time. The network can then be trained using a recursive form of backpropagation[58]. This is the simplest RNN but more sophisticated and powerful variants such as neural Turing machines[59] and long-short term memory networks (LSTM)[60] that incorporate memory into the network are the current leaders. This ability to use previous information has led to their dominance in sequence-based tasks such as machine translation, as previous words define the context and thus, what the next word is likely to be.

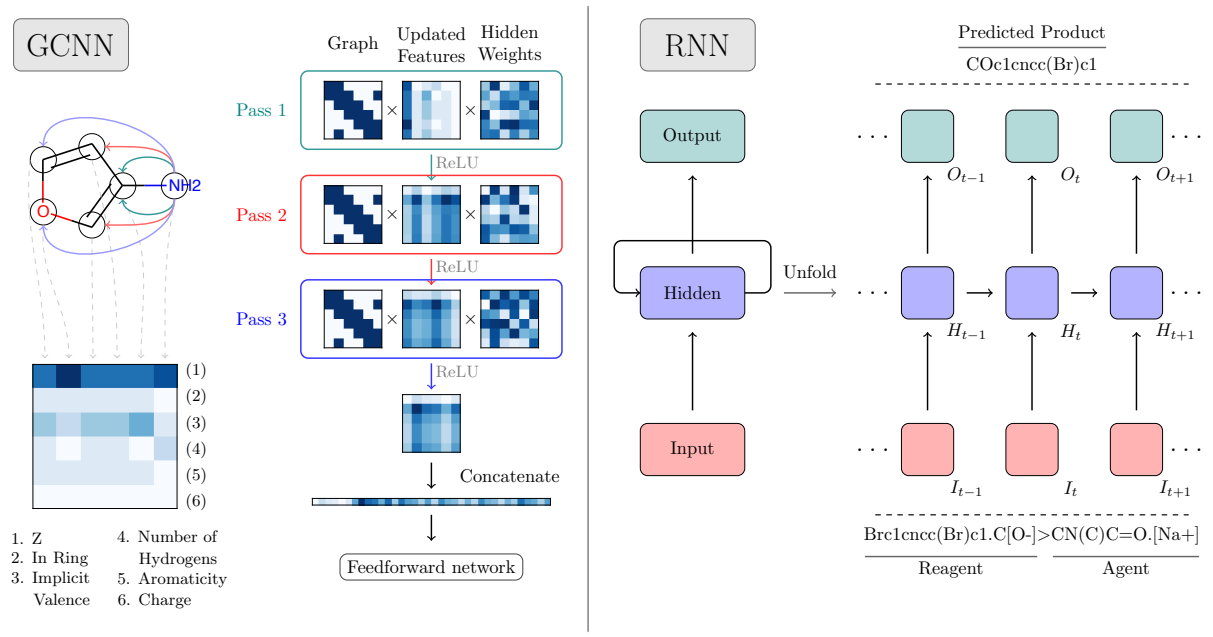

Figure 4**: Illustration of the GCNN and RNN architecture for chemical applications**. Colored arrows stemming from the amine group indicate the information transfer from the nitrogen to other heavy atoms, with the color corresponding to the convolutional pass. Light grey arrows indicate each atom's feature vector in the matrix, importantly properties such as atomic number (Z) are often encoded using one hot vectors, which are binary, but for spatial efficiency the integer is used in its place. The RNN model shows a simplified 'many to many' recurrent network, with the text above and below the dashed lines indicating a stylized reaction prediction system inspired by the work of Schwaller et al.[61] This system takes in reagent and agent SMILES, and predicts the variable length product string, however the LSTM architecture they used is significantly more complex than the one shown above.

The prediction space is the set of all possible outputs for the network. More intuitively, it can be thought of as the utility of the network or the question that the network can produce an output for. As discussed above, supervised learning requires labelled data that allows the model to iteratively improve its predictive performance. This model relies on a quantitative error assessment by the evaluation component, and thus each deep learning problem needs to be framed in such a way that

it can be quantitively evaluated. This creates a significant challenge in chemistry, as questions such as 'what is the best synthetic route?' require systematic analysis to produce a question that can be numerically evaluated, and thus produce quantitively labelled data. In the broader context of artificial intelligence, this means that these systems are weak AI, capable of solving only a single, extremely narrow task, and not being capable of meaningfully answering even slight deviations from the question it was trained on.

Commonly Used Terms. Before concluding this section, a brief explanation of commonly used ideas and terms will be provided. Each term is linked to seminal papers and, where appropriate, accompanied by an example of its application in chemistry.

- **Transfer Learning** – Transfer learning involves using a network that has been trained on a related task, and then tweaking its parameters to adapt to a new task, often with less data[62]. It has been used to adapt a model trained on DFT to a smaller database of higher fidelity calculations by Smith et al.[63].
- **Multitask learning** – This involves training a model on multiple prediction tasks at the same time to decrease the likelihood of overfitting[64]. It has been demonstrated improvements for toxicity or bioactivity prediction[65].
- **One Shot Learning** – A technique used to overcome applications with extremely limited data that uses networks to compress inputs into a continuous latent space and then compares the representation in this space to a larger, trained latent space[66]. It has been used in chemistry for low-data drug discovery[67].
- **Autoencoders** – A network architecture which compresses an input to a real valued vector, commonly referred to as the latent space. A decoder network then takes this vector as its

input and tries to reproduce the original input data[68]. It has been used to design molecules by training the latent space to reflect a particular property, and then navigating it[43].

- **Generative Adversarial Networks (GANs)** – GANs utilize two networks in a competitive scheme. One network has to generate data, and another has to determine if a particular data point is a fake generated by the network, or a real one from the dataset. By competing with one another, the generating network learns to create high quality imitations of the dataset[69]. It has been utilized for the inverse molecular design problem[70].
- **Data Augmentation** – This involves expanding a dataset by creating new training examples through reasonable manipulations of the data. One of the simplest demonstrations of this is rotating images in a dataset, but maintaining the same label in a way that is obvious to humans, i.e., a car is still a car at different angles[71]. This has been used with SMILES to enumerate the different potential orderings and increase the predictive performance[72].
- **Reinforcement Learning** – When the model learns iteratively through trial and error by making its cost function measure its progress towards a particular goal[73]. It has been used to train a model to optimize reactions[74].
- **Supervised Learning** – Supervised learning involves giving the model a labelled dataset, effectively telling it what it needs to learn. While this is currently the dominant learning paradigm in machine learning, it is not representative of how humans tend to learn.
- **Unsupervised Learning** – Unsupervised learning is learning in which the model is not told what to reproduce and instead tries to separate the data into its underlying clusters. Algorithms such as k-mean clustering fall into this category and it is much closer to how humans learn[75].

DEEP LEARNING APPLICATIONS

This section reviews the multiple areas of chemistry that deep learning has thus far impacted, presenting examples in each that highlight particular achievements. To create a logical narrative, this discussion will follow an idealized chemical workflow. To build a molecule with a particular property would first require developing methods to accurately correlate any given structure to the property. These can then be used to intelligently design a molecule that maximizes the desired property. The final step is to design an efficient synthesis from readily available starting materials (Figure 5). This creates a closed feedback cycle in which the synthesized molecule can be experimentally analyzed, and this information can then improve the models that link molecules to properties. Deep learning has influenced every stage of this workflow, beginning with understanding molecules.

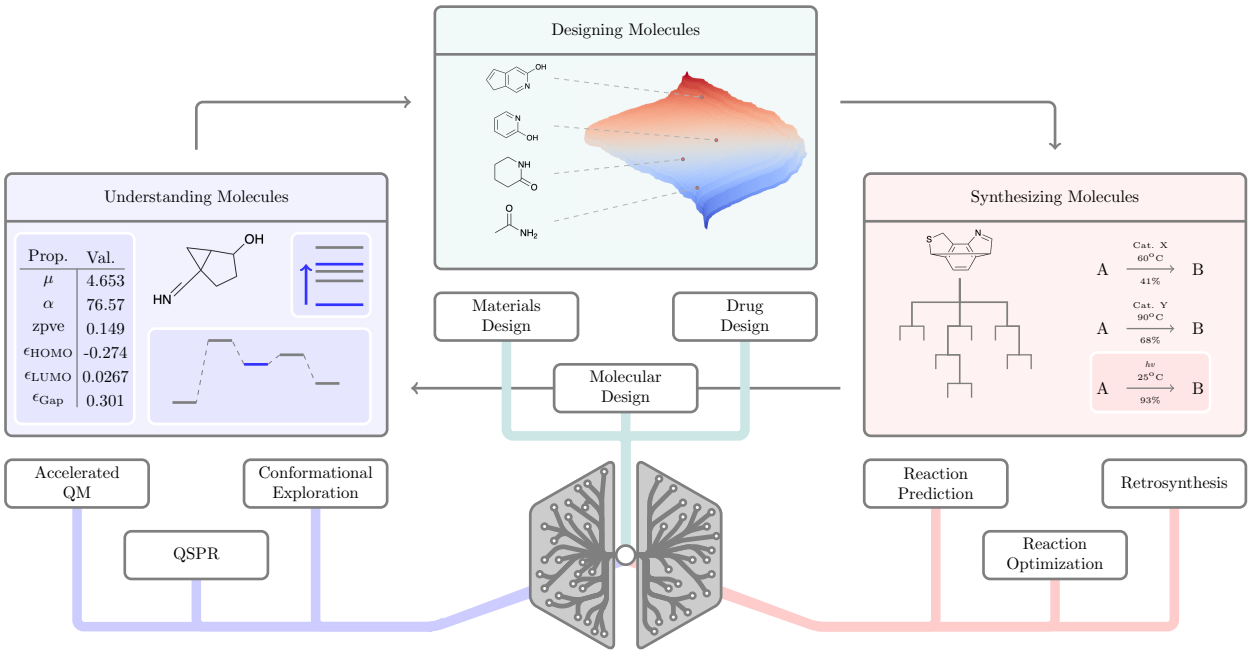

Figure 5: **Deep learning influence on the idealized chemical workflow**. Illustrative examples of each task are shown in the dialogue boxed with arrows indicating the closed cycle that is contained

within the framework. The property values in the blue panel were obtained from the QM9 dataset for a randomly chosen molecule[33].

Accelerated Computational Models. Computational modelling in chemistry seeks to use physics-based calculations to determine the properties and behavior of a given molecular system. There are two distinct ways which deep learning can be used within this space. The first is to integrate the deep learning method with physics style approaches to alleviate computational bottlenecks. The second is directly predicting properties from molecular structures, thereby by-passing physical laws altogether.

Integrating deep learning methodologies with physics-based approaches involves training the network to predict a key component of the overall calculation. These include using the deep learning model to predict potential energy surfaces[52, 76-77], force fields[78], add corrections to ab initio calculations[79] and to bypass expensive stages in both density functional and wavefunction methods[80-81]. There is an excellent review and tutorial on using neural networks for the prediction of potential energy surfaces by Behler[82-83]. Many of these methods adapt a method introduced by Behler and Parrinello[84] in 2007 that determines the energy of the system by summing the energetic contribution of each atom. This method transforms the cartesian coordinates of a molecule using radial symmetry functions, which capture the information of each atom's immediate environment. This transformed representation is then passed through a neural network that predicts the contribution of this atom to the total energy. This general method of using functions to capture an atom's local environment, then predicting its energy through a network and finally summing these contributions has been refined in a variety of ways. Notable work in the field includes that of Schutt et al.[24] which produced size extensive predictions with an average error of 1 kCal/mol, and

the work of Smith et al.[52] which produced errors below 1 kCal/mol and generalization to larger molecules. Schutt et al.'s[85] work has been further refined, and developed into an open source software package (SchNetPack) that can be used to predict properties.

The advantages of this approach are that it is more flexible than mapping a structure to a property, and it is more interpretable due to its physical basis. The difficulty is that, as there are typically still physics-based calculations involved, such methods cannot achieve the same speed as those that map purely from a structure to a particular property. It is important to note that there is a large literature base for using kernel ridge regression as the ML method. This approach has achieved excellent results but is not a deep learning method, and thus is outside the scope of this review. For an overview of these methods the reader is referred to von Lilienfeld's excellent review[86].

Quantitative Structure Property/Activity Relationships. The alternative approach to deep learning in computational chemistry is training a direct map from a simple representation of the molecule through to the desired property. This is a diverse field of research that can broadly be captured under the two fields of quantitative structure property relationship (QSPR) and quantitative structure activity relationships (QSAR). Broadly speaking, QSPR seeks to predict properties of molecular systems, such as thermochemistry, while QSAR seeks to predict the activity of that molecule within a broader context, such as toxicity within biological systems. The goal of these methods is to maximize accuracy of prediction, with chemical accuracy for QSPR commonly being set to 1 kCal/mol or approximately 4 kJ/mol[87-88]. The properties that can be predicted are entirely determined by the available training data, and there are many databases available. There are summaries of available databases in both the review by Butler et al.[27] and the

MoleculeNet paper by Wu, Ramsundar et al.[13]. Typically for properties that can be readily computed, such as ground state energies, ionization energies, or dipole moments, computational datasets are the norm. These are typically computed with a DFT method in order to maximize speed and allow for as much data as possible to be generated. Some of the most commonly utilized are QM9[33], ANI-1[35], and the Materials Project[89]. Properties that are difficult or currently impossible to compute accurately, such as toxicity, free energies of solvation, biological activity, or binding affinities rely instead on experimental datasets that typically contain significantly fewer entries due to the challenge in obtaining them. Frequently used datasets include ChEMBL[90], PubChem[91], and FreeSol[34].

For this type of problem. DNNs were the most widely used network architecture for the first half of this decade. They have been used to effectively predict electronic properties[19, 87, 92], bioactivity[21, 93-95], toxicity[15, 96-97], reactivity[92], as well as other physical properties[98]. Multitask networks are also frequently used due to the increase in predictive performance, as well as increased robustness to overfitting[15, 21, 93-94]. RNNs have been more widely used as the generative networks that produce novel molecules which will be discussed later. For predictive purposes, however, they have utilized both graph type input structures similar to GCNNs to predict aqueous solubility[18] and drug toxicities[99] as well as the more traditional text based inputs of SMILES for general property prediction[100-101].

In almost all cases however, GCNNs and their many variants have demonstrably better predictive performance than either of the other two classes of methods. Due to the focus on improving network architecture, convolutional models are often tested against a variety of benchmarks. However, there has been a particular push to improve the predictions of electronic properties in order to ease the computational stress imparted by physics-based calculations[56, 102-103].

In addition to this GCNNs have shown dominance in predicting bioactivity[104], polymer property predictions[105], and physical properties[55, 106]. Work to increase their predictive abilities is ongoing, but errors below 1 kcal/mol are routinely achieved. The accuracy of these methods brings into question the validity of the training data, particularly the accuracy of the labels, as well as potential bias in the data. DFT is known to have large errors[107-108], while the gold standard methods such as coupled cluster with singles, doubles, and perturbative triples (CCSD(T)) are currently prohibitively expensive for datasets of this size[109]. In order to overcome this deficit, transfer learning has been utilized to fine tune these networks on smaller datasets of calculations performed at significantly higher levels of theory, such as CCSD(T)[63, 110]. Additionally, bias in chemical datasets is a well-known problem[111-112]. While there has been recent work to intelligently design them using deep learning[113], genetic algorithms[114], or techniques such as query by committee[115], the large datasets required for chemical deep learning are largely restricted to small molecules containing only carbon, nitrogen, oxygen, fluorine and hydrogen. As the coverage of chemical space expands, it is critical that the datasets are intelligently designed to maximize coverage of the rapidly expanding combinatorial space.

The final topic to address is interpretability. Deep learning has a reputation for being a 'black box', as it is almost impossible to understand why the network made the decision that it did[116]. Recent work has attempted to overcome this in chemical deep learning by cleverly designing the architectures to allow for extraction of chemical insights from its decision making. In recent work from Goh et al.[117], by changing the information available to the network in their descriptor, they were able to infer that the network was learning a different approach to solve different chemical prediction challenges. Schutt et al. [102] on the other hand demonstrate not how the network is making

decisions, but rather that its predictions align with an understanding of chemical ideas such as aromaticity.

Conformational Exploration. Regardless of how deep learning influences chemical property mappings, effective exploration of chemical space involves navigating not only the species space, but also the conformational space of those species. Conformational screening is an immense challenge in chemistry, as with each new atom, multiple additional local minima appear on the potential energy surface. The aforementioned neural network potentials offer a rapid way to explore the conformational space of a molecule. The leading potential at the moment is the ANI-1 potential that achieved errors below 1 kCal/mol and is trained using off-equilibrium geometries[52]. The dataset it was trained on contains approximately 20 million energies of ~57,000 molecules in different stances[35].

The inverse of conformational screening is to develop a system that can generate equilibrium conformers for a given molecule. This challenge has been undertaken by Gebauer et al.[118] which demonstrated deep learning's ability to generate equilibrium conformers. This method is an adaption of SchNet architecture developed by Schutt et al. that was able to regenerate molecular geometries with a root mean squared deviation of approximately 0.4 Å. Additionally, a novel, but not as rigorously tested method was introduced by Thomas et al.[46] in which 3D point clouds were used to regenerate molecule geometries. This work didn't place the same emphasis on minima structures, but was able to achieve very low errors of approximately 0.15 Å. This field of research is still very young but holds immense potential to minimize the conformational screening bottleneck.

MOLECULAR DESIGN

The second stage of the idealized workflow is the problem of molecular design. This problem, sometimes referred to as inverse QSPR has a history of machine learning applications including Bayesian optimization[119] and genetic algorithms[120]. Recent years have seen the application of generative deep learning models to design molecules. One of the seminal demonstrations of this method is the work of Gómez-Bombarelli et al.[43] which used an autoencoder with a latent space that was optimized by an additional network to reflect a particular property. This 'landscape' can then be explored to identify candidate molecules that maximize the property. There are many other approaches that also use autoencoders[121-124], generative adversarial networks (GANs)[70], or reinforcement learning agents[125-128] to navigate chemical space structured around a particular property. Finally RNNs have also been used for molecular library generation by an adaptation of their text generation capabilities[129]. An excellent review of molecular design is provided by Sanchez-Lengeling and Aspuru-Guzik[130].

General molecular design is seeing a surge of activity, however, there are two special classes of molecules that deserve particular attention: materials and drugs. These are arguably the two most challenging molecule classes to design and optimize, but also offer the greatest potential benefits. Therefore, they have motivated significant research efforts with deep learning.

Materials Design. Many modern technologies such as batteries, aerospace, and renewable energy relying of advanced materials. Deep learning has only recently begun to influence the field, but there has been a rapid growth in applications in the last few years. The distinction between discrete small molecules and crystalline structures has led to a separate set of convolutional descriptors that seek to capture the crystalline structure. Crystal graph convolutional neural networks (CGCNNs) as introduced by Xie and Grossman[131] show much potential in this field.

There has been a push, however, to reconcile the representation systems for these two classes of molecules. SchNet[103] has been demonstrated on both, and MEGNet by Chen, et al.[132] has been developed for this specific purpose.

GCNNs, as well as the CGCNN variant have been used to predict the properties of bulk materials[133], predict thermoelectric properties[134], optimize polymer properties[135], and to explore chemical materials space[136]. These applications are still young; however, it has moved beyond predictive models as demonstrated in work by Li et al.[135] in which they successfully used reinforcement learning to train an agent to experimentally control polymer weight distributions, and thus the polymer's properties. Additionally, the exploration of chemical materials space by Xie and Grossman[136] demonstrated the potential of these methods to uncover previously undetected pockets of materials space. Beyond the properties of materials, work has been done to optimize their synthesis parameters[137] and perform defect detection[138]. Finally, a deep learning method that utilizes tensor networks, similarly to Schutt et al.[24], demonstrated generative design of chiral metamaterials[139]. Most of these applications remain theoretical in nature, and effectively incorporating them with an experimental workflow, such as in the polymer optimization workflow of Li et al.[135] is a key next step to determine their efficacy.

One key subfield of materials design is catalysis design. Machine learning has seen increased use in catalytic research[140-141], however deep learning has seen limited application in this field due to the limited data available, the unique nature of each catalytic process, and the difficulty of representing multimolecular systems. The applications of deep learning within catalyst design largely center around using neural network potentials to model the catalytic system. Recent examples of this include Shakouri et al's.[142] work to model nitrogen gas on a ruthenium surface and the optimization of platinum clusters by Zhai and Alexandrova[143]. Extending this work beyond

using neural network potentials will likely require increased data gathering efforts, as well as the development of newer descriptors to describe interacting, multimolecular systems.

Drug Design. Drug design is arguably one of chemistry's most important applications. Fundamentally it involves identifying molecules that achieve a particular biological function with maximum efficacy. These can either be obtained from natural sources or built from the ground up. In either case the goal typically starts with one, or a set of molecules and the challenge is to optimize its properties to improve potency, specificity, decrease side effects, and decrease production costs. There are a number of reviews on deep learning's impact on this field as it is of great interest to the community[25, 144-145].

The generative models in drug design follow the same trends as general molecular design, with autoencoders[146], GANs[147], and reinforcement learning[148] all being used to try and generate potent drug molecules. In addition to these, there are some novel approaches to drug development rather than molecules design that include predicting anti-cancer drug synergy[149] and developing a benchmarking for generative models in drug design[150]. Drug design approaches struggle from limited data, possibly more so than any other fields due to the expense of obtaining it. Work by Altae-tran et al.[67] utilized one shot learning to address this deficiency and make informed predictions about drug candidates with limited data. Finally, while not a molecule optimizing generative system, work by Segler et al.[151] developed methods to develop focused libraries of drug candidates for screening using RNNs.

SYNTHESIS PLANNING

Synthesis planning is the final stage in this idealized workflow. It can be simplified into three separate components. Retrosynthesis, in which the product is known, and is broken down into a

series of simpler starting materials from which it can be made. Reaction prediction, in which reagents are known, and the dominant product must be determined. Finally, reaction optimization, which involves taking a reaction with known reagents and products and trying to maximize the yield or efficiency of this process. One important distinction to note here is that reaction optimization and reaction prediction both have well established computational approaches, kinetic models and quantum calculations respectively. Both of these can however be expensive, and in the case of quantum calculations, enormously so.

Computational retrosynthesis on the other hand has a long and turbulent history. The original retrosynthesis program was Pensak and Corey's[8] work on the LHASA software. From this point there have been a multitude of assistive software packages[152-154]. The beginning of the 21st century saw a loss of interest in this field due to a variety of factors, but it is largely attributed to a widespread belief that computers could not capture the art of synthesis. This field has had a second wind with the advent of deep learning, with the models beginning to challenge the notion of computational inferiority in synthesis planning[10].

Retrosynthesis. The great challenge of retrosynthesis is the exponential scaling of possible moves in synthetic space from any point. This is a property it shares with traditional board games such as Chess or Go. Formally, this can be expressed as a tree search, where the branching factor is how many possible steps you can take from a particular point. The depth is how many steps it takes to reach the desired position. Compared to the aforementioned games, retrosynthesis has a significantly greater branching factor, but lower depth[155]. Retrosynthesis may present a far greater challenge due to the immense challenging in knowing a priori whether a reaction will be successful and produce the desired material, whereas Chess and Go have a perfectly defined set of possible

moves. However, these games represent a good starting point to consider the problem, and fortunately, both have succumbed to artificial intelligence approaches. It is not surprising then that one of the dominant displays of retrosynthetic AI was heavily inspired by AlphaGo, the seminal AI system Deepmind developed to achieve superhuman Go playing ability[3].

Work by Segler et al.[16] adapted the AlphaGo methodology (Monte Carlo Tree Search with deep neural network policy) to design a state of the art retrosynthetic AI. This system was trained on over 12 million reactions from the Reaxys[156] database and produced human accepted synthesis routes. Assessing synthesis plans is a thorny challenge, and in order to do this, they performed a double-blind study in which graduate chemists were shown the machine's synthetic plan and the original, literature plan. There was no statistically significant difference in their preferences, thus giving a preliminary indication that its synthetic routes are 'human level'. It is also possible, however, to argue that the graduate chemists' do not yet have the necessary expertise to distinguish the human route. Thus, determining when computers achieve human ability in synthesis planning is a decision that can only be made by the entire field. While this method showed great potential, there are other avenues of research such as the use of RNNs in an encoder/decoder setup to perform retrosynthetic analysis of small molecules[157].

Computational retrosynthesis is making enormous strides; however, many problems persist. Firstly, planning a retrosynthesis that looks valid, and experimentally verifying its predictions are different challenges and until these methods are rigorously tested it is unknown whether or not they are useful to chemists. This challenge would likely benefit from a user-friendly software package in order to get chemists' feedback on the computer-generated syntheses. These are beginning to appear with an example being the ASKCOS software developed by the machine learning for pharmaceutical discovery and synthesis consortium[158].

Reaction Prediction. Reaction prediction is the process of taking a set of known reagents and conditions and predicting what products will form; as such, it typically requires greater exploration into uncharted chemical space. Current methods to perform this, such as quantum calculations are exceedingly expensive and thus limited to smaller molecules. Deep learning methods represent an opportunity to alleviate this computational expense, and free up time of trained computational chemists.

Reaction prediction exemplifies the challenge of predicting outliers, due to the frequent need to predict outside of the training space. As a result of this, the majority of reaction prediction machine learning methods either integrate the model with a physics based scheme or apply reaction templates[159]. One of the early works that applied deep learning to reaction prediction involved DNNs with molecular fingerprints to predict what product would form[44]. Additional work has utilized RNN variants[61], as well as more specialized architectures such as neural machine translation[160-161] and Siamese architectures (which take two identical networks given different inputs and determine the similarity between them)[17]. One of the striking challenges for this field is the immense literature bias towards successful reactions. Recently Coley et al.[45] presented by a clever approach to overcome this by recognizing that a successful reaction implicitly defines a large number of unsuccessful reactions that can be added to the database. This was performed by identifying high yielding reactions, and generating viable alternative products that are thus not formed in high yield. These can then be added to the dataset to augment it with negative examples. The current state of the art that also stresses interpretability uses a GCNN to predict reaction outcomes in a manner similar to human intuition[162].

Due to deep learning's relatively new arrival to reaction prediction, there is a history of non-deep learning methods for reaction prediction that is reviewed by Coley et al.[11]. Current developments are reaching a level that is competitive with humans. With further advancements in predictive ability and transitioning it into user friendly software, this is likely to become a key addition to the chemical toolset.

Reaction Optimization. Reaction optimization involves tuning the conditions of a reaction to increase its efficiency. This is often performed via kinetic models, or experimentally through the use of flow chemistry or high throughput combinatorial chemistry. Despite the maturity of these methods, there is scope for a system which can rapidly produce idealized synthetic conditions given a molecule and reaction type. Deep learning has the potential fill this niche, and research has begun to adapt it to this challenge.

The potential of this approach was demonstrated by Zhou et al.[74] in which an RNN variant learned to optimize the conditions of reactions. Their model used an RNN that learned to evolve the conditions of a reaction towards an optimized state. It was trained on simulated reactions and then outperformed other software-based approaches for multiple experimental reaction setups. It is important to acknowledge here that due to limited availability of data, and the need to flexibly update the model, deep learning methods may not be the best choice here, instead a method that uses alternative machine learning methodologies such as random forests has been demonstrated to be a potent alternative[163].

FUTURE DIRECTIONS

To summarize, deep learning is a subfield of machine learning that uses subsequent layers to extract higher level features and use them to learn the patterns present in a dataset so as to predict future behavior. Supervised learning requires large volumes of labelled data and a quantitatively

assessable goal or question. With this, a model uses an interplay of a predictive learner, evaluation, and optimization, in the form of a training cycle to iteratively improve its performance until it begins to overfit the training set, at which point training stops and the model is evaluated.

The last decade has seen explosive growth in the application of these methods across chemistry. Through its applications, deep learning shows promise of being a game changer within chemistry. This review has demonstrated that deep learning has and will continue to impact every stage of the idealized chemistry workflow. Realization of its potential will require a concerted effort to address the major challenges deep learning still faces, many of which have been discussed throughout this review. The three main challenges that must be addressed to maximize the potential of this technique within chemistry are:

1. Obtaining large amounts of high-quality data
2. Developing a standardized framework
3. Effective integration with the broader chemistry community

The first two challenges will be immensely benefitted by increased collaboration, and in particular, continued open sourcing. The push for open sourcing has increased and there is strong evidence of it occurring within deep learning particularly through software packages such as DeepChem[13], TensorMol[164], ANI[52], SchNetPack[85], and chemprop[165]. Addressing the problem of high-quality data also relies on continued advancements in physics based computational chemistry and the accompanying software packages[166-167].

The final challenge requires concerted action from specialists and the broader community. Open sourcing software packages is a step in the right direction, but the chemical community has a long history of resisting assistive software either due to poor usability or unreliable software performance. The latter is demonstrably addressed by these powerful methods, but the former

requires conscious development of usable software packages with feedback from the community. These methods are built to empower chemists first and foremost, and that must be a priority as this field matures.

This review hopes to serve as a gateway to this burgeoning field and encourage chemists, regardless of their specialization, to consider how deep learning could be applied to their work. The following are a set of guidelines to assist in the initial application of these methods;

- Python has become the coding language of choice for deep learning and finding someone proficient in it is invaluable.
- Deep learning requires large volumes of data to outperform traditional machine learning methods. Unless transfer learning is an option, a few thousand data points is a minimum.
- To begin with, employ the open source software packages referenced above with default settings to get a baseline.
- From this baseline, adapt the network architecture using techniques presented in the literature referenced in this review to try and improve performance.
- Utilize the wealth of informative online courses and user-friendly software packages[168-169] provided by the deep learning community to aid in learning these techniques.

Deep learning's contributions to chemistry to date demonstrate that it has a bright future within chemistry, but through effective collaboration between specialists and the broader community, it has the potential to offer a revolution.

AUTHOR INFORMATION

Corresponding Author

*Email: michelle.coote@anu.edu.au

Author Contributions

The manuscript was written through contributions of all authors. All authors have given approval to the final version of the manuscript.

Funding Sources

Australian Research Council (FL170100041)

ACKNOWLEDGMENT

MLC gratefully acknowledges an Australian Research Council Georgina Sweet Laureate Fellowship, while ACM thanks the Australian National University and the Westpac Scholars Trust for PhD scholarships

ABBREVIATIONS

Artificial Neural Networks – ANNs

Coupled Cluster Singles Doubles with perturbative triples - CCSD(T)

Crystal Graph Convolutional Neural Networks - CGCNN

Convolutional Neural Network – CNN

Graph Convolutional Neural Network – GCNN

Long Short Term Memory - LSTM

Recurrent Neural Network – RNN

Rectified Linear Unit - ReLU

Simplified Molecular Input Line Entry System – SMILES

REFERENCES

1. Krizhevsky, A.; Sutskever, I.; Hinton, G.E., ImageNet classification with deep convolutional neural networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, Curran Associates Inc.: Lake Tahoe, Nevada, 2012; pp 1097-1105.
2. Graves, A., Generating Sequences With Recurrent Neural Networks. *eprint arXiv:1308.0850* **2013**, arXiv:1308.0850.
3. Silver, D.; Huang, A.; Maddison, C.J.; Guez, A.; Sifre, L.; van den Driessche, G.; Schrittwieser, J.; Antonoglou, I.; Panneershelvam, V.; Lanctot, M.; Dieleman, S.; Grewe, D.; Nham, J.; Kalchbrenner, N.; Sutskever, I.; Lillicrap, T.; Leach, M.; Kavukcuoglu, K.; Graepel, T.; Hassabis, D., Mastering the game of Go with deep neural networks and tree search. *Nature* **2016,** *529*, 484-489.
4. Taigman, Y.; Yang, M.; Ranzato, M.; Wolf, L. In *DeepFace: Closing the Gap to Human-Level Performance in Face Verification*, 2014 IEEE Conference on Computer Vision and Pattern Recognition, 23-28 June 2014; 2014; pp 1701-1708.
5. Sutskever, I.; Vinyals, O.; Le, Q.V., Sequence to Sequence Learning with Neural Networks. *eprint arXiv:1409.3215* **2014**, arXiv:1409.3215.
6. Hinton, G.; Deng, L.; Yu, D.; Dahl, G.E.; Mohamed, A.; Jaitly, N.; Senior, A.; Vanhoucke, V.; Nguyen, P.; Sainath, T.N.; Kingsbury, B., Deep Neural Networks for Acoustic Modeling in Speech Recognition: The Shared Views of Four Research Groups. *IEEE Signal Processing Magazine* **2012,** *29*, 82-97.
7. Szegedy, C.; Toshev, A.; Erhan, D., Deep neural networks for object detection. In *Proceedings of the 26th International Conference on Neural Information Processing Systems - Volume 2*, Curran Associates Inc.: Lake Tahoe, Nevada, 2013; pp 2553-2561.
8. Pensak, D.A.; Corey, E.J., LHASA—Logic and Heuristics Applied to Synthetic Analysis. In *Computer-Assisted Organic Synthesis*, AMERICAN CHEMICAL SOCIETY: 1977; Vol. 61, pp 1-32.
9. Warr, W.A., A Short Review of Chemical Reaction Database Systems, Computer-Aided Synthesis Design, Reaction Prediction and Synthetic Feasibility. *Molecular Informatics* **2014,** *33*, 469-476.
10. Cook, A.; Johnson, A.P.; Law, J.; Mirzazadeh, M.; Ravitz, O.; Simon, A., Computer-aided synthesis design: 40 years on. *Wiley Interdisciplinary Reviews: Computational Molecular Science* **2012,** *2*, 79-107.
11. Coley, C.W.; Green, W.H.; Jensen, K.F., Machine Learning in Computer-Aided Synthesis Planning. *Accounts of Chemical Research* **2018,** *51*, 1281-1289.
12. LeCun, Y.; Bengio, Y.; Hinton, G., Deep learning. *Nature* **2015,** *521*, 436.
13. 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 *ArXiv e-prints* [Online], 2017. https://ui.adsabs.harvard.edu/#abs/2017arXiv170300564W (accessed March 01, 2017).
14. Goh, G.B.; Hodas, N.O.; Vishnu, A., Deep learning for computational chemistry. *Journal of Computational Chemistry* **2017,** *38*, 1291-1307.
15. Mayr, A.; Klambauer, G.; Unterthiner, T.; Hochreiter, S., DeepTox: Toxicity Prediction using Deep Learning. *Frontiers in Environmental Science* **2016,** *3*.

16. Segler, M.H.S.; Preuss, M.; Waller, M.P., Planning chemical syntheses with deep neural networks and symbolic AI. *Nature* **2018,** *555*, 604.
17. Fooshee, D.; Mood, A.; Gutman, E.; Tavakoli, M.; Urban, G.; Liu, F.; Huynh, N.; Van Vranken, D.; Baldi, P., Deep learning for chemical reaction prediction. *Molecular Systems Design & Engineering* **2018**.
18. Lusci, A.; Pollastri, G.; Baldi, P., Deep Architectures and Deep Learning in Chemoinformatics: The Prediction of Aqueous Solubility for Drug-Like Molecules. *Journal of Chemical Information and Modeling* **2013,** *53*, 1563-1575.
19. Yao, K.; Herr, J.E.; Brown, S.N.; Parkhill, J., Intrinsic Bond Energies from a Bonds-in-Molecules Neural Network. *The Journal of Physical Chemistry Letters* **2017,** *8*, 2689-2694.
20. Faber, F.A.; Hutchison, L.; Huang, B.; Gilmer, J.; Schoenholz, S.S.; Dahl, G.E.; Vinyals, O.; Kearnes, S.; Riley, P.F.; von Lilienfeld, O.A., Prediction errors of molecular machine learning models lower than hybrid DFT error. *Journal of Chemical Theory and Computation* **2017**.
21. Ma, J.; Sheridan, R.P.; Liaw, A.; Dahl, G.E.; Svetnik, V., Deep Neural Nets as a Method for Quantitative Structure–Activity Relationships. *Journal of Chemical Information and Modeling* **2015,** *55*, 263-274.
22. 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. *eprint arXiv:1509.09292* **2015**, arXiv:1509.09292.
23. Kearnes, S.; McCloskey, K.; Berndl, M.; Pande, V.; Riley, P., Molecular graph convolutions: moving beyond fingerprints. *Journal of Computer-Aided Molecular Design* **2016,** *30*, 595-608.
24. Schütt, K.T.; Arbabzadah, F.; Chmiela, S.; Müller, K.R.; Tkatchenko, A., Quantum-chemical insights from deep tensor neural networks. *Nature Communications* **2017,** *8*, 13890.
25. Chen, H.; Engkvist, O.; Wang, Y.; Olivecrona, M.; Blaschke, T., The rise of deep learning in drug discovery. *Drug Discovery Today* **2018,** *23*, 1241-1250.
26. Ekins, S., The Next Era: Deep Learning in Pharmaceutical Research. *Pharmaceutical research* **2016,** *33*, 2594-603.
27. Butler, K.T.; Davies, D.W.; Cartwright, H.; Isayev, O.; Walsh, A., Machine learning for molecular and materials science. *Nature* **2018,** *559*, 547-555.
28. Rupp, M., Machine learning for quantum mechanics in a nutshell. *International Journal of Quantum Chemistry* **2015,** *115*, 1058-1073.
29. Lo, Y.-C.; Rensi, S.E.; Torng, W.; Altman, R.B., Machine learning in chemoinformatics and drug discovery. *Drug Discovery Today* **2018**.
30. Goodfellow, I.; Bengio, Y.; Courville, A., *Deep Learning*. MIT Press: 2016.
31. Le, Q.V.; Ranzato, M.A.; Monga, R.; Devin, M.; Chen, K.; Corrado, G.S.; Dean, J.; Ng, A.Y., Building high-level features using large scale unsupervised learning. *eprint arXiv:1112.6209* **2011**, arXiv:1112.6209.
32. Lowe, D.M. Extraction of Chemical Structures and Reactions from the Literature. University of Cambridge, 2012.
33. Ramakrishnan, R.; Dral, P.O.; Rupp, M.; von Lilienfeld, O.A., Quantum chemistry structures and properties of 134 kilo molecules. *Scientific Data* **2014,** *1*, 140022.
34. Mobley, D.L.; Guthrie, J.P., FreeSolv: a database of experimental and calculated hydration free energies, with input files. *Journal of computer-aided molecular design* **2014,** *28*, 711-720.
35. Smith, J.S.; Isayev, O.; Roitberg, A.E., ANI-1, A data set of 20 million calculated off-equilibrium conformations for organic molecules. *Scientific Data* **2017,** *4*, 170193.

36. Wang, Y.; Xiao, J.; Suzek, T.O.; Zhang, J.; Wang, J.; Zhou, Z.; Han, L.; Karapetyan, K.; Dracheva, S.; Shoemaker, B.A.; Bolton, E.; Gindulyte, A.; Bryant, S.H., PubChem's BioAssay Database. *Nucleic acids research* **2012,** *40*, D400-D412.
37. Domingos, P., A few useful things to know about machine learning. *Commun. ACM* **2012,** *55*, 78-87.
38. Spialter, L., The Atom Connectivity Matrix (ACM) and its Characteristic Polynomial (ACMCP): A New Computer-Oriented Chemical Nomenclature. *Journal of the American Chemical Society* **1963,** *85*, 2012-2013.
39. Rogers, D.; Hahn, M., Extended-Connectivity Fingerprints. *Journal of Chemical Information and Modeling* **2010,** *50*, 742-754.
40. Weininger, D., SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of Chemical Information and Computer Sciences* **1988,** *28*, 31-36.
41. Weininger, D.; Weininger, A.; Weininger, J.L., SMILES. 2. Algorithm for generation of unique SMILES notation. *Journal of Chemical Information and Computer Sciences* **1989,** *29*, 97-101.
42. Heller, S.; McNaught, A.; Stein, S.; Tchekhovskoi, D.; Pletnev, I., InChI - the worldwide chemical structure identifier standard. *Journal of Cheminformatics* **2013,** *5*, 7.
43. 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 Central Science* **2018,** *4*, 725-732.
44. Wei, J.N.; Duvenaud, D.; Aspuru-Guzik, A., Neural Networks for the Prediction of Organic Chemistry Reactions. *ACS Central Science* **2016,** *2*, 725-732.
45. Coley, C.W.; Barzilay, R.; Jaakkola, T.S.; Green, W.H.; Jensen, K.F., Prediction of Organic Reaction Outcomes Using Machine Learning. *ACS Central Science* **2017,** *3*, 434-443.
46. Thomas, N.; Smidt, T.; Kearnes, S.; Yang, L.; Li, L.; Kohlhoff, K.; Riley, P. Tensor field networks: Rotation- and translation-equivariant neural networks for 3D point clouds *ArXiv e-prints* [Online], 2018. https://ui.adsabs.harvard.edu/#abs/2018arXiv180208219T (accessed February 01, 2018).
47. Rupp, M.; Tkatchenko, A.; Müller, K.-R.; von Lilienfeld, O.A., Fast and Accurate Modeling of Molecular Atomization Energies with Machine Learning. *Physical Review Letters* **2012,** *108*, 058301.
48. Staker, J.; Marshall, K.; Abel, R.; McQuaw, C., Molecular Structure Extraction From Documents Using Deep Learning. *eprint arXiv:1802.04903* **2018**, arXiv:1802.04903.
49. Rumelhart, D.E.; Hinton, G.E.; Williams, R.J., Learning representations by back-propagating errors. *Nature* **1986,** *323*, 533.
50. Xavier, G.; Antoine, B.; Yoshua, B., Deep Sparse Rectifier Neural Networks. PMLR: 2011; pp 315-323.
51. Raina, R.; Madhavan, A.; Ng, A.Y., Large-scale deep unsupervised learning using graphics processors. In *Proceedings of the 26th Annual International Conference on Machine Learning*, ACM: Montreal, Quebec, Canada, 2009; pp 873-880.
52. Smith, J.S.; Isayev, O.; Roitberg, A.E., ANI-1: an extensible neural network potential with DFT accuracy at force field computational cost. *Chemical Science* **2017,** *8*, 3192-3203.
53. Gilmer, J.; Schoenholz, S.S.; Riley, P.F.; Vinyals, O.; Dahl, G.E., Neural Message Passing for Quantum Chemistry. *eprint arXiv:1704.01212* **2017**, arXiv:1704.01212.

54. 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. *eprint arXiv:1706.08566* **2017**, arXiv:1706.08566.
55. Cho, H.; Choi, I.S., Three-Dimensionally Embedded Graph Convolutional Network (3DGCN) for Molecule Interpretation. *eprint arXiv:1811.09794* **2018**, arXiv:1811.09794.
56. 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 e-prints* [Online], 2017. https://ui.adsabs.harvard.edu/#abs/2017arXiv170606689G (accessed June 01, 2017).
57. Hopfield, J.J., Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences* **1982,** *79*, 2554.
58. Lipton, Z.C.; Berkowitz, J.; Elkan, C., A Critical Review of Recurrent Neural Networks for Sequence Learning. *eprint arXiv:1506.00019* **2015**, arXiv:1506.00019.
59. Graves, A.; Wayne, G.; Danihelka, I., Neural Turing Machines. *eprint arXiv:1410.5401* **2014**, arXiv:1410.5401.
60. Hochreiter, S.; Schmidhuber, J., *Long Short-term Memory*. 1997; Vol. 9, p 1735-80.
61. Schwaller, P.; Gaudin, T.; Lanyi, D.; Bekas, C.; Laino, T., "Found in Translation": Predicting Outcomes of Complex Organic Chemistry Reactions using Neural Sequence-to-Sequence Models. *eprint arXiv:1711.04810* **2017**, arXiv:1711.04810.
62. Pratt, L.Y., Discriminability-Based Transfer between Neural Networks. In *Advances in Neural Information Processing Systems 5, [NIPS Conference]*, Morgan Kaufmann Publishers Inc.: 1993; pp 204-211.
63. Smith, J.S.; Nebgen, B.T.; Zubatyuk, R.; Lubbers, N.; Devereux, C.; Barros, K.; Tretiak, S.; Isayev, O.; Roitberg, A., *Outsmarting Quantum Chemistry Through Transfer Learning*. 2018.
64. Caruana, R., Multitask Learning. *Machine Learning* **1997,** *28*, 41-75.
65. Ramsundar, B.; Kearnes, S.; Riley, P.; Webster, D.; Konerding, D.; Pande, V., Massively Multitask Networks for Drug Discovery. *eprint arXiv:1502.02072* **2015**, arXiv:1502.02072.
66. Fei-Fei, L.; Fergus, R.; Perona, P., One-shot learning of object categories. *IEEE transactions on pattern analysis and machine intelligence* **2006,** *28*, 594-611.
67. Altae-Tran, H.; Ramsundar, B.; Pappu, A.S.; Pande, V., Low Data Drug Discovery with One-Shot Learning. *ACS Central Science* **2017,** *3*, 283-293.
68. Kingma, D.P.; Welling, M. Auto-Encoding Variational Bayes *ArXiv e-prints* [Online], 2013. https://ui.adsabs.harvard.edu/#abs/2013arXiv1312.6114K (accessed December 01, 2013).
69. Goodfellow, I.J.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; Bengio, Y. Generative Adversarial Networks *ArXiv e-prints* [Online], 2014. http://adsabs.harvard.edu/abs/2014arXiv1406.2661G (accessed June 1, 2014).
70. 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). 2017.
71. Wong, S.C.; Gatt, A.; Stamatescu, V.; McDonnell, M.D. In *Understanding Data Augmentation for Classification: When to Warp?*, 2016 International Conference on Digital Image Computing: Techniques and Applications (DICTA), 30 Nov.-2 Dec. 2016; 2016; pp 1-6.
72. Bjerrum, E.J. SMILES Enumeration as Data Augmentation for Neural Network Modeling of Molecules *ArXiv e-prints* [Online], 2017. https://ui.adsabs.harvard.edu/#abs/2017arXiv170307076J (accessed March 01, 2017).

73. Li, Y., Deep Reinforcement Learning: An Overview. *eprint arXiv:1701.07274* **2017**, arXiv:1701.07274.
74. Zhou, Z.; Li, X.; Zare, R.N., Optimizing Chemical Reactions with Deep Reinforcement Learning. *ACS Central Science* **2017,** *3*, 1337-1344.
75. Längkvist, M.; Karlsson, L.; Loutfi, A., A review of unsupervised feature learning and deep learning for time-series modeling. *Pattern Recognition Letters* **2014,** *42*, 11-24.
76. Behler, J., Atom-centered symmetry functions for constructing high-dimensional neural network potentials. *The Journal of Chemical Physics* **2011,** *134*, 074106.
77. Behler, J., Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations. *Physical Chemistry Chemical Physics* **2011,** *13*, 17930-17955.
78. Zhang, L.; Han, J.; Wang, H.; Car, R.; E, W., Deep Potential Molecular Dynamics: A Scalable Model with the Accuracy of Quantum Mechanics. *Physical Review Letters* **2018,** *120*, 143001.
79. McGibbon, R.T.; Taube, A.G.; Donchev, A.G.; Siva, K.; Hernández, F.; Hargus, C.; Law, K.-H.; Klepeis, J.L.; Shaw, D.E., Improving the accuracy of Møller-Plesset perturbation theory with neural networks. *The Journal of Chemical Physics* **2017,** *147*, 161725.
80. Mills, K.; Spanner, M.; Tamblyn, I., Deep learning and the Schrodinger equation. *Physical Review A* **2017,** *96*, 042113.
81. Yao, K.; Parkhill, J., Kinetic Energy of Hydrocarbons as a Function of Electron Density and Convolutional Neural Networks. *Journal of Chemical Theory and Computation* **2016,** *12*, 1139-1147.
82. Behler, J., First Principles Neural Network Potentials for Reactive Simulations of Large Molecular and Condensed Systems. *Angewandte Chemie International Edition* **2017,** *56*, 12828-12840.
83. Behler, J., Constructing high-dimensional neural network potentials: A tutorial review. *International Journal of Quantum Chemistry* **2015,** *115*, 1032-1050.
84. Behler, J.; Parrinello, M., Generalized Neural-Network Representation of High-Dimensional Potential-Energy Surfaces. *Physical Review Letters* **2007,** *98*, 146401.
85. Schütt, K.T.; Kessel, P.; Gastegger, M.; Nicoli, K.A.; Tkatchenko, A.; Müller, K.R., SchNetPack: A Deep Learning Toolbox For Atomistic Systems. *Journal of Chemical Theory and Computation* **2019,** *15*, 448-455.
86. von Lilienfeld, O.A., Quantum Machine Learning in Chemical Compound Space. *Angewandte Chemie International Edition* **2018,** *57*, 4164-4169.
87. Montavon, G.; Rupp, M.; Gobre, V.; Vazquez-Mayagoitia, A.; Hansen, K.; Tkatchenko, A.; Müller, K.-R.; von Lilienfeld, O.A., Machine learning of molecular electronic properties in chemical compound space. *New Journal of Physics* **2013,** *15*, 095003.
88. Hansen, K.; Biegler, F.; Ramakrishnan, R.; Pronobis, W.; von Lilienfeld, O.A.; Müller, K.-R.; Tkatchenko, A., Machine Learning Predictions of Molecular Properties: Accurate Many-Body Potentials and Nonlocality in Chemical Space. *The Journal of Physical Chemistry Letters* **2015,** *6*, 2326-2331.
89. Jain, A.; Ong, S.P.; Hautier, G.; Chen, W.; Richards, W.D.; Dacek, S.; Cholia, S.; Gunter, D.; Skinner, D.; Ceder, G.; Persson, K.A., Commentary: The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials* **2013,** *1*, 011002.
90. Gaulton, A.; Bellis, L.J.; Bento, A.P.; Chambers, J.; Davies, M.; Hersey, A.; Light, Y.; McGlinchey, S.; Michalovich, D.; Al-Lazikani, B.; Overington, J.P., ChEMBL: a large-scale bioactivity database for drug discovery. *Nucleic Acids Res* **2012,** *40*, D1100-7.

91. Kim, S.; Thiessen, P.A.; Bolton, E.E.; Chen, J.; Fu, G.; Gindulyte, A.; Han, L.; He, J.; He, S.; Shoemaker, B.A.; Wang, J.; Yu, B.; Zhang, J.; Bryant, S.H., PubChem Substance and Compound databases. *Nucleic acids research* **2016,** *44*, D1202-D1213.
92. Hughes, T.B.; Miller, G.P.; Swamidass, S.J., Modeling Epoxidation of Drug-like Molecules with a Deep Machine Learning Network. *ACS Central Science* **2015,** *1*, 168-180.
93. Unterthiner, T.; Mayr, A.; Klambauer, G.; Steijaert, M.; Ceulemans, H.; Wegner, J.; Hochreiter, S., *Deep Learning as an Opportunity in Virtual Screening*. 2014.
94. Dahl, G.E.; Jaitly, N.; Salakhutdinov, R., Multi-task Neural Networks for QSAR Predictions. *eprint arXiv:1406.1231* **2014**, arXiv:1406.1231.
95. Korotcov, A.; Tkachenko, V.; Russo, D.P.; Ekins, S., Comparison of Deep Learning With Multiple Machine Learning Methods and Metrics Using Diverse Drug Discovery Data Sets. *Molecular Pharmaceutics* **2017,** *14*, 4462-4475.
96. Unterthiner, T.; Mayr, A.; Klambauer, G.; Hochreiter, S., Toxicity Prediction using Deep Learning. *eprint arXiv:1503.01445* **2015**, arXiv:1503.01445.
97. Wenzel, J.; Matter, H.; Schmidt, F., Predictive Multitask Deep Neural Network Models for ADME-Tox Properties: Learning from Large Data Sets. *Journal of Chemical Information and Modeling* **2019**.
98. Li, M.; Zhang, H.; Chen, B.; Wu, Y.; Guan, L., Prediction of pKa Values for Neutral and Basic Drugs based on Hybrid Artificial Intelligence Methods. *Scientific Reports* **2018,** *8*, 3991.
99. Xu, Y.; Dai, Z.; Chen, F.; Gao, S.; Pei, J.; Lai, L., Deep Learning for Drug-Induced Liver Injury. *Journal of Chemical Information and Modeling* **2015,** *55*, 2085-2093.
100. Goh, G.B.; Hodas, N.O.; Siegel, C.; Vishnu, A., SMILES2Vec: An Interpretable General-Purpose Deep Neural Network for Predicting Chemical Properties. *ArXiv e-prints* **2017,** *1712*, arXiv:1712.02034.
101. Jastrzębski, S.; Leśniak, D.; Czarnecki, W.M., Learning to SMILE(S). *eprint arXiv:1602.06289* **2016**, arXiv:1602.06289.
102. Schütt, K.T.; Gastegger, M.; Tkatchenko, A.; Müller, K.-R., Quantum-chemical insights from interpretable atomistic neural networks. *eprint arXiv:1806.10349* **2018**, arXiv:1806.10349.
103. Schütt, K.T.; Sauceda, H.E.; Kindermans, P.-J.; Tkatchenko, A.; Müller, K.-R., SchNet - a deep learning architecture for molecules and materials. *ArXiv e-prints* **2017,** *1712*, arXiv:1712.06113.
104. Wallach, I.; Dzamba, M.; Heifets, A. AtomNet: A Deep Convolutional Neural Network for Bioactivity Prediction in Structure-based Drug Discovery *ArXiv e-prints* [Online], 2015. https://ui.adsabs.harvard.edu/#abs/2015arXiv151002855W (accessed October 01, 2015).
105. Zeng, M.; Nitin Kumar, J.; Zeng, Z.; Savitha, R.; Ramaseshan Chandrasekhar, V.; Hippalgaonkar, K., Graph Convolutional Neural Networks for Polymers Property Prediction. *eprint arXiv:1811.06231* **2018**, arXiv:1811.06231.
106. Coley, C.W.; Barzilay, R.; Green, W.H.; Jaakkola, T.S.; Jensen, K.F., Convolutional Embedding of Attributed Molecular Graphs for Physical Property Prediction. *Journal of Chemical Information and Modeling* **2017,** *57*, 1757-1772.
107. Wodrich, M.D.; Corminboeuf, C.; Schleyer, P.v.R., Systematic Errors in Computed Alkane Energies Using B3LYP and Other Popular DFT Functionals. *Organic Letters* **2006,** *8*, 3631-3634.
108. Cohen, A.J.; Mori-Sánchez, P.; Yang, W., Challenges for Density Functional Theory. *Chemical Reviews* **2012,** *112*, 289-320.
109. Purvis, G.D.; Bartlett, R.J., A full coupled-cluster singles and doubles model: The inclusion of disconnected triples. *The Journal of Chemical Physics* **1982,** *76*, 1910-1918.

110. Goh, G.B.; Siegel, C.; Vishnu, A.; Hodas, N.O., Using Rule-Based Labels for Weak Supervised Learning: A ChemNet for Transferable Chemical Property Prediction. *ArXiv e-prints* **2017,** *1712*, arXiv:1712.02734.
111. Ryan-Rhys, G.; Philippe, S.; Alpha, L., Dataset Bias in the Natural Sciences: A Case Study in Chemical Reaction Prediction and Synthesis Design. 2018.
112. Swann, E.T.; Fernandez, M.; Coote, M.L.; Barnard, A.S., Bias-Free Chemically Diverse Test Sets from Machine Learning. *ACS Comb Sci* **2017,** *19*, 544-554.
113. Segler, M.H.S.; Kogej, T.; Tyrchan, C.; Waller, M.P., Generating Focussed Molecule Libraries for Drug Discovery with Recurrent Neural Networks. *ArXiv e-prints* **2017,** *1701*, arXiv:1701.01329.
114. Browning, N.J.; Ramakrishnan, R.; von Lilienfeld, O.A.; Roethlisberger, U., Genetic Optimization of Training Sets for Improved Machine Learning Models of Molecular Properties. *The Journal of Physical Chemistry Letters* **2017,** *8*, 1351-1359.
115. Smith, J.S.; Nebgen, B.; Lubbers, N.; Isayev, O.; Roitberg, A.E., Less is more: Sampling chemical space with active learning. *The Journal of Chemical Physics* **2018,** *148*, 241733.
116. Shwartz-Ziv, R.; Tishby, N., Opening the Black Box of Deep Neural Networks via Information. *eprint arXiv:1703.00810* **2017**, arXiv:1703.00810.
117. B. Goh, G.; Siegel, C.; Vishnu, A.; O. Hodas, N.; Baker, N., How Much Chemistry Does a Deep Neural Network Need to Know to Make Accurate Predictions? 2017.
118. Gebauer, N.W.A.; Gastegger, M.; Schütt, K.T., Generating equilibrium molecules with deep neural networks. *eprint arXiv:1810.11347* **2018**, arXiv:1810.11347.
119. Ikebata, H.; Hongo, K.; Isomura, T.; Maezono, R.; Yoshida, R., Bayesian molecular design with a chemical language model. *Journal of Computer-Aided Molecular Design* **2017,** *31*, 379-391.
120. Kawai, K.; Nagata, N.; Takahashi, Y., De Novo Design of Drug-Like Molecules by a Fragment-Based Molecular Evolutionary Approach. *Journal of Chemical Information and Modeling* **2014,** *54*, 49-56.
121. Blaschke, T.; Olivecrona, M.; Engkvist, O.; Bajorath, J.; Chen, H., Application of Generative Autoencoder in De Novo Molecular Design. *Molecular Informatics* **2018,** *37*, 1700123.
122. Jin, W.; Barzilay, R.; Jaakkola, T., Junction Tree Variational Autoencoder for Molecular Graph Generation. *eprint arXiv:1802.04364* **2018**, arXiv:1802.04364.
123. Dai, H.; Tian, Y.; Dai, B.; Skiena, S.; Song, L., Syntax-Directed Variational Autoencoder for Structured Data. *eprint arXiv:1802.08786* **2018**, arXiv:1802.08786.
124. Lim, J.; Ryu, S.; Kim, J.W.; Kim, W.Y., Molecular generative model based on conditional variational autoencoder for de novo molecular design. *eprint arXiv:1806.05805* **2018**, arXiv:1806.05805.
125. Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H., Molecular de-novo design through deep reinforcement learning. *Journal of Cheminformatics* **2017,** *9*, 48.
126. You, J.; Liu, B.; Ying, R.; Pande, V.; Leskovec, J., Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation. *eprint arXiv:1806.02473* **2018**, arXiv:1806.02473.
127. 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. *Journal of Chemical Information and Modeling* **2018,** *58*, 1194-1204.
128. Zhou, Z.; Kearnes, S.; Li, L.; Zare, R.N.; Riley, P., Optimization of Molecules via Deep Reinforcement Learning. *eprint arXiv:1810.08678* **2018**, arXiv:1810.08678.

129. Bjerrum, E.J.; Threlfall, R., Molecular Generation with Recurrent Neural Networks (RNNs). *eprint arXiv:1705.04612* **2017**, arXiv:1705.04612.
130. Sanchez-Lengeling, B.; Aspuru-Guzik, A., Inverse molecular design using machine learning: Generative models for matter engineering. *Science* **2018,** *361*, 360.
131. Xie, T.; Grossman, J.C., Crystal Graph Convolutional Neural Networks for Accurate and Interpretable Prediction of Material Properties. *ArXiv e-prints* **2017,** *1710*, arXiv:1710.10324.
132. Chen, C.; Ye, W.; Zuo, Y.; Zheng, C.; Ong, S.P., Graph Networks as a Universal Machine Learning Framework for Molecules and Crystals. *eprint arXiv:1812.05055* **2018**, arXiv:1812.05055.
133. Jain, A.; Bligaard, T., Atomic-position independent descriptor for machine learning of material properties. *Physical Review B* **2018,** *98*, 214112.
134. Laugier, L.; Bash, D.; Recatala, J.; Ng, H.K.; Ramasamy, S.; Foo, C.-S.; Chandrasekhar, V.R.; Hippalgaonkar, K., Predicting thermoelectric properties from crystal graphs and material descriptors - first application for functional materials. *eprint arXiv:1811.06219* **2018**, arXiv:1811.06219.
135. Li, H.; Collins, C.R.; Ribelli, T.G.; Matyjaszewski, K.; Gordon, G.J.; Kowalewski, T.; Yaron, D.J., Tuning the molecular weight distribution from atom transfer radical polymerization using deep reinforcement learning. *Molecular Systems Design & Engineering* **2018,** *3*, 496-508.
136. Xie, T.; Grossman, J.C., Hierarchical visualization of materials space with graph convolutional neural networks. *The Journal of Chemical Physics* **2018,** *149*, 174111.
137. Kim, E.; Huang, K.; Jegelka, S.; Olivetti, E., Virtual screening of inorganic materials synthesis parameters with deep learning. *npj Computational Materials* **2017,** *3*, 53.
138. Feng, S.; Zhou, H.; Dong, H., Using deep neural network with small dataset to predict material defects. *Materials & Design* **2019,** *162*, 300-310.
139. Ma, W.; Cheng, F.; Liu, Y., Deep-Learning-Enabled On-Demand Design of Chiral Metamaterials. *ACS Nano* **2018,** *12*, 6326-6334.
140. Kitchin, J.R., Machine learning in catalysis. *Nature Catalysis* **2018,** *1*, 230-232.
141. Goldsmith, B.R.; Esterhuizen, J.; Liu, J.-X.; Bartel, C.J.; Sutton, C., Machine learning for heterogeneous catalyst design and discovery. *AIChE Journal* **2018,** *64*, 2311-2323.
142. Shakouri, K.; Behler, J.; Meyer, J.; Kroes, G.-J., Accurate Neural Network Description of Surface Phonons in Reactive Gas–Surface Dynamics: N2 + Ru(0001). *The Journal of Physical Chemistry Letters* **2017,** *8*, 2131-2136.
143. Zhai, H.; Alexandrova, A.N., Ensemble-Average Representation of Pt Clusters in Conditions of Catalysis Accessed through GPU Accelerated Deep Neural Network Fitting Global Optimization. *Journal of Chemical Theory and Computation* **2016,** *12*, 6213-6226.
144. Smith, J.S.; Roitberg, A.E.; Isayev, O., Transforming Computational Drug Discovery with Machine Learning and AI. *ACS Medicinal Chemistry Letters* **2018,** *9*, 1065-1069.
145. Gawehn, E.; Hiss Jan, A.; Schneider, G., Deep Learning in Drug Discovery. *Molecular Informatics* **2015,** *35*, 3-14.
146. 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. *Molecular Pharmaceutics* **2018,** *15*, 4398-4405.
147. 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. *Molecular Pharmaceutics* **2017,** *14*, 3098-3104.

148. Popova, M.; Isayev, O.; Tropsha, A., Deep Reinforcement Learning for De-Novo Drug Design. *ArXiv e-prints* **2017,** *1711*, arXiv:1711.10907.
149. Preuer, K.; Lewis, R.P.I.; Hochreiter, S.; Bender, A.; Bulusu, K.C.; Klambauer, G., DeepSynergy: predicting anti-cancer drug synergy with Deep Learning. *Bioinformatics* **2017**, btx806-btx806.
150. Preuer, K.; Renz, P.; Unterthiner, T.; Hochreiter, S.; Klambauer, G., Fréchet ChemNet Distance: A Metric for Generative Models for Molecules in Drug Discovery. *Journal of Chemical Information and Modeling* **2018,** *58*, 1736-1741.
151. Segler, M.H.S.; Kogej, T.; Tyrchan, C.; Waller, M.P., Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks. *ACS Central Science* **2018,** *4*, 120-131.
152. Salatin, T.D.; Jorgensen, W.L., Computer-assisted mechanistic evaluation of organic reactions. 1. Overview. *The Journal of Organic Chemistry* **1980,** *45*, 2043-2051.
153. Satoh, H.; Funatsu, K., SOPHIA, a Knowledge Base-Guided Reaction Prediction System - Utilization of a Knowledge Base Derived from a Reaction Database. *Journal of Chemical Information and Computer Sciences* **1995,** *35*, 34-44.
154. Socorro, I.M.; Goodman, J.M., The ROBIA Program for Predicting Organic Reactivity. *Journal of Chemical Information and Modeling* **2006,** *46*, 606-614.
155. Segler, M.; Preuß, M.; Waller, M.P., Towards "AlphaChem": Chemical Synthesis Planning with Tree Search and Deep Neural Network Policies. *ArXiv e-prints* **2017,** *1702*, arXiv:1702.00020.
156. Elsevier Life Sciences, Reaxys. http://www.reaxys.com (accessed March 29, 2019).
157. Liu, B.; Ramsundar, B.; Kawthekar, P.; Shi, J.; Gomes, J.; Luu Nguyen, Q.; Ho, S.; Sloane, J.; Wender, P.; Pande, V., Retrosynthetic Reaction Prediction Using Neural Sequence-to-Sequence Models. *ACS Central Science* **2017,** *3*, 1103-1113.
158. Machine Learning for Pharmaceutical Discovery and Synthesis Symposium, ASKCOS. http://askcos.mit.edu/ (accessed May 08, 2019).
159. Kayala, M.A.; Baldi, P., ReactionPredictor: Prediction of Complex Chemical Reactions at the Mechanistic Level Using Machine Learning. *Journal of Chemical Information and Modeling* **2012,** *52*, 2526-2540.
160. Nam, J.; Kim, J., Linking the Neural Machine Translation and the Prediction of Organic Chemistry Reactions. *eprint arXiv:1612.09529* **2016**, arXiv:1612.09529.
161. Schwaller, P.; Laino, T.; Gaudin, T.; Bolgar, P.; Bekas, C.; Lee, A.A., Molecular Transformer for Chemical Reaction Prediction and Uncertainty Estimation. *eprint arXiv:1811.02633* **2018**, arXiv:1811.02633.
162. Coley, C.W.; Jin, W.; Rogers, L.; Jamison, T.F.; Jaakkola, T.S.; Green, W.H.; Barzilay, R.; Jensen, K.F., A graph-convolutional neural network model for the prediction of chemical reactivity. *Chemical Science* **2019,** *10*, 370-377.
163. Daniel, R.; Gonçalo, B.; Tiago, R., Evolving and Nano Data Enabled Machine Intelligence for Chemical Reaction Optimization. 2018.
164. Yao, K.; Herr, J.E.; Toth, David W.; McKintyre, R.; Parkhill, J., The TensorMol-0.1 model chemistry: a neural network augmented with long-range physics. *Chemical Science* **2018,** *9*, 2261-2269.
165. Yang, K.; Swanson, K.; Jin, W.; Coley, C.; Eiden, P.; Gao, H.; Guzman-Perez, A.; Hopper, T.; Kelley, B.; Mathea, M.; Palmer, A.; Settels, V.; Jaakkola, T.; Jensen, K.; Barzilay, R. Are

Learned Molecular Representations Ready For Prime Time? *arXiv e-prints* [Online], 2019. https://ui.adsabs.harvard.edu/abs/2019arXiv190401561Y (accessed April 01, 2019).
166. Gan, Z.; Epifanovsky, E.; Gilbert, A.T.B.; Wormit, M.; Kussmann, J.; Lange, A.W.; Behn, A.; Deng, J.; Feng, X.; Ghosh, D.; Goldey, M.; Horn, P.R.; Jacobson, L.D.; Kaliman, I.; Khaliullin, R.Z.; Kuś, T.; Landau, A.; Liu, J.; Proynov, E.I.; Rhee, Y.M.; Richard, R.M.; Rohrdanz, M.A.; Steele, R.P.; Sundstrom, E.J.; Woodcock, H.L.; Zimmerman, P.M.; Zuev, D.; Albrecht, B.; Alguire, E.; Austin, B.; Beran, G.J.O.; Bernard, Y.A.; Berquist, E.; Brandhorst, K.; Bravaya, K.B.; Brown, S.T.; Casanova, D.; Chang, C.-M.; Chen, Y.; Chien, S.H.; Closser, K.D.; Crittenden, D.L.; Diedenhofen, M.; DiStasio, R.A.; Do, H.; Dutoi, A.D.; Edgar, R.G.; Fatehi, S.; Fusti-Molnar, L.; Ghysels, A.; Golubeva-Zadorozhnaya, A.; Gomes, J.; Hanson-Heine, M.W.D.; Harbach, P.H.P.; Hauser, A.W.; Hohenstein, E.G.; Holden, Z.C.; Jagau, T.-C.; Ji, H.; Kaduk, B.; Khistyaev, K.; Kim, J.; Kim, J.; King, R.A.; Klunzinger, P.; Kosenkov, D.; Kowalczyk, T.; Krauter, C.M.; Lao, K.U.; Laurent, A.D.; Lawler, K.V.; Levchenko, S.V.; Lin, C.Y.; Liu, F.; Livshits, E.; Lochan, R.C.; Luenser, A.; Manohar, P.; Manzer, S.F.; Mao, S.-P.; Mardirossian, N.; Marenich, A.V.; Maurer, S.A.; Mayhall, N.J.; Neuscamman, E.; Oana, C.M.; Olivares-Amaya, R.; O'Neill, D.P.; Parkhill, J.A.; Perrine, T.M.; Peverati, R.; Prociuk, A.; Rehn, D.R.; Rosta, E.; Russ, N.J.; Sharada, S.M.; Sharma, S.; Small, D.W.; Sodt, A.; Stein, T.; Stück, D.; Su, Y.-C.; Thom, A.J.W.; Tsuchimochi, T.; Vanovschi, V.; Vogt, L.; Vydrov, O.; Wang, T.; Watson, M.A.; Wenzel, J.; White, A.; Williams, C.F.; Yang, J.; Yeganeh, S.; Yost, S.R.; You, Z.-Q.; Zhang, I.Y.; Zhang, X.; Zhao, Y.; Brooks, B.R.; Chan, G.K.L.; Chipman, D.M.; Cramer, C.J.; Goddard, W.A.; Gordon, M.S.; Hehre, W.J.; Klamt, A.; Schaefer, H.F.; Schmidt, M.W.; Sherrill, C.D.; Truhlar, D.G.; Warshel, A.; Xu, X.; Aspuru-Guzik, A.; Baer, R.; Bell, A.T.; Besley, N.A.; Chai, J.-D.; Dreuw, A.; Dunietz, B.D.; Furlani, T.R.; Gwaltney, S.R.; Hsu, C.-P.; Jung, Y.; Kong, J.; Lambrecht, D.S.; Liang, W.; Ochsenfeld, C.; Rassolov, V.A.; Slipchenko, L.V.; Subotnik, J.E.; Van Voorhis, T.; Herbert, J.M.; Krylov, A.I.; Gill, P.M.W.; Head-Gordon, M., Advances in molecular quantum chemistry contained in the Q-Chem 4 program package AU - Shao, Yihan. *Molecular Physics* **2015,** *113*, 184-215.
167. Frisch, M.J.; Trucks, G.W.; Schlegel, H.B.; Scuseria, G.E.; Robb, M.A.; Cheeseman, J.R.; Scalmani, G.; Barone, V.; Petersson, G.A.; Nakatsuji, H.; Li, X.; Caricato, M.; Marenich, A.V.; Bloino, J.; Janesko, B.G.; Gomperts, R.; Mennucci, B.; Hratchian, H.P.; Ortiz, J.V.; Izmaylov, A.F.; Sonnenberg, J.L.; Williams; Ding, F.; Lipparini, F.; Egidi, F.; Goings, J.; Peng, B.; Petrone, A.; Henderson, T.; Ranasinghe, D.; Zakrzewski, V.G.; Gao, J.; Rega, N.; Zheng, G.; Liang, W.; Hada, M.; Ehara, M.; Toyota, K.; Fukuda, R.; Hasegawa, J.; Ishida, M.; Nakajima, T.; Honda, Y.; Kitao, O.; Nakai, H.; Vreven, T.; Throssell, K.; Montgomery Jr., J.A.; Peralta, J.E.; Ogliaro, F.; Bearpark, M.J.; Heyd, J.J.; Brothers, E.N.; Kudin, K.N.; Staroverov, V.N.; Keith, T.A.; Kobayashi, R.; Normand, J.; Raghavachari, K.; Rendell, A.P.; Burant, J.C.; Iyengar, S.S.; Tomasi, J.; Cossi, M.; Millam, J.M.; Klene, M.; Adamo, C.; Cammi, R.; Ochterski, J.W.; Martin, R.L.; Morokuma, K.; Farkas, O.; Foresman, J.B.; Fox, D.J. *Gaussian 16 Rev. B.01*, Wallingford, CT, 2016.
168. Abadi, M.; Barham, P.; Chen, J.; Chen, Z.; Davis, A.; Dean, J.; Devin, M.; Ghemawat, S.; Irving, G.; Isard, M.; Kudlur, M.; Levenberg, J.; Monga, R.; Moore, S.; Murray, D.G.; Steiner, B.; Tucker, P.A.; Vasudevan, V.; Warden, P.; Wicke, M.; Yu, Y.; Zheng, X., TensorFlow: A System for Large-Scale Machine Learning. 2016; pp 265-283.
169. Jia, Y.; Shelhamer, E.; Donahue, J.; Karayev, S.; Long, J.; Girshick, R.; Guadarrama, S.; Darrell, T., Caffe: Convolutional Architecture for Fast Feature Embedding. *eprint arXiv:1408.5093* **2014**, arXiv:1408.5093.

For table of contents use: