

SMILES Enumeration as Data Augmentation for Neural Network Modeling of Molecules

Esben Jannik Bjerrum*

Wildcard Pharmaceutical Consulting, Frødings Allé 41, 2860 Søborg, Denmark

*) esben@wildcardconsulting.dk

Abstract

Simplified Molecular Input Line Entry System (SMILES) is a single line text representation of a unique molecule. One molecule can however have multiple SMILES strings, which is a reason that canonical SMILES have been defined, which ensures a one to one correspondence between SMILES string and molecule. Here the fact that multiple SMILES represent the same molecule is explored as a technique for data augmentation of a molecular QSAR dataset modeled by a long short term memory (LSTM) cell based neural network. The augmented dataset was 130 times bigger than the original. The network trained with the augmented dataset shows better performance on a test set when compared to a model built with only one canonical SMILES string per molecule. The correlation coefficient R2 on the test set was improved from 0.56 to 0.66 when using SMILES enumeration, and the root mean square error (RMS) likewise fell from 0.62 to 0.55. The technique also works in the prediction phase. By taking the average per molecule of the predictions for the enumerated SMILES a further improvement to a correlation coefficient of 0.68 and a RMS of 0.52 was found.

Introduction

Neural networks and deep learning has shown interesting application successes, such as image classification[1], and speech recognition[2]. One of the issues that limits their general applicability in the QSAR domain may be the limited sizes of the labeled datasets available, although successes do appear.[3] Limited datasets necessitates harsh regularization or shallow and narrow architectures. Within image analysis and classification, data augmentation techniques has been used with excellent results.[4, 5, 6, 7] As an example, a dataset of labeled images can be enlarged by operations such as mirroring, rotation, morphing and zooming. The afterwards trained network gets more robust towards such variations and the neural network can recognize the same object in different versions.

Neural networks has also been used on molecular data, where the input may be calculated descriptors,[3] neural network interpretation of the molecular graph[8] or also SMILES representations.[9] Simplified Molecular Input Line Entry System (SMILES) is a single line text based molecular notation format.[10] A single molecule has multiple possible SMILES strings, which has led to the definition of a canonical SMILES,[11] which ensures that a molecule corresponds to a single SMILES string. The possibilities for variation in the SMILES strings of simple molecules are limited. Propane has two possibilities CCC and C(C)C. But as the molecule gets larger in size and more complex in branching,

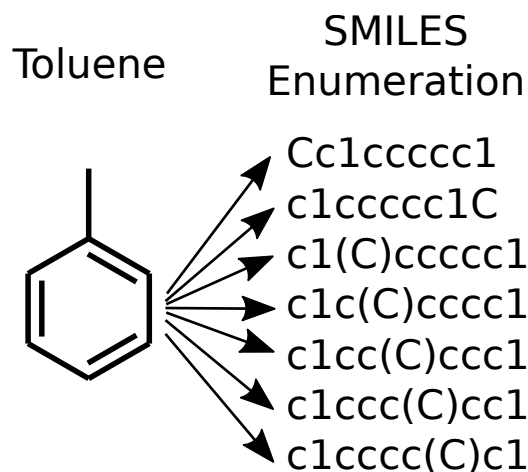


Figure 1: SMILES enumeration enables data augmentation. The molecule toluene corresponds to seven different SMILES, the top one is the canonical smile. One data point with toluene in the dataset would thus leads to seven samples in the augmented dataset.

the number of possible SMILES strings grows rapidly. Toluene with seven atoms, has seven possible SMILES strings (Figure 1).

Here data augmentation of molecular structures with SMILES enumeration for QSAR studies will be investigated using long short term memory (LSTM) cell neural networks inspired by networks used for Twitter tweets sentiment analysis.[12]

Methods

SMILES enumeration

SMILES enumeration was done with a Python script utilizing the cheminformatics library RDKit.[13] The atom ordering of the molecule is scrambled randomly by converting to molfile format[14] and changing the atom order, before converting back to the RDKit mol format. A SMILES is then generated using RDKit with the default option of producing canonical SMILES set to false, where different atom orderings lead to different SMILES. The SMILES strings are then compared and possible added to a growing set of unique SMILES strings. The process is repeated a predefined number of times. The python functions are available on github: <https://github.com/Ebjerrum/SMILES-enumeration>

Molecular dataset

The dataset was obtained from Sutherland et al 2003.[15] It consists of 756 dihydrofolate inhibitors with P. carinii DHFR inhibition data. The dataset was split in test and a training set in a 1:9 ratio. It was expanded with SMILES enumeration and the SMILES strings were padded with spaces to fixed length of 74, which is one character longer than the longest SMILES in the dataset. It was subsequently vectorized by one-hot encoding the characters into a bit matrix with one bit set for the corresponding character in each row using a generated char to int dictionary. Molecules where the associated affinity was not a number were removed. The associated IC50 data was converted to log IC50 and normalized to unit variance and mean zero with utilities from Scikit-learn.[16]

LSTM neural network

Two different neural networks were built and trained using Keras version 1.1.2[17] with Theano v. 0.8.0[18] as back end. One or more LSTM layers were used in batch mode, and the final state fed to a feed-forward neural network with a single linear output neuron. The network layout was optimized using Bayesian optimization with Gaussian processes as implemented in the Python package GpyOpt[19] version 1.0.3, varying the hyper parameters listed in Table 1. 10 initial trainings was done before using the GP_MCMC and the EI_MCMC acquisition function to sample new hyper parameter sets.[20] One network was optimized and trained only using a dataset with canonical SMILES, whereas the other were optimized and trained with the dataset that expanded with SMILES enumeration. In the rest of the publication they will be referred to as the canonical model and enumerated model, respectively.

All computations and training were done on a Linux workstation (Ubuntu Mate 16.04) with 4 GB of

ram, i5-2405S CPU @ 2.50GHz and an Nvidia Geforce GTX1060 graphics card with 6 GB of ram.

Results

Filtering, splitting and SMILES enumeration resulted in a canonical SMILES dataset with 602 train molecules and 71 test molecules, whereas the enumerated dataset had 79143 and 9412 rows for train and test, respectively. This corresponds to an augmentation factor of approximately 130. Each molecule had on average 130 alternative SMILES representations. Optimization of the architecture yielded two different best configurations of hyper parameters, depending on the dataset used. The best hyper parameters found for each dataset are shown in Table 2.

The train history is shown in Figure 2. The best neural network trained on the canonical dataset had a loss of 0.44 including regularization penalty and a mean square error of 0.22 and 0.41 for train and test set, respectively. The curves for the training using the canonical dataset are very noisy (Figure 2A). The best neural network trained on the enumerated dataset loss of 0.18 including regularization penalty and a mean square error of 0.09 and 0.30 for train and test set, respectively. The training curve is significantly less noisy than for the canonical dataset (Figure 2B).

Both neural networks were used to predict the IC50 values from the canonical and enumerated datasets, and the scatter plots are shown in Figure 3.

The correlation coefficients and root mean square deviation (RMS) are tabulated in Table 3. The combination with the worst performance was predicting the test set molecules is using enumerated SMILES neural network model trained on the canonical dataset. Which has a correlation coefficient of 0.26 and an RMS of 0.84. The bad correlation is clearly visible from Figure 3 plot C. The best performance predicting the test set, was seen with the combination of the enumerated model and the enumerated SMILES. Here the correlation coefficient is 0.66 and the RMS 0.55. The two other combinations, canonical model-canonical SMILES and enumerated model, canonical SMILES are close in performance (Table 3).²

Figure 4 show a scatter plot of the average prediction for each molecule obtained with the enumerated model. The calculated correlation coefficient is 0.68 for the test set and the RMS is 0.52.

Discussion

The results clearly suggest that SMILES enumeration as a data augmentation technique for molecular data has benefits. The model trained on canonical data is not able to predict many of the alternative SMILES of the train and test set as is evident for

Table 1: Hyper parameter Search Space

Parameter	Search Space	Type
Number of LSTM layers	[1,2]	Discrete
Number of units in LSTM layers	[32, 64, 128, 256]	Discrete
Dropout for input gates (dropout_W)	0 – 0.2	Continuous
Dropout for recurrent connection (dropout_U)	0 – 0.5	Continuous
Number of dense hidden layers	[0,1]	Discrete
Hidden layer size	[4, 8, 16, 32, 64, 128]	Discrete
Weight regularization on dense layer, L1	0 – 0.2	Continuous
Weight regularization on dense layer, L2	0 – 0.2	Continuous
Learning rate	0.05-0.0001	Continuous

Table 2: Best Hyperparameters found

Parameter	Canonical Model	Enumerated Model
Number of LSTM layers	1	1
Number of units in LSTM layers	128	64
Dropout for input gates (dropout_W)	0.0	0.19
Dropout for recurrent connections (dropout_U)	0.0	0.0
Number of dense hidden layers	0	0
Hidden layer size	N/A	N/A
Weight regularization on dense layer, L1	0.2	0.005
Weight regularization on dense layer, L2	0.2	0.01
Learning rate	0.0001	0.005

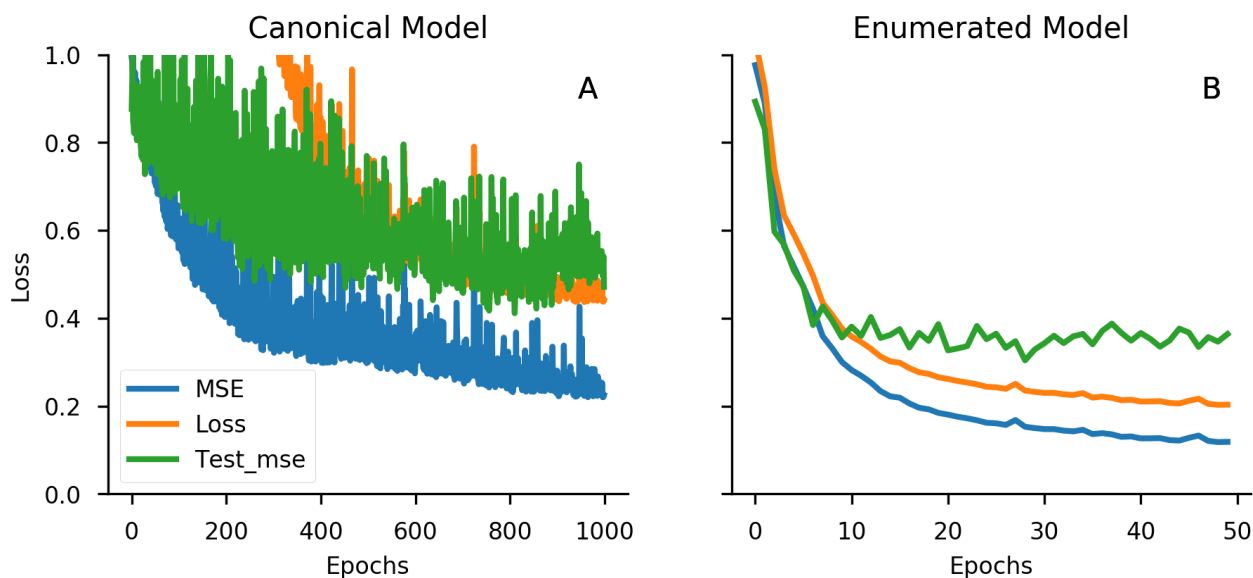


Figure 2: Training history for the two datasets and neural networks. A: Neural network trained on canonical SMILES shows a noisy curve where the best model has a test loss of 0.41. B: Neural network trained on enumerated SMILES obtains the best model with a test loss of 0.30. Blue lines are the mean square error without regularization penalty, green is loss including regularization penalty and the red line is mean square error on the test set.

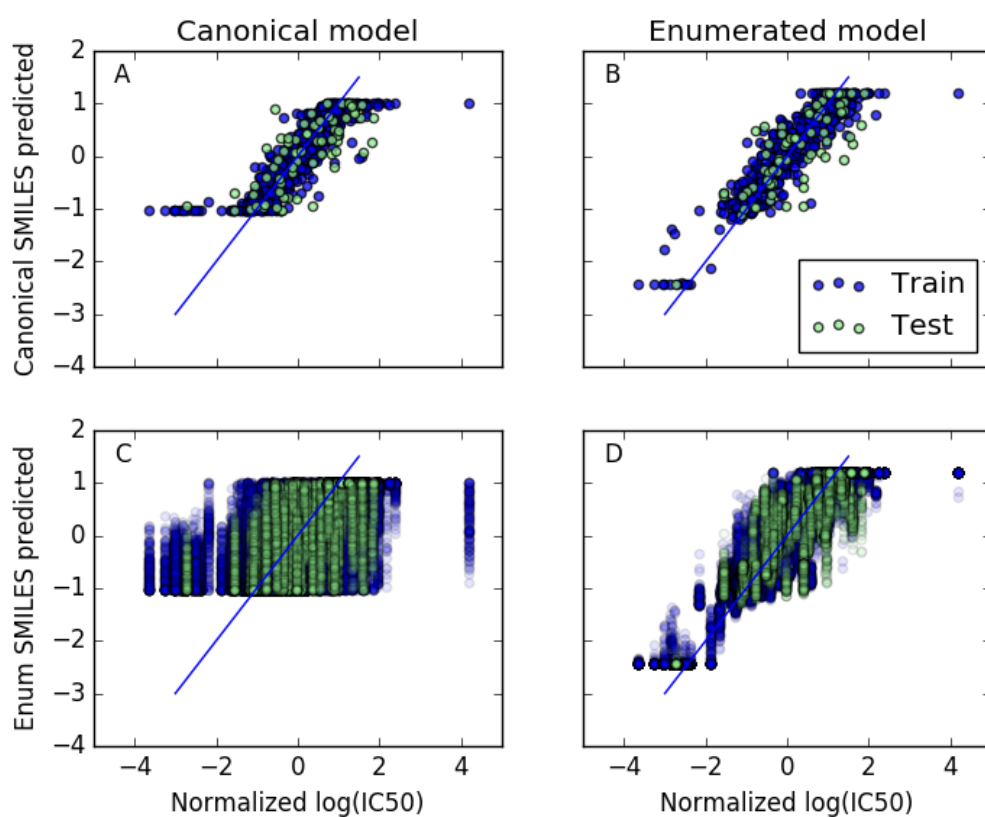


Figure 3: Scatter plots of predicted vs. true values. Left column shows scatter plots obtained with the model trained on canonical SMILES only. Right column shows predictions with the model trained on enumerated data. Top row is scatter plots with only canonical SMILES and bottom row is predictions of the enumerated dataset. The blue line denotes the perfect correlation ($y = x$).

Table 3: Statistics of predicted values, values are for Train/Test set respectively

Dataset		Canonical Model		Enumerated Model	
		R ²	RMS	R ²	RMS
Canonical	Train	0.78	0.46	0.85	0.39
	Test	0.56	0.62	0.63	0.56
Enumerated	Train	0.25	0.88	0.87	0.37
	Test	0.26	0.84	0.66	0.55

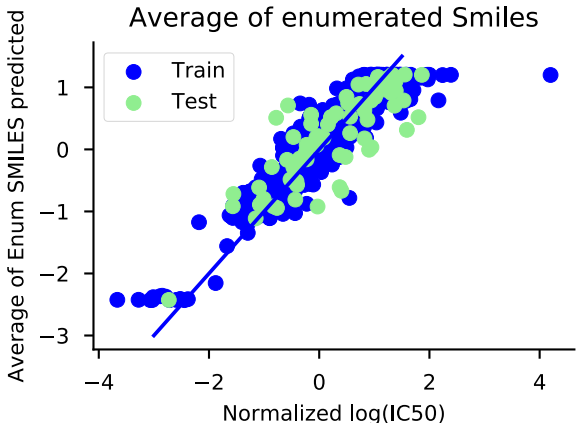


Figure 4: Average of predictions from the enumerated model for each molecule. Train set R² is 0.88 and RMS is 0.38. Test set R² is 0.68 and RMS is 0.52.

Figure 3 plot C, where the bad generalization to non canonical SMILES strings are evident. Instead the best performance was observed by taking the average for each molecule of the predictions of the enumerated SMILES using the enumerated model (Figure 4), which shows that the SMILES enumeration can also be of value in the sampling phase. The canonical model needed a lot more epochs to train, but here it must be considered that the dataset contained 130 times fewer examples. Thus each epoch in the training was only 3 mini batches leading to 3 updates of the weights, whereas the enumerated dataset had approximately 360 updates of the weights of the neural network per epoch. The curves in Figure 2 thus represents 3000 and 18000 updates of the weights. The higher overhead of running more epochs however led to approximately the same wall clock time in training. The hyper parameters found during the optimization of the network architecture and amount of regularization was not entirely as expected. The expectation was that the canonical dataset would prefer a smaller and simpler network with a larger regularization. Instead the canonical dataset has a larger amount of LSTM cells (128) with no dropout, but a much larger regularization of the final weights to the input neuron (L1 and L2 maxed out at 0.2). The enumerated model had fewer LSTM cells (64) and thus fewer connections, but nevertheless found dropout on the input to the LSTM cells to be beneficial. To test if

the differences were due to the Bayesian optimization getting trapped in a local minimum, the network architecture found for the enumerated dataset was test trained with the dataset with the canonical SMILES only. The first try with a learning rate of 0.005 failed (results not shown), but lowering the learning rate to the one found for the canonical SMILES (0.0001), gave a model with a correlation coefficient of 0.5 and RMS of 0.68 on the train set. The predictive performance was even lower with 0.45 and 0.69, for R² and RMS respectively. The differences in hyper parameters after optimization of using the two different datasets thus seems justified. The study lacks the division into train, test and validation set, where the hyper parameters are tuned on the test set, but the final performance evaluated on the validation set. The observed prediction performance of the LSTM-QSAR models are thus likely overestimated to some degree. However, this study is focused on the gains of using SMILES enumeration and not on producing the optimal DHFR QSAR model. The performance on both the train and test set are lower for the canonical model. If the differences in performance had been due to over-fitting, the smaller dataset would probably have had an advantage.

The use of SMILES as descriptors for QSAR is not new[21, 22, 21] and is as an example implemented in the CORAL software.[23] The approach in the CORAL software is however very different from the one in this study. CORAL software breaks down the SMILES into single atoms, double atoms and triple atoms (Sk, SSk and SSSk) as well as some extra manually coded extracted features such as BOND, NOSP, HALO and PAIR.[23, 21] The approach seems close to using a mixture of topological torsions[24] with one, two and three atoms and atom-pair[25] fingerprints. The LSTM-QSAR used in this approach directly uses the SMILES string and supposedly let the model best extract the features from the SMILES strings that best fit with the task at hand, and similar approaches have been shown to outperform other common machine learning algorithms[22], although the details of optimization of the competing algorithms were not completely clear.

SMILES were also used recently in an application of a neural network based auto-encoder.[9] Here the SMILES are used as input to a neural network with the task of recreating the input sequence. The information is passed through a "bottle-neck" layer in between the encoder and the decoder, which limits the direct transfer of information. The bottle neck layer thus ends up as a more continuous floating point vector representation of the molecule, which can be used to explore the chemical space near an input molecule, interpolate between molecules and link the vector representation to physico-chemical properties. The amount of unlabeled molecules for the study already surpassed the needed amount, but could in principle be expanded even more with the SMILES enumera-

tion technique described here. SMILES enumeration could possibly allow the autoencoder to be trained with smaller and more focused datasets of biological interest. Additionally, it would be interesting to see if different SMILES of the same molecule ends up with the same vector representation or in entirely different areas in the continuous molecular representations.

LSTM networks have also been used in QSAR applications demonstrating learning transfer from large datasets to smaller.[26] Here the input was however not SMILES strings but rather molecular graph convolution layers[27] working directly on the molecular graph representation. The approach thus more directly reads in the topology of the molecular model, rather than indirectly letting the network infer the topology from the SMILES branching and ring closures defined by the brackets and numbering in the SMILES strings.

Conclusion

This short investigation has shown promise in using SMILES enumeration as a data augmentation technique for neural network QSAR models based on SMILES data. SMILES enumeration enables the use of more limited sizes of labeled data sets for use in modeling by more complex neural network models. SMILES enumeration gives more robust QSAR models both when predicting single SMILES, but even more when taking the average prediction using enumerated SMILES for the same molecule. The SMILES enumeration code as well as some of the scripts used for generating the LSTM-QSAR models are available on GitHub: <https://github.com/Ebjerrum/SMILES-enumeration>

Conflicts of Interest

E. J. Bjerrum is the owner of Wildcard Pharmaceutical Consulting. The company is usually contracted by biotechnology/pharmaceutical companies to provide third party services

References

- [1] P. Y. Simard, D. Steinkraus, J. C. Platt, et al., Best practices for convolutional neural networks applied to visual document analysis., in: ICDAR, Vol. 3, Citeseer, 2003, pp. 958–962.
- [2] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-r. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, et al., Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups, *IEEE Signal Processing Magazine* 29 (6) (2012) 82–97.
- [3] A. Mayr, G. Klambauer, T. Unterthiner, S. Hochreiter, Deeptox: toxicity prediction using deep learning, *Frontiers in Environmental Science* 3 (2016) 80.
- [4] A.-D. Almási, S. Woźniak, V. Cristea, Y. Leblebici, T. Engbersen, Review of advances in neural networks: Neural design technology stack, *Neurocomputing* 174 (2016) 31–41.
- [5] K. Chatfield, K. Simonyan, A. Vedaldi, A. Zisserman, Return of the devil in the details: Delving deep into convolutional nets, *arXiv preprint arXiv:1405.3531*.
- [6] X. Cui, V. Goel, B. Kingsbury, Data augmentation for deep neural network acoustic modeling, *IEEE/ACM Trans. Audio, Speech and Lang. Proc.* 23 (9) (2015) 1469–1477. doi:10.1109/TASLP.2015.2438544. URL <http://dx.doi.org/10.1109/TASLP.2015.2438544>
- [7] J. Schmidhuber, Deep learning in neural networks: An overview, *Neural networks* 61 (2015) 85–117.
- [8] S. Kearnes, K. McCloskey, M. Berndl, V. Pande, P. Riley, Molecular graph convolutions: moving beyond fingerprints, *Journal of computer-aided molecular design* 30 (8) (2016) 595–608.
- [9] R. Gómez-Bombarelli, D. Duvenaud, J. M. Hernández-Lobato, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, A. Aspuru-Guzik, Automatic chemical design using a data-driven continuous representation of molecules, *arXiv preprint arXiv:1610.02415*.
- [10] D. Weininger, Smiles, a chemical language and information system. 1. introduction to methodology and encoding rules, in: *Proc. Edinburgh Math. SOC*, Vol. 17, 1970, pp. 1–14.
- [11] N. M. O’Boyle, Towards a universal smiles representation - a standard method to generate canonical smiles based on the inchi., *Journal of cheminformatics* 4 (2012) 22. doi:10.1186/1758-2946-4-22.
- [12] D. Tang, B. Qin, X. Feng, T. Liu, Target-dependent sentiment classification with long short term memory, *CoRR*, abs/1512.01100.
- [13] G. A. Landrum, Rdkit: Open-source cheminformatics software (2016). URL <http://www.rdkit.org/>, <https://github.com/rdkit/rdkit>
- [14] Ctf file formats, <http://accelrys.com/products/-informatics/cheminformatics/ctfile-formats/no-fee.php> (Dec 2011).

- URL <http://accelrys.com/products/informatics/cheminformatics/ctfile-formats/no-fee.php>
- [15] J. J. Sutherland, L. A. O'Brien, D. F. Weaver, Spline-fitting with a genetic algorithm: a method for developing classification structure-activity relationships., *Journal of chemical information and computer sciences* 43 (2003) 1906–1915. doi: 10.1021/ci034143r.
- [16] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, E. Duchesnay, Scikit-learn: Machine learning in Python, *Journal of Machine Learning Research* 12 (2011) 2825–2830.
- [17] F. Chollet, keras, <https://github.com/fchollet/keras> (2015).
- [18] R. Al-Rfou, G. Alain, A. Almahairi, C. Angermueller, D. Bahdanau, N. Ballas, F. Bastien, J. Bayer, A. Belikov, A. Belopolsky, Y. Bengio, A. Bergeron, J. Bergstra, V. Bisson, J. Blecher Snyder, N. Bouchard, N. Boulanger-Lewandowski, X. Bouthillier, A. de Brébisson, O. Breuleux, P.-L. Carrier, K. Cho, J. Chorowski, P. Christiano, T. Cooijmans, M.-A. Côté, M. Côté, A. Courville, Y. N. Dauphin, O. Delalleau, J. Demouth, G. Desjardins, S. Dieleman, L. Dinh, M. Ducoffe, V. Dumoulin, S. Ebrahimi Kahou, D. Erhan, Z. Fan, O. Firat, M. Germain, X. Glorot, I. Goodfellow, M. Graham, C. Gulcehre, P. Hamel, I. Harlouchet, J.-P. Heng, B. Hidas, S. Honari, A. Jain, S. Jean, K. Jia, M. Korobov, V. Kulkarni, A. Lamb, P. Lamblin, E. Larsen, C. Laurent, S. Lee, S. Lefrançois, S. Lemieux, N. Léonard, Z. Lin, J. A. Livezey, C. Lorenz, J. Lowin, Q. Ma, P.-A. Manzagol, O. Mastropietro, R. T. McGibbon, R. Memisevic, B. van Merriënboer, V. Michalski, M. Mirza, A. Orlandi, C. Pal, R. Pascanu, M. Pezeshki, C. Raffel, D. Renshaw, M. Rocklin, A. Romero, M. Roth, P. Sadowski, J. Salvatier, F. Savard, J. Schlüter, J. Schulman, G. Schwartz, I. V. Serban, D. Serdyuk, S. Shabianian, E. Simon, S. Spieckermann, S. R. Subramanyam, J. Sygnowski, J. Tanguay, G. van Tulder, J. Turian, S. Urban, P. Vincent, F. Visin, H. de Vries, D. Warde-Farley, D. J. Webb, M. Willson, K. Xu, L. Xue, L. Yao, S. Zhang, Y. Zhang, Theano: A Python framework for fast computation of mathematical expressions, *arXiv e-prints* abs/1605.02688. URL <http://arxiv.org/abs/1605.02688>
- [19] T. G. authors, Gpyopt: A bayesian optimization framework in python, <http://github.com/SheffieldML/GPyOpt> (2016).
- [20] C. Wang, R. M. Neal, Mcmc methods for gaussian process models using fast approximations for the likelihood, *arXiv preprint arXiv:1305.2235*.
- [21] A. Worachartcheewan, P. Mandi, V. Prachayasittikul, A. P. Toropova, A. A. Toropov, C. Nantasenamat, Large-scale qsar study of aromatase inhibitors using smiles-based descriptors, *Chemometrics and Intelligent Laboratory Systems* 138 (2014) 120–126.
- [22] S. Jastrzebski, D. Lesniak, W. M. Czarnecki, Learning to SMILE(S), *CoRR* abs/1602.06289. URL <http://arxiv.org/abs/1602.06289>
- [23] A. P. Toropova, A. A. Toropov, Coral software: prediction of carcinogenicity of drugs by means of the monte carlo method, *European Journal of Pharmaceutical Sciences* 52 (2014) 21–25.
- [24] R. Nilakantan, N. Bauman, J. S. Dixon, R. Venkataraghavan, Topological torsion: a new molecular descriptor for sar applications. comparison with other descriptors, *Journal of Chemical Information and Computer Sciences* 27 (2) (1987) 82–85.
- [25] R. E. Carhart, D. H. Smith, R. Venkataraghavan, Atom pairs as molecular features in structure-activity studies: definition and applications, *Journal of Chemical Information and Computer Sciences* 25 (2) (1985) 64–73.
- [26] H. Altae-Tran, B. Ramsundar, A. S. Pappu, V. Pande, Low data drug discovery with one-shot learning, *arXiv preprint arXiv:1611.03199*.
- [27] D. K. Duvenaud, D. Maclaurin, J. Iparraguirre, R. Bombarell, T. Hirzel, A. Aspuru-Guzik, R. P. Adams, Convolutional networks on graphs for learning molecular fingerprints, in: *Advances in neural information processing systems*, 2015, pp. 2224–2232.