
THERE AND BACK AGAIN: A RETROSYNTHESIS TALE

A PREPRINT

Pavel Karpov*

Institute of Structural Biology
Helmholtz Zentrum München
Germany, Munich
carpovpv@gmail.com

Igor V. Tetko†

Institute of Structural Biology
Helmholtz Zentrum München
Germany, Munich
itetko@vccclab.org

Guillaume Godin

Firmenich International SA,
Research&Development Division,
Switzerland, Geneva
guillaume.godin@firmenich.com

April 1, 2019

ABSTRACT

We describe a Transformer model for a retrosynthetic reaction prediction task. The model is trained end-to-end on 45033 experimental reaction examples extracted from USA patents. Our final model can successfully predict the reactants set for 44.2% of cases on the external test set. During the training procedure, we applied different learning rate schedules and knowledge distillation. These techniques can prevent overfitting and thus can be a reason to get rid of internal validation dataset that is advantageous for deep models with millions of parameters. We investigate the influence of the temperature while decoding the model output probabilities and find that at $T=1.3$ the final model behaves like a consensus of models and improve accuracy up to 6%. We also discuss the synergy of applying together the retrosynthetic model to predict reagents and a direct reaction model trained on the same data which makes a prognosis for the product base on reactants.

Keywords Reaction outcomes prediction · Computer retrosynthesis · Character-based models · Transformer

1 Introduction

New chemical compounds drive technological advances in material, agricultural, environmental, and medical sciences, thus, embracing all fields of scientific activities which have been bringing social and economic benefits throughout human history. Design of chemicals with predefined properties is an arena of QSAR/QSPR (Quantitative Structure Activity/Property Relationships) approaches aimed at finding correlations between molecular structures and their desired outcomes and then applying these models to optimise activity/property of compounds.

The advent of deep learning[1] gave a new impulse for virtual modeling and also opened a venue for a promising set of generative methods based on Recurrent Neural Networks[2], Variational Autoencoders[3], and Generative Adversarial Networks trained with reinforcement learning[4, 5]. These techniques are changing the course of QSAR studies from the observation to the invention: from a virtual screening of available compounds to direct synthesis of new candidates. Generative models can produce big sets of promising molecules and impaired with SMILES-based QSAR methods[6] provide a strong foundation for creating highly optimized focussed libraries, but estimation of synthetic availability of these compounds is an open question though several scores based on fragmentation[7] and machine learning[8] approaches have been developed. To synthesize a molecule, one should have a plan of a multi-step synthesis and also a set of available reactants. Finding an optimal combination of reactants, reactions, and conditions to obtain the compound with good yield, sufficient quality, and quantity is not a trivial task even for experts in organic chemistry. Recent advances in the computer-aided synthesis planning are reviewed in [9, 10].

The retrosynthetic analysis worked out by E. J. Corey [11] tries to account for all factors while deriving the synthetic route. It iteratively decomposes the molecule on simpler blocks till all of them become available either by purchase or

*Helmholtz Zentrum München – Research Center for Environmental Health (GmbH), Institute of Structural Biology, Ingolstädter Landstraße 1, D-85764 Neuherberg, Germany, <https://www.helmholtz-muenchen.de/stb/index.html>

†BigChem GmbH, <https://ochem.eu>

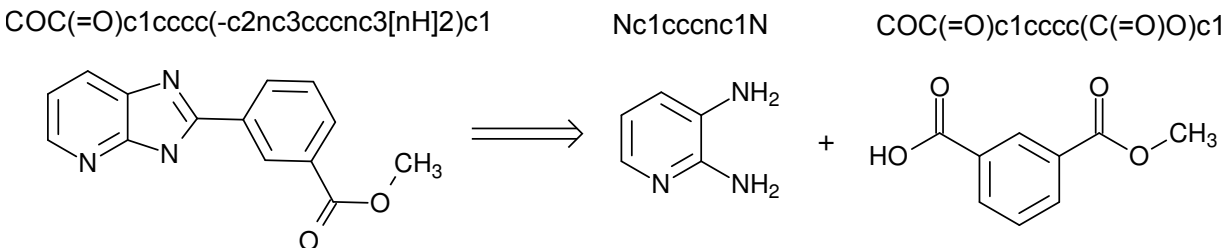


Figure 1: An example of a retrosynthetic reaction: on the left side of the arrow the target molecule is depicted, and on the right side the one possible set of reactants that can lead to the target is shown in common chemistry-like scheme and using SMILES notation. Here two successive amidation reactions result in cyclisation and aromatization.

by synthesis described in the literature. At each step, fig. 2, all possible disconnections (rules) with known reactions simplify the target molecule bringing to the scene less complex compounds. Some of them may be already available, while the others undergo the next step of retrosynthesis decomposition. Due to the recursive nature of the procedure, it can deal with thousands of putative compounds so computational retrosynthetic approaches can greatly help chemists in finding the best routes. Managing of the database of such rules is complicated and more critical the models based on it are not ready to accommodate new reactions and will always be outdated. Almost more than 60 years of developing rule-based systems ended with no remarkable success in synthesis planning programs. Another approach to tackle the problem is to use so-called template-free methods inspired by the success of machine-translation. They don't require the database of templates and rules due to an inherent possibility to derive this information during training directly from a database of organic reactions with clearly designated roles of reactants, products, reagents, and conditions.

The analogy between machine translation and retrosynthesis is evident: each target molecule has its predecessors from which it can be synthesized as every meaningful sentence one can translate from source language to target one. If all parts of a reaction are written in SMILES notation, then our source and target sentence are composed of valid SMILES tokens as words. The main goal of the work is to build a model which could for a given target molecule for example³ COC(=O)c1cccc(-c2nc3ccnc3[nH]2)c1 in fig. 2 correctly predict the set of reactants. Namely, it should predict Nc1ccnc1N.COC(=O)c1cccc(C(=O)O)c1 in this case.

Neural sequence-to-sequence (seq2seq) approach has been recently applied for a direct reaction prediction task [12, 13] with outstanding statistical parameters of final models – 90.4% of accuracy on test set. Seq2seq modeling has been also tested on retrosynthesis task[14], but due to the complex nature of retrosynthesis itself and difficulty in estimating the correct predictions of reactants⁴, accuracy on the test set was moderate 37.4% but still comparable to rule-based systems 35.4%. We questioned about the possibility of improvement models for one-step retrosynthesis utilizing modern neural network architectures and training techniques. Applying the Transformer Model[15], together with cyclical learning rate schedule[16], and knowledge distillation[17] resulted in a single model with accuracy 44.2%, that is more than 6% higher compare to the baseline model[14].

Our main contributions are:

- We show that for this particular task there is no need to use a validation dataset for early-stopping or other parameters optimization. We trained all the parameters directly from the training dataset.
- Applying weights averaging, snapshot learning and knowledge distillation helped us to train the most precise model for one-step retrosynthesis prediction.
- We show that different chemistry-relevant SMILES tokenization approaches cannot improve the accuracy of the models compared to the simplest letter based schema.
- Increasing the temperature while performing a beam-search procedure improves the accuracy up to 4%. Almost the same effect as consensus does.
- Using both retrosynthetic model for reactant prediction and a direct model trained on the same data for reaction outcome prognosis allows improving precision of the retrosynthesis model.

³ This reaction is in the test set and it was correctly predicted by our model. Probability landscapes in the following figures are drawn from this example.

⁴ A target molecule usually can be synthesized with different reactions starting from different sets of reactants. The predictions of the model may be correct from organic chemist point of view but differ from the reactant set in ground truth. This may lead to underestimation of effectiveness of models.

2.1 Dataset

2.2 Model input

layers is then mixed with original data, layer-wise normalized, and passed position-wise through a couple of ordinary dense layers to go further either in next level of self-attention layers or to a decoder as an information-rich vector representing the input. The decoder part of Transformer resembles the encoder but has an additional self-attention layer which corresponds to encoder’s output.

Transformer model shows the state-of-the-art results in machine translation and reaction prediction outcomes [13]. The latter work showed that training the Transformer on large and noisy datasets results in a model that can outperform not only other machine models but also well qualified and experienced organic chemists.

2.4 Model inference

The model tries to estimate the probability of the next symbol over the model’s vocabulary given all previous symbols in the string. Technically, the Transformer model first calculates logits, z_i , and then transforms them to probabilities.

$$z_i = \text{Transformer}(\{x_1, x_2, x_3, \dots, x_L\}, \{y_1, y_2, y_3, \dots, y_{i-1}\}) \quad (1)$$

Here x_i is the input of the models at i position; L – the length of the input string; y_i is the decoded output of the model up to position $(i - 1)$; and z_i – logits that are to be converted to probabilities:

$$q_i = \frac{\exp(z_i/T)}{\sum_{j=0}^V \exp(z_j/T)} \quad (2)$$

where V is the size of the vocabulary (66 in this work) and T stands for the temperature⁸ usually assigned to 1.0 in standard softmax layers. With higher T the landscape of the probability distribution becomes more smooth. During the training the model adapts its weights to better predict q_i , so $y_i = q_i$.

During the inference however we have several possibilities how to convert q_i into y_i , namely greedy and beam search. The first one picks up a symbol with maximum probability whereas the second one at each step holds $top - K$ (K = beam’s size) suggestions of the model and summarises the overall likelihood for each of K final decodings. The beam search allows better inference and the probability landscape exploration compared to the greedy search because at a particular step of decoding it may choose a symbol with less than maximum probability, but the total likelihood of the result can be higher due to more significant probabilities on the next steps.

2.5 Training heuristics

Training a Transformer model is a challenge, and several heuristics have been proposed[16]. The most useful we used in this work are:

Using as bigger batch size as possible. However, due to our hardware limitations we could not set the batch size greater than 64⁹;

Increasing the learning rate at the beginning of training up to warmup steps¹⁰. The authors of the original Transformer paper [15] used 4 000 steps for warming. The Transformer model for reaction prediction task from [13] used 8 000 steps. We have also tried different values for warmup and eventually found that 16 000 works well with our model.

Applying cyclic learning rate schedules. This tips can generally improve any model[20] through better loss landscape exploration with bigger learning rates after the optimiser felt down to some local minima. For this study we used the following scheme for learning rate calculation depending on the step:

$$u(\text{step}) = \begin{cases} \text{warmup} + (\text{step} \bmod \text{cycle}), & \text{if } \text{step} \geq \text{cycle} \\ \text{step}, & \text{otherwise} \end{cases}$$

where cycle stands for the number of steps while the learning rate is decreasing before raising to the maximum again.

$$\lambda(\text{step}) = \text{factor} * \frac{\min(1.0, u(\text{step})/\text{warmup})}{\max(u(\text{step}), \text{warmup})} \quad (3)$$

where factor is just a constant. Big values of factor introduce numerical instability during training, so after several trials we set $\text{factor} = 20.0$. The curve for learning rate in this study is shown in fig. 2, plot (4, f).

⁸Similar to formula of Boltzmann (Gibbs) distribution used in statistical mechanics.

⁹Our first implementation of the model required a lot of memory to deal with masks of reactants and products. Though later we improved the code we still remained this size for consistency of the results.

¹⁰In our implementation 1 step is equivalent to 1 batch. The number of reactions for training is 40 029 + 5 004, so one epoch is equal to 704 batches.

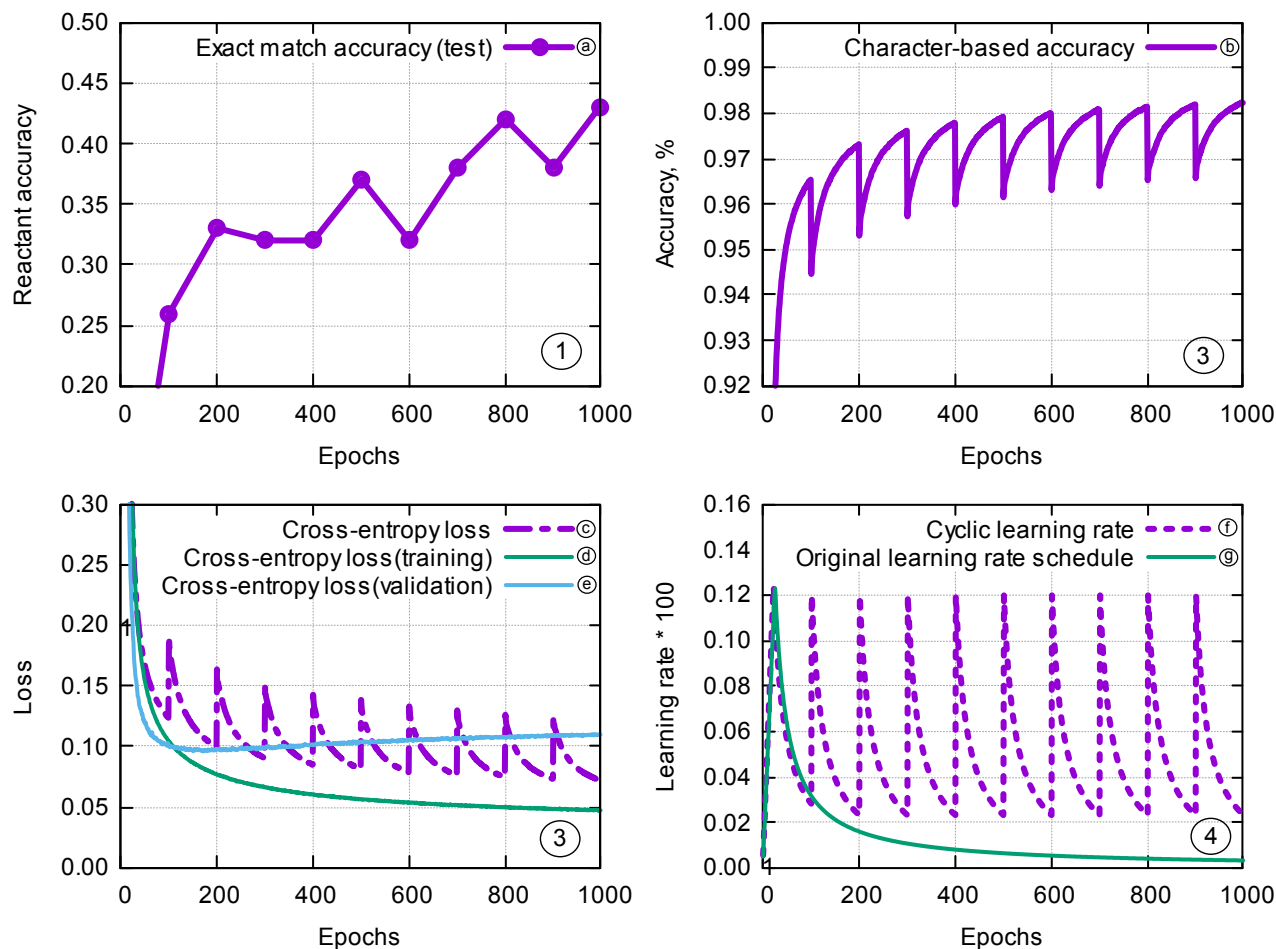


Figure 2: Summary of learning curves for the Transformer model: 1) Accuracy of correctly predicted reactants in test set (a); 2) character based accuracy (b); 3) cross-entropy loss with cycling learning rate (f) for the entire dataset (c) and losses for training (d) and validation (e) sets with monotonically decreasing learning rate (g); 4) learning rate schedules as a function of epoch.

Averaging weights during last steps of training or at minima of learning rates.

2.6 Knowledge distillation

After training a (teacher) model it is possible to get predictions for the training set and then train another (student) model on these predictions. This second model usually outperforms the first one. This approach was further developed in [17] where the algorithm for knowledge distillation was proposed. The key idea is to calculate probabilities for training dataset at higher temperature T , eq. 2, and then use these values as targets along with the correct labels. The final loss function:

$$Loss = - \sum y \log(q) - T^2 * \sum q_{teacher} \log(q) \quad (4)$$

contains terms for hard and soft labels and, thus, the model will learn to distinguish subtle differences in final probability distribution. We applied this technique for further training our models and found that this simple approach can improve the accuracy up to 2%.

Table 1: Accuracy (%) of the models on test set when all reactants were correctly predicted.

Model	top-1	top-3	top-5	Description
Seq2Seq	37.4	52.4	57.0	Literature result from [14] based on Seq2Seq architecture.
T1	38.27	58.23	64.96	Transformer Model trained with validation control set (early stopping, 200 epochs).
T1 ₁	40.16	58.95	65.54	The same as T1, but without early stopping (1000 epochs).
T2				Transformer Model trained on both training and validation sets for 1000 epochs.
T2 ₁	40.36			T = 1.3 during the decoding stage of the T2 model.
T3				Transformer Model trained with cyclic learning rate schedule for 6000 epochs.
T3 ₁	42.28			T = 1.3 during the decoding stage of T3 model
T4	42.25	61.31	68.11	Transformer Model trained with distillation approach [17] for 1000 epochs.
T4 ₁	42.91	62.11	68.92	The same as T4 but decoding at T=1.3
T4 ₁	44.42	63.66	70.38	The same as T4 but decoding at T=1.3, rdkit version 2017.09.3

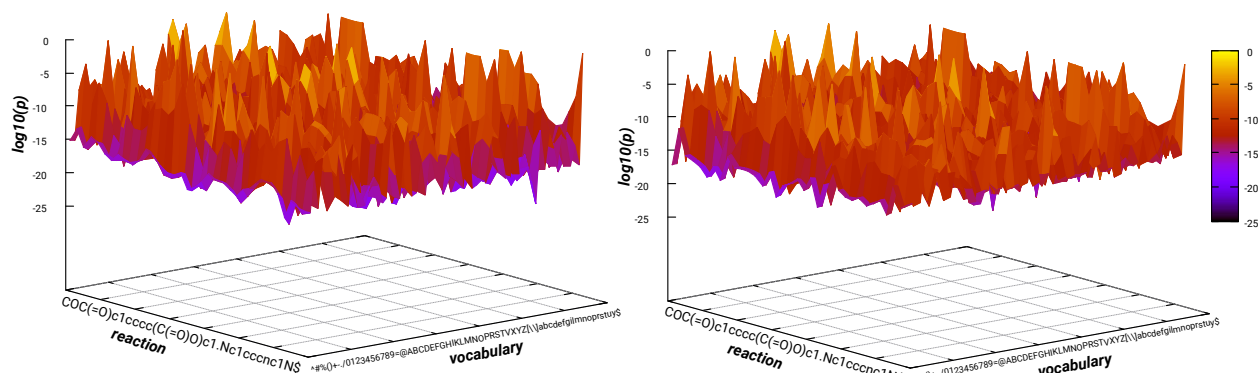


Figure 3: Landscapes of probabilities for predicting example reaction, fig. 2, with T3 model(left) and T4 (distilled) model (right).

3 Results

4 Discussion

All source code and also models built are available online via github

<https://github.com/bigchem/retrosynthesis>

Acknowledgments

This study has been partially supported by ERA-CVD (<http://era-cvd.eu>) "Cardio-Oncology" project, BMBF 01KL1710. The authors thank NVIDIA Corporation for donating Quadro P6000 and Titan Xp graphics cards for this research.

References

- [1] Chen, H., Engkvist, O., Wang, Y., Olivecrona, M. & Blaschke, T. The rise of deep learning in drug discovery. *Drug Discovery Today* **23**, 1241 – 1250 (2018).
- [2] Ertl, P., Lewis, R., Martin, E. & Polyakov, V. In silico generation of novel, drug-like chemical matter using the lstm neural network. *arXiv* (2017). 1712.07449v2.

- [3] Gómez-Bombarelli, R. *et al.* Automatic chemical design using a data-driven continuous representation of molecules. *ACS Central Science* **4**, 268–276 (2018).
- [4] Guimaraes, G. L., Sanchez-Lengeling, B., Outeiral, C., Farias, P. L. C. & Aspuru-Guzik, A. Objective-reinforced generative adversarial networks (organ) for sequence generation models. *arXiv* (2017). 1705.10843v3.
- [5] Olivecrona, M., Blaschke, T. & hongming Chen, O. E. Molecular de-novo design through deep reinforcement learning. *J. Cheminform.* **9**, 1758–2946 (2017).
- [6] Kimber, T. B., Engelke, S., Tetko, I. V., Bruno, E. & Godin, G. Synergy effect between convolutional neural networks and the multiplicity of smiles for improvement of molecular prediction. *arXiv* (2018). 1812.04439v1.
- [7] Ertl, P. & Schuffenhauer, A. Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions. *J. Cheminform.* **1**, 8 (2009).
- [8] Coley, C. W., Rogers, L., Green, W. H. & Jensen, K. F. Scscore: Synthetic complexity learned from a reaction corpus. *Journal of Chemical Information and Modeling* **58**, 252–261 (2018). URL <https://doi.org/10.1021/acs.jcim.7b00622>. PMID: 29309147.
- [9] Coley, C. W., Green, W. H. & Jensen, K. F. Machine learning in computer-aided synthesis planning. *Accounts of Chemical Research* **51**, 1281–1289 (2018). PMID: 29715002.
- [10] Engkvist, O. *et al.* Computational prediction of chemical reactions: current status and outlook. *Drug Discovery Today* **23**, 1203 – 1218 (2018).
- [11] Corey, E. J. & Cheng, X.-M. *The Logic of Chemical Synthesis* (Wiley-Interscience, 1995).
- [12] Schwaller, P., Gaudin, T., Lanyi, D., Bekas, C. & Laino, T. Found in translation: Predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models. *ArXiv* (2018). 1711.04810v2.
- [13] Schwaller, P. *et al.* Molecular transformer for chemical reaction prediction and uncertainty estimation. *arXiv* (2018). 1811.02633v1.
- [14] Liu, B. *et al.* Retrosynthetic reaction prediction using neural sequence-to-sequence models. *ACS Central Science* **3**, 1103–1113 (2017).
- [15] Vaswani, A. *et al.* Attention is all you need. *ArXiv* (2017). 1706.03762.
- [16] Popel, M. & Bojar, O. Training tips for the transformer model. *arXiv* (2018). 1804.00247v2.
- [17] Hinton, G., Vinyals, O. & Dean, J. Distilling the knowledge in a neural network. *arXiv* (2015). 1503.02531v1.
- [18] Lowe, D. M. *Extraction of chemical structures and reactions from the literature*. Ph.D. thesis (2012). URL <https://www.repository.cam.ac.uk/handle/1810/244727>.
- [19] Schneider, N., Stiefl, N. & Landrum, G. A. What’s what: The (nearly) definitive guide to reaction role assignment. *Journal of Chemical Information and Modeling* **56**, 2336–2346 (2016). PMID: 28024398, <https://doi.org/10.1021/acs.jcim.6b00564>.
- [20] Izmailov, P., Podoprikin, D., Garipov, T., Vetrov, D. & Wilson, A. G. Averaging weights leads to wider optima and better generalization 1803.05407v3.