

Hard-threshold-Neural-Network based Prediction of Organic Synthetic Outcomes

Haoyang Hu

Tsinghua University

Zhihong Yuan (✉ zhihongyuan@mail.tsinghua.edu.cn)

Tsinghua University

Research article

Keywords: Medicine development; Retrosynthetic analysis; Outcome prediction; Hard-threshold neural network; Combinatorial optimization

Posted Date: January 2nd, 2020

DOI: <https://doi.org/10.21203/rs.2.16734/v2>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Abstract

Retrosynthetic analysis is the canonical technique to plan the synthesis route of organic molecules in drug discovery and development. In this technique, the screening of synthetic tree branches requires accurate forward reaction prediction, but existing software is still far from completing this step independently. Previous studies have attempted to apply neural network in the forward reaction prediction, but the accuracy is not satisfying. Through using the Edit Vector based description and Extended-Connectivity Fingerprints to transform reaction into vector, the presented work focuses on the update of neural network to improve the template-based forward reaction prediction. Hard-threshold activation and target propagation algorithm are implemented by introducing the mixed-convex combinatorial optimization. Comparative tests are conducted to explore the optimal hyperparameter set. Using 15,000 experimental reaction extracted from granted United States patents, the proposed hard-threshold neural network is systematically trained and tested. The results demonstrate that a higher prediction accuracy is obtained when compared to the traditional neural network with backpropagation algorithm. Some successfully predicted reaction examples are also briefly illustrated.

Background

Drug discovery and development is one of the most important components of the pharmaceutical industry. In order to meet the customers' increasing demands while guaranteeing the good potency and minimal side effects, structures of new drug molecules have been becoming more and more complicated. In other words, extensive pharma R&D activities are dramatically required, and relative short discovery-development-deployment cycle is desirable^[1].

Over the past decades, as a rate-limiting factor, innovations in organic synthesis have significantly enabled the discovery and development of important life-changing medicines, improving the health of patients worldwide^[2,3]. Nevertheless, innovations and excellence in organic synthesis are expected to be the most powerful driver for all phases of drug discovery and development. Very recently, chemists have enthusiastically applied advanced machine learning and artificial intelligence (AI) technologies towards the preparation of drug molecules. For example, AI driven discovery of drug molecules^[4-6], automated planning of synthetic routes^[7-9], machine learning driven optimization of reaction conditions^[10-12], and autonomous assembly of synthetic processes^[13-15].

Synthesis planning, which is regarded as one of the central elements of organic synthesis, can be traced back to 1960s^[16]. Traditional computer-aided approaches for synthesis planning are labeled with different disadvantages such as low efficiency, poor repeatability and high experimental cost. Especially, more and more compounds and reactions have been discovered. For example, the number of reactions and compounds that Reaxys database contained has exceeded 40 million and 100 million, respectively. Novel synthesis planning approaches are highly required. During the past 3 years, a variety of machine learning and artificial intelligence methods such as random forest, automated reasoning, support vector machines and more recently deep learning have demonstrated their capacity for organic molecules

discovery, design, and production^[13, 17-19]. Clearly, the application of the aforementioned new technologies to end-to-end organic molecules discovery and development will be the key point for realizing the fully automated synthesis planning^[20].

Retrosynthetic analysis is the canonical technique used to plan the synthesis of small organic molecules for drug discovery^[21]. In General, retrosynthesis analysis consists of four steps: step 1. determine the target compound; step 2. disconnect certain bonds which are thought to be easy to form according to known chemical knowledge in the target compound as the reverse of reaction, and search for possible precursors in this way; step 3. repeat step 2 for all the precursors to form a synthesis tree, and expand it until all precursors are available; step 4. evaluate all the branches of the synthetic tree one by one, then take the most possible one as the optimal route. Since different groups of reaction sites in the same group of precursors may exist, in other words, the difference between real reactions and the expected ones in the branches may occur, step 4 is therefore very critical but also the most difficult one. Forward reaction prediction is a necessity point to guarantee the correct evaluation of each branch. The main aim of the paper is to present the machine learning approaches for assisting the forward reaction prediction.

The existing approaches for forward reaction prediction can be categorized as template-based and template-free method. For example, Coley et al.^[17] applied reaction templates to reactants to generate candidate reactions as many as possible, which then are used to train a neural network. While for template-free methods, Schwaller et al.^[22] compared chemical reactions from reactants to products to translations from one language to another, so that forward reaction prediction can be transformed into machine translation and solved by training a seq-to-seq recurrent neural network which directly takes reactant SMILES (Simplified Molecular Input Line Entry Specification) as input.

As the research target is to discover new reactions meaningful for organic synthesis according to existing reaction mechanisms, to fully reuse the discovered reaction rules summarized from experimental results so far, the presented paper adopts the template-based approach. Specifically, the concentration of this paper is to adopt a novel hard-threshold based deep neural network for improve the prediction accuracy for the forward reaction prediction. In detail, hard-threshold activation and target propagation algorithm are implemented by introducing the mixed-convex combinatorial optimization. Comparative tests are conducted to explore the optimal hyperparameter set. The reminder of the paper is as follows: forward reaction prediction will be described firstly, then hard-threshold neural network is touched upon. Results and discussions are then given. Conclusion is finally presented.

Forward reaction prediction

The above reaction can be decomposed into three Edits: atom 1(nitrogen) loses a hydrogen, atom 2(carbon) and 3(chlorine) loses a single bond, and atom 1 and 3 respectively gains a single bond. In the reactant, atom 1 has one hydrogen and two non-hydrogen neighbors, atom 2 has no hydrogen but is surrounded by three non-hydrogen neighbors, while atom 3 has one non-hydrogen neighbor without any

hydrogen. Therefore, feature vectors of atom 1, atom 2, atom 3, along with two bonds can be expressed as: (see formula set 1 in the Supplementary files)

The Edit Vectors of hydrogen loss and gain (e_1 , and e_2 , respectively) are directly taken as the feature vector of the corresponding atom, and those of the bond loss and gain (e_3 , and e_4 , respectively) are taken as the connection of the feature vectors of corresponding atoms and the bonds: (see formula set 2 in the Supplementary Files)

Where "/" represents the connection of feature vectors, and the outermost brackets represent that Edit Vector is the combination of all the vectors of basic Edit in a reaction. For instance, if one reaction has four atoms losing hydrogen in the reactant, its Edit Vector of hydrogen loss (e_1) will have four 6D feature vectors.

Candidate reaction selection

The selection step uses a complex neural network consisting of several subnetworks, as shown in Fig.3. For each candidate reaction, the above mentioned four Edit Vectors are calculated and severed as input to four corresponding subnetworks. The sum of outputs from four subnetworks is then fed to the lower integrating subnetwork to produce scalar probability scores. The above steps are repeated for all candidate reactions, and then all the probability scores are normalized by the softmax method to estimate the probability of occurrence of each candidate reaction. Finally, all candidate reactions are sorted by the probability score, and the candidate ranking first is regarded as the prediction result. The prediction is right if its outcome has the same SMILES as the recorded one, and vice versa.

In order to improve the prediction accuracy, a hybrid model using the Edit Vector and the Extended-Connectivity Fingerprint (ECFP)^[23] is also considered in our work. The only difference between two models is that an extra subnetwork without hidden layer which evaluates the ECFP is added to the hybrid model, as shown in the middle of Fig.3. The output of ECFP subnetwork is multiplied by ε when subnetwork outputs are summed before input to the final integrating subnetwork, where ε is the mixing factor. By adjusting ε , the proportion of ECFP subnetwork can be precisely controlled.

Hard-threshold neural network

Neural network, especially deep neural network learning, is currently the most popular machine learning algorithm with powerful fitting capability^[24]. However, with the ceaseless expansion of the size of the neural network, a series of problems, particularly gradient vanishing and gradient explosion would often occur. To get rid of this dilemma, the paper tends to apply the hard-threshold neural network for predicting the outcomes of organic synthesis.

Constructing hard-threshold neural network

“Hard-threshold neural network” means neural network with hard-threshold activation, including Step activation and Staircase activation, which is shown in Figs 4a and 4b, respectively. Indeed, staircase activation is the sum of some step activations. Actually, hard-threshold activation was used in the early contributions to deal with the binary classification when the neural network had not been proposed. Hard-threshold activation has a constant derivative as 0, which can effectively avoid gradient vanishing and gradient explosion. Besides, the scale of output is almost fixed and insensitive to the scale of the input, which helps avoid certain abnormal propagation and simplify the computation. However, the zero derivative of hard-threshold activation also prevents it from being trained with traditional backpropagation.

Target propagation algorithm

A new backpropagation algorithm is required to train a hard-threshold neural network bypassing the zero derivative of hard-threshold activation.

Based on the “target propagation” concept^[25], a new target propagation algorithm named FTPROP-MB was recently proposed^[26]. Basically, since perceptron with Step activation is trainable, hard-threshold neural network could also be trainable if it can be decomposed into perceptron. Specifically, a target vector t_d is introduced to represent what the d^{th} layer is supposed to be the output for all hard-threshold activation layers. After the normal forward propagation procedure for each layer, FTPROP-MB determines t_d firstly, then introduces a layer loss L_d which is used to compute gradient just like training perceptron so that weights can be updated.

Considering that the output of the hard-threshold activation is a set of discrete values, the determination of t_d can be reduced to a combinatorial optimization problem. In detail, the question of how to optimize t_d regarding overall loss and layer loss can be expressed in standard form as follows: (see Equation 1 in the Supplementary Files)

where W_d and z_d represents the weight and the pre-activation output of the d^{th} layer. The search space is large and discrete because all the components of t_d is restricted to 0 and 1, so it is tough for common search algorithms to find the optimal solution in reasonable time horizon. Since layer loss is usually convex, FTPROP-MB determines the target vector t_d in the d^{th} layer with a heuristic method: compute the derivative of layer loss in the $(d+1)^{th}$ layer L_{d+1} with respect to the d^{th} layer’s output h_d , and then set t_d according to the opposite sign of this derivative. This method can be mathematically formulated as:(see Equation 2 in the Supplementary Files)

When layer loss function is convex, the negative partial derivative of L_{d+1} on h_{dj} points to the global minimal of L_{d+1} . Let’s take $h_{dj}=-1$ as an example, if $r(h_{dj})=-1$, which indicate that the partial derivative of L_{d+1} is positive, it is clear that L_{d+1} will increase by making $h_{dj}=+1$ when fixing other components of \mathbf{h}_d so there is no doubt that $t_{dj}=r(h_{dj})=-1$. On the other hand, when $r(h_{dj})=+1$, which means the partial derivative

of L_{d+1} is negative, we don't know exactly whether L_{d+1} would increase or decrease by making $h_{dj}=+1$. But the difference between h_{dj} and $r(h_{dj})$ indicates that the current value of h_{dj} is lack of confidence, so a natural choice is to lead z_{dj} to 0 by adjusting t_{dj} to make it more possible for h_{dj} to flip, w.r.t $t_{dj}=r(h_{dj})=+1$.

In summary, the training process of a n-layer hard-threshold neural network has both optimization problem on weights and convex-combinatorial optimization problem on target vectors, so a mixed convex-combinatorial optimization problem is formed. The block diagram of target propagation algorithm is shown in Fig.5.

Layer loss function

Till now we are still facing a problem of choosing layer loss function. According to related work^[27], it is acceptable to adopt soft hinge loss and weighing according to the gradient, which is shown in Fig.6.

Methods

Preparing reaction database

The reaction data sets are originally extracted from the 1976-2013 USPTO dataset compiled by Lowe^[28]. Based on the popular template sets^[29, 30], the original extracted reaction data sets are reduced to 15, 000 groups of reactants corresponding to 15, 000 real reactions. Then, approximately 5 million candidate reactions including real and fake ones are generated by aforementioned data augmentation strategy and stored with MongoDB format.

Structure of Edit Vector

Atom features used in this paper is much more complex than the simplified example illustrated above, while bond features are the same. Specific structure is shown in Tab.1.

Tab.1 Structure of Edit Vector

Object	Index	Feature
Atom	0	Crippen logP contribution
	1	Crippen MR contribution
	2	TPSA contribution
	3	Labute ASA contribution
	4	Estate index
	5	Gasteiger partial charge
	6	Gasteiger H partial charge
	7-17	atomic number (1-hot)
	18-23	number of neighbors (1-hot)
	24-28	number of hydrogens (1-hot)
	29	formal charge
	30	is in ring
	31	is aromatic
Bond	0	is single bond
	1	is aromatic bond
	2	is double bond
	3	is triple bond

Structure of ECFP for hybrid model

Molecular fingerprint is also a common method for vectorizing molecules. A fingerprint is usually a 0-1 vector with an adjustable dimension and equivalent to the hash of a molecule. The Extended Connectivity Fingerprints (ECFP) proposed by Rogers et al.^[23] in 2010 is a circular fingerprint based on the Morgan algorithm, which has become the de facto standard in the industry. In this paper, the ECFP of the reactants and products is used with a radius of 2 and a dimension of 1024 as a supplement to the Edit Vector to construct the hybrid model, where the radius and dimension are determined by convention.

Building training platform

PyTorch is one of the most popular deep learning frameworks, and its high customizability brings much convenience to implement all kinds of "non-standard" backpropagation algorithm. Therefore, this paper uses PyTorch and a NVIDIA GeForce GTX 1070 GPU to conduct all experiments.

The dataset containing 15000 groups of reactants (5 million candidate reactions) is divided by convention: the latter 20% (3000 groups with 1 million candidate reactions) is the test set; 12.5% of the former 12,000 groups is randomly taken as the validation set (1500 groups with 0.5 million candidate reactions); the rest 10,500 groups (3.5 million candidate reactions) are considered as the training set.

The initial hyperparameters for the hard-threshold neural network are given as follows: the hidden nodes structure of four subnetwork evaluating Edit Vector is [200/100/50], while that of integrating subnetwork is [50/1]; the activation is Tanh, and the optimizer is AdaDelta ($\rho=0.95$); each batch contains 20 groups of reactants, and each model is trained for 85 epochs.

Results And Discussion

Through the extensive observations on the prediction accuracy, we conclude that the model tends to be stable after 100 epochs, and the AdaM optimizer makes training more stable compared to AdaDelta. Therefore, the following results and discussions are according to the experiments performed based on AdaM optimizer.

Edit Vector based model

The core of the hard-threshold neural network is activation, so its influence on the model is examined firstly. The results are shown in Tab.2, where the "(Soft/Hard)" in the first column marks the kind of the activation.

Tab.2 Influence of hard-threshold activation on Edit Vector based model

Activation Type	Training Accuracy	Validation Accuracy	Test Accuracy
Tanh(Soft)	80.0%	71.1%	70.0%
Step(Hard)	72.7%	71.5%	69.1%
3-Staircase(Hard)	76.4%	69.3%	69.8%
5-Staircase(Hard)	78.0%	68.9%	68.8%
7-Staircase(Hard)	80.1%	70.0%	71.2%
10-Staircase(Hard)	80.0%	69.8%	70.8%

Although innocent Step activation works even worse than Tanh activation, the prediction accuracy gradually improves as the order of the hard-threshold activation increases. When the order reaches 7, Staircase activation has achieved a higher prediction accuracy than traditional soft activation. However, prediction accuracy would decrease on the contrary when the order continues to increase, especially, the test accuracy drops a lot. Thus in this paper, following experiments are performed with 7-Staircase activation.

Furthermore, influences of subnetwork structure are shown in Tab.3.

Tab.3 Influence of subnetwork structure on Edit Vector based model

Subnetwork structure	Training Accuracy	Validation Accuracy	Test Accuracy
200/100/50	80.1%	70.0%	71.2%
250/100/50	80.0%	72.0%	71.3%
250/125/50	78.8%	71.7%	70.9%
300/100/50	78.7%	69.8%	69.8%
100/100/50/50/50	76.0%	70.4%	69.9%
100/100/50/50/50	74.7%	70.1%	69.5%

It can be seen that the prediction accuracy cannot be significantly improved via deepen or to widen the structure of subnetworks, we identify that the subnetwork structure of 200/100/50 is complicated enough in this task. In other words, what limits the accuracy is overfitting rather than underfitting, and more hidden nodes will only disturb training. Therefore, the following experiments will focus on how to avoid overfitting using original subnetwork structure.

Dropout is a common and convenient strategy to avoid overfitting^[31]. The original idea of dropout is very simple: in each forward propagation step, some outputs of hidden nodes are forced to zero. Then hidden nodes are prevented from connecting incorrect partners and overfitting can thus be avoided. Very similar to regularization, the principle of dropout is also to reduce the number of non-zero parameters in the model. However, two additional advantages of dropout should be emphasized here: first, the meaning of the dropout rate is relatively intuitive, so we can directly adjust the proportion of parameters set to zero in the model via dropout rate; second, different from regularization that penalties all non-zero parameters, the parameters are set to zero by dropout in a random manner during each forward propagation in training, which improves the robustness of the model. It should be noticed that the dropout rate must be

set carefully: a dropout rate too low cannot avoid overfitting, while one too high will lead to underfitting. Experimental results of dropout rate are shown in Tab.4.

Tab.4 Influence of dropout rate on Edit Vector based model

Dropout rate	Training Accuracy	Validation Accuracy	Test Accuracy
0	80.1%	70.0%	71.2%
0.01	77.3%	69.8%	70.1%
0.02	79.9%	72.5%	72.7%
0.1	75.4%	69.7%	70.8%

As shown in Tab.4, high dropout rate (such as 0.1) damages the model significantly, while low dropout rate (such as 0.01) cannot solve the overfitting problem properly, and the prediction accuracy does not improve in both cases. A dropout rate of 0.02 can achieve a balance between the above two cases, in other words, it can moderate overfitting while not damaging the model too much. After the extensive experiments, the test accuracy can reach as high as 72.7%. Compared with the published contribution with 68.5% of the test accuracy^[17], more than 120 reactions can be correctly predicted additionally.

Moreover, the training and validation processes of our model along with the model provided by the published literature^[17], are shown in Fig.7. It is obvious that hard-threshold neural network has the potential to approach higher prediction accuracy while reducing the instability of the running processes.

As for training efficiency, experiments above show the hard-threshold neural network and the corresponding optimization algorithm do not require much extra computing resources during training. It takes about 13-14 hours with 100 epochs, which is very similar when compared with traditional neural networks.

Hybrid model

As mentioned in the section "Candidate reaction selection", the mixing factor ε determines the proportion of the ECFP subnetwork's output in the sum fed to the integrating subnetwork, which directly determines how much the model relies on ECFP. According to the results in the section "Edit Vector based model", a more complex subnetwork is meaningless for prediction, so only the influence of mixing factor and dropout rate is examined for the hybrid model, and the results are shown in Tab.5.

Tab.5 Effect of mixing factor ε and deactivation rate on hybrid model

ε	Dropout rate	Training Accuracy	Validation Accuracy	Test Accuracy
0	0	80.1%	70.0%	71.2%
1.0000	0	99.9%	63.1%	61.6%
0.1000	0	99.7%	65.1%	66.9%
0.0200	0	98.8%	71.0%	70.8%
0.0010	0	85.3%	71.9%	72.5%
0.0008	0	85.5%	75.3%	72.6%
0.0005	0	83.6%	70.1%	72.3%
0.0010	0	85.3%	71.9%	72.5%
0.0010	0.01	85.7%	73.8%	73.0%
0.0010	0.02	85.4%	73.7%	73.9%
0.0010	0.05	83.4%	73.1%	73.1%
0.0010	0.1	82.6%	70.9%	72.7%
0.0010	0.2	77.9%	68.7%	70.3%

As shown in Tab.5, large mixing factor ε (such as 1) would cause a very serious overfitting. However, when mixing factor is gradually reduced to around 0.001, overfitting in the model almost disappears. Meanwhile, the prediction accuracy is even better than the Edit Vector based model ($\varepsilon=0$), which means that the extra information introduced by the ECFP does bring positive effects on the prediction.

Prediction examples

Here we illustrate a couple of very important reactions that our proposed model can successfully predict but the model in the literature failed.

The reaction in Fig.8 is taken from the synthesis route of certain substituted 3,4-diarylpyrazole compounds, which modulate the activity of protein kinases^[32]. These compounds are very useful in therapy and in the treatment of diseases associated with a dysregulated protein kinase activity, like cancer. For this reaction, the substitution should occur on the pyrazole ring due to the strong electron withdrawing effect of the nitro group. Our propose model assigns a probability of 33.1% to the true product. On the other hand, the model of the published literature assigned a probability of 1.7% to the true product, and a probability of 31.6% to the wrong product.

The reaction in Fig.9 is extracted from the synthesis route of a novel P2X3 receptor antagonists that play a critical role in treating disease states associated with pain, in particular peripheral pain, inflammatory pain, or tissue injury pain^[33]. For this reaction, since the hydrochloric acid-pyridine condition is weakly acidic, the imine hydroxyl group on the product should not dehydrate to form a cyano group. Our propose model assigns a probability of 70.1% to the true product. On the other hand, the model of the published literature assigned a probability of 47.1% to the true product, and a probability of 48.5% to the wrong product.

Conclusion

In this paper, we implemented the vectorized description of reaction by using the Edit Vector and ECFP, and applied the hard-threshold neural network with target propagation algorithm to the template-based forward reaction prediction thereby. For the pure Edit Vector based model, the prediction accuracy

reached as high as 72.7% which is higher than the published accuracy with 68.5%. We also found that the prediction accuracy could benefit from the utilization of ECFP with the proper mixer factor. Although the implemented hard-threshold neural network, whose hyperparameters were adjusted via heuristic approach, only improved the prediction accuracy by 4.2%, it did provide a new alternative approach for computer-aided template-based forward reaction prediction of organic synthesis for drug discovery purpose. An automatic approach for adjusting the hyperparameters for improving the prediction accuracy is under investigation. Furthermore, novel approaches for describing the reaction for prediction purpose are also under consideration.

Abbreviations

SMILES: Simplified Molecular Input Line Entry Specification

SMARTS: SMiles ARbitrary Target Specification

ECFP: Extended-Connectivity FingerPrint

Declarations

Availability of data and material

Reaction database can be downloaded from:

https://figshare.com/articles/MongoDB_dump_compressed_/4833482

An implementation of FTPROP-MB can be found in:

<https://github.com/afriesen/ftprop>

Funding

The authors gratefully acknowledge financial support from the National Scientific Foundations of China (NSFC, Grant No. 21706143) and the State Key Laboratory of Chemical Engineering (Grant No. SKL-ChE-18T01)

Authors' contributions

Dr. ZY conceived and guided the project, and contributed to writing the manuscript. Mr. HH built the model, performed all experiments, and wrote the manuscript. All authors have read and approved the final manuscript.

Acknowledgements

We would like to acknowledge the contributions of ZS for introducing PyTorch and YC for providing organic chemistry support.

Competing Interests

The authors declare no competing financial interest.

References

1. Scannell J W, et al. Diagnosing the decline in pharmaceutical R&D efficiency[J]. *Nature Reviews Drug Discovery*. 2012, 11: 191-200.
2. Campos K R, et al. The importance of synthetic chemistry in the pharmaceutical industry[J]. *Science*. 2019, 363: eaat0805.
3. Blakemore D C, et al. Organic synthesis provides opportunities to transform drug discovery[J]. *Science*. 2018, 10: 383-394.
4. Schneider G. Automating drug discovery[J]. *Nature Reviews Drug Discovery*. 2018, 17: 97-113.
5. Button A, et al. Automated de novo molecular design by hybrid machine intelligence and rule-driven chemical synthesis[J]. *Nature machine intelligence*. 2019, 1: 307-315.
6. Elton D C, et al. Deep learning for molecular design-a review of the state of the art. *Mol. Syst. Des. Eng.* 2019, 4: 828-849.
7. Segler M, Preuss M, Waller M. Planning chemical syntheses with deep neural networks and symbolic AI[J]. *Nature*. 2018, 555: 604-610.
8. Ahneman D, et al. Predicting reaction performance in C-N cross-coupling using machine learning[J]. *Science*. 2018, 360: 186-190.
9. Vamathevan J, et al. Applications of machine learning in drug discovery and development[J]. *Nature Reviews Drug Discovery*. 2019, 18: 463-477.
10. Butler K T, et al. Machine learning for molecular and materials science[J]. *Science*. 2018, 559: 547-555.
11. Zhou Z, et al. Optimizing chemical reactions with deep reinforcement learning[J]. *ACS central science*. 2017; 3: 1337-1344.
12. Gao H, et al. Using machine learning to predict suitable conditions for organic reactions[J]. *ACS central science*. 2018, 4: 1465-1476.
13. Coley C, et al. A robotic platform for flow synthesis of organic compounds informed by AI planning[J]. *Science*. 2019, 365: eaax1566.
14. Steiner S, et al. Organic synthesis in a modular robotic system driven by a chemical programming language[J]. *Science*. 2019; 363: eaav 2211.
15. Trobe M, Burke M. The molecular Industrial Revolution: Automated Synthesis of small molecules[J]. *Angewandte Chemie International Edition*. 2018, 57: 4192-4214.

16. Corey E J, Wipke W T. Computer-Assisted Design of Complex Organic Syntheses[J]. Science, 1969, 166: 178-192.
17. Coley C W, Barzilay R, Jaakkola T S, et al. Prediction of Organic Reaction Outcomes Using Machine Learning[J]. ACS central science, 2017, 3(5): 434-443.
18. Segler M, et al. Modeling chemical reasoning to predict and invent reactions[J]. Chem. – Eur. J. 2017, 23: 6118-6128
19. Segler M, et al. Neural-symbolic machine learning for retrosynthesis and reaction prediction[J]. Chem. – Eur. J. 2017, 23: 5966-5971.
20. Ekins S, et al. Exploiting machine learning for end-to-end drug discovery and development[J]. Nature Materials. 2019, 18: 435-441.
21. Corey E, et al. Computer-assisted analysis in organic synthesis[J]. Science. 1985; 228: 408-418.
22. Schwaller P, Gaudin T, Lanyi D, et al. “Found in Translation”: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models[J]. Chemical Science, 2018, 9: 6091-6098.
23. Rogers D, Hahn M. Extended-connectivity fingerprints[J]. Journal of Chemical Information and Modeling, 2010, 50: 742-754.
24. LeCun Y, et al. Deep learning[J]. Nature, 2015, 521: 436-444.
25. LeCun Y. Learning process in an asymmetric threshold network[M]//Disordered systems and biological organization. Springer, Berlin, Heidelberg, 1986: 233-240.
26. Friesen A L, Domingos P M. Deep Learning as a Mixed Convex-Combinatorial Optimization Problem[J]. international conference on learning representations, 2018.
27. Wu Y, Liu Y. Robust Truncated Hinge Loss Support Vector Machines[J]. Journal of the American Statistical Association, 2007, 102(479): 974-983.
28. Lowe D M. Extraction of chemical structures and reactions from the literature[D]. University of Cambridge, 2012.
29. Law J, et al. A retrosynthetic analysis tool utilizing automated retrosynthetic rule generation[J]. Journal of Chemical Information and modeling, 2009, 49: 593-602.
30. Borgevig A, et al. Route design in the 21st century: the ICSYNTH software tool as an idea generator for synthesis prediction[J]. Organic Process Research & Development. 2015, 19: 357-368.
31. Srivastava N, et al. Dropout: A Simple Way to Prevent Neural Networks from Overfitting[J]. Journal of Machine Learning Research. 2014, 15: 1929-1958.
32. Maurizio P, et al. 3,4-diarylpyrazoles as protein kinase inhibitors: WO2010010154A1[P]., 2010-01-28
33. Christopher S. B, et al. P2x3 receptor antagonists for treatment of pain: WO2009058299A1[P]. 2009-05-07

Figures

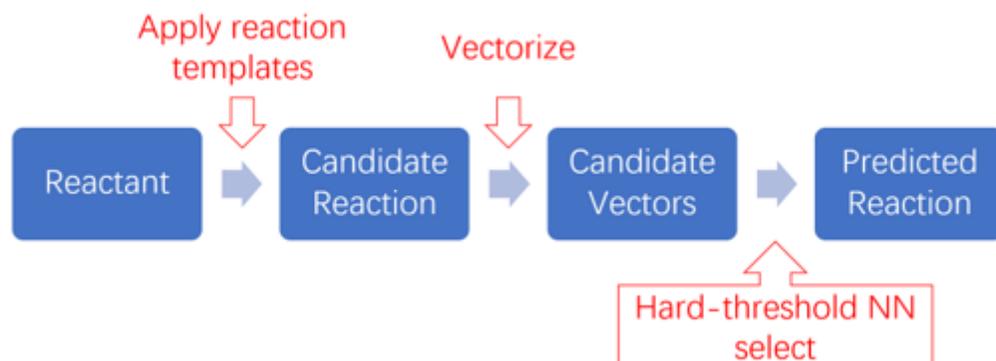


Figure 1

The illustration of template-based forward reaction prediction process

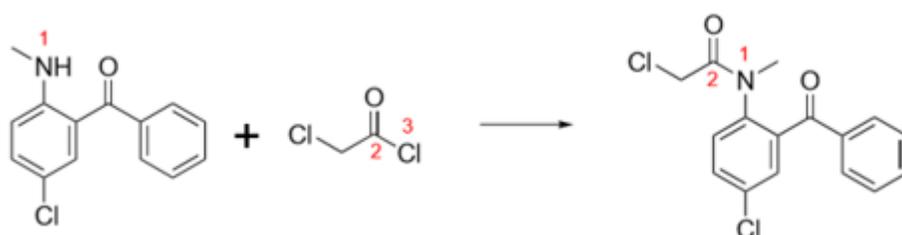


Figure 2

Reaction between chloroacetyl chloride and 2-methylamino-5-chlorobenzophenone.

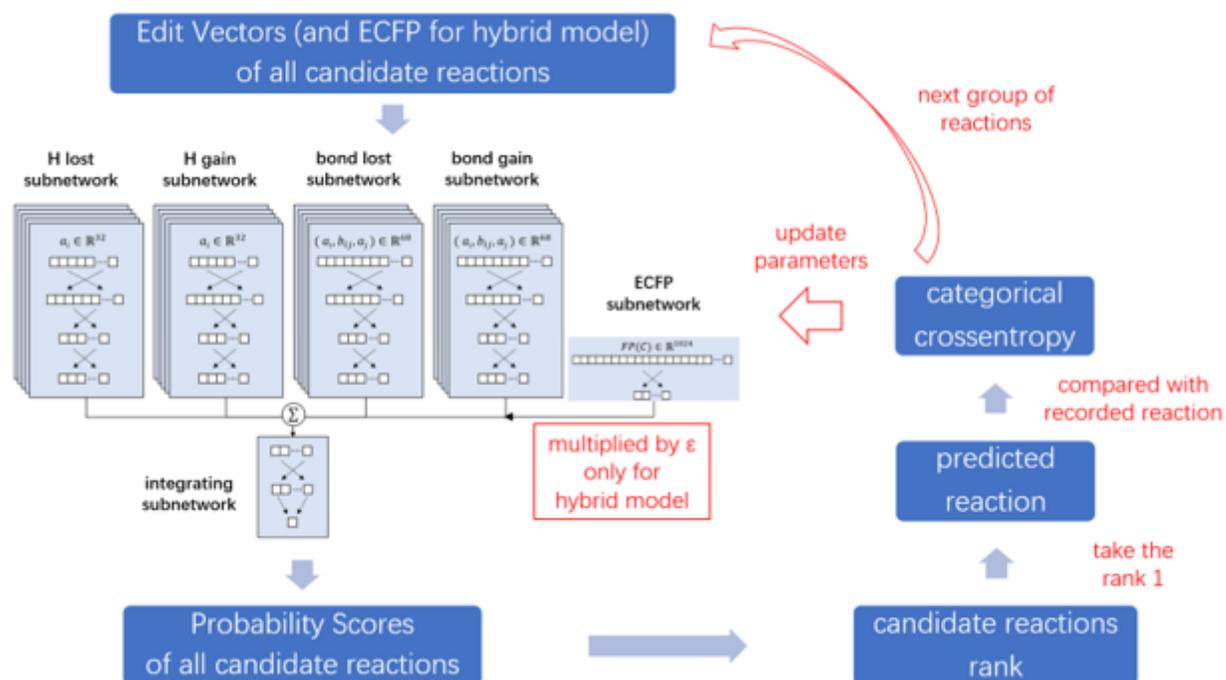


Figure 3

The illustration of selection process.

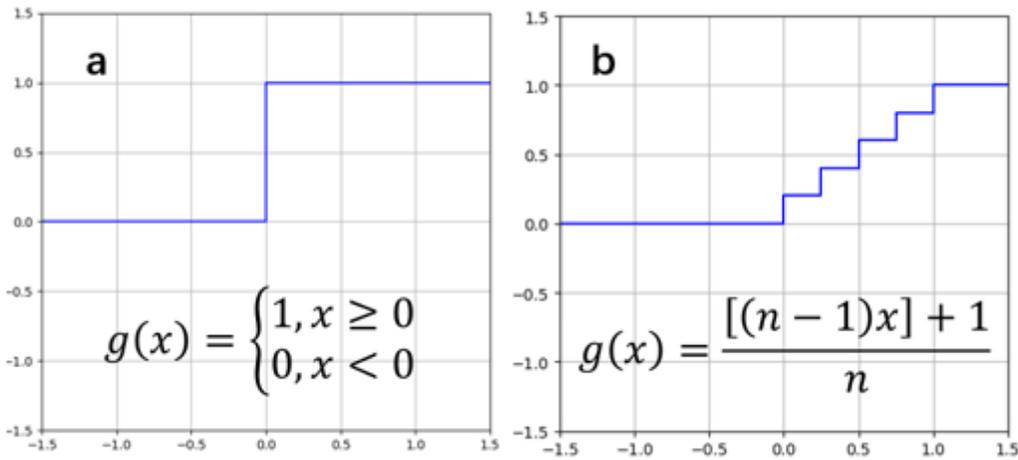


Figure 4

Hard-threshold activations. (a) Step activation. (b) Staircase activation.

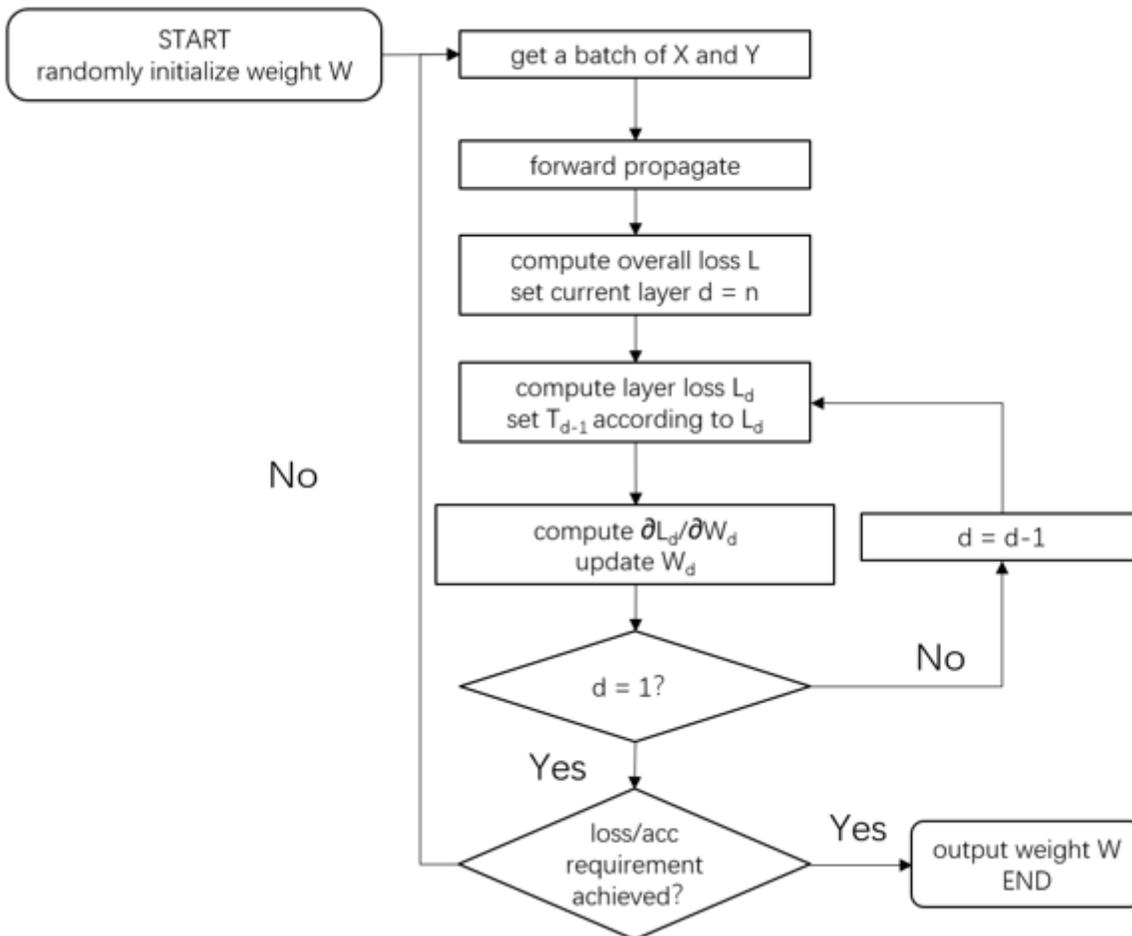


Figure 5

Block diagram of target propagation algorithm.

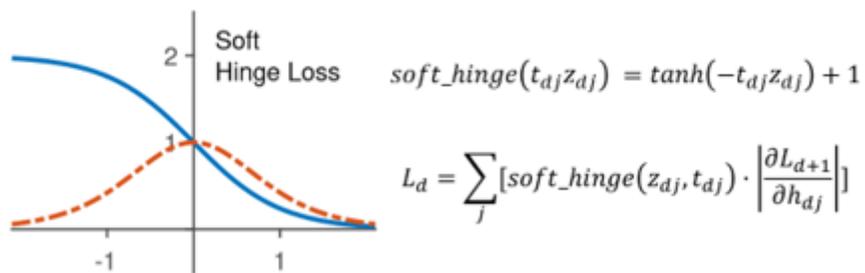


Figure 6

Weighted soft hinge layer loss function.

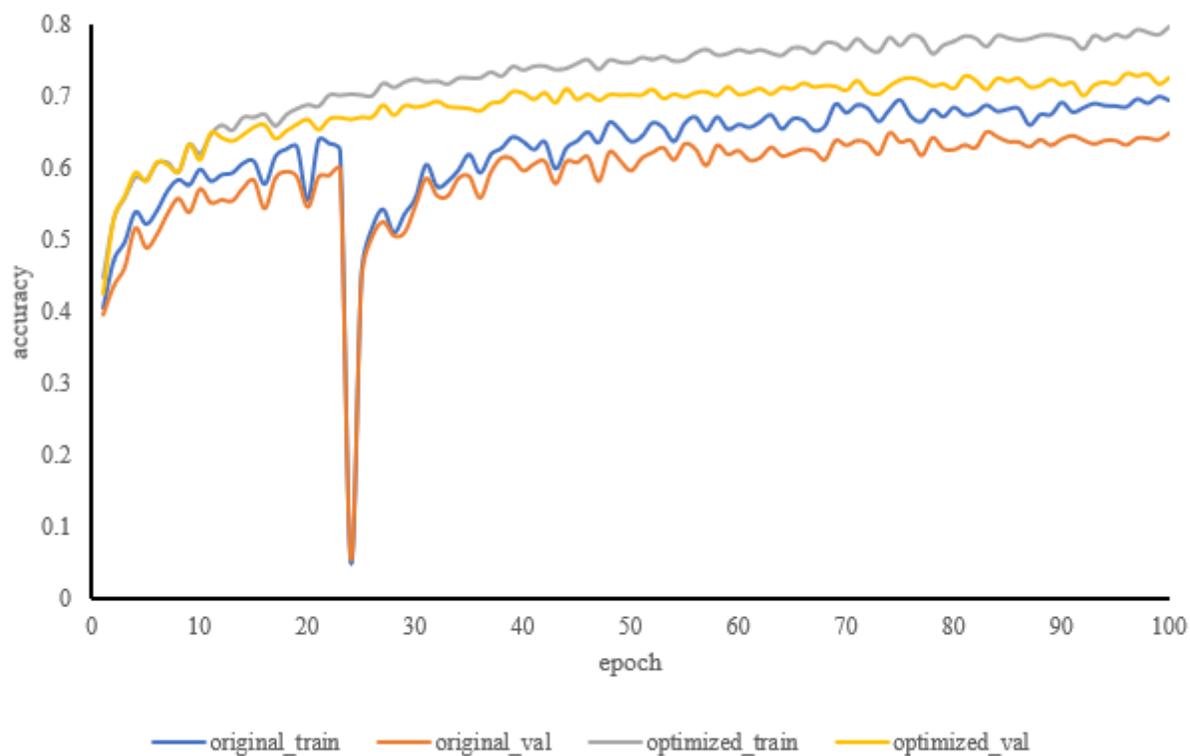


Figure 7

Training and validation accuracy history of original model and optimized model.

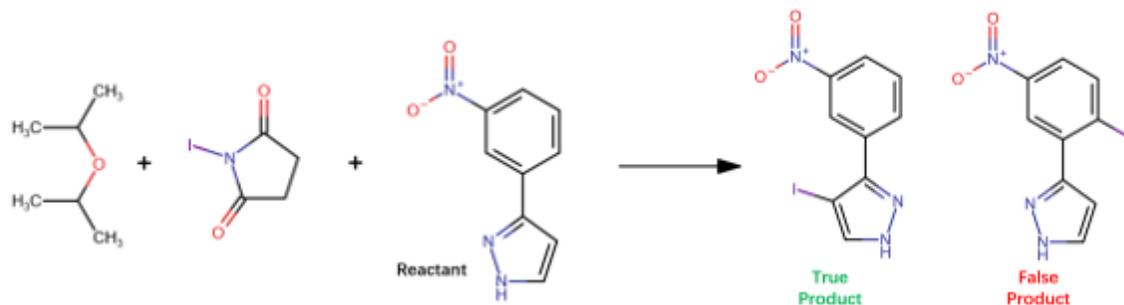


Figure 8

Iodine reaction in 2-propan-2-yloxypropane between NIS and 3-(3-Nitrophenyl)-1H-pyrazole

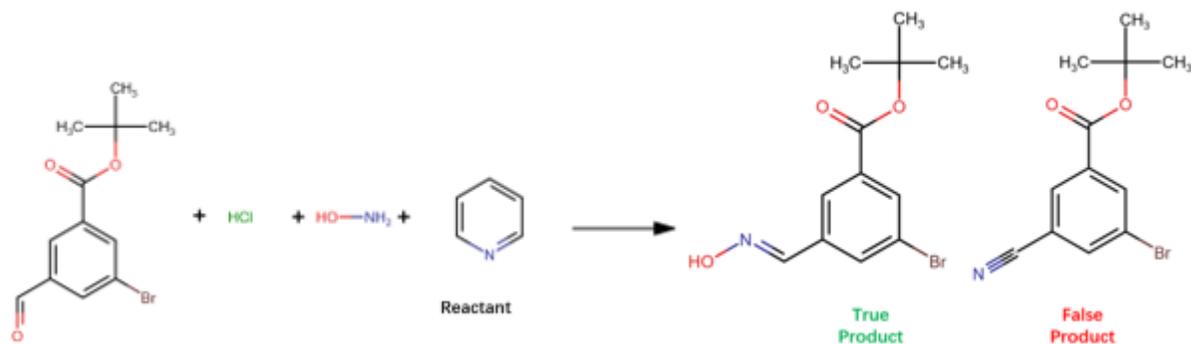


Figure 9

Amination of hydrazines with aromatic aldehydes under hydrochloric acid-pyridine condition.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Equation1.jpg](#)
- [Equation2.jpg](#)
- [formula1.jpg](#)
- [formula2.jpg](#)