PAPER • **OPEN ACCESS**

SMILES-X: autonomous molecular compounds characterization for small datasets without descriptors

MACHINE
LEARNING
Science and Technology

OPEN ACCESS

RECEIVED
30 July 2019

REVISED
24 October 2019

ACCEPTED FOR PUBLICATION
14 November 2019

PUBLISHED
18 March 2020

PAPER

SMILES-X: autonomous molecular compounds characterization for small datasets without descriptors

Guillaume Lambard[1] and **Ekaterina Gracheva**[2,3]

[1] Research and Services Division of Materials Data and Integrated System, Energy Materials Design Group, National Institute for Materials Science, 1-2-1 Sengen, Tsukuba, Ibaraki, 305-0047, Japan

[2] International Center for Materials Nanoarchitectonics, National Institute for Materials Science, 1-2-1 Sengen, Tsukuba, Ibaraki, 305-0047 Japan

[3] University of Tsukuba, 1-1-1 Tennodai, Tsukuba, Ibaraki, 305-8577 Japan

E-mail: LAMBARD.Guillaume@nims.go.jp

Keywords: cheminformatics, small molecules, SMILES, natural language processing, machine learning, attention mechanism, small datasets

Abstract

There is more and more evidence that machine learning can be successfully applied in materials science and related fields. However, datasets in these fields are often quite small (from tens to several thousands of samples). This means the most advanced machine learning techniques remain neglected, as they are considered to be applicable to big data only. Moreover, materials informatics methods often rely on human-engineered descriptors, that should be carefully chosen, or even created, to fit the physicochemical property that one intends to predict. In this article, we propose a new method that tackles both the issue of small datasets and the difficulty of developing task-specific descriptors. The SMILES-X is an autonomous pipeline for molecular compounds characterisation based on a {Embed-Encode-Attend-Predict} neural architecture with a data-specific Bayesian hyper-parameters optimisation. The only input to the architecture—the SMILES strings—are de-canonicalised in order to efficiently augment the data. One of the key features of the architecture is the attention mechanism, which enables the interpretation of output predictions without extra computational cost. The SMILES-X achieves state-of-the-art results in the inference of aqueous solubility ($\overline{RMSE}_{\text{test}} \simeq 0.57 \pm 0.07$ mols/L), hydration free energy ($\overline{RMSE}_{\text{test}} \simeq 0.81 \pm 0.22$ kcal/mol, which is ~24.5% better than molecular dynamics simulations), and octanol/water distribution coefficient ($\overline{RMSE}_{\text{test}} \simeq 0.59 \pm 0.02$ for LogD at pH 7.4) of molecular compounds. The SMILES-X is intended to become an important asset in the toolkit of materials scientists and chemists. The source code for the SMILES-X is available at github.com/GLambard/SMILES-X.

1. Introduction

In the fields of bio- and cheminformatics, machine learning (ML) algorithms combined with human-engineered molecular descriptors [1] have shown great potential in tasks of predicting physicochemical properties of molecular compounds. In practice, however, it is often necessary to run a blind scan through a large number of such combinations (there are over 6000 descriptors available) in order to find the most accurate inference model, which still may not lead to success. Most of the descriptors are task- or domain-specific, and their format makes it impossible to use them for more general problems, such as virtual screening, similarity searching, clustering and structure-activity modelling [2–5].

For these purposes molecular fingerprints have been developed [6]. A fingerprint is a binary representation of a molecule: its structural or functional features are translated into a string of bits as a way to keep the fingerprint invariant to rotations, translations and property-preserving atomic permutations (see, e.g., extended circular fingerprints [7]). Even though fingerprints are known to be beneficial to tasks like similarity searches, their development requires some level of simplification, information loss and/or bias towards the field and molecular target they have been designed for.

In 2014 Cadeddu *et al* [8] demonstrated that chemical language is quantitatively similar to a natural language, which implies that molecules can be considered as chemical sentences. This finding encouraged the use of machine learning algorithms based on the text representations of the molecules. The most common molecular text representation is SMILES (simplified molecular input line entry system) [9, 10]. Applying natural language processing (NLP) techniques to the SMILES allowed, for example, to build a powerful generative model for de novo drug design [11]. Another study shows state-of-the-art results in reaction prediction problem [12]. These and other works prove that one can achieve best to date results by applying machine learning directly on SMILES and that fully data-driven machine learning approaches can outperform the methods that depend on hand-encoded features [11–15]. Yet it is usually implied that the usage of deep neural architectures (NAs) requires the presence of big data.

In fields like materials science, however, it is common to have datasets containing from several tens to several thousands of samples, which is considered to be too small for a direct deep learning application. Some research groups apply them for secondary tasks such as building novel high-level features as non-linear combinations of molecular descriptors [16–18]. Others apply deep learning to learn features based on 2D/3D images [19, 20], molecular graphs [21], N-gram graphs [22] or a combination of mentioned inputs [23], similar to computer vision (CV). Still none of them intend to develop a deep NA specifically for small datasets. There are some works on transfer learning [11, 16, 24, 25], but the results vary greatly depending on the correlation between the tasks—which is often unknown *a priori*. This situation is comparable to the fields of CV or NLP, where most of the used NAs are trained on big data and impose architectures that do not fit small datasets.

Aside from the lack of data, another bottleneck on the way to using NAs in physics and chemistry is the lack of interpretability of the trained models. A method for explaining neural networks has been recently proposed [19]. It consists of training an additional neural network to generate a mask identifying the most important SMILES characters. Despite the respectable coherence in the interpretation of the chemical solubility, the explanation network is entirely correlated to its prediction network, which forces the training phase to be doubled for each dataset. Moreover, even though the explanation network allows us to identify the groups that have the highest weight in the property prediction, there is no evidence that the original prediction network has also learned the known chemistry concepts in order to make proper characterisation.

In this article we propose a method allowing to overpass the issues of data scarcity, descriptors engineering and the prediction interpretation ambiguity at the same time. To achieve this, we borrowed the latest techniques from the CV and NLP fields to build an entirely autonomous system—the SMILES-X. As the name implies, we use SMILES molecular representation as the sole input. It allowed us to implement an augmentation procedure similar to Bjerrum [26]. The key feature of the algorithm is the attention mechanism [27]. Not only does it allow us to get more information out of small data by reading deeper into the SMILES, but also it provides a straightforward interpretation of the model's output at no extra cost. The attention layer also allows us to reduce the number of trainable parameters, keeping the architecture relatively light. We use a simplified version of attention applicable to feed-forward networks [28]. The attention mechanism has already been applied in the field of cheminformatics [12]. But to the best of our knowledge, this is the first time in materials science related fields when an NA is specifically designed to manage small datasets, and the first attempt to integrate an NLP-based attention mechanism for predicting physicochemical properties of molecular compounds. The SMILES-X can be used to predict any physicochemical property given the molecule's SMILES and is intended to become an important asset in the toolkit of materials scientists and chemists. The algorithm achieves the state-of-the-art results on three benchmark datasets.

The structure of the article is as follows. First, we describe the entire pipeline of the SMILES-X in section 2. The SMILES augmentation and formatting are detailed in sections 2.1 and 2.22.1, respectively, while the procedures of building the NA frame and its data-specific optimisation are presented in the section 2.3. Section 3.1 is dedicated to the performance of the SMILES-X based on three benchmark datasets for regression tasks from the MoleculeNet [29]: ESOL [30], FreeSolv [31] and Lipophilicity [32]. There are three modes of interpretation of the results of the SMILES-X, which are discussed in section 3.2. Finally, we conclude and discuss further possible improvements of the SMILES-X, as well as propose more potential target properties to be inferred using the algorithm, in section 4.

2. The SMILES-X pipeline

The SMILES-X has been conceived to meet the following requirements: (i) to use the SMILES format as the only representation of a molecular compound; computable characteristics, such as the fingerprints or physical descriptors, are left out. (ii) Remove the SMILES canonicalization [9] in order to exploit the full capacity of the

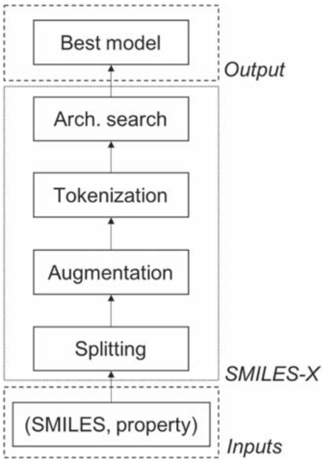


Figure 1. The SMILES-X pipeline.

molecular compound representation. (iii) The core architecture is simple enough to handle small datasets without sacrificing the prediction accuracy. (iv) Outcomes of the SMILES-X are interpretable.

Figure 1 is a sketch of the main steps within the SMILES-X pipeline. The primary input is a list of SMILES strings with corresponding property values. Then, a splitting into training, validation and test sets is performed via equiprobable sampling. The subsequent steps are detailed below.

2.1. Augmentation

It has been shown in CV that data augmentation approaches such as flipping, rotation, scaling, cropping and other image transformations are effective to reduce the error rate on classification tasks and improve generalisation [33]. Here, we introduce a technique called SMILES augmentation, similar to Bjerrum [26]. The first step consists in removing canonicalization [9] of the SMILES. Canonicalization is the default procedure to standardise the SMILES across the databases, therefore removing it leads to an expanded number of SMILES individual representations. Then, augmentation is done by iterating over the following two steps: (i) Renumber the atoms of a given SMILES by rotation of their index. (ii) For each renumbering, reconstruct grammatically correct SMILES under the condition of conserving the initial molecule's isomerism and prohibiting Kekulisation [9, 10]. In the end, one obtains an expanded list of SMILES together with their corresponding property and cardinality $n_{augm}(s_i)$ (number of augmentations for a SMILES s_i), if any. Duplicated SMILES are removed. The SMILES augmentation is individually performed after splitting into training, validation and test sets to avoid any information leakage. The procedure is performed using the RDKit library [34].

2.2. Tokenisation

Tokenisation consists in dividing the SMILES into unique tokens, each token being a set of characters. The procedure of SMILES tokenisation is as follows [9, 10]: (i) Aliphatic and aromatic organic atoms (B, C, N, O, S, P, F, Cl, Br, I, b, c, n, o, s, p), bounds, branches and rings (−,=, #, $, /, \, ., (,), %digits, digit) are set as individual tokens. (ii) The characters between squared brackets, that may include inorganic and aromatic organic atoms, isotopes, chirality, hydrogen count, charges or class number, form a single token (brackets included, e.g., $[NH^{4+}]$). (iii) Unlike the NLP analysis, the beginning token is not different from the termination one: both of them are represented by a white space, which is added at both ends of a tokenized SMILES. This is important to keep its reading direction invariant. Finally, a set of unique tokens is extracted to form the representative chemical vocabulary for a given dataset. To become an interpretable NA input, this vocabulary is then mapped into integers, and is conserved into memory for future usage.

2.3. Architecture search

The neural architecture search has recently reached a new milestone in finding the optimal NA for a given task, by using, e.g., reinforcement learning techniques [35, 36] or evolutionary algorithms [37]. However, not only these techniques are computationally expensive but also they do not necessarily deal with the recurrent blocks. It has therefore been decided to fix the overall NA geometry (figure 2) and search for the best set of the hyperparameters through the Bayesian optimisation [38]. As it was mentioned earlier in section 2, this geometry is NLP-oriented and treats SMILES strings as sentences in the chemical language; it has low complexity so as to be applicable to small datasets, and its outcomes are interpretable. Inspired by the hierarchical neural

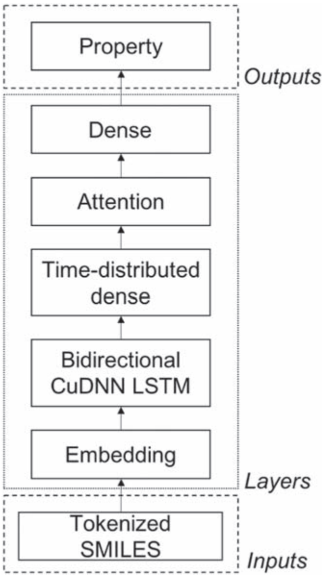


Figure 2. Fixed skeleton of the neural architecture in the SMILES-X.

architecture [39], which allows us to get cutting edge results on document classification, we have built the SMILES-X frame based on a four-step formula: {Embed, Encode, Attend, Predict} [40].

1. **Embed** The embedding layer [41] transforms the tokens, derived from the dataset's vocabulary in form of integers, into dense $\mathrm{n_{embed}}$-dimensional float vectors. Unlike arbitrary ordinal numbers, these vectors encapsulate the semantic meaning of tokens and their relations. This operation transforms SMILES into series of $\mathrm{n_{embed}} \times 1$ vectors, or $\mathrm{n_{tokens}} \times \mathrm{n_{embed}}$ tensor, where $\mathrm{n_{tokens}}$ corresponds to the number of tokens in a tokenised SMILES string.

2. **Encode** The encoding phase is responsible for modifying the embedding, so that it captures the relationships between tokens in the context of the dataset. It consists of two neural layers: a bidirectional CuDNN long short-term memory (LSTM) layer [42, 43] is followed by a time-distributed fully connected one. The former consists of $\mathrm{n_{LSTM}}$ LSTM blocks and maps the input SMILES, represented now by a $\mathrm{n_{tokens}} \times \mathrm{n_{embed}}$ tensor, into a context-aware $\mathrm{n_{tokens}} \times \mathrm{n_{LSTM}}$ tensor. After training, each row of the tensor represents the meaning of a given token within the context of the rest of the SMILES string containing it. The bidirectionality forces the embedded SMILES to be sequentially passed forwards and backwards, conserving the invariance of their reading direction. The forward and backward encodings of a SMILES are then concatenated, resulting in a $\mathrm{n_{tokens}} \times 2\mathrm{n_{LSTM}}$ output tensor. The time-distributed dense layer is then applied to each of $\mathrm{n_{tokens}}$ tokens. This allows us to capture the relationships between tokens in greater detail, or in other words to deepen the LSTM layer (similar to the effect of adding an extra dense layer to a vanilla neural network). Given that the number of hidden units in this layer is $\mathrm{n_{dense}}$, the output after encoding is a $\mathrm{n_{tokens}} \times \mathrm{n_{dense}}$ tensor. It should be noted that we specifically use CuDNN LSTM [44] blocks for efficient optimization and training phases on GPU from NVIDIA Corporation. Without the CuDNN version of LSTM, the speed of training would drop by a factor of ~10, making the optimisation phase intractable.

3. **Attend** The attention layer detects the salient tokens, compressing tensor $\mathrm{H} \in \mathbb{R}^{\mathrm{n_{tokens}} \times \mathrm{n_{dense}}}$ into an $\mathrm{n_{dense}}$ vector c with minimum information loss [28]:

$$\begin{aligned} e &= \tanh(\mathrm{H} \cdot \mathrm{W_a} + \mathrm{b_a}), \\ \alpha &= \frac{\exp(e)}{\sum_{i=1}^{\mathrm{n_{tokens}}} \exp(e_i)}, \\ c &= \mathrm{H}^{\mathrm{T}} \cdot \alpha, \end{aligned} \tag{1}$$

where $W_a \in \mathbb{R}^{\mathrm{n_{dense}} \times 1}$ and $b_a \in \mathbb{R}^{\mathrm{n_{tokens}} \times 1}$ are trainable parameters, $\alpha \in \mathbb{R}^{\mathrm{n_{tokens}} \times 1}$ is the attention vector and $c \in \mathbb{R}^{\mathrm{n_{dense}} \times 1}$ is the output. Thus, the attention layer performs two important tasks at once: (1) it collapses the representation H of a variable length chain of tokens into a fixed length vector c by applying a weighted sum over the tokens to fit the final property best, with (2) the weights in α which represent the importance of each token towards the final property prediction, bringing to a straightforward interpretation. Therefore, the attention layer has two modes, one returning the output vector c, and the other—the attention vector α (see section 3). The two modes are switchable at will without extra computational cost.

4. **Predict** The final NA layer transforms the attention layer output c into a single property value $\mathrm{Prop}(s_i)$ by a simple linear operation:

$$\mathrm{Prop}(s_i) = W_p^T \cdot c + b_p, \tag{2}$$

The interpretation from α in equation (1) and the prediction are thus linearly connected and are accessible without any additional treatments on the input data or NA, unlike the pipelines in other works [14, 45, 46].

It should be noted that all the above tensors or vectors have one additional dimension, n_{SMILES}, omitted for the sake of simplicity. This dimension corresponds to the batch size of a single iteration passed to the network, i.e. the maximum number of SMILES that it processes at once. All of the steps above are implemented in Keras API [47] and Tensorflow [48] with GPU support.

3. Results and discussion

To evaluate the regression performance of the SMILES-X, it was chosen to test it on three benchmark physical chemistry datasets issued from the MoleculeNet [29]. These datasets are considered as small, with less than 5000 compound-property pairs, and therefore present a challenge to machine learning models. The ESOL [30] dataset contains the logarithmic aqueous solubility (mols/L) for 1128 organic small molecules; the FreeSolv [31] consists of the calculated and experimental hydration free energies (kcal/mol) for 642 small neutral molecules in water; and the Lipophilicity [32] stores the experimental data on octanol/water distribution coefficient (logD at pH 7.4) for 4200 compounds.

In present report the splitting ratio for training/validation/test is set to 0.8/0.1/0.1. The results are reported based on 8 splits, each split using new seed for the random data sampling. The seeds have been fixed for the sake of reproducibility. We use the averaged RMSE over the 8 test sets as the comparison metric of performance.

The optimal model architecture is determined via Bayesian optimisation individually for each split. We used the python library GPyOpt [49] for this purpose. The search bounds are as follows: $(\mathrm{n_{embed}}, n_{\mathrm{LSTM}}, n_{\mathrm{dense}}\ and\ n_{\mathrm{SMILES}}) \in \{8, 16, 32, 64, 128, 512, 1024\}$, $\gamma \in [2; 4]$ with a step of 0.1, where γ is related to the optimiser learning rate as $\mathrm{lr} \equiv 10^{-\gamma}$, making a total of 50421 configurations. For the Lipophilicity dataset, n_{SMILES} and learning rate are fixed to 1024 and 10^{-3}, respectively, leaving 343 potential architectures to search among. First, 25 architectures are randomly sampled and trained. Next, up to 25 architectures are proposed via the expected improvement acquisition function [50]. Each of the architectures is sequentially trained for 30 epochs for ESOL [30] and FreeSolv [31], and 10 for the Lipophilicity [32] set (these values have been chosen based on the speed/efficiency ratio). The best proposed architecture is finally trained using a standard Adam optimiser [51] with checkpoint and early stopping. The early stopping is configured to stop the training if the validation loss is not improving for 50 consecutive epochs, and a checkpoint saves the parameters of the model with the minimal validation loss. The maximum number of epochs is set to 300, but because of the early stopping condition this value has never been reached. Depending on whether the SMILES augmentation is requested or not, the code needs from 1 to 4 GPUs running in parallel.

3.1. Predictions

We compare the performance of SMILES-X against the best-to-date results from MoleculeNet [29], and for the FreeSolv [31] additionally to the calculations based on the molecular dynamics simulations [31] (table 1). The results in MoleculeNet [29] are reported for the molecular graph-based models that achieved the best results on a given dataset: concretely, a message passing neural network [52] for the ESOL and FreeSolv datasets, and a graph convolutional model [53] for the Lipophilicity [32] dataset. Bayesian optimisation is also used there for the layers size, batch size and learning rate. We include both the results on canonicalised SMILES (Can) and on SMILES that have been augmented (Augm) (see section 2.1). When a SMILES string s_i is augmented to n_{augm} strings, its predicted property value is averaged over n_{augm} predictions.

Note that the uncertainty on the RMSE values reported in table 1 have different sources for each of the three methods. Molecular dynamics calculations [29] derive it principally from the experimental errors. MoleculeNet [29] and SMILES-X do not take experimental errors into account, but report instead the standard deviation of RMSEs obtained after several runs (3 for the MoleculeNet, 8 for SMILES-X). Moreover, the MoleculeNet [29] performed runs with random data splitting and a fixed model and its hyperparameters. As for the SMILES-X, not only is the data re-sampled, but also the architecture search is performed from scratch for every run, most of the time resulting in a different architecture. This explains larger error bars for SMILES-X. To compare the performances, we conducted a two-sample t-test for mean values and computed one-tailed p-values (under

Table 1. Comparison of physicochemical properties predictions from the SMILES-X (Can, Augm) to the best performances in MoleculeNet [29] on the ESOL [30], FreeSolv [31] and Lipophilicity [32] datasets, and to molecular dynamics calculations [31] for the FreeSolv dataset only. Molecular dynamics calculations [31] report the error on RMSE based on the experimental error. MoleculeNet [29] and SMILES-X do not use experimental error, and report instead the standard deviation of RMSEs obtained after several runs. MoleculeNet [29] performed 3 runs with random data splitting, but a fixed neural architecture. SMILES-X performed 8 runs with both random data splitting and neural architecture search through Bayesian optimisation.

	$\overline{RMSE}_{\text{test}}$		
Method	ESOL	FreeSolv	Lipophilicity
MoleculeNet [29]	0.58 ± 0.03	1.15 ± 0.12	0.65 ± 0.04
Molecular dynamics [31]	—	1.51 ± 0.07	—
SMILES-X (Can)	0.70 ± 0.05	1.14 ± 0.17	0.68 ± 0.05
SMILES-X (Augm)	**0.57 ± 0.07**	**0.81 ± 0.22**	**0.60 ± 0.04**

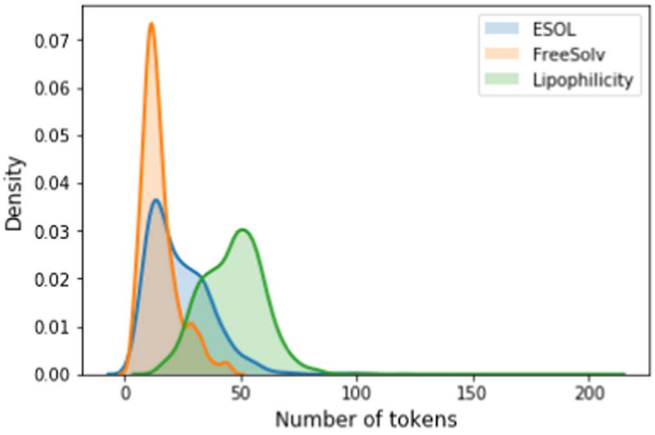


Figure 3. Number of tokens per SMILES for the datasets ESOL [30], FreeSolv [31], and Lipophilicity [32].

alternative hypothesis that SMILES-X outperforms MoleculeNet [29]). The obtained p-values are 0.103, 0.041 and 0.386 for Lipophilicity [32], FreeSolv [31] and ESOL [30] datasets, respectively. We conclude that the SMILES-X sets state-of-the-art results on the FreeSolv [31] dataset, while having comparable performance on the ESOL [30] and Lipophilicity [32] datasets.

It is unclear why our algorithm fails to improve on the ESOL data. We thought that the number of tokens per SMILES may be the culprit. However, figure 3 shows that this is not the case. Note that even using the standard canonicalised SMILES strings, the property can be predicted quite well without employing any chemical knowledge (i.e., using no descriptors).

To validate the performance of the SMILES-X on small data, we ran the algorithm on the FreeSolv dataset changing the training data size from 10% to 90% of the whole dataset. Figure 4 shows the obtained out-of-sample performances. Each datapoint corresponds to the average $\overline{RMSE}_{\text{test}}$ of 8 independent runs, with error bars indicating standard deviation. Already with a dataset containing as little as 200 points SMILES-X outperforms the molecular dynamics calculations.

There are three main reasons that we think allowed SMILES-X to achieve these results:

i. The success is mainly attributed to the attention layer, that shows similar improvements in document classification tasks [39]. Comparing our performance to a similar NA without an attention layer [14], we see some 32.5% improvement on accuracy.

ii. Bayesian optimisation is a valuable tool that allows us to efficiently find the best hyper-parameters in a short time.

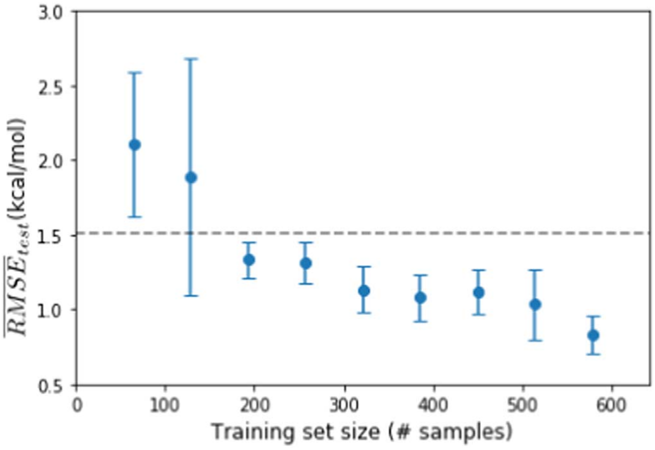


Figure 4. Out-of-sample performances of the SMILES-X on the FreeSolv [31] data. The algorithm is trained using the training set size ranging from 10% to 90% of the full dataset. Each datapoint corresponds to the average $\overline{RMSE}_{\text{test}}$ of 8 independent runs, with error bars indicating one standard deviation.

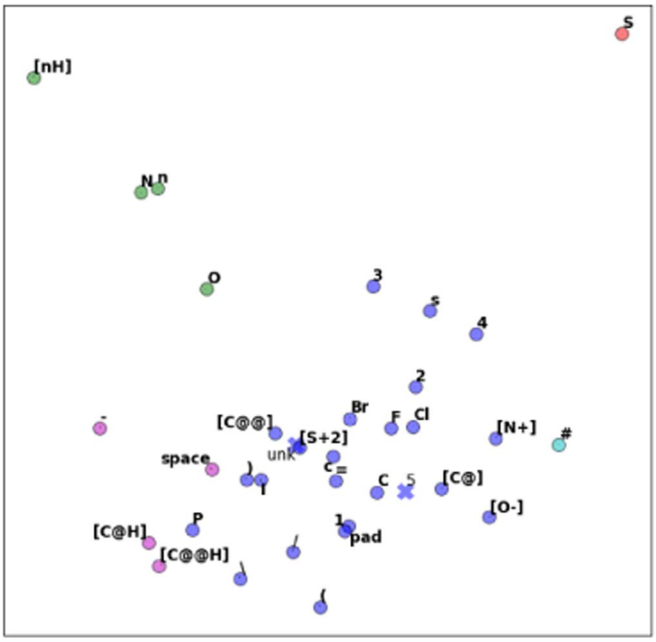
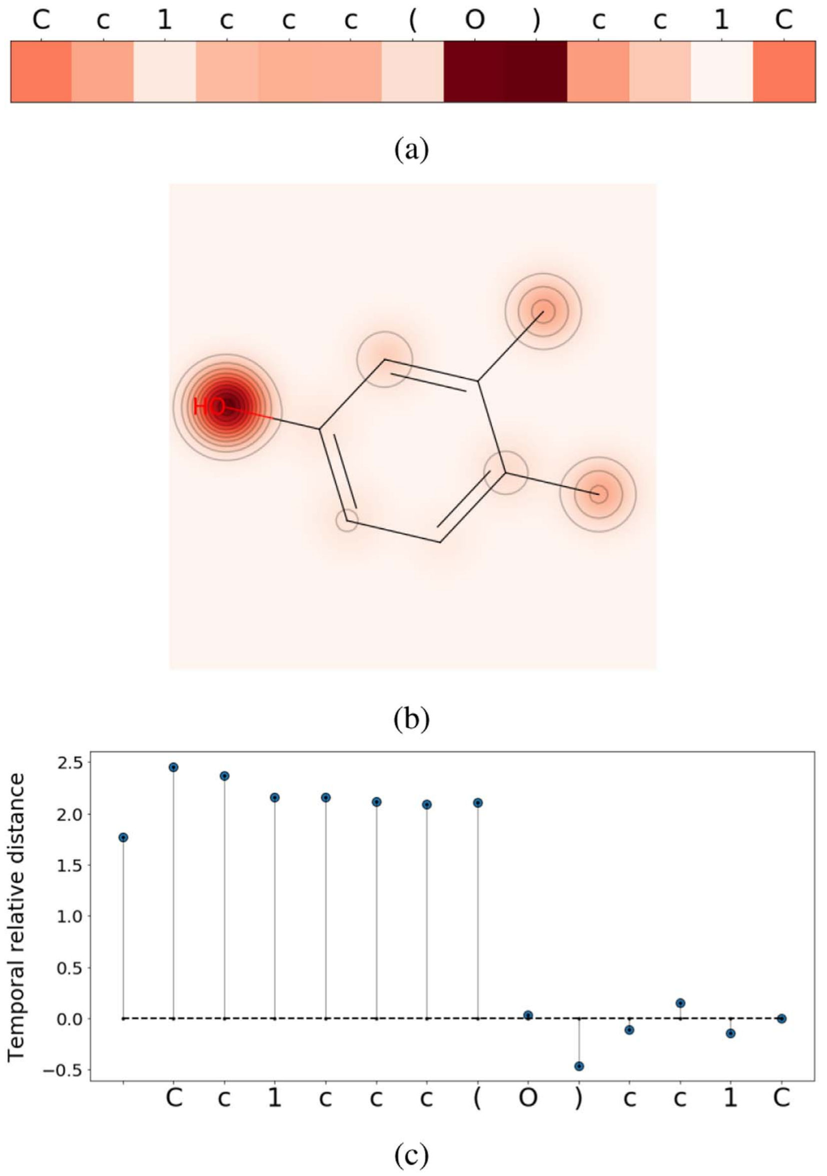


Figure 5. Visualisation of a representation of SMILES tokens from the embedding layer for the FreeSolv [31] dataset.

iii. It is obvious that SMILES augmentation shows great improvement (Can versus Augm in table 1), and was necessary to achieve the best current results. Also, one can note that a graph-based NA would not allow such data augmentation.

3.2. Interpretability

As it was mentioned before, one of the great advantages of our method is its interpretability. The figure 5 shows an example of the trained token embeddings for the FreeSolv [31] dataset. We used a principal component analysis (PCA [54, 55]) to reduce dimensionality from $\text{n}_{\text{embed}} = 1024$ down to two, for the purpose of visualisation. The tokens that are not included in the training set, and are therefore randomly assigned, are represented by a cross. One can see that halogens Br, F, Cl are located near each other. Other distinguishable sets are, for example, $\{[C\,@@],\ [S + 2],\ c,\ C,\ [C@]\}$ and $\{n,\ N\}$, that have the same valence and bonds type within the group. The model also puts $\{[N +],\ [O -]\}$ close to each other, which reveals their regular coexistence in compounds within the FreeSolv data. Some other tokens' placements, however, are not obvious to chemically qualify. In any case, the principle aim of clustering is to smooth out the chemical relations; it serves as a trainable look-up table for the further context-aware processing of tokens. We should not, thus, expect too great a degree



(a)

(b)

(c)

Figure 6. Visualisation of the importance of each token within the SMILES towards the final prediction of the property of interest. The illustration is done on the structure Cc1ccc(O)cc1C from the FreeSolv [31] dataset, with hydration free energy as the corresponding property. The 1D (a) and 2D (b) attention maps show the projections of the attention vector α on the SMILES string and molecular graph, respectively. The redder and darker the colour is, the stronger is the attention on a given token. The temporal relative distance $\mathcal{T}_{\text{dist}}$ is shown in (c). The closer to zero is the distance value, the closer is the temporary prediction on the SMILES fragment to the whole SMILES prediction.

of interpretability at this step. Representation of the individual tokens out of their chemical context is not the objective of the SMILES-X.

Instead, we are interested in the interpretation of the network property prediction. With the SMILES-X, we are able to visualise the importance of each single token towards the final prediction of the property of interest (figure 6).

There are three ways of visualisation available: (a) a 1D map built from the attention vector α (see equation (1)) juxtaposed with the SMILES string, (b) a similar 2D version for the molecular graph and (c) temporal relative distance $\mathcal{T}_{\text{dist}}$ to the predicted property. For the first two, the redder and darker the colour is the stronger is the attention on a given token.

$\mathcal{T}_{\text{dist}}(\text{n})$ shows the evolution of the prediction for the SMILES while reading it token by token from left to right. It is inspired by Lanchantin [56] and defined as:

$$\mathcal{T}_{\text{dist}}(\text{n}) = \frac{\text{Prop}(\text{n}) - \text{Prop}(\text{n}_{\text{tokens}})}{|\text{Prop}(\text{n}_{\text{tokens}})|}, \quad (3)$$

where Prop(n) is the property predicted value based on the first n tokens of the SMILES for $\text{n} \in [1,\ldots,\text{n}_{\text{tokens}}]$. Note that it converges to the final prediction $\text{Prop}(\text{n}_{\text{tokens}}) \equiv \text{Prop}(\text{s}_\text{i})$ (prediction based on the entire SMILES).

This also allows us to judge as to how much a token influences the property of a compound. In this example, the prediction based on fragment 'Cc1ccc(O' is almost identical to the final prediction on the whole structure.

For the compound that we used as an example, the oxygen atom ('O') is considered to be the most influential element of the molecule for the hydration free energy prediction, which reflects chemical reality.

4. Conclusions

A new neural architecture for the chemical compounds characterisation, the SMILES-X, has been developed. In this article, we have presented the pipeline and performance of the SMILES-X. We demonstrate its aptitude to provide state-of-the-art results on the inference of several physicochemical properties, concretely the logarithmic aqueous solubility ($\overline{RMSE}_{\text{test}} \simeq 0.57 \pm 0.07$ mols L^{-1}), hydration free energy ($\overline{RMSE}_{\text{test}} \simeq 0.81 \pm 0.22$ kcal mol^{-1}) and octanol/water distribution coefficient ($\overline{RMSE}_{\text{test}} \simeq 0.60 \pm 0.04$ for LogD at pH 7.4). These results prove that it is now possible to successfully predict a physicochemical property employing no chemical intuition, even with a small dataset at hand. The success of the SMILES-X rides on three key factors: (i) The Embed-Encode-Attend-Predict architecture, that allows us to simplify the whole architecture thanks to the attention mechanism (i.e., to have less trainable parameters), and therefore reduces the risk of over-fitting. (ii) The Bayesian optimisation of the neural network's hyper-parameters allows us to achieve close-to-optimal representation of the molecular compounds, per task and dataset. (iii) The use of SMILES strings as a sole input representation of chemical compounds allows efficient data augmentation.

Thanks to the attention mechanism, the SMILES-X comes with three modes of interpretation of the inference outcomes. This provides the end-user with the insights on which fragments of the chemical structure have the highest (or the lowest) influence on the property of interest. This kind of artificial intuition is a valuable asset not only for the tasks of characterisation and design of novel compounds, but also to re-purpose already-known materials.

As for the future improvement on the SMILES-X, we plan to use BERT-like [57] NA's skeleton for the sake of reducing the accuracy gap existing between the ESOL, FreeSolv and Lipophilicity datasets studied here. The LSTM blocks are known to have memory problems with very distant dependencies within long sentences, and an architecture that is entirely based on the attention mechanism, i.e. free from LSTM blocks, like BERT, may overcome this weakness. Another way to improve the inference accuracy may be via informative sampling [58].

In our forthcoming article we will address the tasks of classification, still using the MoleculeNet's datasets [29]. That means that the SMILES-X will be modified in order to handle single-to-many, many-to-many and many-to-single classification tasks.

Acknowledgments

The authors gratefully acknowledge the NVIDIA Corporation for the Titan V and Titan Xp GPUs, without which this research would not have been possible.

Conflicts of interest

There are no conflicts to declare.

Data availability

The data that support the findings of this study are openly available at DOI 10.5281/zenodo.3517791.

ORCID iDs

Guillaume Lambard https://orcid.org/0000-0003-0275-4079
Ekaterina Gracheva https://orcid.org/0000-0002-9704-5939

References

[1] Todeschini R and Consonni V (ed) 2008 *Handbook of Molecular Descriptors* (Weinheim: Wiley-VHC)
[2] Willett P, Barnard J M and Downs G M 1998 *J. of Chem. Inf. Comput. Sci.* **38** 983–96
[3] Cereto-Massagué A, Ojeda M J, Valls C, Mulero M, Garcia-Vallvé S and Pujadas G 2015 *Methods* **71** 58–63
[4] McGregor M J and Pallai P V 1997 *J. Chem. Inf. Comput. Sci.* **37** 443–8

[5] Li H, Yap C, Ung C, Xue Y, Li Z, Han L, Lin H and Chen Y 2007 *J. Pharm. Sci.* **96** 2838–60

[6] Morgan H L 1965 *J. Chem. Doc.* **5** 107–13

[7] Rogers D and Hahn M 2010 *J. Chem. Inf. Model.* **50** 742–54

[8] Cadeddu A, Wylie E K, Jurczak J, Wampler-Doty M and Grzybowski B A 2014 *Angew. Chem.* **53** 8108–12

[9] Weininger D 1988 *J. Chem. Inf. Comp. Sci.* **28** 31–6

[10] OpenSMILES URL http://opensmiles.org/opensmiles.html

[11] Segler M H S, Kogej T, Tyrchan C and Waller M P 2018 *ACS Cent. Sci.* **4** 120–31

[12] Schwaller P, Gaudin T, Lányi D, Bekas C and Laino T 2018 *Chem. Sci.* **9** 6091–8

[13] Segler M H S, Preuss M and Waller M P 2018 *Nature* **555** 604–10

[14] Goh G B, Hodas N O, Siegel C and Vishnu A 2017 SMILES2Vec: an interpretable general-purpose deep neural network for predicting chemical properties (arXiv:1712.02034)

[15] Kimber T B, Engelke S, Tetko I V, Bruno E and Godin G 2018 Synergy effect between convolutional neural networks and the multiplicity of SMILES for improvement of molecular prediction (arXiv:1812.04439)

[16] Coley C W, Barzilay R, Green W H, Jaakkola T S and Jensen K F 2017 *J. Chem. Inf. Model.* **57** 1757–72

[17] Mayr A, Klambauer G, Unterthiner T and Hochreiter S 2016 *Front. Environ. Sci.* **3** 80

[18] Ramsundar B, Kearnes S, Riley P, Webster D, Konerding D and Pande V 2015 Massively multitask networks for drug discovery (arXiv:1502.02072)

[19] Goh G B, Siegel C, Vishnu A and Hodas N O 2017 Using rule-based labels for weak supervised learning: a ChemNet for transferable chemical property prediction (arXiv:1712.02734)

[20] Wallach I, Dzamba M and Heifets A 2015 AtomNet: a deep convolutional neural network for bioactivity prediction in structure-based drug discovery (arXiv:1510.02855)

[21] Duvenaud D K, Maclaurin D, Iparraguirre J, Bombarell R, Hirzel T, Aspuru-Guzik A and Adams R P 2015 Convolutional networks on graphs for learning molecular fingerprints *Advances in Neural Information Processing Systems 28* ed C Cortes *et al* (Curran Associates) pp 2224–32

[22] Liu S, Chandereng T and Liang Y 2019 N-Gram Graph: Simple Unsupervised Representation for Graphs, with Applications to Molecules *Advances in Neural Information Processing Systems* (32) ed H Wallach *et al* (Red Hook, NY: Curran Associates) 8466–78

[23] Paul A, Jha D, Al-Bahrani R, Keng Liao W, Choudhary A and Agrawal A 2018 CheMixNet: mixed DNN architectures for predicting chemical properties using multiple molecular representations (arXiv:1811.08283)

[24] Hutchinson M L, Antono E, Gibbons B M, Paradiso S, Ling J and Meredig B 2017 Overcoming data scarcity with transfer learning (arXiv:1711.05099)

[25] John P C S, Phillips C, Kemper T W, Wilson A N, Crowley M F, Nimlos M R and Larsen R E 2019 Message-passing neural networks for high-throughput polymer screening *J. Chem. Phys.* **150** 234111

[26] Bjerrum E J 2017 SMILES enumeration as data augmentation for neural network modeling of molecules (arXiv:1703.07076)

[27] Bahdanau D, Cho K and Bengio Y 2015 Neural machine translation by jointly learning to align and translate *III International Conference on Learning Representations, ICLR 2015 (San Diego, California)*

[28] Raffel C and Ellis D P W 2015 Feed-forward networks with attention can solve some long-term memory problems (arXiv:1512.08756)

[29] Wu Z, Ramsundar B, Feinberg E, Gomes J, Geniesse C, Pappu A S, Leswing K and Pande V 2018 *Chem. Sci.* **9** 513–30

[30] Delaney J S 2004 *J. Chem. Inf. Comp. Sci.* **44** 1000–5

[31] Mobley D L and Guthrie J P 2014 *J. Comput. Aid. Mol. Des.* **28** 711–20

[32] Gaulton A *et al* 2016 *Nucleic Acids Res.* **45** D945–54

[33] Perez L and Wang J 2017 The effectiveness of data augmentation in image classification using deep learning (arXiv:1712.04621)

[34] Landrum G RDKit: Open-source cheminformatics http://www.rdkit.org

[35] Zoph B and Le Q V 2016 Neural architecture search with reinforcement learning (arXiv:1611.01578)

[36] Pham H, Guan M Y, Zoph B, Le Q V and Dean J 2018 Efficient neural architecture search via parameter sharing *Proc. of the 35th Int. Conf. on Machine Learning* 80 *(Stockholm, Sweden, 10–15 July 2018)*

[37] Real E, Aggarwal A, Huang Y and Le Q V 2018 Regularized evolution for image classifier architecture search *Proc. of the AAAI Conf. on Artificial Intelligence* 33

[38] Frazier P I 2018 A tutorial on Bayesian optimization (arXiv:1807.02811)

[39] Yang Z, Yang D, Dyer C, He X, Smola A and Hovy E 2016 Hierarchical attention networks for document classification *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (San Diego, California: Association for Computational Linguistics)* pp 1480–9

[40] Honnibal M Embed, encode, attend, predict: the new deep learning formula for state-of-the-art NLP models URL https://explosion.ai/blog/deep-learning-formula-nlp

[41] Gal Y and Ghahramani Z 2016 A theoretically grounded application of dropout in recurrent neural networks *Proc. of the 30th Int. Conf. on Neural Information Processing Systems* 1027–35

[42] Hochreiter S and Schmidhuber J 1997 *Neural Comput.* **9** 1735–80

[43] Gers F A, Schmidhuber J and Cummins F A 2000 *Neural Comput.* **12** 2451–71

[44] Appleyard J, Kocisky T and Blunsom P 2016 Optimizing performance of recurrent neural networks on GPUs (arXiv:1604.01946)

[45] Montavon G, Samek W and Müller K R 2018 *Digit. Signal Process.* **73** 1–15

[46] Schütt K T, Arbabzadah F, Chmiela S, Müller K R and Tkatchenko A 2017 *Nat. Commun.* **8** 13890

[47] Chollet F *et al* 2015 Keras URL https://keras.io

[48] TensorFlow: large-scale machine learning on heterogeneous systems URL https://tensorflow.org

[49] The GyOpt Authors 2016 GPyOpt: a Bayesian optimization framework in Python URL https://github.com/SheffieldML/GPyOpt

[50] Jones D R, Schonlau M and Welch W J 1998 *J. Global Optim.* **13** 455–92

[51] Kingma D P and Ba J 2014 Adam: a method for stochastic optimization (arXiv:1412.6980)

[52] Gilmer J, Schoenholz S S, Riley P F, Vinyals O and Dahl G E 2017 Neural message passing for quantum chemistry *Proc. of the 34th Int. Conf. on Machine Learning* 70, 1263–72 (arXiv:1704.01212)

[53] Altae-Tran H, Ramsundar B, Pappu A S and Pande V 2017 *ACS Central Sci.* **3** 283–93

[54] Frs K P 1901 *Philos. Mag.* **2** 559–72

[55] Hotelling H 1933 *J Educ. Psychol.* **24** 417–41

[56] Lanchantin J, Singh R, Wang B and Qi Y 2016 *Biocomputing 2017* **22** 254–65

[57] Devlin J, Chang M W, Lee K and Toutanova K 2019 BERT: pre-training of deep bidirectional transformers for language understanding *Proc. of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies* 1, 4171–86

[58] Fernandez J and Downey D C 2018 Sampling informative training data for RNN language models *ACL 2018-LVI Annual Meeting of the Association for Computational Linguistics, Proceedings of the Student Research Workshop (Association for Computational Linguistics)* pp 9–13