

De Novo Molecule Design by Translating from Reduced Graphs to SMILES

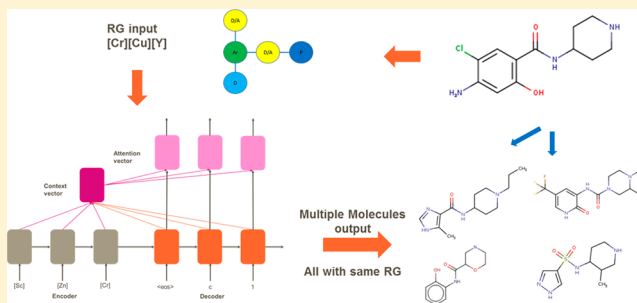
Peter Pogány,[#] Navot Arad,[§] Sam Genway,[§] and Stephen D. Pickett^{*,#}

[#]Computational and Modeling Sciences, GlaxoSmithKline, Gunnels Wood Road, Stevenage, Herts SG1 2NY, United Kingdom

[§]GlaxoSmithKline-Tessella Analytics Partnership, Tessella Ltd, Walkern Road, Stevenage, Herts SG1 3QP, United Kingdom

S Supporting Information

ABSTRACT: A key component of automated molecular design is the generation of compound ideas for subsequent filtering and assessment. Recently deep learning approaches have been explored as alternatives to traditional de novo molecular design techniques. Deep learning algorithms rely on learning from large pools of molecules represented as molecular graphs (generally SMILES), and several approaches can be used to tailor the generated molecules to defined regions of chemical space. Cheminformatics has developed alternative higher-level representations that capture the key properties of a set of molecules, and it would be of interest to understand whether such representations can be used to constrain the output of molecule generation algorithms. In this work we explore the use of one such representation, the Reduced Graph, as a definition of target chemical space for a deep learning molecule generator. The Reduced Graph replaces functional groups with superatoms representing the pharmacophoric features. Assigning these superatoms to specific nonorganic element types allows the Reduced Graph to be represented as a valid SMILES string. The mapping from standard SMILES to Reduced Graph SMILES is well-defined, however, the inverse is not true, and this presents a particular challenge. Here we present the results of a novel seq-to-seq approach to molecule generation, where the one to many mapping of Reduced Graph to SMILES is learned on a large training set. This training needs to be performed only once. In a subsequent step, this model can be used to generate arbitrary numbers of compounds that have the same Reduced Graph as any input molecule. Through analysis of data sets in ChEMBL we show that the approach generates valid molecules and can extrapolate to Reduced Graphs unseen in the training set. The method offers an alternative deep learning approach to molecule generation that does not rely on transfer learning, latent space generation, or adversarial networks and is applicable to scaffold hopping and other cheminformatics applications in drug discovery.



INTRODUCTION

Drug discovery is a complex multiparameter optimization process.^{1,2} Traditionally the design–make–test cycle involves multidisciplinary teams coming together to understand the data and make decisions about the next iteration of compounds to synthesize. Recent advances in algorithms and computer hardware have led to investment in systems that automate large parts of this process, and there have been some notable successes.³ This data driven approach to drug discovery involves a number of key technologies:

1. Automated design of compounds in the relevant chemical space.
2. Automated methods to build and update models and to filter and score compounds.
3. Algorithms for the selection of compounds for the next iteration.

There have been advances in all three of these areas in recent years. (1) De novo design is a well-established field of cheminformatics.⁴ Algorithms include evolving structures to predefined constraints,⁵ fragment growing particularly in the

context of a protein structure,^{6–8} fragmenting and recombining using reaction based schemes such as RECAP⁹ and BRICS,¹⁰ and forward reaction prediction with tools such as ICFRP.¹¹ The availability of large publicly available chemical databases such as ChEMBL,¹² PubChem,¹³ and ZINC¹⁴ combined with recent advances in neural network based learning methods has led to the development of a number of deep learning approaches^{15–17} for de novo structure generation of small molecules^{16,18–24} and peptides^{25–27} and has reinvigorated the interest in de novo design. (2) Several groups have published approaches to automate model building that can be used in the design–make–test cycle to update models as new data are generated.^{28–32} (3) Active learning^{33–35} is emerging as an approach to tackle the challenge of compound selection, as it combines the requirements for model improvement and chemical space exploration.

Special Issue: Machine Learning in Drug Discovery

Received: September 13, 2018

Published: December 11, 2018



Our focus in this paper is on the first step, the automated design of compounds to a defined chemical space. Current deep learning approaches require data sets of diverse active molecules to define this target chemical space. However, it is often the case that high throughput screening on a novel target will deliver a singleton or small number of active analogues. Alternatively, there may be interest in lead hopping from a lead series. To address these issues, we employ a higher-level description of compounds that captures the key features of both molecular properties and drug–target interactions, the Reduced Graph.^{36–38} The Reduced Graph defines a target chemical space as many molecules can share the same Reduced Graph. The problem is how to translate from a Reduced Graph representation to a standard molecular graph. We have borrowed from the field of language translation to develop a novel deep learning seq-to-seq approach that achieves this translation. Once trained there is no transfer learning or optimization required. Rather, a Reduced Graph is entered and molecules are generated that represent it. The method has particular applicability to generating novel molecules around a single or small number of related hit molecules or in lead hopping.

The paper is arranged as follows. Below we briefly review deep learning algorithms for molecule generation to set this work into context. We review Reduced Graphs and then describe the seq-to-seq algorithm implementation. Results are presented where ChEMBL is used to learn the Reduced Graph to SMILES³⁹ translation. We apply the method to structures with Reduced Graphs unseen by the training set to show that the method generalizes. We conclude with a summary and outline potential future work.

Deep Learning for Molecule Generation. Long short-term memory (LSTM) neural networks⁴⁰ have found application in areas ranging from time-series⁴¹ to speech recognition.⁴² Recently, it has been demonstrated that LSTM networks can generate coherent text in a particular language and style, word-by-word, following training on a large corpus of similar text.⁴³ Such approaches to generate sequences of words have been successfully applied to the problem of generating molecular libraries, by generating SMILES string representations of molecules character-by-character and training on large databases of known molecules.²¹ Employing transfer learning on a smaller set of molecules known to show activity against a specific target, several groups have shown effective generation of focused molecular libraries.¹⁹ An alternative approach to focus the chemical space of the generated molecules is to use reinforcement learning.⁴⁴ In this case predictive models are used to score generated molecules in an iterative approach so that the model learns “good” molecules.

All of the approaches above utilize the molecular graph as represented by the SMILES string.³⁹ A relatively large number (hundreds to thousands)^{19,21} of molecules is required for transfer learning or to build effective models to focus the chemical space. While such information may be available during lead optimization, it is not uncommon to have only limited data available at the start of a drug discovery program. In addition, for novel targets or phenotypic end points there may be little or no published information. Lead hopping may also be a challenge as it is unlikely that project specific models will extrapolate to new chemotypes and global models will likely lack the resolution to effectively rank ideas. One-shot learning⁴⁵ has been applied to attempt to learn structure

activity relationship models with small amounts of data, but this is distinct from the molecule generation use case. Here we describe an alternative approach to molecule generation that addresses the limitations above by using a higher-order representation of the chemical structure, the Reduced Graph, to represent the target chemical space.

Reduced Graphs. In the Reduced Graph, the full molecular graph is reduced to pharmacophore feature type nodes as shown in Figure 1. The nodes may be further merged into superatoms, as in this example, where the amide is represented by a single donor/acceptor feature.

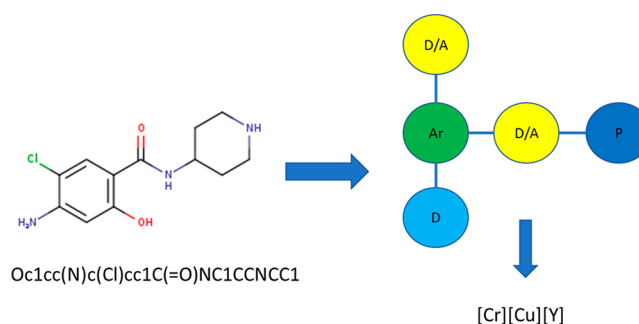
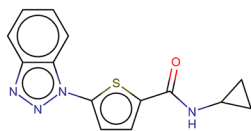
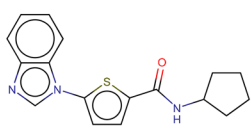
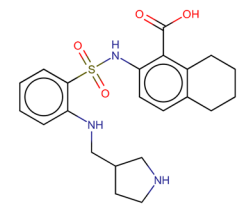
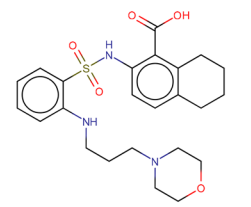
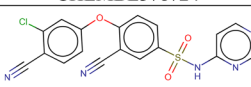
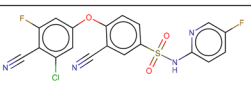
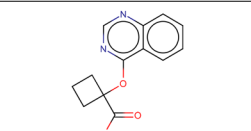
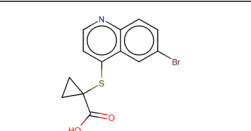
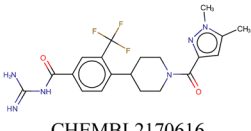
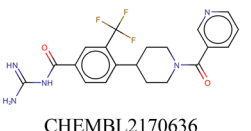
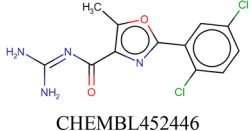

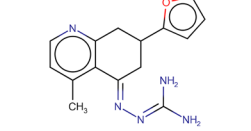
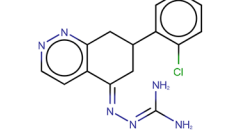
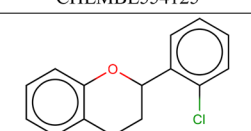
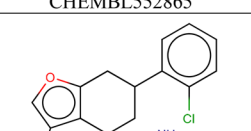
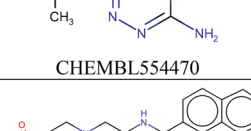
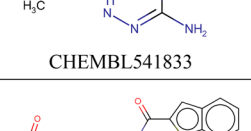


Figure 1. Molecule and corresponding Reduced Graph. Ar, aromatic; D/A, donor/acceptor; P, basic ring; D, donor. Further merging the nodes on the aromatic ring leads to the Reduced Graph SMILES. Cr, donor/acceptor aromatic ring; Cu, donor/acceptor feature node; Y, positively ionizable aliphatic ring.

The Reduced Graph³⁶ and related representations^{46,47} have been used in a variety of applications related to cheminformatics and drug discovery including summarizing HTS data,³⁸ similarity searching and virtual screening,^{37,46,48,49} scaffold hopping,^{47,50} bioisosterism,⁵¹ and SAR modeling.^{52–54}

Molecules with different core structures but the same Reduced Graph have the same pharmacophoric representation and hence have an increased probability of being active toward a biological target. Examples from ChEMBL are shown in Table 1, which depicts molecules with different cores but the same Reduced Graph. In most cases the standard 2D similarity between the molecules is low. However, while the mapping from the full molecular graph to the Reduced Graph is well-defined algorithmically, the inverse is not. For example, the Reduced Graph may be used to cluster HTS output³⁸ and identify interesting groups of molecules. However, there is no direct method for generating alternative molecules that have the same Reduced Graph for idea generation. We present a novel application of the deep learning seq-to-seq approach popularized in the field of neural machine translation. We show that the seq-to-seq approach can generate chemically sensible molecules that share the same Reduced Graph, as exemplified by application to several data sets from ChEMBL. The coupled LSTM neural network architecture has been developed specifically for this task and involves several state-of-the-art features as described below. Since the Reduced Graph defines the region of interest in chemical space, once the initial translation has been learned, this removes the need for large training sets or extensive models required by other deep learning approaches. This approach has applications primarily in early lead discovery, expanding on a hit from HTS and in core hopping.

Table 1. Examples from ChEMBL of Molecular Structures Tested against the Same Target That Have the Same Reduced Graph but Different Core Structures^a

Target	RG	Exemplar 1	Exemplar 2	CFP7 similarity
DHODH	[Hf][Cu][Sc][V]=[Sc]	 CHEMBL3694248	 CHEMBL2012827	0.60
METAP2	[Y][Zn][Co][Sc][Mo] [Sc]([Mo])=[Hf]	 CHEMBL378724	 CHEMBL212733	0.91
SLC22A12	[V][Mo][Sc]([Ni][Sc] [Zn][Ni])[Zn][Ni]	 CHEMBL3683371	 CHEMBL3688188	0.70
SLC22A12	[Mo][Hf][Zn][V]=[Sc]	 CHEMBL3746459	 CHEMBL3747024	0.23
SLC9A1	[V][Ni][Hf][Sc][Ni] [Nb]	 CHEMBL2170616	 CHEMBL2170636	0.63
SLC9A1	[Sc][V][Ni][Nb]	 CHEMBL452446	 CHEMBL352933	0.25
SLC9A1	[Sc][Hf](=[V])[Ni][Nb]	 CHEMBL554125	 CHEMBL552865	0.51
SLC9A1	[Sc][Hf](=[Sc])[Ni] [Nb]	 CHEMBL554470	 CHEMBL541833	0.42
PLD1	[Sc][Re]=[Y][Zn] [Cu][Sc]=[Sc]	 CHEMBL1891232	 CHEMBL471257	0.67

^aThe Tanimoto similarity coefficient using the ChemAxon chemical fingerprint (see [Implementation details](#)) is also given.

Table 2. Reduced Graph Node Definitions and Occurrence in the Training Set

RG node type	description	occurrence	RG node type	description	occurrence
[Sc]	aromatic ring nonfeature	1151853	[Mo]	feature node negatively ionizable	85055
[Zn]	linker	741949	[Nb]	feature node positively ionizable	80844
[V]	aromatic ring acceptor	525751	[Co]	feature node donor	74960
[Ni]	feature node acceptor	453074	[Re]	aliphatic ring donor/acceptor	48780
[Cu]	feature node donor/acceptor	414019	[Mn]	aromatic ring positively ionizable	24178
[Hf]	aliphatic ring nonfeature	253289	[Ti]	aromatic ring donor	13797
[W]	aliphatic ring acceptor	143868	[Ta]	aliphatic ring donor	5298
[Y]	aliphatic ring positively ionizable	128942	[Zr]	aliphatic ring negatively ionizable	2247
[Cr]	aromatic ring donor/acceptor	109325	[Fe]	aromatic ring negatively ionizable	1700

METHODS

Representation of Molecules and Reduced Graphs.

The well-established SMILES representation^{39,55} allows for a single text string description of a molecule. SMILES uses a dictionary of characters to represent the graph of a molecule, with atoms as nodes and the aromatic, single, double, or triple bonds as the edges. By representing ring closures with pairs of numerical indices along the text string and containing side chains within parentheses, the linear SMILES string represents the full network graph of a molecule. The text nature of SMILES has been exploited in previous work^{19,21} using language models for molecular library generation.

Molecular Reduced Graphs are found by collapsing groups of connected atoms to a single superatom. Groups of atoms are characterized by chemical properties and assigned a node label. For ease of use of standard chemical toolkits, the node labels are normally chosen as elements not found in organic molecules: for example, [W] is chosen to represent an aliphatic ring acceptor and [Sc] is chosen to denote an aromatic ring with no additional features. Here we use the definitions of Harper et al.³⁸ as shown in Table 2. This also shows the Reduced Graph node occurrence in the training set (described below). Thus Reduced Graphs can be represented using the standard SMILES grammar. The resulting string has fewer characters than the corresponding SMILES string because of the collapse of atoms to a single superatom. However, this process leads to a many to one mapping, with many molecules having the same Reduced Graph. The conversion from Reduced Graph to SMILES is thus an ill-defined problem and there are currently no approaches to this other than through enumeration and filtering of large chemical spaces. We recast the problem as a language translation problem and use deep learning approaches to provide a robust methodology for converting from Reduced Graph to SMILES.

Effective language modeling uses tokenization of the text.⁵⁶ In the context of natural language processing, this usually involves creating a dictionary of words, such that sequences of characters are replaced with sequences of tokens which represent multiple characters in the sequence. Tokenization is used to simplify the text description of molecule and Reduced Graph SMILES representations, by capturing substrings of characters which necessarily occur together in a single token. For example, the atom chlorine is represented by the two characters Cl in a SMILES string, but for language modeling, can be collapsed into a single token. Similarly, the pseudo atoms in the Reduced Graph SMILES representation use three or four characters, such as [Cu], but can be represented by a single symbol. While language models can learn that the four characters “[”, “C”, “u”, and “]” need to occur together for a [Cu] node, the explicit tokenization

simplifies the representation and allows for improved language models.^{21,56}

Molecular Generation. The method used to generate molecular SMILES strings from Reduced Graphs builds on the techniques used to generate molecules using recurrent neural networks (RNNs). RNNs are neural networks where the outputs of layers contribute to the input of the same layer. When considering such networks applied to sequential data, it is common to express the recurrent loops as unfolded in time, with the output of the layer feeding laterally into the same layer at the next time step (see Figure 2). Approaches to SMILES generation using recurrent neural networks have centered on learning a probability distribution over all next possible symbols that extend an incomplete, tokenized, text string.^{19,21} It is possible to generate molecules by sampling from a learned probability distribution over symbols S , $P_\theta(S)$ symbol-by-symbol, noting that

$$P_\theta(S) = \prod_{j=1}^N P_\theta(s_j | s_{j-1}, s_{j-2}, \dots, s_0) \quad (1)$$

where parameters θ are learned from the training set, and s_j is the j th symbol in the string S . For the example of benzene, $S = \text{c1ccccc1}$, with $s_1 = \text{c}$, $s_2 = 1$, $s_3 = \text{c}$, and so on. Equation 1 shows that sampling a distribution of SMILES strings is equivalent to sampling a distribution of SMILES string tokens one-by-one, with the distribution of each token conditioned on the previous tokens in the sequence. Generative models using recurrent neural networks generate molecules by exploiting this equivalence and, for convenience, beginning with a special start-of-sequence symbol $s_0 = \langle \text{sos} \rangle$ and then evaluating the probabilities for each of the possible next symbols in sequence (see Figure 2). Sampling from this distribution, a second symbol is chosen and then the probabilities for the third symbol, conditioned on the first two selected symbols, are evaluated using the model. This process is completed until a special end-of-sequence symbol is selected.

In the context of a generator recurrent neural network, the probability $P_\theta(s_j | s_{j-1}, s_{j-2}, \dots, s_1)$ can be evaluated from the hidden state of the ultimate recurrent neural network layer via

$$P_\theta(s_j | s_{j-1}, s_{j-2}, \dots, s_1) = \text{softmax}(g(\mathbf{h}_j)) \quad (2)$$

where \mathbf{h}_j is the RNN hidden state and g is the projection which allows the neural network to output a vector whose length is equal to the size of the dictionary. The hidden state vector itself is updated by the RNN after each new character has been generated. The softmax function is a generalization of the logistic function to arbitrary dimension. Training RNNs is performed using a training set where the inputs are incomplete

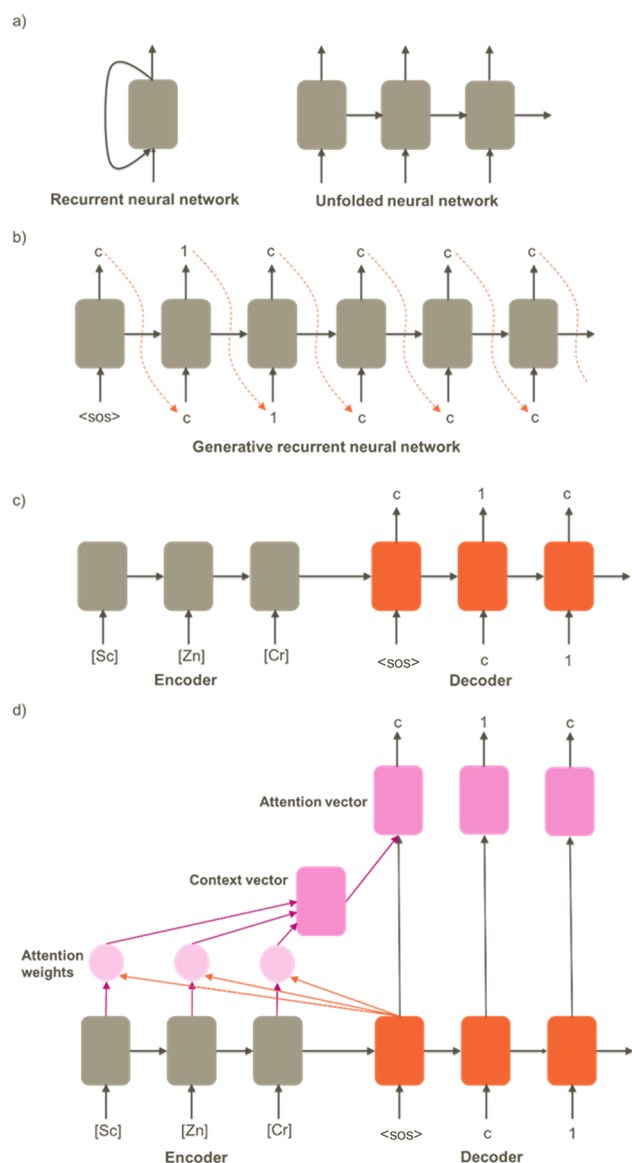


Figure 2. Schematic diagrams of generative models using recurrent neural networks. (a) Recurrent unit shown explicitly and unfolded in time. (b) Text generation network generating SMILES strings symbol-by-symbol by feeding the sampled output character back in at the next time step. (c) Sequence-to-sequence language translation model, “translating” a Reduced Graph SMILES string into a molecular SMILES string. Reduced Graph tokens (shown as [Sc], [Zn], and [Cr]) are fed into the input of the encoder network at consecutive time steps, and the decoder provides a token-by-token translation of the molecular SMILES string conditioned on previous tokens generated and the Reduced Graph encoding. (d) Sequence-to-sequence language translation model with attention mechanism addition. The input layer and ultimate projection layer on to the space of string tokens are omitted for simplicity.

and tokenized molecular SMILES strings and the targets are the next symbols in the string.

Reduced Graph Translation. Generation of a molecular SMILES string S which matches a specific Reduced Graph R requires us to be able to sample from

$$P(\text{SIR}) = P(s_1, s_2, \dots, s_N | r_1, r_2, \dots, r_L) \quad (3)$$

where r_j is the j th symbol in the Reduced Graph. To use a symbol-by-symbol generation approach, it is necessary to

estimate the conditional probability for a subsequent symbol conditioned not only on the previous selected symbols in the SMILES string, but also on a specific Reduced Graph R and use the following relationship:

$$P(\text{SIR}) = \prod_{j=1}^N P(s_j | s_{j-1}, s_{j-2}, \dots, s_0, R) \quad (4)$$

For this to be possible, the hidden state in the generator neural network needs also to capture the Reduced Graph R . Because the representation of a Reduced Graph is not of fixed length, an encoding of the Reduced Graph is found using an encoder recurrent neural network. The encoder network takes each of L Reduced Graph symbols as inputs in order and provides a fixed-length encoding via its hidden state \tilde{h}_L . In the most straightforward case,⁵⁶ the hidden state from the encoder neural network is passed to the generator or decoder network, such that $h_0 = \tilde{h}_L$.

Neural Machine Translation Training. A natural question which arises is how to train the encoder network to produce good quality fixed length encodings which allow the decoder network to generate molecular SMILES conditioned on a specific Reduced Graph SMILES. The answer is that both networks can be trained together using two input data sets and one target data set. Taking matched pairs of molecules and Reduced Graphs in SMILES format, the molecule data is preprocessed such that partial molecular SMILES strings containing all symbols up to the j th symbol can be provided to the networks with the associated Reduced Graph. The target data is the $j + 1$ th symbol in the molecular SMILES string associated with the input Reduced Graph. Training examples for each value of j up to N are created for each molecule. The Reduced Graph is supplied to the encoder which, after all the Reduced Graph symbols have been input, provides the initial internal state for the decoder. This initializes the state of the decoder which is then fed the first j symbols in the molecular SMILES string (see Figure 2). This network combination is trained to predict the $j + 1$ th symbol using categorical cross entropy and backpropagation through the coupled networks.

Generation of molecular SMILES using the trained model involves supplying the Reduced Graph to the encoder network to derive an initial hidden state for the decoder and then generating molecular SMILES symbol-by-symbol by sampling from the learned probability distribution over next symbols at each step, as in the molecular generation process described above. By sampling in this way, it is possible to generate a diverse set of molecules associated with a specific Reduced Graph.

Assessing the validity of neural machine translations is, in general, a further challenge. For example, in the case of translating French into English, there are multiple valid translations for each French phrase such that it is not possible to check the translated text against a single gold standard. This is not the case for molecule generation from Reduced Graphs: the inverse problem of generating a Reduced Graph is well-defined and each molecule has a unique Reduced Graph associated with it. This has two consequences: the first is that we can assess the performance of the model directly as the proportion of generated molecules which match the intended Reduced Graph. Furthermore, because we can quickly validate that a molecule collapses to the correct Reduced Graph, 100% accuracy is not required for the model to be valuable. Incorrect molecules can be filtered quickly.

Table 3. Composition of the Test Set with Respect to Its Overlap with the Training Set for Each Reduced Graph^a

Target	Reduced Graph	Total no. of cmpds	Total no. of RGs	Smiles/ RG	Categ ory	Exemplar
Alcohol dehydrogenase class III	[Sc][Re]([V])[Cr][Ni]	14	2	8	3	
	[Sc][Re]([V])[Cr][Mo]			6	3	
Ribonucleoside-diphosphate reductase M2 chain	[Co][V][Zn][Ni][Nb][Co]	7	1	7	3	
Apolipoprotein B-100	[Sc][Sc][Cu][Sc]=[Hf][Cu]	7	1	7	1	
C3H/3T3	[Sc][Co][Y][Sc]	7	1	7	1	
Glutamyl-peptide cyclotransferase-like protein	[Sc][Co][Zn]([Ni])[Co][Zn][V]	7	1	7	1	
Phosphodiesterase 4	[Sc][Zn][Cu][Sc][Zn][Hf]	63	4	14	1	
	[Sc][Zn]([Sc])[Ni]			20	1	
	[Sc][Zn][Sc]=[Sc][Cu][V]			14	2	
	[V][Cu][Sc]=[V]			15	1	
Uridine phosphorylase 1	[Sc][Zn][Cr]	26	1	26	1	

^aOne exemplar molecule is shown.**LSTM, Bidirectionality, and Attention Mechanisms.**

The approach described above can be modified in several ways to improve performance. The general description applies to all RNNs; in practice, the LSTM network⁴⁰ is used because of its ability to capture relationships between far-separated input symbols—something which is necessary for ring closures and branches. LSTM neurons have cell states in addition to the recurrent hidden state; the cell state is passed from encoder to decoder networks with the hidden state

Although Reduced Graph SMILES representations can take a canonical form, where the ordering of symbols in the representation is uniquely defined, graphs in general do not have a uniquely defined first and last node. This suggests that it might be sensible to reduce the constraint of learning from Reduced Graph SMILES from left to right. Bidirectional LSTM networks⁵⁷ use paired layers of LSTM neurons that read the input string from opposing directions. Hidden state vectors, generated by the encoder, are thus composed of the hidden states of both forward and backward LSTM neurons.

The final neural translation model enhancement used in this work is the attention mechanism.^{58,59} A limitation of the use of

the final encoder hidden state \tilde{h}_L for the encoding is that for long sequences it can be difficult to capture information about symbols within the sequence far from the end; this issue is only somewhat improved with the bidirectional LSTM. Attention mechanisms offer an improvement by using the full set of encoder hidden states \tilde{h}_i found after input of each Reduced Graph symbol r_i at each step. Specifically, a context vector with c_j at decoding step j is used in combination with the decoder hidden state h_j to create a new *attention* hidden state h'_j (see Figure 2). The context vector is a weighted average of encoder hidden states

$$c_j = \sum_i \alpha_{ij} \tilde{h}_i \quad (5)$$

where the attention weights α_{ij} are found using a scoring function following the multiplicative form of Luong et al.⁵⁸

$$d(h_j, \tilde{h}_i) = h_j^T W \tilde{h}_i \quad (6)$$

The attention weights are normalized using the softmax normalization

$$\alpha_{ij} = \frac{\exp d(\mathbf{h}_j, \tilde{\mathbf{h}}_i)}{\sum_i \exp d(\mathbf{h}_j, \tilde{\mathbf{h}}_i)} \quad (7)$$

Finally, an *attention vector* is found from the context vector \mathbf{c}_j via

$$\mathbf{h}'_j = \tanh(\mathbf{W}_c[\mathbf{c}_j; \mathbf{h}_j]) \quad (8)$$

where the hidden state \mathbf{h}_j and context vector \mathbf{c}_j are concatenated. The results presented in this work make use of an encoder network with a bidirectional LSTM with 1024 neurons in total and a decoder network with 1024 LSTM neurons. The attention mechanism of Luong et al. discussed above is used in the results presented.

Data Sets. Compounds were taken from ChEMBL 23.¹² Reduced Graphs were created using an in-house implementation with the ChemAxon JChem toolkit (17.11.0)⁶⁰ using the approach described in the work of Harper et al.³⁸ All SMILES were converted to parent structures after removing any salts and keeping only the largest fragments. Structures with bad valency, isotope labeled compounds, compounds containing atoms other than N, O, C, S, F, Cl, Br, and I, and compounds with poorly defined bonds were removed. SMILES were canonicalized, InChIs were calculated, and Rule-of-5⁶¹ filtering criteria were applied. The final list was deduplicated using the InChIKeys to give 798 374 unique SMILES. This set was divided into two parts: a training set containing 798 243 unique SMILES and 139 312 unique Reduced Graphs. A test set containing 131 unique SMILES was created. From the training set 1597 compounds (0.2%) were selected at random as a development set to monitor training performance. A test set was constructed by selecting 131 compound IDs having assay data for at least one target and meeting one of the criteria defined below. These were compounds associated with the following targets: alcohol dehydrogenase class III, apolipoprotein B-100, C3H/3T3, glutamyl-peptide cyclotransferase-like protein, phosphodiesterase 4, ribonucleoside-diphosphate reductase M2 chain, uridine phosphorylase 1. The test set contains 11 unique Reduced Graphs that can be assigned to three categories depending on their occurrence in the training set:

1. Reduced Graphs seen previously in the training data. (7 Reduced Graphs)
2. A Reduced Graph that is not present in the training data but is a subgraph of a Reduced Graph in the training data. (1 Reduced Graph)
3. Neither the Reduced Graph nor its supergraph is present in the training data. (3 Reduced Graphs)

The test set Reduced Graphs with their targets and characteristics are listed in Table 3.

Training and Evaluation. The model was trained until the loss function evaluated on the small development data set, withheld from the training set, showed no further decrease. When evaluating the performance of the model, three tasks of increasing complexity were considered:

1. Can the model generate novel molecules for a Reduced Graph seen previously in the training data?
2. Can the model generate novel molecules for a Reduced Graph not seen previously in the training data but that exists as a subgraph of another Reduced Graph in the training set?
3. Can the model generate novel molecules for a Reduced Graph not seen previously in the training data and which is not a subgraph of any within the training set?

3. Can the model generate novel molecules for a Reduced Graph not seen previously in the training data and which is not a subgraph of any within the training set?

Evaluation consists of inputting the Reduced Graph to the trained network and generating a user-defined number of SMILES strings. The SMILES strings are canonicalized and checked for uniqueness. The Reduced Graph for each SMILES is also generated to allow comparison with the input Reduced Graph.

Implementation Details. The algorithms were encoded in Python 2.7 on RedHat 7 using the deep learning libraries Tensorflow version 1.4 and the open source seq-to-seq framework provided by Britz et al.⁶² The default training parameters of the framework were used to train the neural network. Computations were run on a single Tesla-P100 GPU node. BIOVIA Pipeline Pilot⁶³ Molecule from SMILES component was used to filter lexically correct SMILES within the generation code.

Similarity calculations used the Tanimoto similarity coefficient using the ChemAxon path-based fingerprint (2048 bits, up to path length 7, 4 bits per path).

RESULTS

The initial network was trained for 20 epochs using the SMILES/Reduced Graph pairs of the training set, using the development set to monitor performance. The loss curve is shown in Figure 3. The final model was then used to generate

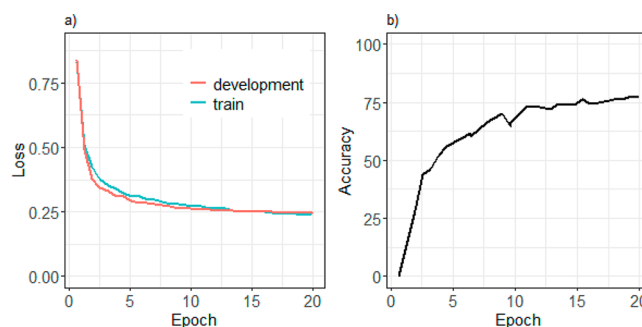


Figure 3. (a) Loss for the train and development data sets. (b) Percentage of generated SMILES with the correct Reduced Graph on the development data set.

SMILES based on each Reduced Graph in the test set (Table 3). Here, 250 000 SMILES strings were generated for each input Reduced Graph. The results for each Reduced Graph are shown in Table 4. It should be noted that the generated SMILES strings may not be unique (the same string can be generated multiple times) and are not necessarily canonical SMILES. Thus, the strings were further processed by canonicalizing with ChemAxon and removing duplicates. Counts after processing are shown in Table 4.

The algorithm was able to produce SMILES that match the target Reduced Graph for all the input Reduced Graphs. The proportion of lexically correct SMILES generated, 96.7% across all test set reduced graphs, is on a par with other RNN based approaches.¹⁹ The challenge is generating SMILES corresponding to the input Reduced Graph. However, as the generated SMILES can be filtered readily by comparison with the input Reduced Graph and SMILES generation is fast (5 min for 100 000 SMILES), this is not a problem. In fact, it is often the case that SMILES not matching the input Reduced

Table 4. Measures of the Performance for the Test Set against Each Reduced Graph^a

no.	reduced graph	no. of SMILES in training set ^b	category	number (%) of unique canonical SMILES strings	number of unique canonical SMILES matching target RG	total no. of test set compounds generated	highest similarity to test set compound
1	[Sc][Re]([V]) [Cr][Ni]	0	3	146024 (58%)	20350	0/8	0.92
2	[Sc][Re]([V]) [Cr][Mo]	0	3	99160 (40%)	18726	0/6	0.86
3	[Co][V][Zn][Ni] [Nb][Co]	0	3	98876 (40%)	4248	0/7	0.93
4	[Sc][Zn][Sc] =[Sc][Cu][V]	0/2	2	105134 (42%)	68001	1/14	1
5	[Sc][Sc][Cu][Sc] =[Hf][Cu]	1/1	1	85224 (34%)	58368	0/7	0.97
6	[Sc][Co][Y][Sc]	15/12	1	68243 (27%)	22629	6/7	1
7	[V][Cu][Sc]=[V]	231/140	1	87613 (35%)	64472	4/15	1
8	[Sc][Zn]([Sc]) [Ni]	309/1079	1	107954 (43%)	84922	8/20	1
9	[Sc][Co][Zn] ([Ni])[Co][Zn] [V]	7/0	1	139304 (56%)	29283	0/7	0.94
10	[Sc][Zn][Cu][Sc] [Zn][Hf]	3/7	1	68720 (27%)	56672	8/14	1
11	[Sc][Zn][Cr]	1471/2462	1	98404 (39%)	85248	17/26	1

^aThe total number of generated SMILES was 250 000 for each structure. ^bNo. of SMILES in training set shows the number of SMILES with that Reduced Graph/number of SMILES where the Reduced Graph is a superstructure of the query Reduced Graph. Generated SMILES count refers to unique SMILES strings after canonicalization.

Graph differ by just a single node so they may in themselves be interesting compounds.

The diversity of the generated SMILES has been evaluated in two ways:

1. To assess the internal diversity of the generated molecules, for each input Reduced Graph, the set of unique canonical SMILES with the correct Reduced Graph was clustered using the sphere exclusion algorithm^{64,65} at two radii, 0.75 and 0.85, using ChemAxon path based fingerprints (vide supra). The latter cutoff was selected as it is used in the GSK collection model,⁶⁶ and the lower threshold gives an indication of the breadth of the distribution. Table 5 shows the number of clusters for each set of generated SMILES. On this criterion the generated compounds can be considered as internally diverse, covering large numbers of clusters even at quite a low similarity threshold. Thus, the algorithm is not simply generating very close analogues, for example by adding methyl groups or halogen atoms.
2. To assess whether the algorithm is simply making small modifications to compounds in the training set, the generated SMILES were compared to the training set. Figure 4 shows the distribution of the nearest neighbor similarity of each generated compound to the training set, categorized by each input Reduced Graph. The nearest neighbor similarity values show a broad range of values with the peak well below 0.8 in most cases, the commonly accepted threshold for this fingerprint.^{67,68} This demonstrates that the compounds are diverse with respect to the training set, and the algorithm is not memorizing whole molecules.

DISCUSSION

The results show that the method is successful in translating from Reduced Graph to SMILES. To our knowledge this is the

Table 5. Cluster Counts for the Generated Compounds That Match the Input Reduced Graph^a

no.	reduced graph	number of compounds	number of clusters at 0.85 Tanimoto	number of clusters at 0.75 Tanimoto
1	[Sc][Re]([V]) [Cr][Ni]	20350	6071	3001
2	[Sc][Re]([V]) [Cr][Mo]	18726	5049	2389
3	[Co][V][Zn] [Ni][Nb] [Co]	4248	1572	991
4	[Sc][Zn][Sc] =[Sc][Cu] [V]	68001	19505	8443
5	[Sc][Sc][Cu] [Sc]=[Hf] [Cu]	58368	12842	5222
6	[Sc][Co][Y] [Sc]	22629	6763	2937
7	[V][Cu][Sc] =[V]	64472	26454	13146
8	[Sc][Zn]([Sc]) [Ni]	84922	35668	18697
9	[Sc][Co][Zn] ([Ni])[Co] [Zn][V]	29283	12715	7208
10	[Sc][Zn][Cu] [Sc][Zn][Hf]	56672	12586	4883
11	[Sc][Zn][Cr]	85248	36117	18876

^aClustering was performed using the sphere exclusion algorithm at two Tanimoto thresholds.

first time that this sort of translation has been done explicitly in the context of cheminformatics. The deep learning algorithm that has been presented uses several state-of-the-art technical details that are novel in the context of molecule generation. Of particular interest is the relative simplicity of the approach compared to larger more complex networks required for autoencoders or reinforcement learning approaches. Although, combining with reinforcement learning could be an interesting

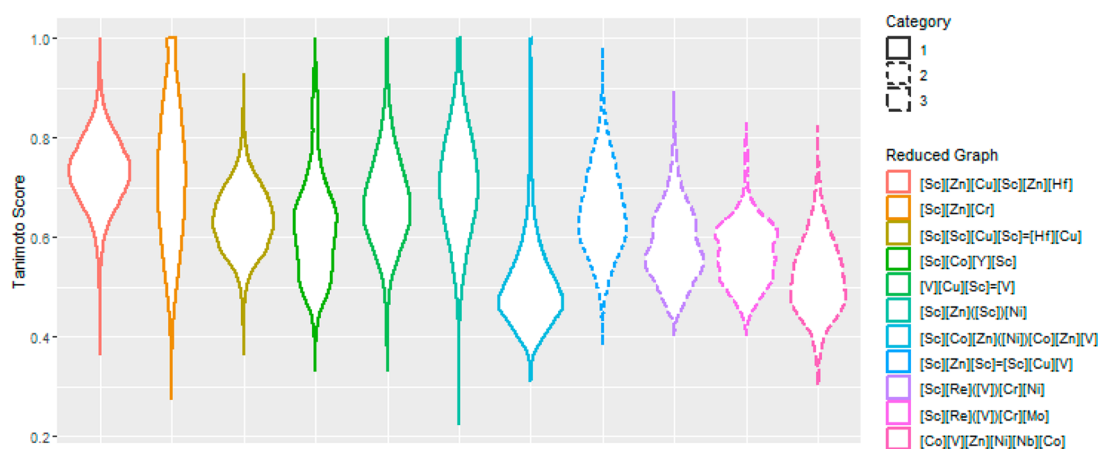


Figure 4. Distribution of nearest neighbor similarity values for the generated compounds to the training set. Border types correspond to the Reduced Graph category shown in Table 4.

avenue for further exploration to further focus the properties of the generated molecules.

It is apparent from Table 4 that the generation of molecules is not the challenge here. Generating molecules with the specific input Reduced Graph can be more difficult, as illustrated by the proportion of molecules that match the required Reduced Graph. However, the generative aspects of the algorithm are robust and fast enough for this not to be an issue. In fact, even the worst case of 4000 unique compounds from 250 000 requested would be sufficient for most programs to select ideas for follow-up. As shown in Table 5 and Figure 4, even with the constraints, the generated molecules are diverse within themselves and with respect to the training set. Therefore, the algorithm is not overfit or memorizing the training set.

There are interesting trends that can be extracted from Table 4. The results indicate that the proportion of unique SMILES matching the target Reduced Graph is larger for category 1 and 2 Reduced Graphs (those found in training set, Reduced Graph nos. 4–11) compared to category 3 Reduced Graphs (not found in training set, Reduced Graph nos. 1–3). Even then, some of the Reduced Graphs which were present in the training set did not perform as well as others (nos. 6 and 9). There does not appear to be a clear explanation for this: while Reduced Graph nos. 6 and 9 are poorly represented in the data set, nos. 5 and 10 are as well. Reduced Graph 9 has nine nodes, so it is complex; however, Reduced Graph 6 only has four nodes.

To investigate this further, the occurrence of each Reduced Graph node in the training set is shown in Table 2. Input Reduced Graphs of category 3 (not present in the training set) produced unique canonical SMILES in similar proportion to the other Reduced Graphs. However, the number of generated SMILES matching the input Reduced Graph is generally lower. Two of the category 3 Reduced Graphs (nos. 1 and 2) are very similar, differing only in one superatom. These two Reduced Graphs have 5 nodes and 1 branching point. They generate similar numbers of unique canonicalized SMILES strings matching the input Reduced Graph. (20 402 and 18 726, respectively). Interestingly, there is a much larger difference in the total number of unique canonical SMILES generated (~146 000 and ~99 000, respectively). Referring to Table 2, the node [Ni] in Reduced Graph no. 1 has about five times the occurrence of the node [Mo] in Reduced Graph no. 2. Perhaps

this lower occurrence is compromising the generation. On the other hand, the chemistry associated with the terminal [Mo], feature node negatively ionizable, is also very restrictive being limited to carboxylate and terminal acidic sulphonamides. A similar tendency can be seen for all the substructures containing [Mo]. Reduced Graph no. 3 is larger than the previous Reduced Graphs by 1 node, containing the superatom [Nb] with similar occurrence to [Mo]. This Reduced Graph produces similar valid SMILES ratio as Reduced Graph no. 2. Reduced Graph no. 4 (category 2) produced a large number of SMILES satisfying the input Reduced Graph. The high success ratio (68 001/250 000 correct unique SMILES) of this Reduced Graph can be attributed to the fact that the component Reduced Graph nodes all have a high occurrence in the training set, even if there are no examples of this Reduced Graph in the training set and only 2 SMILES with a Reduced Graph superstructure of Reduced Graph no. 4. This illustrates further the ability of the algorithm to generalize. A more detailed analysis of training set occurrence of Reduced Graph substructures for the input Reduced Graphs is given in the Supporting Information. This can be informative for the success rates on some of the other queries. Reduced Graph nos. 6–8 all have 4 nodes, and their order in performance correlates with their occurrence in the training set. For example, [Co][Y], present in Reduced Graph no. 6, is one of the least abundant Reduced Graph substructures and could explain the lower number of generated SMILES that match the input Reduced Graph in this case.

In summary, the performance in terms of number of generated SMILES that match the input Reduced Graph seems to have some relationship to both the occurrence of the node types in the training set and the specificity of the chemistry defined in a particular node. However, the results show that the generated molecules are diverse both within themselves and relative to the training set. The algorithm is capable of generalization and is useful for exploration of chemistry ideas both at the hit to lead phase, when only a small number of analogues are known, and in scaffold hopping in lead optimization.

CONCLUSIONS

We have presented a novel deep learning based approach to molecule generation based on methods originally developed for language translation. This work has led to a solution to one

of the challenges of cheminformatics, that is converting from Reduced Graph to SMILES, which is a one-to-many problem. The approach offers an alternative to autoencoders and reinforcement learning for generating compounds in a defined region of chemical space that rely on relatively large numbers of training molecules. Instead the chemical space is implicitly defined by the Reduced Graph so only a single active molecule is required.

We have shown that the method is able to generate valid SMILES for Reduced Graphs not present in the training set. The generated SMILES are diverse both within themselves and relative to the training set, thus demonstrating the ability of the method to generalize.

■ ASSOCIATED CONTENT

■ Supporting Information

The Supporting Information is available free of charge on the ACS Publications website at DOI: 10.1021/acs.jcim.8b00626.

Distribution of Reduced Graph subgraphs in the training data (XLSX)

■ AUTHOR INFORMATION

Corresponding Author

*Email: stephen.d.pickett@gsk.com.

ORCID

Stephen D. Pickett: 0000-0002-0958-9830

Funding

S.D.P. and P.P. are employees of GlaxoSmithKline. N.A. and S.G. are employees of Tessella.

Notes

The authors declare no competing financial interest.

■ ACKNOWLEDGMENTS

This work was conducted at GSK as part of the GSK-Tessella Analytics Partnership.

■ ABBREVIATIONS

LSTM, long short-term memory; RNN, recurrent neural network

■ REFERENCES

- (1) Manas, E. S.; Green, D. V. S. CADD medicine: design is the potion that can cure my disease. *J. Comput.-Aided Mol. Des.* **2017**, *31*, 249–253.
- (2) Segall, M. Advances in multiparameter optimization methods for de novo drug design. *Expert Opin. Drug Discovery* **2014**, *9*, 803–817.
- (3) Schneider, G. Automating drug discovery. *Nat. Rev. Drug Discovery* **2017**, *17*, 97.
- (4) Hartenfeller, M.; Schneider, G. De Novo Drug Design. In *Chemoinformatics and Computational Chemical Biology*, Bajorath, J., Ed.; Humana Press: Totowa, NJ, 2011; pp 299–323.
- (5) Westhead, D. R.; Clark, D. E.; Frenkel, D.; Li, J.; Murray, C. W.; Robson, B.; Waszkowycz, B. PRO_LIGAND: An Approach to De Novo Molecular Design. 3. A Genetic Algorithm for Structure Refinement. *J. Comput.-Aided Mol. Des.* **1995**, *9*, 139–148.
- (6) Bohm, H. J. The computer program LUDI: A new method for the de novo design of enzyme inhibitors. *J. Comput.-Aided Mol. Des.* **1992**, *6*, 61–78.
- (7) Bohm, H. J. LUDI: Rule-based Automatic Design of New Substituents for Enzyme Inhibitor Leads. *J. Comput.-Aided Mol. Des.* **1992**, *6*, 593–606.
- (8) Bohm, H. J. A Novel Computational Tool for Automated Structure-Based Drug Design. *J. Mol. Recognit.* **1993**, *6*, 131–137.

(9) Lewell, X. Q.; Judd, D. B.; Watson, S. P.; Hann, M. M. RECAP - Retrosynthetic Combinatorial Analysis Procedure: A Powerful New Technique for Identifying Privileged Molecular Fragments with Useful Applications in Combinatorial Chemistry. *J. Chem. Inf. Comput. Sci.* **1998**, *38*, 511–522.

(10) Degen, J.; Wegscheid-Gerlach, C.; Zaliani, A.; Rarey, M. On the Art of Compiling and Using 'Drug-Like' Chemical Fragment Spaces. *ChemMedChem* **2008**, *3*, 1503–1507.

(11) Knowledge-based de novo molecular design using icsynth frp <http://www.haxel.com/icic/2014/Programme/monday-13-oct-2014> (accessed 22 August 2018).

(12) Gaulton, A.; Hersey, A.; Nowotka, M.; Bento, A. P.; Chambers, J.; Mendez, D.; Mutowo, P.; Atkinson, F.; Bellis, L. J.; Cibrián-Uhalte, E.; Davies, M.; Dedman, N.; Karlsson, A.; Magariños, M. P.; Overington, J. P.; Papadatos, G.; Smit, I.; Leach, A. R. The ChEMBL database in 2017. *Nucleic Acids Res.* **2017**, *45*, D945–D954.

(13) 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 Res.* **2016**, *44*, D1202–D1213.

(14) Sterling, T.; Irwin, J. J. ZINC 15 – Ligand Discovery for Everyone. *J. Chem. Inf. Model.* **2015**, *55*, 2324–2337.

(15) Xu, Y.; Ma, J.; Liaw, A.; Sheridan, R. P.; Svetnik, V. Demystifying Multitask Deep Neural Networks for Quantitative Structure–Activity Relationships. *J. Chem. Inf. Model.* **2017**, *57*, 2490.

(16) Blaschke, T.; Olivecrona, M.; Engkvist, O.; Bajorath, J.; Chen, H. Application of Generative Autoencoder in De Novo Molecular Design. *Mol. Inf.* **2018**, *37*, 1700123.

(17) 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.

(18) Neil, D.; Segler, M.; Guasch, L.; Ahmed, M.; Plumbley, D.; Sellwood, M.; Brown, N. Exploring Deep Recurrent Models with Reinforcement Learning for Molecule Design. In *ICLR 2018 Workshop. 6th International Conference on Learning Representations*, Vancouver, BC, Canada, 2018.

(19) Segler, M. H. S.; Kogej, T.; Tyrchan, C.; Waller, M. P. Generating Focused Molecule Libraries for Drug Discovery with Recurrent Neural Networks. *ACS Cent. Sci.* **2018**, *4*, 120–131.

(20) Ertl, P.; Lewis, R.; Martin, E.; Polyakov, V. In silico generation of novel, drug-like chemical matter using the LSTM neural network. *arXiv.org* **2017**, 1712.07449.

(21) Gupta, A.; Müller, A. T.; Huisman, B. J. H.; Fuchs, J. A.; Schneider, P.; Schneider, G. Generative Recurrent Networks for De Novo Drug Design. *Mol. Inf.* **2018**, *37*, 1700111.

(22) Jørgensen, P. B.; Schmidt, M. N.; Winther, O. Deep Generative Models for Molecular Science. *Mol. Inf.* **2018**, *37*, 1700133.

(23) Merk, D.; Friedrich, L.; Grisoni, F.; Schneider, G. De Novo Design of Bioactive Small Molecules by Artificial Intelligence. *Mol. Inf.* **2018**, *37*, 1700153.

(24) Putin, E.; Asadulaev, A.; Ivanenkov, Y.; Aladinskiy, V.; Sanchez-Lengeling, B.; Aspuru-Guzik, A.; Zhavoronkov, A. Reinforced Adversarial Neural Computer for de Novo Molecular Design. *J. Chem. Inf. Model.* **2018**, *58*, 1194–1204.

(25) Müller, A. T.; Hiss, J. A.; Schneider, G. Recurrent Neural Network Model for Constructive Peptide Design. *J. Chem. Inf. Model.* **2018**, *58*, 472–479.

(26) Nagarajan, D.; Nagarajan, T.; Roy, N.; Kulkarni, O.; Ravichandran, S.; Mishra, M.; Chakravorty, D.; Chandra, N. Computational antimicrobial peptide design and evaluation against multidrug-resistant clinical isolates of bacteria. *J. Biol. Chem.* **2018**, *293*, 3492–3509.

(27) Grisoni, F.; Neuhaus, C. S.; Gabernet, G.; Müller, A. T.; Hiss, J. A.; Schneider, G. Designing Anticancer Peptides by Constructive Machine Learning. *ChemMedChem* **2018**, *13*, 1300–1302.

(28) Cox, R.; Green, D. V. S.; Luscombe, C. N.; Malcolm, N.; Pickett, S. D. QSAR workbench: automating QSAR modeling to drive compound design. *J. Comput.-Aided Mol. Des.* **2013**, *27*, 321–336.

- (29) Davis, A. M.; Wood, D. J. Quantitative Structure-Activity Relationship Models That Stand the Test of Time. *Mol. Pharmaceutics* **2013**, *10*, 1183–1190.
- (30) Dixon, S. L.; Duan, J.; Smith, E.; Barga, C. D. V.; Sherman, W.; Repasky, M. P. AutoQSAR: an automated machine learning tool for best-practice quantitative structure–activity relationship modeling. *Future Med. Chem.* **2016**, *8*, 1825–1839.
- (31) Nantasenamat, C.; Worachartcheewan, A.; Jamsak, S.; Preeyanon, L.; Shoombuatong, W.; Simeon, S.; Mandi, P.; Isarankura-Na-Ayudhya, C.; Prachayasittikul, V. AutoWeka: Toward an Automated Data Mining Software for QSAR and QSPR Studies. In *Artificial Neural Networks*, Cartwright, H., Ed.; Springer New York: New York, NY, 2015; pp 119–147.
- (32) Carrió, P.; López, O.; Sanz, F.; Pastor, M. eTOXlab, an open source modeling framework for implementing predictive models in production environments. *J. Cheminf.* **2015**, *7*, 8.
- (33) Lang, T.; Flachsenberg, F.; von Luxburg, U.; Rarey, M. Feasibility of Active Machine Learning for Multiclass Compound Classification. *J. Chem. Inf. Model.* **2016**, *56*, 12–20.
- (34) Oglic, D.; Oatley, S. A.; Macdonald, S. J. F.; McInally, T.; Garnett, R.; Hirst, J. D.; Gärtner, T. Active Search for Computer-aided Drug Design. *Mol. Inf.* **2018**, *37*, 1700130.
- (35) Reker, D.; Schneider, P.; Schneider, G.; Brown, J. Active learning for computational chemogenomics. *Future Med. Chem.* **2017**, *9*, 381–402.
- (36) Gillet, V. J.; Downs, G. M.; Holliday, J. D.; Lynch, M. F.; Dethlefsen, W. Computer storage and retrieval of generic chemical structures in patents. 13. Reduced graph generation. *J. Chem. Inf. Model.* **1991**, *31*, 260–270.
- (37) Gillet, V. J.; Willett, P.; Bradshaw, J. Similarity Searching Using Reduced Graphs. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 338–345.
- (38) Harper, G.; Bravi, G. S.; Pickett, S. D.; Hussain, J.; Green, D. V. S. The Reduced Graph Descriptor in Virtual Screening and Data-Driven Clustering of High-Throughput Screening Data. *J. Chem. Inf. Comput. Sci.* **2004**, *44*, 2145–2156.
- (39) Weininger, D. SMILES, a Chemical Language and Information System. 1. Introduction to Methodology and Encoding Rules. *J. Chem. Inf. Model.* **1988**, *28*, 31–36.
- (40) Hochreiter, S.; Schmidhuber, J. Long Short-Term Memory. *Neural Computation* **1997**, *9*, 1735–1780.
- (41) Malhotra, P.; Ramakrishnan, A.; Anand, G.; Vig, L.; Agarwal, P.; Shroff, G. LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection. *arXiv.org* **2016**, 1607.00148.
- (42) Graves, A.; Mohamed, A. r.; Hinton, G. Speech recognition with deep recurrent neural networks. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, May 26–31, 2013; pp 6645–6649.
- (43) Graves, A. Generating Sequences With Recurrent Neural Networks. *arXiv.org* **2013**, 1308.0850.
- (44) Olivecrona, M.; Blaschke, T.; Engkvist, O.; Chen, H. Molecular de-novo design through deep reinforcement learning. *J. Cheminf.* **2017**, *9*, 48.
- (45) Altae-Tran, H.; Ramsundar, B.; Pappu, A. S.; Pande, V. Low Data Drug Discovery with One-Shot Learning. *ACS Cent. Sci.* **2017**, *3*, 283–293.
- (46) Rarey, M.; Dixon, J. S. Feature Trees: A New Molecular Similarity Measure Based On Tree Matching. *J. Comput.-Aided Mol. Des.* **1998**, *12*, 471–490.
- (47) Stiefl, N.; Watson, I. A.; Baumann, K.; Zaliani, A. ErG: 2D Pharmacophore Descriptors for Scaffold Hopping. *J. Chem. Inf. Model.* **2006**, *46*, 208–220.
- (48) Barker, E. J.; Gardiner, E. J.; Gillet, V. J.; Kitts, P.; Morris, J. Further Development of Reduced Graphs for Identifying Bioactive Compounds. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 346–356.
- (49) Gunera, J.; Kolb, P. Fragment-based similarity searching with infinite color space. *J. Comput. Chem.* **2015**, *36*, 1597–1608.
- (50) Barker, E. J.; Buttar, D.; Cosgrove, D. A.; Gardiner, E. J.; Kitts, P.; Willett, P.; Gillet, V. J. Scaffold Hopping Using Clique Detection Applied to Reduced Graphs. *J. Chem. Inf. Model.* **2006**, *46*, 503–511.
- (51) Birchall, K.; Gillet, V. J.; Willett, P.; Ducrot, P.; Luttmann, C. Use of Reduced Graphs To Encode Bioisosterism for Similarity-Based Virtual Screening. *J. Chem. Inf. Model.* **2009**, *49*, 1330–1346.
- (52) Birchall, K.; Gillet, V. J.; Harper, G.; Pickett, S. D. Evolving Interpretable Structure-Activity Relationships. 1. Reduced Graph Queries. *J. Chem. Inf. Model.* **2008**, *48*, 1543–1557.
- (53) Birchall, K.; Gillet, V. J.; Harper, G.; Pickett, S. D. Evolving Interpretable Structure-Activity Relationship Models. 2. Using Multiobjective Optimization To Derive Multiple Models. *J. Chem. Inf. Model.* **2008**, *48*, 1558–1570.
- (54) Wollenhaupt, S.; Baumann, K. inSARA: Intuitive and Interactive SAR Interpretation by Reduced Graphs and Hierarchical MCS-Based Network Navigation. *J. Chem. Inf. Model.* **2014**, *54*, 1578–1595.
- (55) Weininger, D.; Weininger, A.; Weininger, J. L. Smiles. 2. Algorithm for the Generation of Unique SMILES Notation. *J. Chem. Inf. Model.* **1989**, *29*, 97–101.
- (56) Cho, K.; van Merriënboer, B.; Gulcehre, C.; Bahdanau, D.; Bougares, F.; Schwenk, H.; Bengio, Y. Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv.org* **2014**, 1406.1078.
- (57) Graves, A.; Fernandez, S.; Schmidhuber, J. Bidirectional LSTM Networks for Improved Phoneme Classification and Recognition. In *Artificial Neural Networks: Formal Models and Their Applications – ICANN 2005*; Duch, W.; Kacprzyk, J.; Oja, E.; Zdrożny, S., Eds.; Springer: Berlin, 2005; pp 799–804.
- (58) Luong, T.; Pham, H.; Manning, C. D. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, Association for Computational Linguistics: Lisbon, Portugal, 2015; pp 1412–1421.
- (59) Bahdanau, D.; Cho, K.; Bengio, Y. Neural machine translation by jointly learning to align and translate. *arXiv.org* **2014**, 1409.0473.
- (60) ChemAxon. <http://www.chemaxon.com>. (accessed 14 Nov 2018).
- (61) Lipinski, C. A.; Lombardo, F.; Dominy, B. W.; Feeney, P. J. Experimental and Computational Approaches to Estimate Solubility and Permeability in Drug Discovery and Development Settings. *Adv. Drug Delivery Rev.* **1997**, *23*, 3–25.
- (62) Britz, D.; Goldie, A.; Luong, M.-T.; Le, Q. Massive exploration of neural machine translation architectures. *arXiv.org* **2017**, 1703.03906.
- (63) Dassault Systèmes BIOVIA. *BIOVIA Pipeline Pilot 17.2.0.1361*, Release 2017; San Diego: Dassault Systèmes, 2017.
- (64) Taylor, R. Simulation Analysis of Experimental Design Strategies for Screening Random Compounds as Potential New Drugs and Agrochemicals. *J. Chem. Inf. Model.* **1995**, *35*, 59–67.
- (65) Butina, D. Unsupervised Data Base Clustering Based on Daylight's Fingerprint and Tanimoto Similarity: A Fast and Automated Way To Cluster Small and Large Data Sets. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 747–750.
- (66) Harper, G.; Pickett, S. D.; Green, D. V. S. Design of a Compound Screening Collection for use in High Throughput Screening. *Comb. Chem. High Throughput Screening* **2004**, *7*, 63–70.
- (67) Brown, R. D.; Martin, Y. C. An Evaluation of Structural Descriptors and Clustering Methods for use in Diversity Selection. *SAR QSAR Environ. Res.* **1998**, *8*, 23–39.
- (68) Papadatos, G.; Cooper, A. W. J.; Kadirkamanathan, V.; Macdonald, S. J. F.; McLay, I. M.; Pickett, S. D.; Pritchard, J. M.; Willett, P.; Gillet, V. J. Analysis of Neighborhood Behavior in Lead Optimization and Array Design. *J. Chem. Inf. Model.* **2009**, *49*, 195–208.