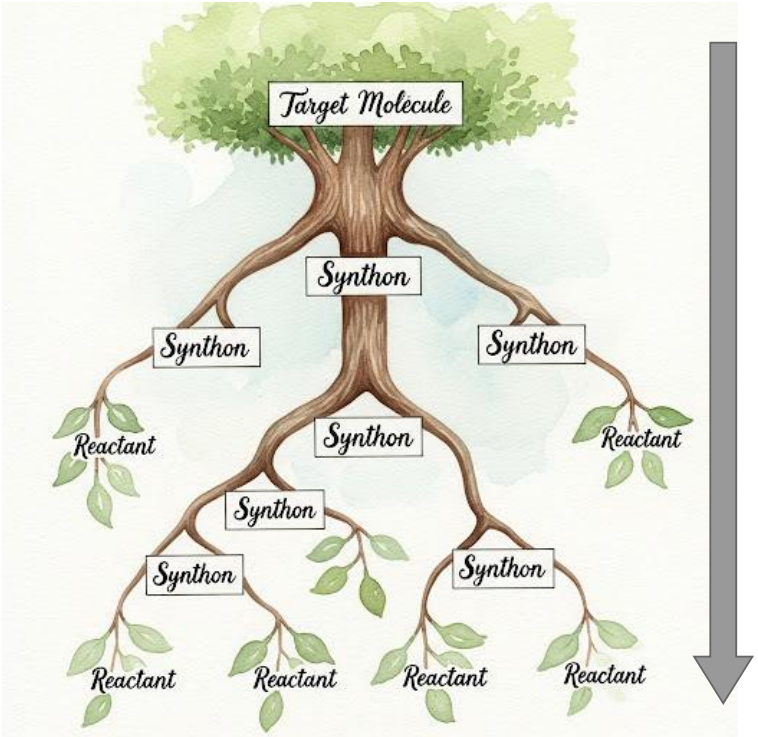
Synthony

Novel Synthon-based Retrosynthesis Approach

RDKit UGM 2025 - Prague

Mher Matevosyan

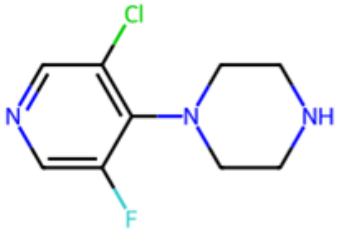
Retrosynthetic Tree



Gemini 2.5 Flash: Completely normal tree

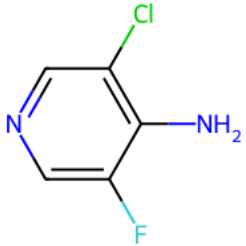
Target Molecule:

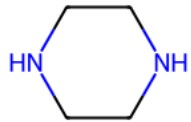
Final desired compound



Synthon:

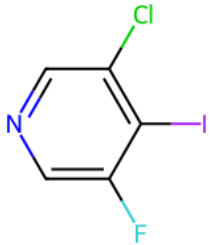
A hypothetical unit within a target molecule that represents a potential starting reactant in the retrosynthetic pathway of that target molecule

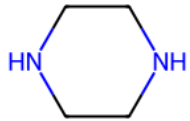




Reactant:

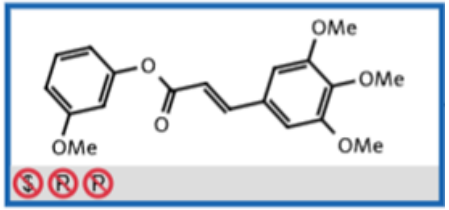
Starting material in a reaction that undergo a chemical change to form a product





Retrosynthesis with ML

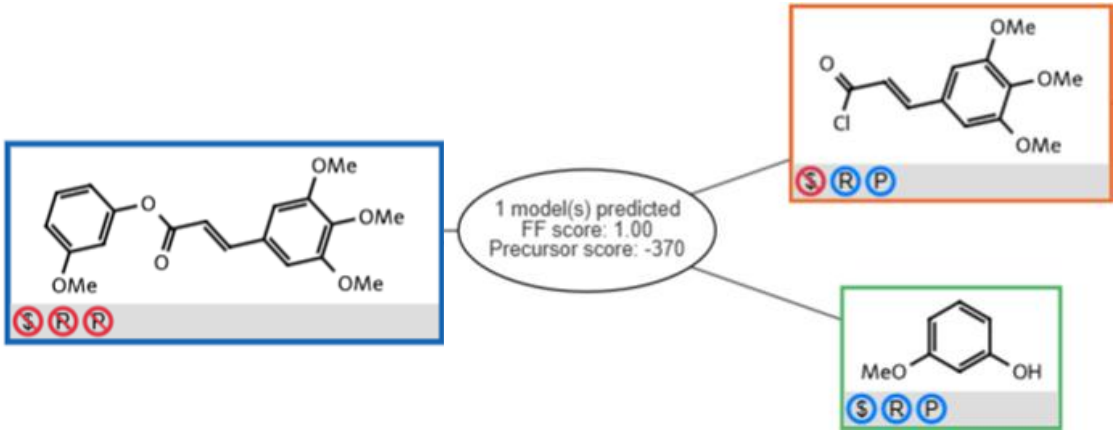
The standard approach is to train a 1-step retrosynthesis ML prediction model and recursively apply it to (hopefully) find a complete pathway.



Source: https://askcos.mit.edu/

Retrosynthesis with ML

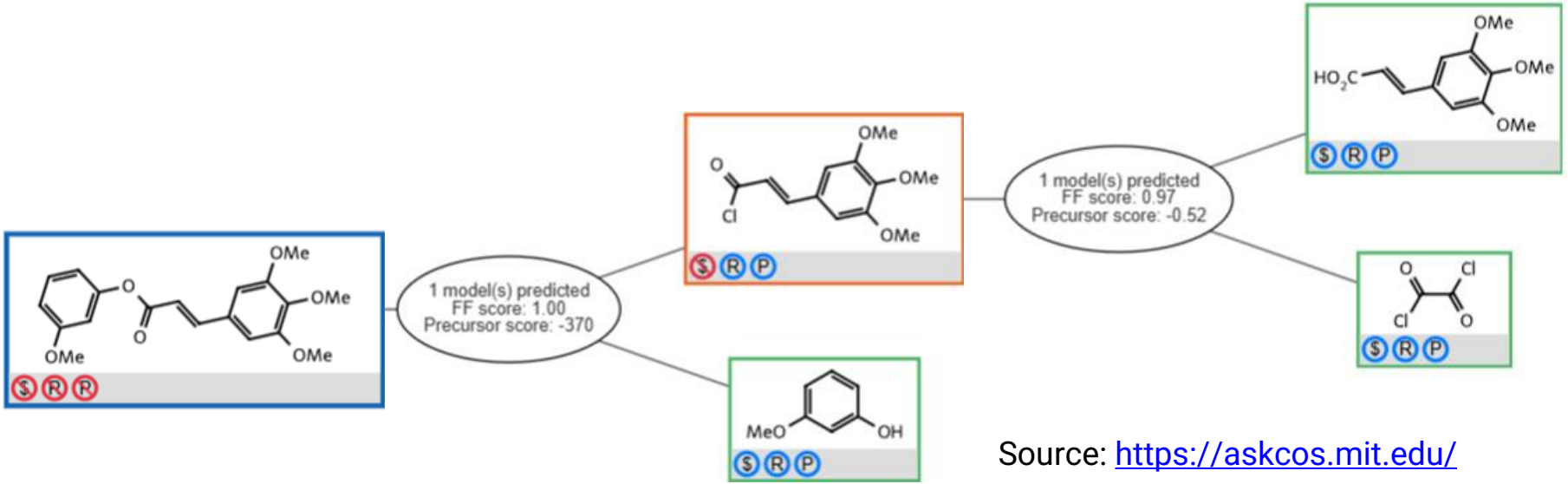
The standard approach is to train a 1-step retrosynthesis ML prediction model and recursively apply it to (hopefully) find a complete pathway.



Source: https://askcos.mit.edu/

Retrosynthesis with ML

The standard approach is to train a 1-step retrosynthesis ML prediction model and recursively apply it to (hopefully) find a complete pathway.



Source: https://askcos.mit.edu/

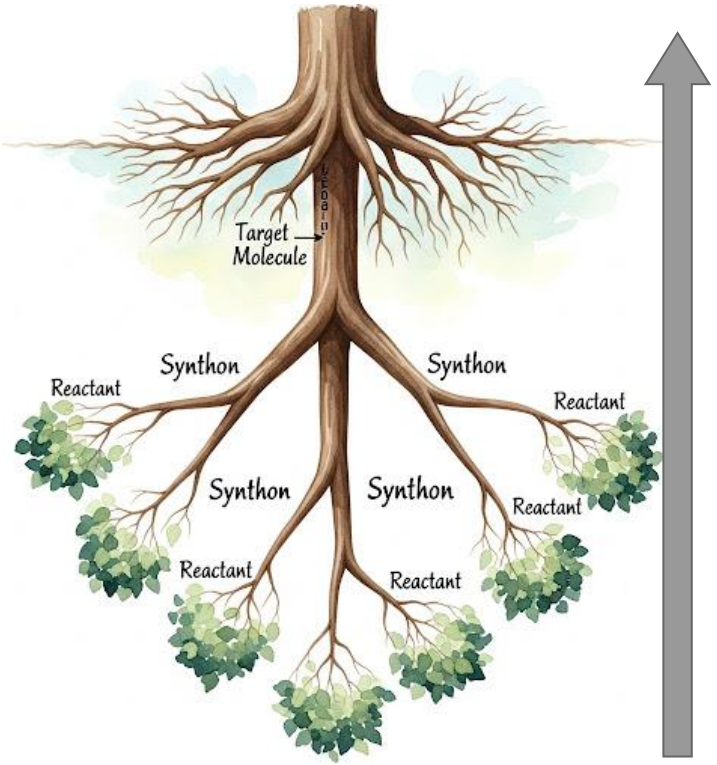
Retrosynthetic Tree but it is Upside Down?

Idea: Figure out the potential reactants from the target molecule first, then build the synthesis tree from leaves to the root — kind of like selecting the exact LEGO pieces before building the house.

Similar approaches are usually categorized under selection-based retrosynthesis.

Guo Z, Wu S, Ohno M, Yoshida R. Bayesian algorithm for retrosynthesis. J Chem Inf Model. 2020; 60: 4474–4486.

Lee H, Ahn S, Seo S-W, Song YY, Yang E, Hwang S-J, et al. RetCL: a selection-based approach for retrosynthesis via contrastive learning. Montreal: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence (IJCAI-21); 2021.



Gemini 2.5 Flash: Normal Tree #2

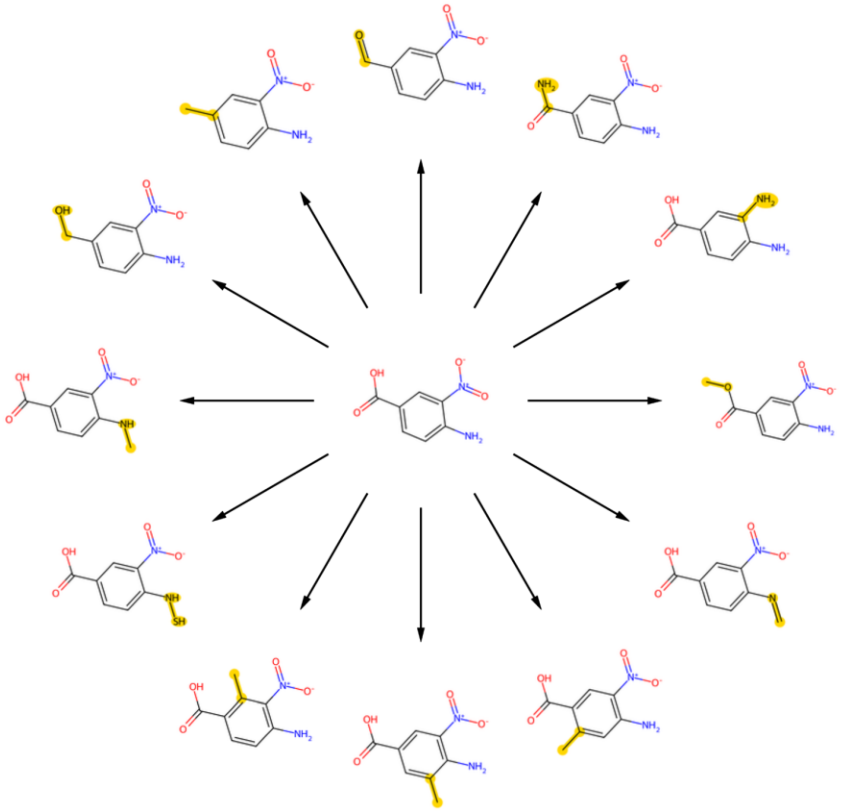
Synthony: General Overview

1. **Database Creation**: Generate a database of possible synthons from given stock and reaction databases
2. **Superstructure Search**: Do a superstructure search on target molecule from synthon database
3. **Synthon Combination**: Find all synthon combinations that cover target molecule without overlapping (Exact Cover Problem)
4. **Combination Ranking**: Rank the synthon combinations
5. **Tree Building**: Build retrosynthetivc tree from leaves to the root for each synthon combination

Custom C ++ scripts and its Python bindings for RDKit in 2 mins in Google Colab

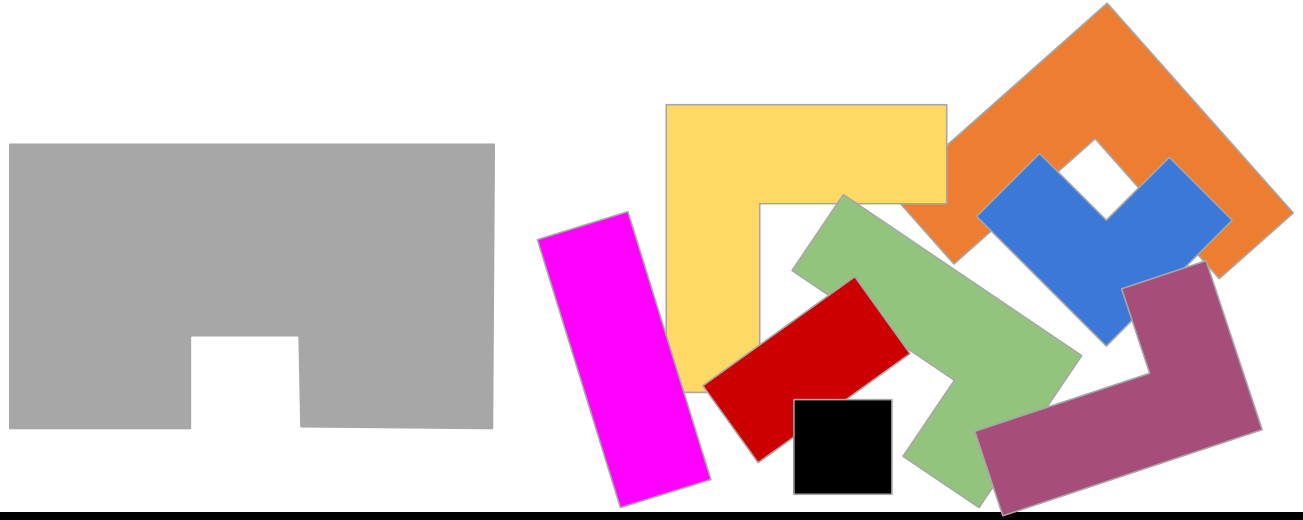
Step 1: Database Creation

A. Get atom mapped reaction database (e.g. USPTO50k)
B. Get stock (building block) database (e.g. reactants from USPTO-full)
C. Extract reaction templates using rdchiral
D. Separate reaction templates into a “synthon template” for each participating reactant
E. Match synthon templates to each building block and generate synthons



Step 2: Superstructure Search

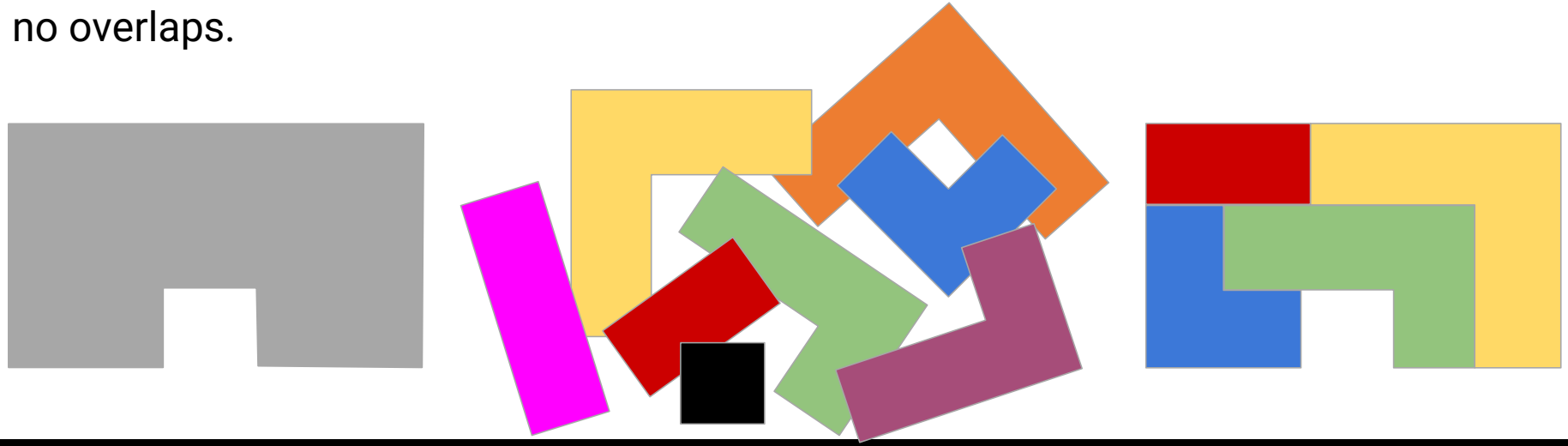
Superstructure search returns molecules that comprise or make up the provided chemical structure query (that is, substructures that is contained in the query superstructure).



Step 2: Superstructure Search Step 3: Combine Synthons

Superstructure search returns molecules that comprise or make up the provided chemical structure query (that is, substructures that is contained in the query superstructure).

The Exact Cover problem asks whether, given a set and a collection of its subsets, you can select some subsets so that every element is covered exactly once with no overlaps.



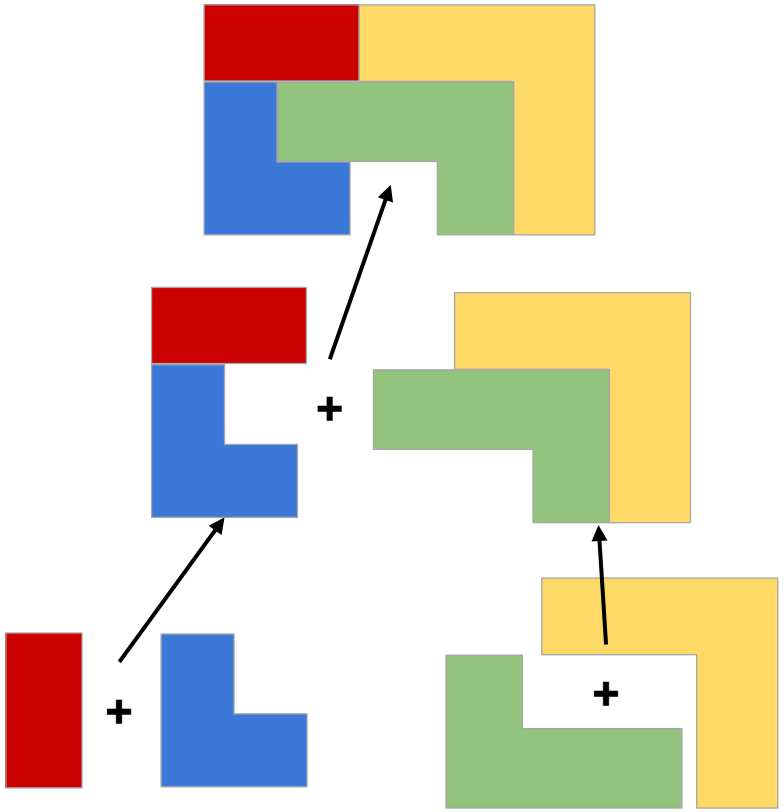
Step 4: Rank Combinations

Hybrid Ranking: Frequency + ML

Step 4: Rank Combinations

Step 5: Build Tree

Hybrid Ranking: Frequency + ML



Results: Top-k Accuracy

Top-k Accuracy measures the percentage of cases where the true reactants appear in the top-k predictions. The standard benchmark is the USPTO-50k dataset with a fixed train/validation/test split, and results are reported on the test set.

Model	Top 1	Top 3	Top 5	Top 10	Top 50	Top 100	Top 1000
Synthony[+]	**73.3%**	**91.2%**	**95.2%**	**96.9%**	**97.4%**	**97.7%**	**97.8%**
RetCL*[+]	71.3%	86.4%	92.0%	94.1%	96.4%	96.7%	-
GLN*	52.5%	69.0%	75.6%	83.7%	-	92.4%	-
LocalRetro**	51.8%	76.6%	84.5%	90.9%	96.7%	96.8%	-
MEGAN**	48.7%	72.4%	79.5%	86.8%	93.5%	94.0%	-

* Results are taken from Lee et al.

[+] Selection based algorithm

** From running locally via [syntheseus]

Results: Top-k Accuracy

Since our model (like some others) only considers **given building blocks**, we also compare cases where predictions are ignored if any reactant in the top-k is **outside the stock**.
Note: While none of these comparisons are fair, they provide a better picture.

Model	Top 1	Top 3	Top 5	Top 10	Top 50	Top 100	Top 1000
Synthony[+]	73.3%	91.2%	95.2%	**96.9%**	**97.4%**	**97.7%**	**97.8%**
RetCL*[+]	71.3%	86.4%	92.0%	94.1%	96.4%	96.7%	-
GLN*	77.3%	-	90.0%	93.3%	93.3%	93.3%	-
LocalRetro**	**78.6%**	**93.7%**	**96.0%**	96.6%	96.8%	96.8%	-
MEGAN**	76.1%	90.7%	92.9%	93.8%	94.0%	94.0%	-

* Results are taken from Lee et al.

[+] Selection based algorithm

** From running locally via [syntheseus]

Thank You!

Contact us at:

denovosciences.ai

linkedin.com/company/denovo-sciences/

mherm@denovosciences.ai