Hosein Fooladi (November 2020)

State of The Art Molecule Synthetic Accessibility Tool Review

In this document, I want to review different approaches (solutions) towards the problem of retrosynthesis prediction and planning. I can divide (classify) papers based on different criteria. Papers can be classified based on template usage:

1- Template-based methods
2- Template-free methods

Also, papers can be classified based on the steps required for retrosynthesis:

1- One-step retrosynthesis
2- Multi-step retrosynthesis

So, I think we should classify papers based on these, and consider them when we are talking about performance and accuracy of each methods.

In section 1, I just want to list the papers based on template and steps required for retrosynthesis.

Template-based methods and multi-step retrosynthesis

1- Planning chemical syntheses with deep neural networks and symbolic AI
2- Learning retrosynthetic planning through self-play
3- Depth-First Proof-Number Search with Heuristic Edge Cost and Application to Chemical Synthesis Planning
4- Retro*: Learning Retrosynthetic Planning with Neural Guided A* Search

Template-free methods and one-step retrosynthesis

1- Learning Graph Models for Template-Free Retrosynthesis
2- Molecular Transformer unifies reaction prediction and retrosynthesis across pharma chemical space Predicting
3- Retrosynthetic Reaction using Self-Corrected Transformer Neural Networks
4- Automatic retrosynthetic route planning using template-free models

5- Energy-based View of Retrosynthesis

Template-based methods and one-step retrosynthesis!

1- Retrosynthesis Prediction with Conditional Graph Logic Network
2- Computer-Assisted Retrosynthesis Based on Molecular Similarity
3- Neural-Symbolic Machine Learning for Retrosynthesis and Reaction Prediction

On the other hand, researchers have tried to tackle the even harder problem of multi-step retrosynthesis using single-step retrosynthesis as a subroutine. So, any improvement in single-step retrosynthesis could directly transfer into improvement of multi-step retrosynthesis.

In general (independent of particular paper), there are some advantage and disadvantage for using template-based/template-free methods.

Template-based methods

First, there is an inevitable trade-off between generalization and specificity in template-based methods. Second, current template extraction algorithms consider reaction centers and their neighboring atoms, but not the global chemical environment of molecules. Moreover, mapping the atoms between products and reactants remains a nontrivial problem for all template-based methods.

Template-free methods

They don't need a template, and usually these papers are framing retrosynthesis problem as a translation problem. Given input (product SMILES), the goal is to translate this input to output (reactants SMILES). Clearly, the advantage is they don't need to stick to predefined rules and templates. But the main disadvantage is the invalidity rate of the predicted SMILES (because the generation process is producing SMILES character by character and this is prone to error). Moreover, the process is not interpretable.

Note: I have barely seen the template free methods for multi-step retrosynthesis (of course, there are some effort, but I did not find them promising). Most of the methods for multi-step are based on template-based methods.

Personal opinion: I think template-based multi-step retrosynthesis planning is the most promising approach for the task. Maybe we can enhance incorporated one-step subroutines that is used in these methods.

All the papers I skim or read

1- Planning chemical syntheses with deep neural networks and symbolic AI
2- Learning retrosynthetic planning through self-play
3- Depth-First Proof-Number Search with Heuristic Edge Cost and Application to Chemical Synthesis Planning
4- Retro*: Learning Retrosynthetic Planning with Neural Guided A* Search
5- Retrosynthetic Reaction Prediction Using Neural Sequence-to-Sequence Models
6- Learning Graph Models for Template-Free Retrosynthesis
7- A Graph to Graphs Framework for Retrosynthesis Prediction
8- Molecular Transformer unifies reaction prediction and retrosynthesis across pharma chemical space Predicting
9- Retrosynthetic Reaction using Self-Corrected Transformer Neural Networks
10-Molecular Transformer – A Model for Uncertainty-Calibrated Chemical Reaction Prediction
11-Machine Learning in Computer-Aided Synthesis Planning.
12- Automatic retrosynthetic route planning using template-free models
13- Retrosynthesis Prediction with Conditional Graph Logic Network
14- Computer-Assisted Retrosynthesis Based on Molecular Similarity
15- Neural-Symbolic Machine Learning for Retrosynthesis and Reaction Prediction

So, In the next section, I am choosing the best methods based on my opinion, and I am trying to provide more explanation about each paper.

Retro*: Learning Retrosynthetic Planning with Neural Guided A* Search

Template-based multi-step retrosynthesis. I think this is state of the art in this method class.

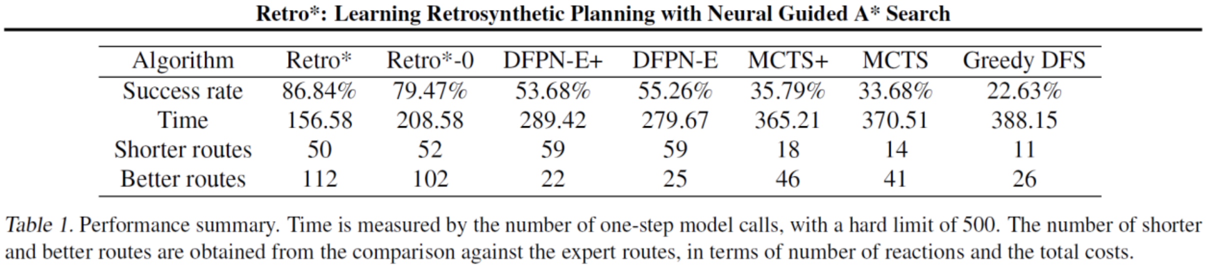
Dataset: USPTO

Preprocessing and using about 1.3 Million reaction. Also, using about 380K distinct template for template prediction task.

Code availability:

Yes, the code is available. But the algorithm is a little difficult, and I think needs some time to become familiar with the code.

Performance:



Retro*: Learning Retrosynthetic Planning with Neural Guided A* Search

Algorithm	Retro*	Retro*-0	DFPN-E+	DFPN-E	MCTS+	MCTS	Greedy DFS
Success rate	86.84%	79.47%	53.68%	55.26%	35.79%	33.68%	22.63%
Time	156.58	208.58	289.42	279.67	365.21	370.51	388.15
Shorter routes	50	52	59	59	18	14	11
Better routes	112	102	22	25	46	41	26

Table 1. Performance summary. Time is measured by the number of one-step model calls, with a hard limit of 500. The number of shorter and better routes are obtained from the comparison against the expert routes, in terms of number of reactions and the total costs.

Predicting Retrosynthetic Reaction using Self Corrected Transformer Neural Networks

Template-free one-step retrosynthesis.

Dataset: USPTO-50k

The dataset was derived from USPTO granted patents that includes 50, 000 reactions that was later classified into 10 reaction classes by Schneider et al, namely USPTO-50K.

Code availability:

Yes, the code is available. I should look more carefully to the code, but in a nutshell, it is a transformer-based model, and should not be very hard to reproduce.

Performance:

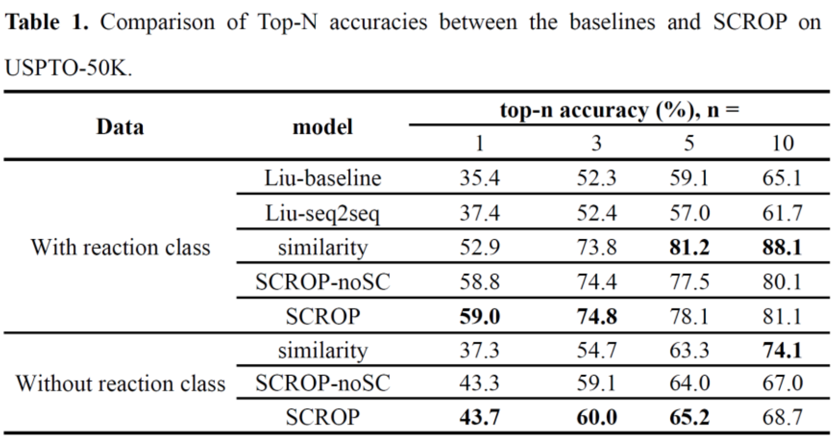


Table 1. Comparison of Top-N accuracies between the baselines and SCROP on USPTO-50K.

Data	model	top-n accuracy (%), n =			
		1	3	5	10
With reaction class	Liu-baseline	35.4	52.3	59.1	65.1
	Liu-seq2seq	37.4	52.4	57.0	61.7
	similarity	52.9	73.8	**81.2**	**88.1**
	SCROP-noSC	58.8	74.4	77.5	80.1
	SCROP	**59.0**	**74.8**	78.1	81.1
Without reaction class	similarity	37.3	54.7	63.3	**74.1**
	SCROP-noSC	43.3	59.1	64.0	67.0
	SCROP	**43.7**	**60.0**	**65.2**	68.7

Learning Graph Models for Template-Free Retrosynthesis

Template-free one-step retrosynthesis. It is state of the art in most of one-step retrosynthesis tasks.

Dataset: USPTO-50k

The dataset was derived from USPTO granted patents that includes 50, 000 reactions that was later classified into 10 reaction classes by Schneider et al, namely USPTO-50K.

Code availability:

No, the code is not available now. But the authors have mentioned that they will release the code.

Performance:

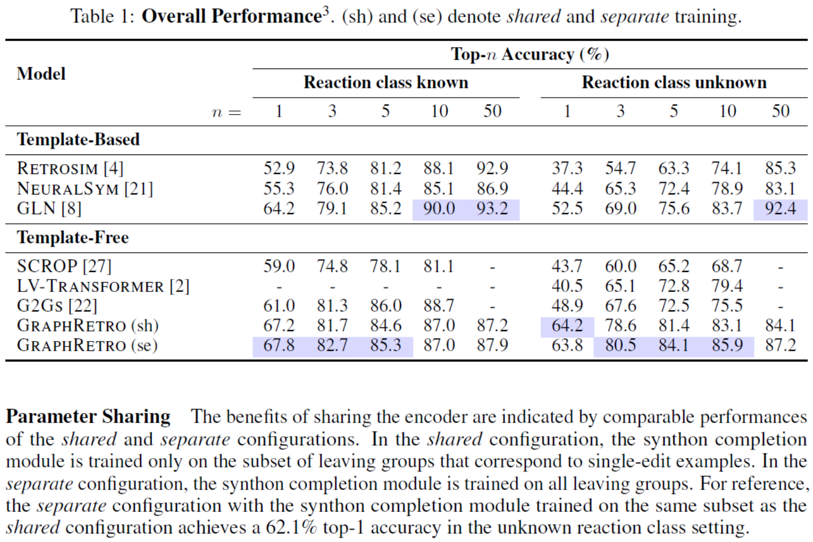


Table 1: **Overall Performance**[3]. (sh) and (se) denote *shared* and *separate* training.

Model	Top-n Accuracy (%)									
	Reaction class known					Reaction class unknown				
$n =$	1	3	5	10	50	1	3	5	10	50
Template-Based										
RETROSIM [4]	52.9	73.8	81.2	88.1	92.9	37.3	54.7	63.3	74.1	85.3
NEURALSYM [21]	55.3	76.0	81.4	85.1	86.9	44.4	65.3	72.4	78.9	83.1
GLN [8]	64.2	79.1	85.2	90.0	93.2	52.5	69.0	75.6	83.7	92.4
Template-Free										
SCROP [27]	59.0	74.8	78.1	81.1	-	43.7	60.0	65.2	68.7	-
LV-TRANSFORMER [2]	-	-	-	-	-	40.5	65.1	72.8	79.4	-
G2GS [22]	61.0	81.3	86.0	88.7	-	48.9	67.6	72.5	75.5	-
GRAPHRETRO (sh)	67.2	81.7	84.6	87.0	87.2	64.2	78.6	81.4	83.1	84.1
GRAPHRETRO (se)	67.8	82.7	85.3	87.0	87.9	63.8	80.5	84.1	85.9	87.2

Parameter Sharing The benefits of sharing the encoder are indicated by comparable performances of the *shared* and *separate* configurations. In the *shared* configuration, the synthon completion module is trained only on the subset of leaving groups that correspond to single-edit examples. In the *separate* configuration, the synthon completion module is trained on all leaving groups. For reference, the *separate* configuration with the synthon completion module trained on the same subset as the *shared* configuration achieves a 62.1% top-1 accuracy in the unknown reaction class setting.