SELFIES: a robust representation of semantically constrained graphs with an example application in chemistry

Mario Krenn[1,2,3*], **Florian Häse**[1,2,3,4], **AkshatKumar Nigam**[2], **Pascal Friederich**[1,5], **Alán Aspuru-Guzik**[1,2,3,6 *]

[1]Department of Chemistry, University of Toronto, Canada.
[2]Department of Computer Science, University of Toronto, Canada.
[3]Vector Institute for Artificial Intelligence, Toronto, Canada.
[4]Department of Chemistry and Chemical Biology, Harvard University, Cambridge, USA.
[5]Institute of Nanotechnology, Karlsruhe Institute of Technology, Germany.
[6]Canadian Institute for Advanced Research (CIFAR) Senior Fellow, Toronto, Canada

Abstract

Graphs are ideal representations of complex, relational information. Their applications span diverse areas of science and engineering, such as Feynman diagrams in fundamental physics, the structures of molecules in chemistry or transport systems in urban planning. Recently, many of these examples turned into the spotlight as applications of machine learning (ML). There, common challenges to the successful deployment of ML are domain-specific constraints, which lead to semantically constrained graphs. While much progress has been achieved in the generation of valid graphs for domain- and model-specific applications, a general approach has not been demonstrated yet. Here, we present a general-purpose, sequence-based, robust representation of semantically constrained graphs, which we call SELFIES (SELF-referencIng Embedded Strings). SELFIES are based on a Chomsky type-2 grammar, augmented with two self-referencing functions. We demonstrate their applicability to represent chemical compound structures and compare them to perhaps the most popular 2D representation, SMILES, and other important baselines. We find stronger robustness against character mutations while still maintaining similar chemical properties. Even entirely random SELFIES produce semantically valid graphs in most of the cases. As feature representation in variational autoencoders, SELFIES provide a substantial improvement in the task of in reconstruction, validity, and diversity. We anticipate that SELFIES allow for direct applications in ML, without the need for domain-specific adaptation of model architectures. SELFIES are not limited to the structures of small molecules, and we show how to apply them to two other examples from the sciences: representations of DNA and interaction graphs for quantum mechanical experiments.

1 Introduction

Deep generative models have gained considerable attention in recent years with notable results in generating images [1], sound [2], text [3]. These developments have sparked interest in the field of chemistry and materials science, as it allows for the automated computational design of new

*Correspondence to: mario.krenn@utoronto.ca, alan@aspuru.com; Source: https://github.com/aspuru-guzik-group/selfies

Preprint. Under review.

drugs and materials [4, 5]. Different generative models have been suggested for this task, such as variational autoencoders (VAEs) [6], generative adversarial networks (GANs) [7, 8, 9] or sampling from recurrent neural networks (RNNs) [10].

Objects of interest in the natural sciences are often complex structures, which can be expressed as graphs with additional domain-specific semantic constraints. Some concrete examples are the structures of molecules in chemistry (element dependent bond limitations), quantum optical experiments in physics (component dependent connectivity) or DNA and RNA in biology (nucleobase-dependent connectivity).

These constraints pose major challenges for generative models, as their violation leads to unphysical and thus invalid results. A popular research question has been: *How to design deep generative models for semantically constrained graphs?* (see, for instance, [11, 12]).

Here we ask a related, but conceptually different question: *How can we represent the information encoded in a semantically constrained graph in a simple, robust, deterministic, domain-independent, model-independent way?* An answer to this question would allow us to use, as a direct input, our representation into existing (and even future) models without any model-dependent adaptation, and thus has the potential to be transferable across a spectrum of applications.

In this work, we present SELFIES (SELF-referencIng Embedded Strings), a sequence-based representation of semantically constrained graphs that fulfills all of these criteria. At the heart of SELFIES is a formal Chomsky type-2 grammar [13], which is augmented with two self-referencing, recursive functions to ensure the generation of syntactically and semantically valid graphs. Every symbol of a SELFIES sequence corresponds to a vector of rules in the grammar, and the states of the grammar are used to encode and memorize semantic constraints. A SELFIES sequence can be directly obtained from and transformed into a graph.

We show that the SELFIES representation can be used as a direct input to deep learning models on the example of a variational autoencoder. Due to its robustness, SELFIES enable the application of a range of ML algorithms to semantically constrained graphs that have been challenging before. Concrete examples are evolutionary strategies, which can be applied as a robust alternative to Q-learning [14], and as direct generative models for the *de novo* design of molecules [15, 16, 17, 18].

2 Related Work

A large amount of effort has been put into the construction of representations of semantically constrained graphs, due to their significance for natural science. The application of VAEs in chemistry has seen a rapid evolution of robustness. In the first application of VAEs in chemistry, SMILES (Simplified Molecular Input Line Entry Specification) have been used to represent chemical compounds [6]. SMILES were designed in the late 1980s to represent molecular structure information in cheminformatics applications [19]. Even though they are fragile, *i.e.* small variations often lead to invalid molecules, SMILES are still one of the standard representations used today and are one of the baselines in our experiments.

The original SMILES representation has since been extended, most notable by *Daylight Chemical Information Systems* in the form of SMARTS [20] and SMIRKS [21], and by *Blue Obelisk* who defined the open-source standard *OpenSMILES* [22]. Other sequence-based representation are *Wiswesser line notation (WLN)* [23], SLN [24], and InChI [25]. None of these representations have been constructed for robust representation of information in the context of ML.

DeepSMILES [26] is an effort to extend SMILES by introducing a more robust encoding of rings and branches of molecules. We use DeepSMILES as another baseline for our numerical experiments.

To improve the robustness of molecular representations, parse trees have been employed to formally derive SMILES strings [27]. This approach has been introduced as GrammarVAE, and while the work presented here shares the use of a formal grammar, both approaches are diametrically opposite. GrammarVAE uses the well-defined grammar of SMILES which has been defined to *construct* all possible graph structures of chemical elements – a class which contains much more than just all valid molecules. We use the deterministic rewriting rule representation of GrammarVAE as one of our baselines.

Further improvements over the GrammarVAE came from introducing syntax-directed VAEs which use *attribute grammars* to introduce semantic meaning [28]. Junction-Tree VAEs introduce supernodes to transform molecular graphs into trees [29]. These methods are domain-specific, and extensions beyond chemistry are not obvious. It was shown how the use of general semantically constrained graphs combined with regularization can lead to high validity of the decoded molecules from a VAE [11]. A range of other techniques for generating semantically valid graphs have been presented in the last two years, for example, [30, 31, 12, 32], to name a few. The objective of these works is the demonstration of generative models for semantically valid graphs, especially in the settings of VAEs – while our motivation is the definition of a model-independent, simple, and robust representation of semantically constrained graphs.

A different field of study in mathematics (denoted as *graph grammars*) concerns itself with formal grammars to describe transformations of graphs in a systematic way (see [33] for a comprehensive overview). However, graph grammars have not been studies with the objective of robustness, and are therefore often very brittle[12, 34].

3 Robust representation of semantically constrained graphs

We take advantage of a **formal grammar** to derive *words*, which will represent semantically valid graphs. A formal grammar is a tuple $G(V, \Sigma, R, S)$, where $v \in V$ are non-terminal symbols that are replaced using rules, $r \in R$, into non-terminal or terminal symbols $t \in \Sigma$. S is a start symbol. When the resulting string only consists of terminal symbols, the derivation of a new word is completed [35].

The SELFIES representation is a Chomsky type-2, context-free grammar with self-referencing functions for valid generation of branches in graphs. The rule system is shown in Table 1.

	Vertices			Branches			Rings	
	$\mathbf{A}_0$	$\mathbf{A}_1$	$\mathbf{A}_n$	$\mathbf{A}_{n+1}$		$\mathbf{A}_{n+m}$	$\mathbf{A}_{n+m+1}$	$\mathbf{A}_{n+m+p}$
$\mathbf{X}_0 \rightarrow \epsilon$	$\mid t_{0,1}\,\mathbf{X}_{h_{0,1}}$	… $\mid t_{0,n}\,\mathbf{X}_{h_{r,0}}$	$\mid$ B(**N**, $\mathrm{X}_{i_{0,1}}$) $\mathbf{X}_{j_{0,1}}$	…	$\mid$B(**N**, $\mathrm{X}_{i_{0,m}}$) $\mathbf{X}_{j_{0,m}}$	$\mid$ R(**N**) $\mathbf{X}_{k_{0,1}}$	… $\mid$R(**N**) $\mathbf{X}_{k_{0,p}}$
$\mathbf{X}_1 \rightarrow \epsilon$	$\mid t_{1,1}\,\mathbf{X}_{h_{1,1}}$	… $\mid t_{1,n}\,\mathbf{X}_{h_{r,1}}$	$\mid$ B(**N**, $\mathrm{X}_{i_{1,1}}$) $\mathbf{X}_{j_{1,1}}$	…	$\mid$B(**N**, $\mathrm{X}_{i_{1,m}}$) $\mathbf{X}_{j_{1,m}}$	$\mid$ R(**N**) $\mathbf{X}_{k_{1,1}}$	… $\mid$R(**N**) $\mathbf{X}_{k_{1,p}}$
				…				
$\mathbf{X}_r \rightarrow \epsilon$	$\mid t_{r,1}\,\mathbf{X}_{h_{r,1}}$	… $\mid t_{r,n}\,\mathbf{X}_{h_{r,n}}$	$\mid$ B(**N**, $\mathrm{X}_{i_{r,1}}$) $\mathbf{X}_{j_{r,1}}$	…	$\mid$B(**N**, $\mathrm{X}_{i_{r,m}}$) $\mathbf{X}_{j_{r,m}}$	$\mid$ R(**N**) $\mathbf{X}_{k_{r,1}}$	… $\mid$R(**N**) $\mathbf{X}_{k_{r,p}}$
$\mathbf{N} \rightarrow 0$	$\mid$ 1	… $\mid$ n	$\mid$ n+1	…	$\mid$ n+m	$\mid$ n+m+1	… $\mid$n+m+p

Table 1: System of the character production rules of SELFIES in form of a grammar $G(V, \Sigma, R, S)$ with recursion, and $\mathbf{S} \rightarrow \mathbf{X}_0$.

In SELFIES, $V = \{\mathbf{X}_0, \ldots, \mathbf{X}_r, \mathbf{N}\}$ are called non-terminal symbols or states. $\Sigma = \{t_{0,1}, \ldots, t_{r,n}\}$ are terminal symbols. The derivation rules R has exactly $(n + m + p + 1) \times (r + 2)$ elements, corresponding to n rules for vertex production, m rules for producing branches, p rules for rings and r non-terminal symbols (or states) in V. The subscripts $h_{a,b}$, $i_{a,b}$, $j_{a,b}$ and $k_{a,b}$ have values from 1 to r. The semantic and syntactical constrained are encoded into the rule vectors, which guarantees strong robustness. There are $n + m + p + 1$ rule vectors $\mathbf{A}_i$, each with a dimension $(r + 2)$.

3.1 Self-referencing functions for syntactic validity

In order to account for **syntactic validity** of the graph, we augment the context-free grammar with **branching functions** and **ring functions**. $B(\mathbf{N}, X_i)$ is the branching function, that recursively starts another grammar derivation with subsequent $\mathbf{N}$ SELFIES symbols in state X_i. Here, $\mathbf{N}$ is a non-terminal symbol, which leads to a numerical value after its derivation. After the full derivation of a new word (which is, again, a graph), the branch function returns the graph, and connects it to the current vertex.

The ring function $R(\mathbf{N})$ is used to establish edges between the current vertex and the $(\mathbf{N} + 1)$-th last derived vertex, including vertices obtained by the recursive branch function, for which access to the already derived string is necessary. In general, both the branching functions and ring functions are self-referencing, in the sense that they have access to the SELFIES string and the derived string. A further specialty is that the non-terminal symbol $\mathbf{N}$ is derived via the grammar and acts as an argument of the two self-referencing functions.

3.2 Rule vectors for semantic validity

To incorporate **semantic validity**, we denote A_i as the i-th vector of rules, with dimension $d_{A_i} = |V| = r+2$. The **conceptual idea** is to interpret a symbol of a SELFIES string, $s_i \in \{0, \ldots, n+m+p\}$ as an index of a rule vector, A_{s_i}. In the derivation of a symbol, the rule vector is defined by the symbol of the SELFIES string (external state) while the specific rule is chosen by the non-terminal symbol (internal state). Thereby, we can encode semantic information into the rule vector A_i, which is *memorized* by the internal state during derivation. In the SI, we show a domain-independent toy example, which shows the action of the rule vector concerning semantic validity.

3.3 Example in Chemistry

We now show a concrete alphabet for the application in chemistry in Table 2. In particular, we use it to represent molecules in the benchmark dataset QM9 (or *GDB-9*, which contains all possible organic molecules of up to 9 heavy atoms.) [36, 37]. We limit ourselves to non-ionic molecules for simplicity, which reduced the dataset by 1,5%. However, the grammar can readily be extended to include ions by introducing additional rule vectors (i.e. by adding additional rules for each non-terminal symbol). The system has 14×9 rules encoding the semantic and syntactic restrictions given by chemistry.

	A	**B**	**C**	**D**	**E**	**F**	**G**	**H**	**I**	**J**	**K**	**L**	**M**	**N**
$\mathbf{X}_0$	$\rightarrow \mathbf{X}_0$	F $\mathbf{X}_1$	O $\mathbf{X}_2$	N $\mathbf{X}_3$	O $\mathbf{X}_2$	N $\mathbf{X}_3$	N $\mathbf{X}_3$	C $\mathbf{X}_4$	C $\mathbf{X}_4$	C $\mathbf{X}_4$	ign $\mathbf{X}_0$	ign $\mathbf{X}_0$	ign $\mathbf{X}_0$	ign $\mathbf{X}_0$
$\mathbf{X}_1$	$\rightarrow \epsilon$	F	O	N	O $\mathbf{X}_1$	N $\mathbf{X}_2$	N $\mathbf{X}_2$	C $\mathbf{X}_3$	C $\mathbf{X}_3$	C $\mathbf{X}_3$	ign $\mathbf{X}_1$	ign $\mathbf{X}_1$	ign $\mathbf{X}_1$	R(**N**)
$\mathbf{X}_2$	$\rightarrow \epsilon$	F	=O	=N	O $\mathbf{X}_1$	N $\mathbf{X}_2$	=N $\mathbf{X}_1$	C $\mathbf{X}_3$	=C $\mathbf{X}_2$	=C $\mathbf{X}_2$	B(**N**,$\mathbf{X}_5$)$\mathbf{X}_1$	B(**N**,$\mathbf{X}_5$)$\mathbf{X}_1$	B(**N**,$\mathbf{X}_5$)$\mathbf{X}_1$	R(**N**) $\mathbf{X}_1$
$\mathbf{X}_3$	$\rightarrow \epsilon$	F	=O	#N	O $\mathbf{X}_1$	N $\mathbf{X}_2$	=N $\mathbf{X}_1$	C $\mathbf{X}_3$	=C $\mathbf{X}_2$	#C $\mathbf{X}_1$	B(**N**,$\mathbf{X}_5$)$\mathbf{X}_2$	B(**N**,$\mathbf{X}_6$)$\mathbf{X}_1$	B(**N**,$\mathbf{X}_5$)$\mathbf{X}_2$	R(**N**) $\mathbf{X}_2$
$\mathbf{X}_4$	$\rightarrow \epsilon$	F	=O	#N	O $\mathbf{X}_1$	N $\mathbf{X}_2$	=N $\mathbf{X}_1$	C $\mathbf{X}_3$	=C $\mathbf{X}_2$	#C $\mathbf{X}_1$	B(**N**,$\mathbf{X}_5$)$\mathbf{X}_3$	B(**N**,$\mathbf{X}_7$)$\mathbf{X}_1$	B(**N**,$\mathbf{X}_6$)$\mathbf{X}_2$	R(**N**) $\mathbf{X}_3$
$\mathbf{X}_5$	$\rightarrow$ C	F	O	N	O $\mathbf{X}_1$	N $\mathbf{X}_2$	N $\mathbf{X}_2$	C $\mathbf{X}_3$	C $\mathbf{X}_3$	C $\mathbf{X}_3$	$\mathbf{X}_5$	$\mathbf{X}_5$	$\mathbf{X}_5$	$\mathbf{X}_5$
$\mathbf{X}_6$	$\rightarrow$ C	F	=O	=N	O $\mathbf{X}_1$	N $\mathbf{X}_2$	=N $\mathbf{X}_1$	C $\mathbf{X}_3$	=C $\mathbf{X}_2$	=C $\mathbf{X}_2$	$\mathbf{X}_6$	$\mathbf{X}_6$	$\mathbf{X}_6$	$\mathbf{X}_6$
$\mathbf{X}_7$	$\rightarrow$ C	F	=O	#N	O $\mathbf{X}_1$	N $\mathbf{X}_2$	=N $\mathbf{X}_1$	C $\mathbf{X}_3$	=C $\mathbf{X}_2$	#C $\mathbf{X}_1$	$\mathbf{X}_7$	$\mathbf{X}_7$	$\mathbf{X}_7$	$\mathbf{X}_7$
N	$\rightarrow$ 1	2	3	4	5	6	7	8	9	10	11	12	13	14

Table 2: Derivation rules of SELFIES for molecules in the QM9 dataset. The semantic constraints are encoded into the rule vectors while the syntactic constraints are encoded by the branching and ring functions. For states $\mathbf{X}_0$ and $\mathbf{X}_1$, branchings and rings are invalid, thus the corresponding functions ignore the subsequent SELFIES symbol which would stand for **N** (denoted as `ign`).

As a concrete example, we show the derivation of the molecular graph of 1,1-diethyl-cyclopropane (DECP) in Figure 1. The SELFIES for DECP is `HHHKBHHHHHHNA`, where each symbol corresponds to a rule vector in the grammar in Table 2.

The full grammar of SELFIES for chemistry involves, in addition to Table 2, also aromatic symbols, stereochemistry, and ions. This generalisation extends the grammar by additional rule vectors.

4 Experiments

In this section, we examine the robustness of SELFIES and their application in a VAEs. We chose chemistry as the domain in which we apply and test SELFIES, as it has become a widely used

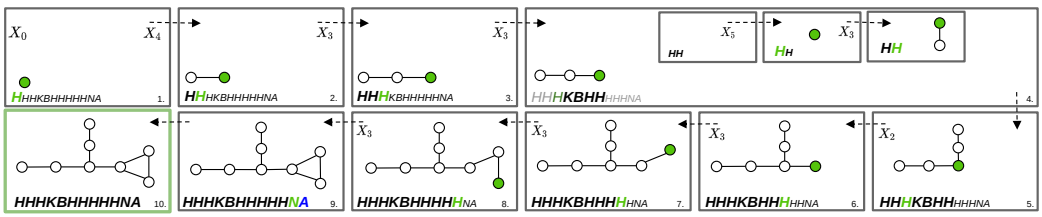

Figure 1: Derivation of a 1,1-diethyl-cyclopropane (DECP) molecule. The nodes marked in green are currently active, meaning that the next edges will be connected to them. In the fourth step, a branch is created in a recursive way: A SELFIES word with two symbols is derived, and added in the seventh step. The last step derives an edge that creates a ring. All vertices are carbon atoms, thus the final graph is an abstract representation of the molecule DECP.

	QM9 dataset		organic semiconductor	
	Alphabet	**largest Molecule**	**Alphabet**	**largest Molecule**
SMILES	**14**	22	**19**	153
DeepSMILES	15	**18**	20	174
GrammarVAE	28	70	40	402
SELFIES	**14**	21	22	**127**

Table 3: Size of encodings of 134.000 molecules from QM9, and 50.000 molecules for solar cells, with different representations. The bold numbers show the smallest encoding sizes.

application for ML tasks. As a baseline, we use deterministic representations of molecular structures – namely SMILES, DeepSMILES, and GrammarVAE.

We use two benchmark datasets containing in total of 184.000 molecules. One of the datasets, denoted as QM9, consists of 134.000 molecules with up to 9 heavy atoms (C, O, N and F) [36, 37], while the other contains 50.000 small molecule organic semiconductors [38]. We encode them in the four different representations – the sizes of their alphabets and encodings is shown in Table 9

For the QM9 dataset, SMILES, DeepSMILES, and SELFIES have a similar alphabet size, and size of the largest molecule (between 18 and 22 symbols are required to encode the largest molecule in the dataset). For organic semiconductor molecules, SELFIES has the shortest encoding. The encoding using GrammarVAE is less efficient in terms of size.

4.1 Validity after mutations

A straight-forward, model-independent way to evaluate the robustness of a representation is to introduce random mutations in the encoding sequence. We start from an encoding of valid molecules in the QM9 dataset, and perform one mutation, and evaluate the validity (for which we use RDKit [39]). Afterwards, we investigate the validity of random strings.

In Table 4, we show that the validity of the competing representations drops significantly when mutations are introduced, while SELFIES are valid in more than 80% of the cases, even when random strings are produced. All representations except for GrammarVAE use stop-symbols in the alphabet which terminate the derivation of the string (for padding and to satisfy semantic constraints). We remove all stop symbols, thus all strings are derived until the last element. In order to verify that the robustness does not originate merely from shorter derivations, we restrict ourself to replacements with non-stop symbols. Again, we observe again that SELFIES have significantly higher robustness than all other representations.

	full alphabet				only non-terminals			
	single mutation		**random string**		**single mutation**		**random string**	
	Validity	Length	Validity	Length	Validity	Length	Validity	Length
SMILES	18.1%	**13.9**	2.8%	1.4	17.6%	**14.9**	0.0%	-
DeepSMILES	38.2%	13.4	3.0%	1.4	34.3%	14.7	0.0%	-
GrammarVAE	9.5%	12.9	17.2%	1.0	9.5%	12.9	17.2%	1.0
SELFIES	**88.1%**	12.4	**84.0%**	**4.7**	**85.3%**	13.6	**67.8%**	**9.5**

Table 4: Validity and average Length of valid molecules of four molecule representations after random mutations, starting from valid molecules from QM9. The length of valid molecules is measured in symbols of the corresponding SMILES string. The average size of molecule SMILES without mutations is 15.6 symbols. Bold numbers show show the largest validity and the longest average molecules.

4.2 Similarity of the chemical neighborhood

Robustness is an essential characteristic of a representation. However, it alone does not allow to draw conclusions on the applicability, because we can find pathological functions that are robust, but in which very similar structures have very different phenotypic properties. An example for a

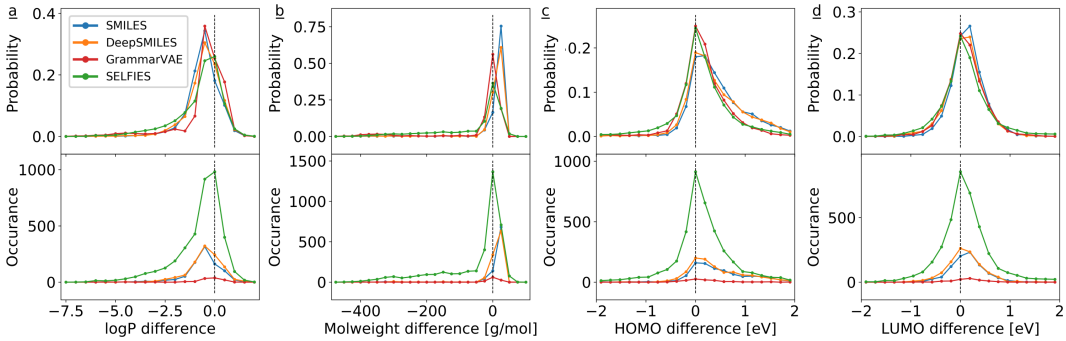

Figure 2: Differences of chemical properties between nearest neighbors of the representation. a) Water octanol partition coefficient (*solubility*, logP), b) molecular weight, c) Highest Occupied Molecular Orbital (HOMO) energy and d) lowest unoccupied molecular orbital (LUMO) energy. The upper row shows the frequency of the differences (conditioned on valid output) while the lower row shows the occurrences. It demonstrates that the SELFIES are not only significantly more robust but the difference that nearest neighbors have a similarity that is comparable with the baselines.

pathological function that translates a sequence into a graph is the following: If the input sequence is translated to a valid semantic-constrained graph, the function will return the sequence. If the sequence is invalid, a randomly generated, valid graph is returned. Although this function yields 100 % valid graphs, most small changes in the sequence have severe effects on the properties of the graph. In the following, we show that our representation is not of that kind, instead of that its neighborhood shares similar properties with the original graph.

To show that small variations of SELFIES leads in most cases to molecules with similar domain-dependent properties, we conduct an experiment on organic semiconductor molecules, which are used for organic solar cells [38]. We employ 5.000 randomly chosen molecules with less than 60 symbols, counted in their canonical SMILES representation. We translate each of these molecules into four representations. Then we investigate the neighborhood in the high-dimensional representation space: We replace one randomly chosen symbol with another symbol of the alphabet of the respective representation, determine whether the new molecule is valid, and if it is, we calculate a number of chemical properties. These involve the water-octanol partition coefficient (logP), the molecular weight and the energies of the highest occupied molecular orbital (HOMO) as well as the lowest unoccupied molecular orbital (LUMO). HOMO and LUMO are actual properties of interest for organic solar cells, and we calculate them with GFN2-xTB [40].

The results are shown in Figure 2. The upper row shows the probability density of the difference of the chemical properties of the modified and the unmodified molecule, if the molecule was valid. The lower row shows the actual occurrence of these cases, within 5.000 iterations. For this application, SELFIES are significantly more robust upon random modifications and, importantly, find that small variations in the sequence produce molecules with similar chemical properties (see SI for more detail).

4.3 Application in a Variational Autoencoder

We further demonstrate the robustness and practicality of SELFIES for molecule generation in VAE trained on the QM9 dataset. Specifically, we evaluate the performance of each representation based on the reconstruction quality, validity, and diversity of the generated molecules. The reconstruction quality is determined as a per-character matching between the encoder input and the decoder output. For the validity and diversity, we decode 2.500 random samples from the latent space. The fraction of valid molecules (as before, according to RDKit) we call *validity*, and the fraction of valid molecules with different SMILES strings we call *diversity*.

To account for conceptual differences in representing information, we perform a hyperparameter search that optimizes the model architecture of the VAE. The hyperparameter search is based on Bayesian optimization using *GPyOpt* [41]. For each representation, we aim to identify the architectures which simultaneously improve on validity, reconstruction quality, and diversity. To this end, we employ the *Chimera* scalarizing function (see SI) [42].

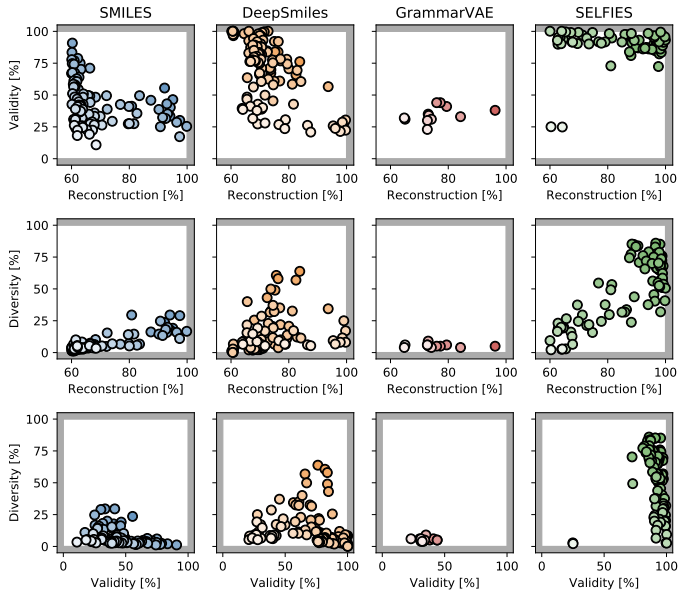

Figure 3: Model quality during hyperparameter optimization of VAE architectures, and correlations between pair-wise combinations of the objectives: validity, reconstruction quality and diversity. It demonstrates that SELFIES has a consistently high validity and has large diversity for large reconstruction qualities. All representations are trained for the same amount of time. GrammarVAE takes significantly longer to train, and has reached an early stopping criteria more often than other representations, thus there are less results plotted.

Each autoencoder is trained on a randomly chosen 50% subset of the QM9 dataset, while the remaining 50% is used as a test set. Overall, we observe that SELFIES maintains high reconstruction qualities and validity scores during the hyperparameter optimization while keeping the computational cost of training a single VAE at a level comparable to the standard SMILES. Results of all models investigated during the optimization show a clear general trend, see Figure 3. Best scores achieved for each representation are reported in Table 5. We find that SELFIES not only enable a VAE to generate valid molecules with high confidence but also that the generated molecules are significantly more diverse than those generated from VAEs trained on other representations.

4.3.1 Prediction of chemical properties from latent space

Up to now, we have demonstrated that SELFIES have high validity, reconstruction quality, and diversity when used as inputs for a VAE. For their applications in the *inverse design* of molecules, they need to enable the prediction of molecular properties from latent space points. Our target property is the solubility (denoted as logP). For this, we split the QM9 dataset into training and validation set. Besides, to investigate a generalization of the property beyond the domain of the training set, we include all molecules with logP$<$-1 in the validation set. For a fair comparison, we perform hyperparameter optimization of the model architecture again, with scalarization of the objectives using *Chimera* (see SI).

The results in Figure 4 demonstrate that SELFIES accurately predicts properties of unseen molecules, thus we can approach inverse design questions in chemistry and other domains.

	Validity	**Reconstruction**	**Diversity**
SMILES	71.9%	66.2%	5.9%
DeepSMILES	81.4%	79.8%	67.3%
GrammarVAE	34.0%	84.0%	4.0%
SELFIES	**95.2%**	**98.2%**	**85.7%**

Table 5: Scores of best performing variational autoencoder architectures (VAE) trained on each of the four representations. SELFIES achieve the best scores across all three performance metrics.

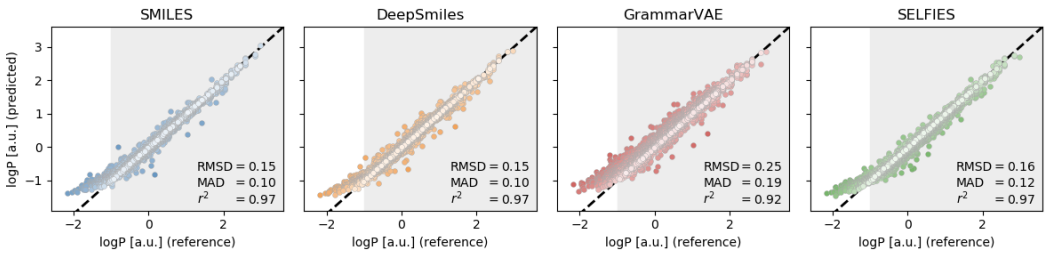

Figure 4: Prediction of chemically relevant properties (in this case, the solubility of the molecule, denoted as logP) of points in the latent space of a VAE. We show that SELFIES can robustly predict chemical properties of molecules it has never encountered before. Here the training set is restricted to molecules with logP>-1, and molecules in the white region are outside of this region. We find that all representations have similar prediction quality, while SELFIES, in addition, solve the problem of robustness. We plot 2.000 out of 65.000 randomly chosen molecules, each data point corresponds to one molecule in the validation set. RMSD stands for root-mean-square deviation, MAD is mean absolute deviation, and r^2 is the coefficient of determination.

5 Conclusion and Future Work

SELFIES is a robust, general-purpose representation of graphs with domain-specific constraints. It enables the application of new deep learning methods in the natural sciences, such as chemistry, without the necessity to adapt models with domain-specific constraints. The robustness allows the application of models that are critical to validity, such as evolutionary strategies, and in particular reinforcement learning based on evolutionary strategies [14].

When deep learning models are applied in the natural sciences, scientists are not only interested in excellent predictive or generative results. In the best case, they understand what the model has learned – interpret its result [43, 44, 45], in particular interpreting its internal representations [46]. This task seems highly challenging when most of the latent space leads to invalid, chemical or physical, results. SELFIES might be a way to enable the interpretation of internal representations, by combining it with recent methods to shape latent spaces [47].

In the near future, the authors of the manuscript will convene a 1-day workshop for interested parties to discuss the formal standardization of molecular SELFIES to make sure the grammar to discuss chemical concepts such as hypervalency and chirality that are trivial additions but need to be discussed carefully to cover all the potential chemical space. Further discussions will hopefully be carried out online in a multidisciplinary working group.

5.1 Potential applications to other domains in the physical sciences

The principles of SELFIES are not limited to the chemical domain, but the method can also be applied in various scientific design questions with syntactic and semantic constraints. For example, many questions in machine learning for physics [48] have similar character and could benefit from robust representations enforced by SELFIES. Two specific examples are the automated design of quantum optical experiments [49, 50, 51], and representation of DNA (deoxyribonucleic acid) in biology (for example, to design DNA origami [52, 53]).

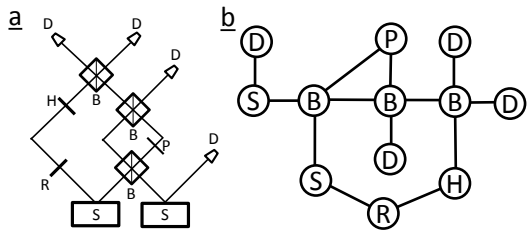

Figure 5: Quantum optical experiments as Graphs. A recent high-dimensional multi-partite quantum experiment [54] is depicted in (a). Capital letters stand for optical components (for details, see SI and [55]). In (b), the abstract graph which encodes the connectivity of the optical elements. In the SI, this graph is derived using a quantum-optics specific SELFIES representation.

The grammar of SELFIES for designing quantum optical experiments is shown in the SI. The rule vectors encode semantic information of connectivities of quantum optical components. In the SI we demonstrate a concrete encoding of a recent, complex quantum optical setup [54]. The corresponding quantum optical circuit and the semantically constrained graph is shown in Figure 5.

Another concrete grammar, for the application of SELFIES on DNA, is presented in the SI. With these examples, we show that SELFIES can be applied in multiple fields of science.

Acknowledgments

The authors thank Theophile Gaudin for useful discussions. A. A.-G. acknowledges generous support from the Canada 150 Research Chair Program, Tata Steel, Anders G. Froseth, and the Office of Naval Research. We acknowledge supercomputing support from SciNet. M.K. acknowledges support from the Austrian Science Fund (FWF) through the Erwin Schrödinger fellowship No. J4309. F.H. acknowledges support from the Herchel Smith Graduate Fellowship and the Jacques-Emile Dubois Student Dissertation Fellowship. P.F. has received funding from the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement No 795206.

References

[1] A. Brock, J. Donahue and K. Simonyan, Large scale gan training for high fidelity natural image synthesis. *arXiv:1809.11096* (2018).

[2] A. Van Den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A.W. Senior and K. Kavukcuoglu, WaveNet: A generative model for raw audio.. *SSW* **125**, (2016).

[3] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei and I. Sutskever, Language models are unsupervised multitask learners. *OpenAI Blog* **1**, 8 (2019).

[4] B. Sánchez-Lengeling and A. Aspuru-Guzik, Inverse molecular design using machine learning: Generative models for matter engineering. *Science* **361**, 360–365 (2018).

[5] D.C. Elton, Z. Boukouvalas, M.D. Fuge and P.W. Chung, Deep learning for molecular generation and optimization-a review of the state of the art. *arXiv:1903.04388* (2019).

[6] R. Gómez-Bombarelli, J.N. Wei, D. Duvenaud, J.M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T.D. Hirzel, R.P. Adams and A. Aspuru-Guzik, Automatic chemical design using a data-driven continuous representation of molecules. *ACS central science* **4**, 268–276 (2018).

[7] G.L. Guimaraes, B. Sánchez-Lengeling, C. Outeiral, P.L.C. Farias and A. Aspuru-Guzik, Objective-reinforced generative adversarial networks (ORGAN) for sequence generation models. *arXiv:1705.10843* (2017).

[8] B. Sánchez-Lengeling, C. Outeiral, G.L. Guimaraes and A. Aspuru-Guzik, Optimizing distributions over molecular space. An objective-reinforced generative adversarial network for inverse-design chemistry (ORGANIC). *Harvard University, Chem Rxiv.* (2017).

[9] E. Putin, A. Asadulaev, Y. Ivanenkov, V. Aladinskiy, B. Sánchez-Lengeling, A. Aspuru-Guzik and A. Zhavoronkov, Reinforced adversarial neural computer for de novo molecular design. *Journal of chemical information and modeling* **58**, 1194–1204 (2018).

[10] M. Popova, O. Isayev and A. Tropsha, Deep reinforcement learning for de novo drug design. *Science advances* **4**, eaap7885 (2018).

[11] T. Ma, J. Chen and C. Xiao, Constrained generation of semantically valid graphs via regularizing variational autoencoders. *Advances in Neural Information Processing Systems* 7113–7124 (2018).

[12] Y. Li, O. Vinyals, C. Dyer, R. Pascanu and P. Battaglia, Learning deep generative models of graphs. *arXiv:1803.03324* (2018).

[13] N. Chomsky, Three models for the description of language. *IRE Transactions on information theory* **2**, 113–124 (1956).

[14] T. Salimans, J. Ho, X. Chen, S. Sidor and I. Sutskever, Evolution strategies as a scalable alternative to reinforcement learning. *arXiv:1703.03864* (2017).

[15] I.Y. Kanal and G.R. Hutchison, Rapid computational optimization of molecular properties using genetic algorithms: Searching across millions of compounds for organic photovoltaic materials. *arXiv arXiv:1707.02949* (2017).

[16] N. Yoshikawa, K. Terayama, M. Sumita, T. Homma, K. Oono and K. Tsuda, Population-based de novo molecule generation, using grammatical evolution. *Chemistry Letters* **47**, 1431–1434 (2018).

[17] J.H. Jensen, A graph-based genetic algorithm and generative model/Monte Carlo tree search for the exploration of chemical space. *Chemical Science* **10**, 3567–3572 (2019).

[18] C. Rupakheti, A. Virshup, W. Yang and D.N. Beratan, Strategy to discover diverse optimal molecules in the small molecule universe. *Journal of chemical information and modeling* **55**, 529–537 (2015).

[19] D. Weininger, SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *Journal of chemical information and computer sciences* **28**, 31–36 (1988).

[20] D.C.I. Systems, SMARTS - A Language for Describing Molecular Patterns. *https://www.daylight.com/dayhtml/doc/theory/theory.smarts.html* (2007).

[21] D.C.I. Systems, SMIRKS - A Reaction Transform Language. *https://www.daylight.com/dayhtml/doc/theory/theory.smirks.html* (2007).

[22] C.A. James, OpenSMILES Specification. *http://opensmiles.org/spec/open-smiles.html* (2007).

[23] W.J. Wiswesser, How the WLN began in 1949 and how it might be in 1999. *Journal of Chemical Information and Computer Sciences* **22**, 88-93 (1982).

[24] R.W. Homer, J. Swanson, R.J. Jilek, T. Hurst and R.D. Clark, SYBYL Line Notation (SLN): A Single Notation To Represent Chemical Structures, Queries, Reactions, and Virtual Libraries. *Journal of Chemical Information and Modeling* **48**, 88-93 (2008).

[25] S. Heller, A. McNaught, S. Stein, D. Tchekhovskoi and I. Pletnev, InChI - the worldwide chemical structure identifier standard. *Journal of Cheminformatics* **5**, 7 (2013).

[26] N. O'Boyle and A. Dalke, DeepSMILES: An Adaptation of SMILES for Use in machine-learing chemical structures. *ChemRxiv* (2018).

[27] M.J. Kusner, B. Paige and J.M. Hernández-Lobato, Grammar variational autoencoder. *Proceedings of the 34th International Conference on Machine Learning-Volume 70* 1945–1954 (2017).

[28] H. Dai, Y. Tian, B. Dai, S. Skiena and L. Song, Syntax-directed variational autoencoder for structured data. *arXiv:1802.08786* (2018).

[29] W. Jin, R. Barzilay and T. Jaakkola, Junction tree variational autoencoder for molecular graph generation. *arXiv:1802.04364* (2018).

[30] M. Simonovsky and N. Komodakis, Graphvae: Towards generation of small graphs using variational autoencoders. *International Conference on Artificial Neural Networks* 412–422 (2018).

[31] Q. Liu, M. Allamanis, M. Brockschmidt and A. Gaunt, Constrained graph variational autoencoders for molecule design. *Advances in Neural Information Processing Systems* 7795–7804 (2018).

[32] B. Samanta, A. De, G. Jana, P.K. Chattaraj, N. Ganguly and M. Gomez-Rodriguez, NeVAE: A Deep Generative Model for Molecular Graphs. *arXiv:1802.05283* (2018).

[33] R. Grzegorz, Handbook Of Graph Grammars And Computing By Graph Transformation, Vol 1: Foundations. (1997).

[34] S. Aguiñaga, R. Palacios, D. Chiang and T. Weninger, Growing Graphs from Hyperedge Replacement Graph Grammars. *Proceedings of the 25th ACM International on Conference on Information and Knowledge Management* 469–478 (2016).

[35] J.E. Hopcroft, R. Motwani and J.D. Ullman, Introduction to Automata Theory, Languages, and Computation (3rd Edition). (2006).

[36] R. Ramakrishnan, P.O. Dral, M. Rupp and O.A. Von Lilienfeld, Quantum chemistry structures and properties of 134 kilo molecules. *Scientific data* **1**, 140022 (2014).

[37] L. Ruddigkeit, R. Van Deursen, L.C. Blum and J.L. Reymond, Enumeration of 166 billion organic small molecules in the chemical universe database GDB-17. *Journal of chemical information and modeling* **52**, 2864–2875 (2012).

[38] S.A. Lopez, B. Sánchez-Lengeling, J. Goes Soares and A. Aspuru-Guzik, Design principles and top non-fullerene acceptor candidates for organic photovoltaics. *Joule* **1**, 857–870 (2017).

[39] G. Landrum and others, RDKit: Open-source cheminformatics. *Journal of chemical information and modeling* (2006).

[40] C. Bannwarth, S. Ehlert and S. Grimme, GFN2-xTB - An Accurate and Broadly Parametrized Self-Consistent Tight-Binding Quantum Chemical Method with Multipole Electrostatics and Density-Dependent Dispersion Contributions. *Journal of chemical theory and computation* (2018).

[41] T.G. authors, GPyOpt: A Bayesian Optimization framework in python. *http://github.com/SheffieldML/GPyOpt* (2016).

[42] F. Häse, L.M. Roch and A. Aspuru-Guzik, Chimera: enabling hierarchy based multi-objective optimization for self-driving laboratories. *Chemical science* **9**, 7642–7655 (2018).

[43] K.T. Schütt, F. Arbabzadah, S. Chmiela, K.R. Müller and A. Tkatchenko, Quantum-chemical insights from deep tensor neural networks. *Nature communications* **8**, 13890 (2017).

[44] K. Preuer, G. Klambauer, F. Rippmann, S. Hochreiter and T. Unterthiner, Interpretable Deep Learning in Drug Discovery. *arXiv:1903.02788* (2019).

[45] F. Häse, I.F. Galván, A. Aspuru-Guzik, R. Lindh and M. Vacher, How machine learning can assist the interpretation of ab initio molecular dynamics simulations and conceptual understanding of chemistry. *Chemical Science* **10**, 2298–2307 (2019).

[46] R. Iten, T. Metger, H. Wilming, L. Del Rio and R. Renner, Discovering physical concepts with neural networks. *arXiv:1807.10300* (2018).

[47] T.Q. Chen, X. Li, R.B. Grosse and D.K. Duvenaud, Isolating sources of disentanglement in variational autoencoders. *Advances in Neural Information Processing Systems* 2610–2620 (2018).

[48] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto and L. Zdeborová, Machine learning and the physical sciences. *arXiv:1903.10563* (2019).

[49] M. Krenn, M. Malik, R. Fickler, R. Lapkiewicz and A. Zeilinger, Automated search for new quantum experiments. *Physical review letters* **116**, 090405 (2016).

[50] L. O'Driscoll, R. Nichols and P. Knott, A hybrid machine learning algorithm for designing quantum experiments. *Quantum Machine Intelligence* 1–11 (2018).

[51] A.A. Melnikov, H.P. Nautrup, M. Krenn, V. Dunjko, M. Tiersch, A. Zeilinger and H.J. Briegel, Active learning machine learns to create new quantum experiments. *Proceedings of the National Academy of Sciences* **115**, 1221–1226 (2018).

[52] T. Gerling, K.F. Wagenbauer, A.M. Neuner and H. Dietz, Dynamic DNA devices and assemblies formed by shape-complementary, non-base pairing 3D components. *Science* **347**, 1446-1452 (2015).

[53] F. Praetorius, B. Kick, K.L. Behler, M.N. Honemann, D. Weuster-Botz and H. Dietz, Biotechnological mass production of DNA origami. *Nature* **552**, 84-87 (2017).

[54] M. Erhard, M. Malik, M. Krenn and A. Zeilinger, Experimental Greenberger–Horne–Zeilinger entanglement beyond qubits. *Nature Photonics* **12**, 759 (2018).

[55] J.W. Pan, Z.B. Chen, C.Y. Lu, H. Weinfurter, A. Zeilinger and M. Żukowski, Multiphoton entanglement and interferometry. *Reviews of Modern Physics* **84**, 777 (2012).

[56] K. Cho, B. Merrienboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk and Y. Bengio, Learning Phrase Representations using RNN Encoder-Decoder for Statistical Machine Translation. *arXiv:1406.1078* (2014).

[57] R. Pascanu, T. Mikolov and Y. Bengio, On the difficulty of training Recurrent Neural Networks. *arXiv:1211.5063* (2012).

Supplementary Information

Toy example for rule vector and semantic validity

As an example for the action of the rule vector, we consider the following: We want to construct graphs with three different types of vertices (O, N and C), with the following constraints: the degree of an O-vertex is maximally 2 (denoted as $deg(\mathtt{O}) \leq 2$), $deg(\mathtt{N}) \leq 3$ and $deg(\mathtt{C}) \leq 4$. A restricted SELFIES grammar for these constraints are seen in Table 6.

		$\mathbf{A}_0$	$\mathbf{A}_1$	$\mathbf{A}_2$	$\mathbf{A}_3$	$\mathbf{A}_4$	$\mathbf{A}_5$	$\mathbf{A}_6$
R	$\rightarrow$	ϵ	| O **R**	| N **G**	| C **B**	| N **G**	| C **B**	| C **B**
G	$\rightarrow$	ϵ	| O **R**	| N **G**	| C **B**	| =N **R**	| =C **G**	| =C **G**
B	$\rightarrow$	ϵ	| O **R**	| N **G**	| C **B**	| =N **R**	| =C **G**	| #C **R**

Table 6: Derivation rules of SELFIES for a semantically restricted graph, with $\mathbf{S} \rightarrow \mathbf{B}$. A double edge between two vertices is denoted as = and a triple edge is shown as #.

Now we derive the SELFIES $A_1A_2A_6A_0$. The derivation starts in the state B:

$$\mathbf{S} \rightarrow \mathbf{B} \xrightarrow{A_1} \mathtt{O}\mathbf{R} \xrightarrow{A_2} \mathtt{ON}\mathbf{G} \xrightarrow{A_6} \mathtt{ON{=}C}\mathbf{G} \xrightarrow{A_0} \mathtt{ON{=}C} \tag{1}$$

The **action of the rule vector** can be seen here: At derivation of symbol A_6, the rule vector contained a triple bond, but this would violate the constraints of N. That information has been encoded into the state X_1, by which it enforced the construction of a semantically valid graph.

The derivation can be interpreted as transitions between different states (which are encoded in color) using rule A_i. The different states encode the domain-specific semantic restrictions. Here, the domain-specific constraint is the vertex-degree (i.e. the number of edges connected to the vertex) of the specific vertex. In chemistry, for example, the domain-specific constraints are limited numbers of bonds which each atom can form. The process is visualized in Figure 6.

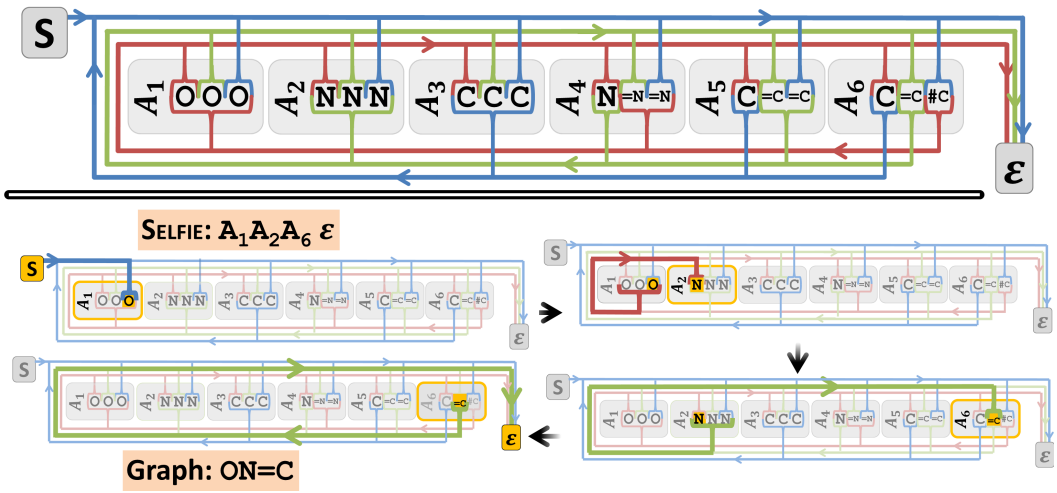

Figure 6: Decoding of SELFIES $A_1A_2A_6A_0$ to the graph ON=C, using the grammar in Table 2 in the main text. The third symbols is A_6, which contains a triple edge connection. However, this triple connection would violate the semantic constrained for the vertex N, which has $deg(\mathtt{N}) \leq 3$. This constrained is memorized in the internal state **G**, thus is not violated.

Chemical neighborhood interpretation of representation

Here we present a more detailed chemical interpretation of Figure 3 of the main text, about the chemical phenotype of molecules that are in the neighborhood of the representation.

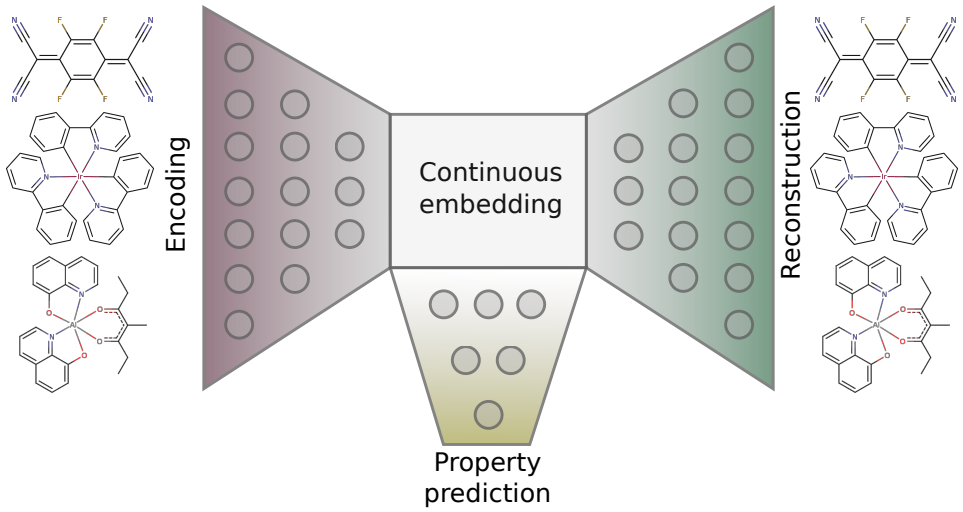

Figure 7: Conceptual structure of a variational autoencoder, which consists of three coupled neural networks – the encoder, the decoder and the prediction network. The encoder (left) maps a discrete representation (in our case: molecules) into a continuous latent space. The decoder (right) translates a point of the latent space back to a discrete representation. The encoder and decoder are trained together for optimizing the reconstruction quality. The input of the property prediction network (bottom) is a point of the latent space. The prediction network is trained together with the encoder, which leads to a property-dependent shaping of the latent space, which is significant for *inverse design* tasks.

The logP coefficient on average decreases slightly for all representations, indicating a slightly higher water solubility for the mutated molecules. This result is more pronounced in the case of SMILES and DeepSMILES representations than in case of the GrammarVAE and SELFIES. The molecular weight changes are centered around zero, indicating no systematic change. In the case of SELFIES, there is, however, an elongated tail to molecules with significantly lower molecular weight, which can be attributed to the introduction of terminal groups in the SELFIES string which ends the molecule. HOMO and LUMO energy changes are centered around zero with most of the molecules being in an interval of 2 eV width. The distribution of HOMO changes is slightly asymmetric, in particular in case of SMILES and DeepSMILES, favoring positive changes. This corresponds to a destabilization of the aromatic structure of the non-fullerene molecules which typically shifts the HOMO to higher energies.

Variational Auto Encoders

A variational autoencoder utilizes three simultaneously trained neural networks - an encoder, a decoder, and a prediction network, see Figure 7. The encoder produces a continuous, fixed-dimensional latent dimension of the representations, while the decoder reconstructs the original input from points in the latent space. The latent space is encouraged to follow a Normal distribution by adding Kullback-Leibler divergence to the loss function.

Decoding samples from the latent-normal distribution leads to the production of new samples. The model is trained to learn the distribution of high dimensional data - maximizing the reconstruction probability of the training set.

In our experiment, the encoder receives input strings (molecules in one of the four representations) as one-hot encodings of characters, and employs a fully-connected network to form the latent space. From a point of the latent space, a recurrent neural network (RNN) decoder is employed to reconstruct the input on a character-by-character basis. Over successive time steps, the network outputs a distribution over possible characters and is trained to reconstruct the training sample. The Problem of exploding & vanishing gradients is addressed using stacked GRU Cells [56] and gradient clipping [57]. For the purpose of inverse design, a fully-connected neural network is trained on the

latent space for predicting property values. In all cases we split the dataset into 50% training set and 50% test set.

Hyperparameter optimization

For a fair comparison of the four representations (SMILES, DeepSMILES, GrammarVAE, SELFIES), the architecture of the three networks has been optimized for each of the representation.

Hyperparameter optimizations of the VAE architectures were conducted to improve on the validity, reconstruction quality, and diversity of the decoded molecules in decreasing order of importance. As a loss function, we cannot use a simple weighted sum of objectives, as a weighted sum cannot resolve the *Pareto surface* [42]. Objective scores were scalarized with *Chimera*, where tolerances of 30 % were applied to each objective, and a fourth (least important) objective has been constructed from the weighted sum of the validity (50 % weight), reconstruction quality (30 % weight) and diversity (20 % weight) scores. For the experiment which involves the prediction network, the objective for optimizing the model was the reconstruction and prediction quality. The hyperparameters where optimized over the following domains:

- size of encoder layer 1/2/3: [250-2000] / [200-1000] / [100-500]
- size of latent space: [100-300]
- number of GRU layers: [1-3]
- size of GRU layers: [20-200]
- learning rate: 10^{-1}-10^{-5}
- KL divergence strength: 10^{-1}-10^{-8}

The hyperparameters for the fuly connected prediction network are:

- size of prediction layer 1 / 2: [5-250] / [5-250]
- learning rate of prediction: 10^{-2}-10^{-5}
- prediction start epoch: [1-200]
- prediction update frequency: [1-10]

We run the hyperparameter optimization for each of the representation for the same amount of time, namely 7 days, each of them on a single GPU (i.e. 4 GPUs in total).

Potential applications to other domains in the physical sciences

SELFIES can be used independently of the domain, which we demonstrate here. Ideal targets for SELFIES grammar are different types of objects (which for the vertices) with vertex-dependent connectivity restrictions. In that case, rule vectors of grammars can be used to encode the restrictions on connectivities. Rings and Branches could be dependent on vertices as well, for example in the case of DNA, where nucleobases can only connect to their conjugate base. We now show now two different examples, one from physics and one from biology.

5.2 Quantum Optical Experiments

	S	**B**	**H**	**P**	**R**	**D**	**Y**	**Z**
$\mathbf{X}_0 \rightarrow$	[SPDC] $\mathbf{X}_2$	[BS] $\mathbf{X}_3$	[Holo] $\mathbf{X}_1$	[DP] $\mathbf{X}_1$	[Ref] $\mathbf{X}_1$	[Det]	$\mathbf{X}_0$	$\mathbf{X}_0$
$\mathbf{X}_1 \rightarrow$	[SPDC] $\mathbf{X}_1$	[BS] $\mathbf{X}_3$	[Holo] $\mathbf{X}_1$	[DP] $\mathbf{X}_1$	[Ref] $\mathbf{X}_1$	[Det]	$\mathbf{X}_1$	R($\mathbf{N}$)
$\mathbf{X}_2 \rightarrow$	[SPDC] $\mathbf{X}_1$	[BS] $\mathbf{X}_3$	[Holo] $\mathbf{X}_1$	[DP] $\mathbf{X}_1$	[Ref] $\mathbf{X}_1$	[Det]	B($\mathbf{N}$,$\mathbf{X}_0$)$\mathbf{X}_1$	R($\mathbf{N}$)$\mathbf{X}_1$
$\mathbf{X}_3 \rightarrow$	[SPDC] $\mathbf{X}_1$	[BS] $\mathbf{X}_3$	[Holo] $\mathbf{X}_1$	[DP] $\mathbf{X}_1$	[Ref] $\mathbf{X}_1$	[Det]	B($\mathbf{N}$,$\mathbf{X}_0$)$\mathbf{X}_2$	R($\mathbf{N}$)$\mathbf{X}_2$
$\mathbf{N} \rightarrow$	1	2	3	4	5	6	8	9

Table 7: Derivation rules of SELFIES for a semantically restricted graph that represents quantum optical experiments, with the derivation starting in $\mathbf{X}_0$.

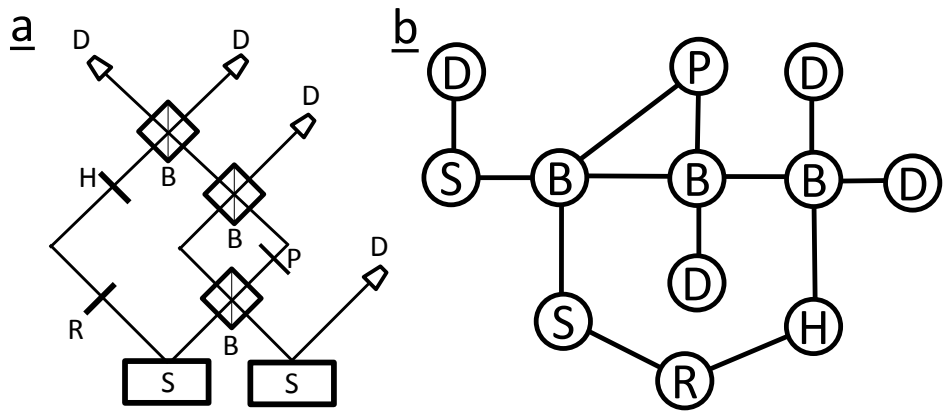

Figure 8: SELFIES for quantum optical experiments. In (a) we see the graph generated from SELFIES for a recent high-dimensional multipartite quantum experiment [54]. In (b), the structure of the experimental configuration.

A grammar for the generation of quantum optical experments can be written in Table 8.

There, the non-terminal symbols stand for quantum optical components that are used in experiments (see [55] for more information), [SPDC] stands for a non-linear crystal that undergoes spontaneous parametric down-conversion to produce photon pairs, [BS] stands for beam splitters, [Holo] stands for holograms to modify the quantum state, [DP] stands for Dove prism which introduces mode dependent phases, [Ref] stand for mirrors which modify mode numbers and phases, and [Det] are single-photon detector. B($\mathbf{N}$,$\mathbf{X}_0$) and R($\mathbf{N}$) are branch functions and ring functions as defined in the main text. Now we derive a recent complex quantum optical experiment (which has been designed by a computer algorithm), which demonstrates high-dimensional multi-partite quantum entanglement [54]. The SELFIES string for the experimental configuration is SYSDBBYHPZSYSDBYSDYSDHRSZD, the derivation is performed in the following way:

$$
\begin{aligned}
&\mathbf{X_0} \xrightarrow{S} \texttt{[SPDC]}\mathbf{X_2} \\
&\xrightarrow{YSD} \texttt{[SPDC]([D])}\mathbf{X_1} \\
&\xrightarrow{B} \texttt{[SPDC]([D])[BS]}\mathbf{X_3} \\
&\xrightarrow{B} \texttt{[SPDC]([D])[BS][BS]}\mathbf{X_3} \\
&\xrightarrow{YHPZS} \texttt{[SPDC]([D])[BS]1[BS]([P]1)}\mathbf{X_2} \\
&\xrightarrow{YSD} \texttt{[SPDC]([D])[BS]1[BS]([P]1)([Det])}\mathbf{X_1} \\
&\xrightarrow{B} \texttt{[SPDC]([D])[BS]1[BS]([P]1)([Det])[BS]}\mathbf{X_3} \\
&\xrightarrow{YSD} \texttt{[SPDC]([D])[BS]1[BS]([P]1)([Det])[BS]([Det])}\mathbf{X_2} \\
&\xrightarrow{YSD} \texttt{[SPDC]([D])[BS]1[BS]([P]1)([Det])[BS]([Det])([Det])}\mathbf{X_1} \\
&\xrightarrow{H} \texttt{[SPDC]([D])[BS]1[BS]([P]1)([Det])[BS]([Det])([Det])[Holo]}\mathbf{X_1} \\
&\xrightarrow{R} \texttt{[SPDC]([D])[BS]1[BS]([P]1)([Det])[BS]([Det])([Det])[Holo][R]}\mathbf{X_1} \\
&\xrightarrow{S} \texttt{[SPDC]([D])[BS]1[BS]([P]1)([Det])[BS]([Det])([Det])[Holo][R][SPDC]}\mathbf{X_1} \\
&\xrightarrow{ZD} \texttt{[SPDC]([D])[BS]12[BS]([P]1)([Det])[BS]([Det])([Det])[Holo][R][SPDC]2}
\end{aligned}
\tag{2}
$$

The derived graph and the corresponding setup can be seen in Figure 8.

If **N** cannot represent the length of the ring or the length of a branch, one can introduce a second symbol which represents longer rings, and which derives the next two SELFIES symbols for reconstructing a number.

Representation of DNA and RNA

		A	**G**	**C**	**T**	**R**
$\mathbf{X}_0$	$\rightarrow$	`A` $\mathbf{X}_1$	| `G` $\mathbf{X}_1$	| `C` $\mathbf{X}_1$	| `T` $\mathbf{X}_1$	| $\mathbf{X}_0$
$\mathbf{X}_1$	$\rightarrow$	`A` $\mathbf{X}_1$	| `G` $\mathbf{X}_1$	| `C` $\mathbf{X}_1$	| `T` $\mathbf{X}_1$	| R(**N**)$\mathbf{X}_0$
N	$\rightarrow$	1	| 2	| 3	| 4	| 5

Table 8: Derivation rules of SELFIES for a semantically restricted graph that represents DNA. `A` is Adenin, `C` is cytosin, `G` is Guanin and `T` is Thyim. The DNA is a single strain, and the rings R(**N**) can connect the nucleobase with its conjugate base **N+1** bases before. This grammar and represent, for example DNA origamis or tRNA (by replacing `T` with `U` for Uracil.)

An important structure, which can be represented as a semantically constrained graph, is DNA and RNA. We demonstrate here a simple variation of SELFIES which can be used for design questions in DNA nanotechnologies.

Acronyms for representing chemical compound systems

Akronym	**full name**	**reference**
WLN	Wiswesser Line Notation	[23]
SNL	SYBYL Line Notation	[24]
SMILES	Simplified Molecular Input Line Entry Specification	[19]
SMARTS	SMILES arbitrary target specification	[20]
SMIRKS	SMILES for reactions	[21]
InChI	International Chemical Identifier	[25]
SELFIES	SELF-referencIng Embedded Strings	here

Table 9: A list of common encoding systems in chemistry, with its corresponding full name.