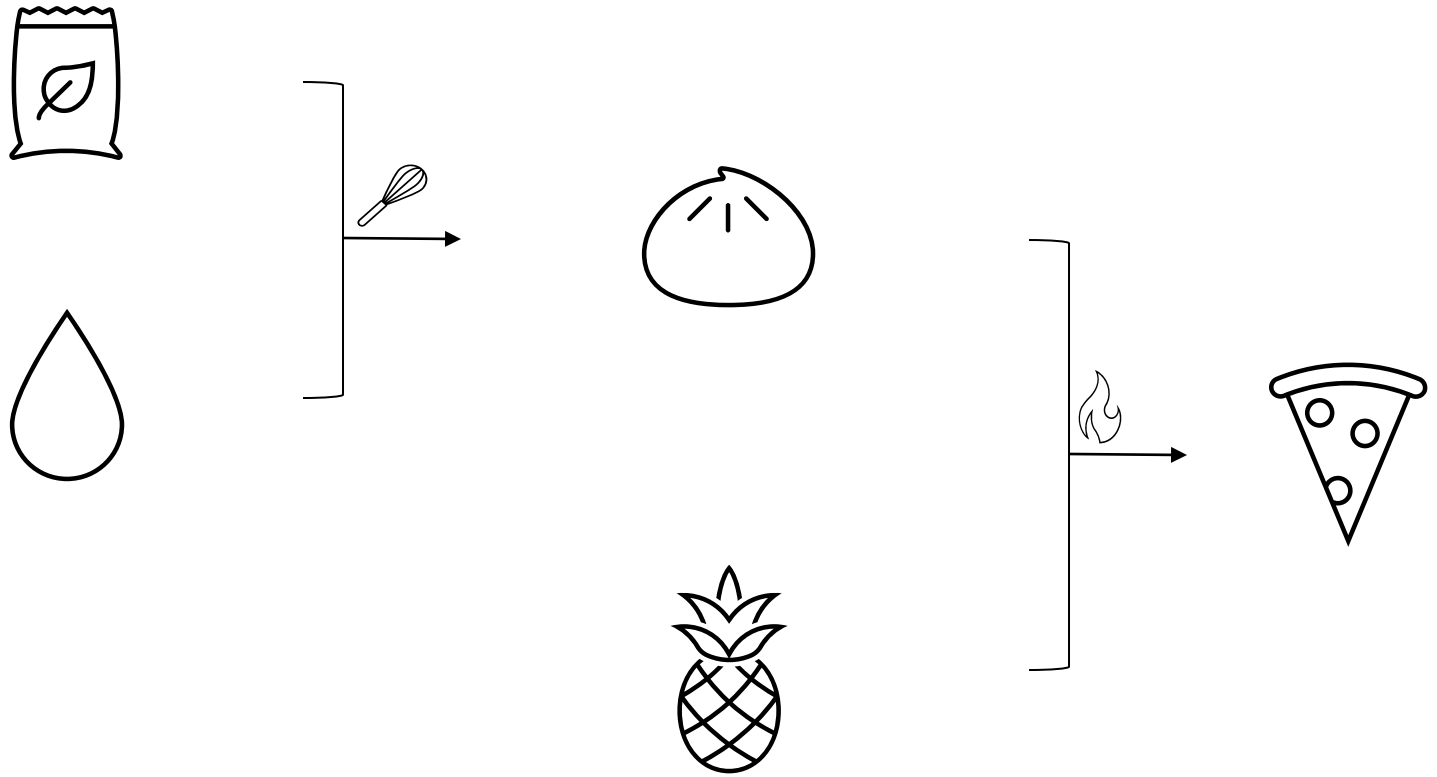
Microsoft

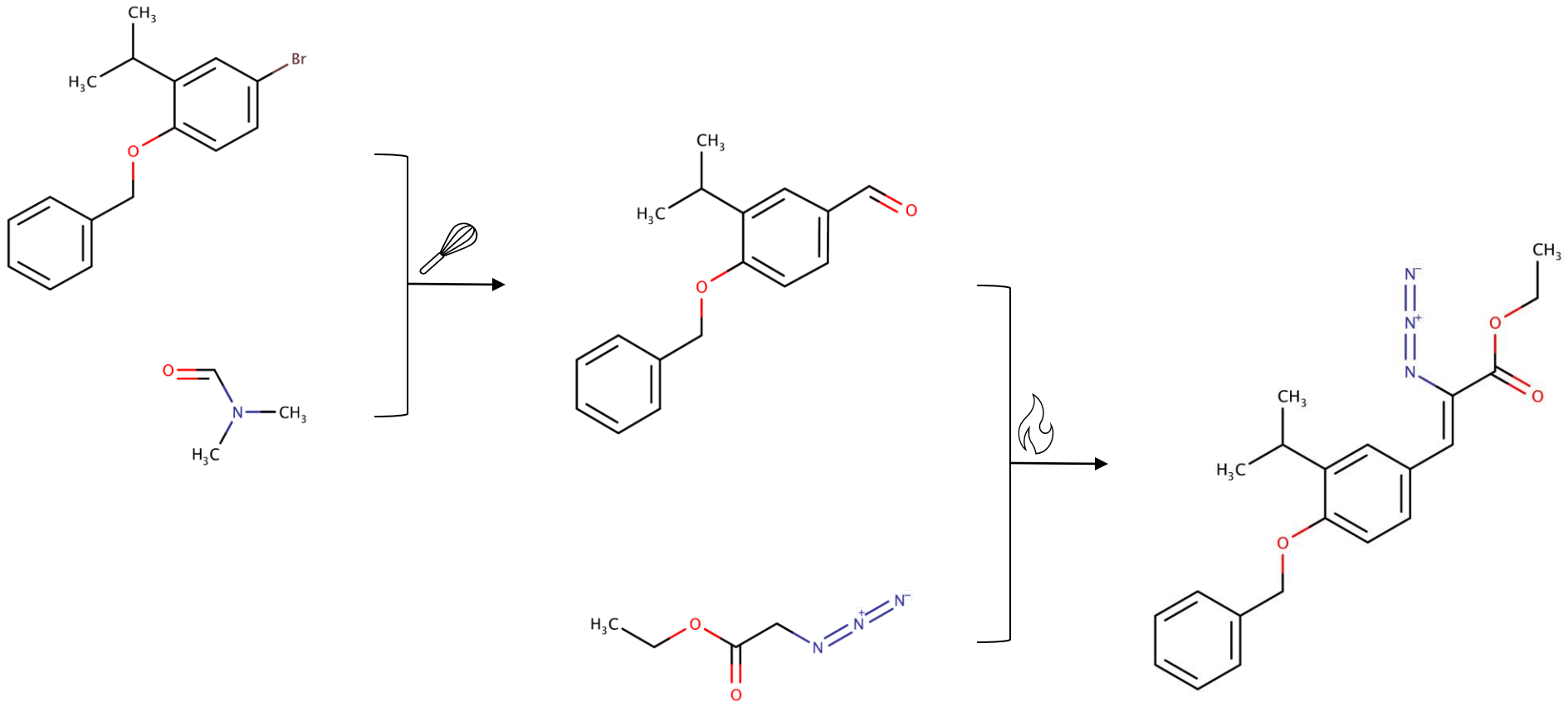
Pushing the boundaries of chemical synthesis with RetroChimera

Krzysztof Maziarz
Principal Researcher @ Microsoft Research AI for Science

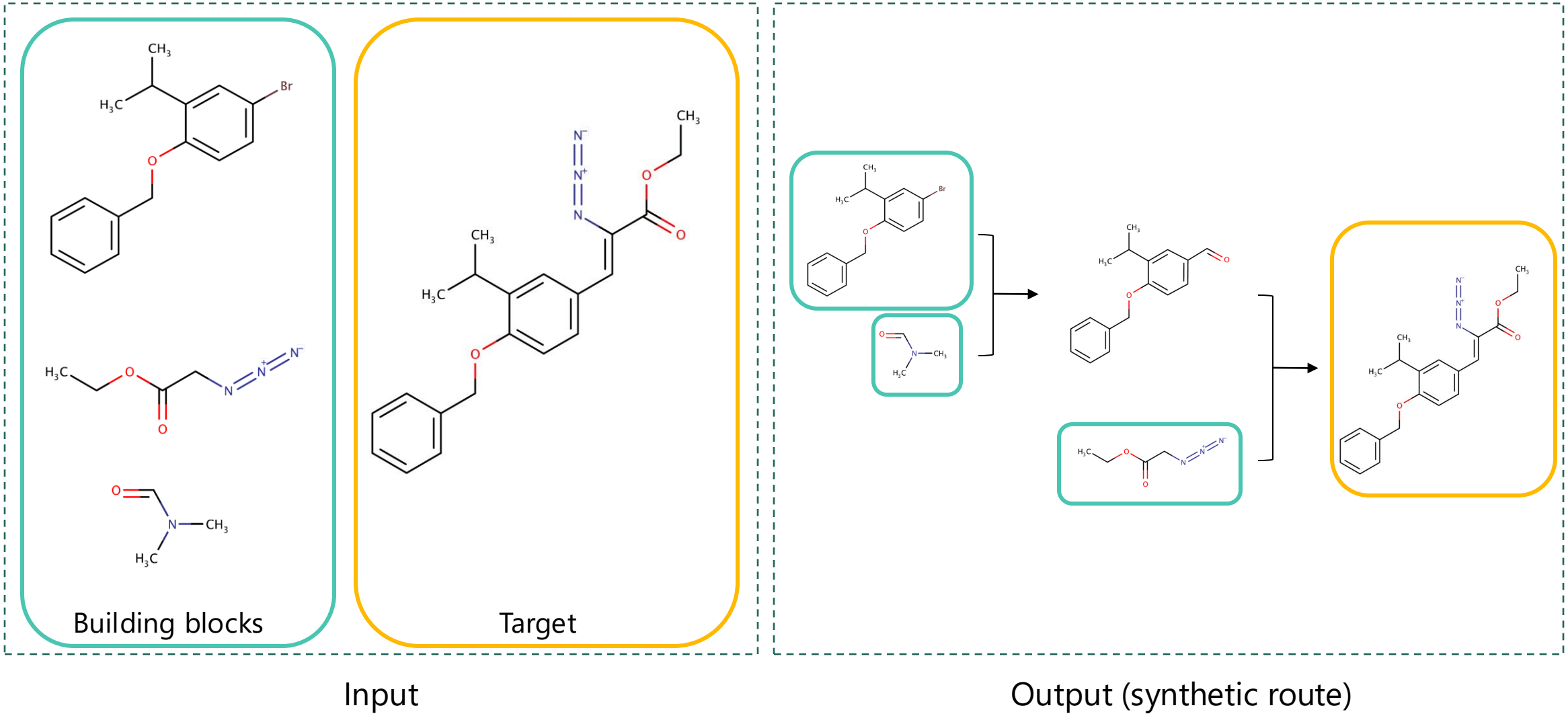
Synthesis: cooking with molecules



Synthesis: cooking with molecules



Synthesis planning



Microsoft

Demo time

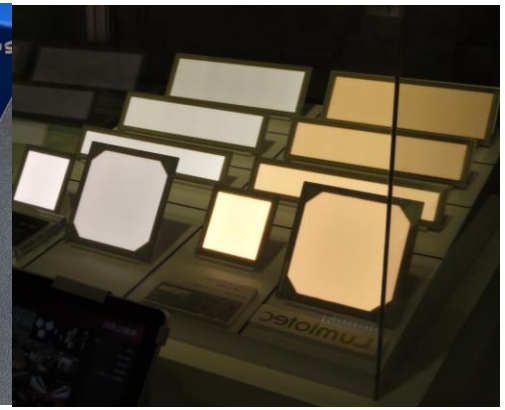
Microsoft

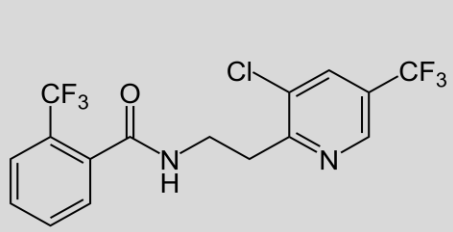
aka.ms/RetroChimeraNeurIPS

Why is synthesis planning important?

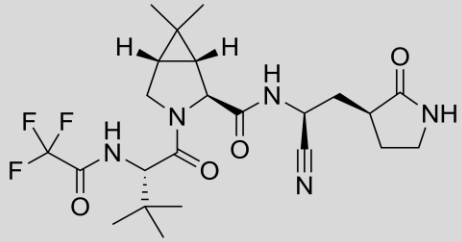
Small molecules are central to our wellbeing

aka.ms/RetroChimeraNeurIPS

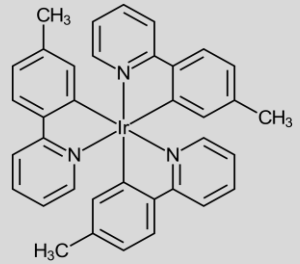




Agrochemicals



Drugs

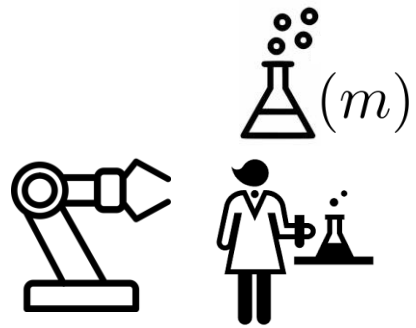


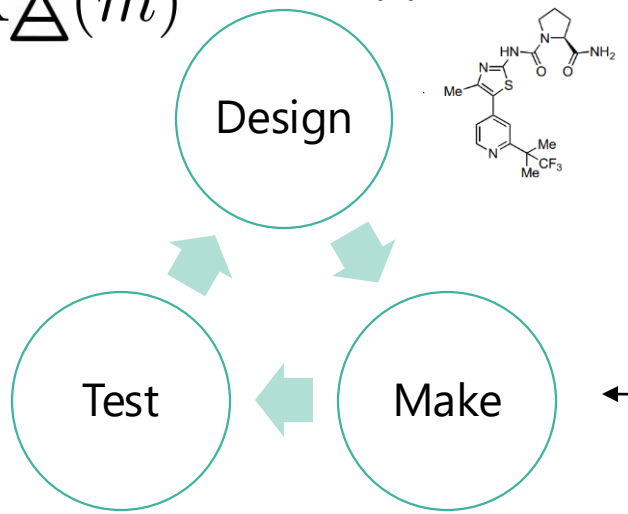
Materials

Synthesis is a bottleneck in drug discovery

aka.ms/RetroChimeraNeurIPS

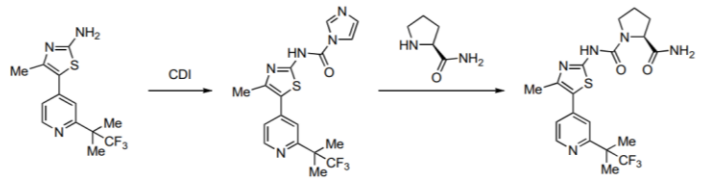
$$\arg\max_{m \in \mathcal{M}} (m) \quad (1)$$





Half of the designs fail here because:

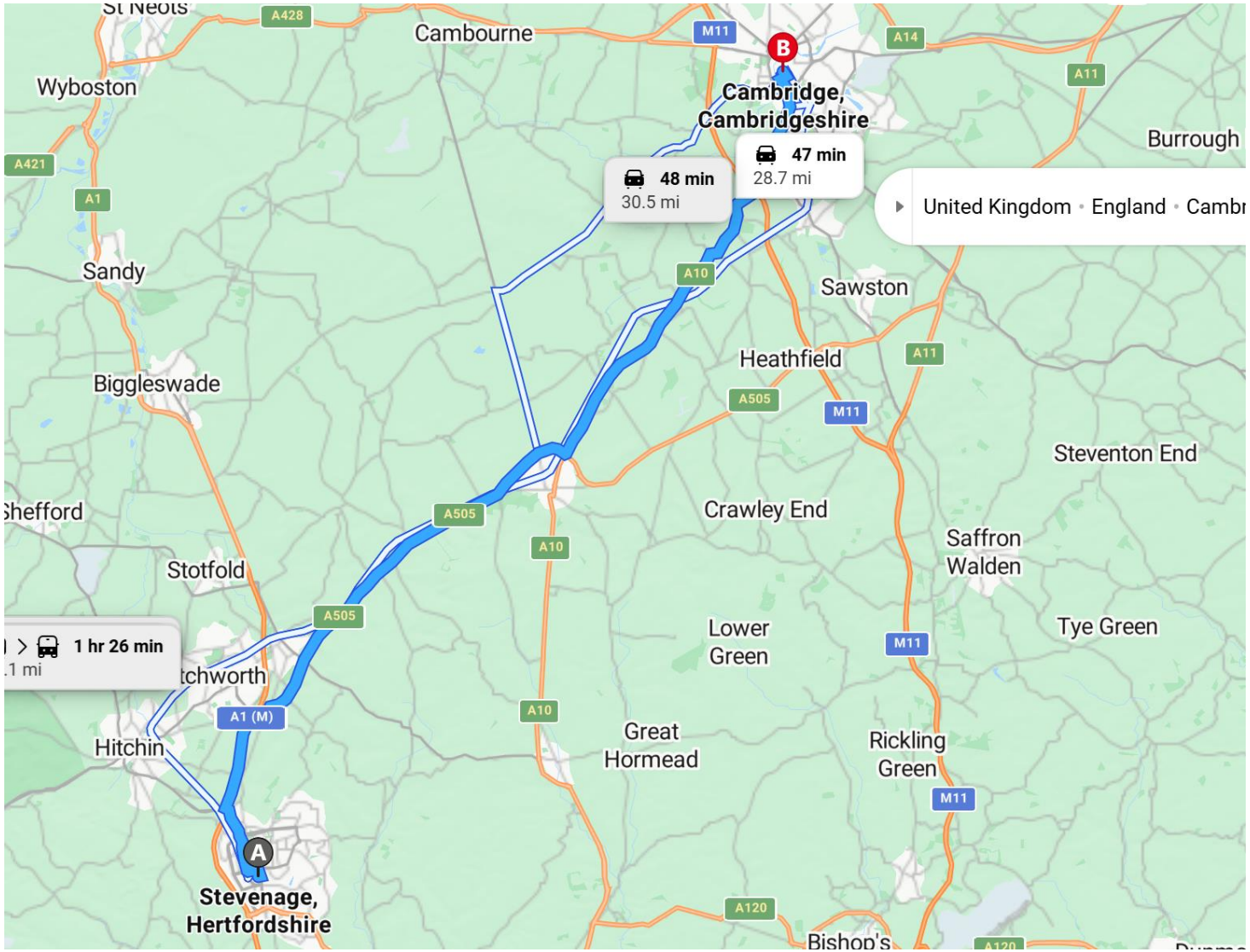
- Synthesis is a multi-step process
- It's time consuming & expensive
- Many reactions fail



(1) Baskin et al *D. Akad. Nauk* SSSR **1989**; Maziarz et al, *ICLR* **2021**; Wu et al. *Nat. Comm.* **2024**

GPS for molecules?

aka.ms/RetroChimeraNeurIPS



Microsoft

aka.ms/RetroChimeraNeurIPS

What kind of modelling is RetroChimera based on?

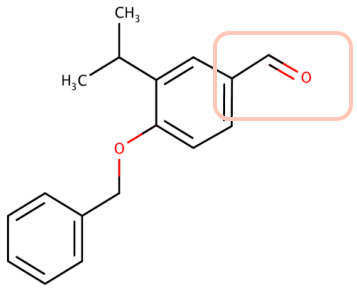
aka.ms/RetroChimeraNeurIPS

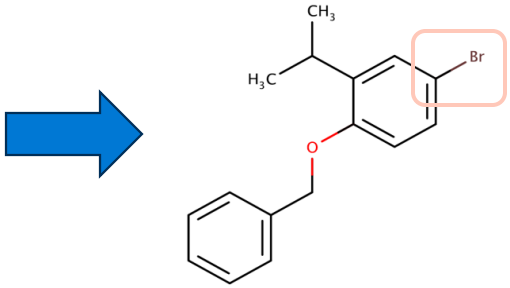
How to model chemical reactions

Input (Product)

Output (Reactants)

Molecular Graph





SMILES
(Graph as Token Sequence)

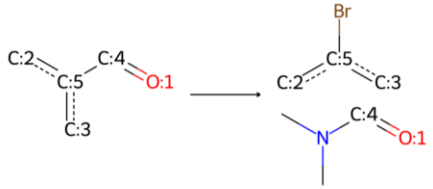
`CC(C)c1cc(C=O)ccc1OCc1ccccc1`

`CN(C)C=O.CC(C)c1cc(Br)ccc1OCc1ccccc1`

Predict Reactants *De Novo*

`CN(C)C=O.CC(C)c1cc(Br)ccc1OCc1ccccc1`

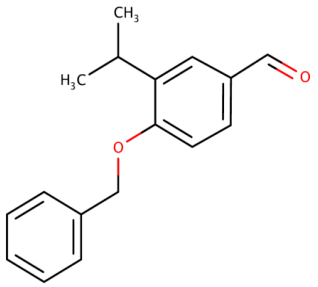
Predict Molecular *Edits*



Applying Edit Template to product yields reactants

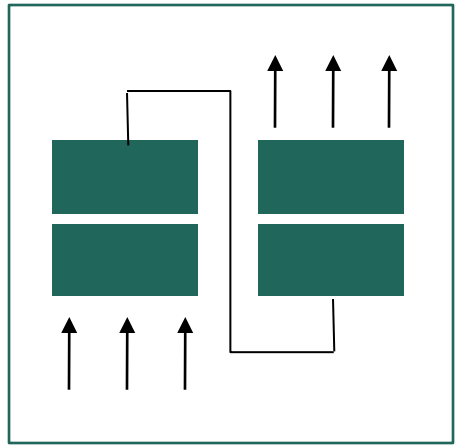
Approach #1: end-to-end Transformer

aka.ms/RetroChimeraNeurIPS

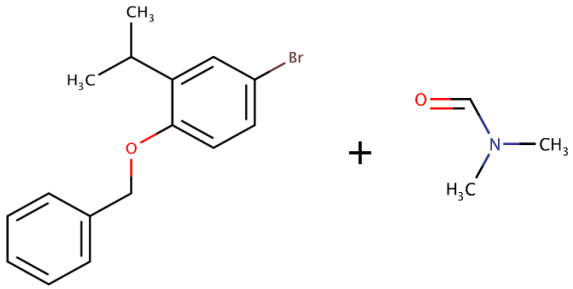


CC(C)C1=CC(C=O)=CC=C1OCC1=CC=CC=C1

Input: Product in SMILES notation



Encoder-Decoder Transformer



CC(C)C1=CC(Br)...

Output: Reactants in SMILES notation

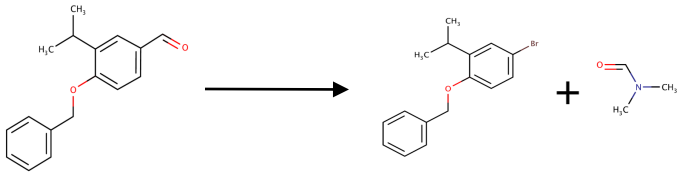
Approach #1: end-to-end Transformer

aka.ms/RetroChimeraNeurIPS

1. R-SMILES alignment and augmentation

Product **Reactants**

Molecular Graph:



1. Canonical SMILES:

CC(C)c1cc(C=O)ccc1OCc1ccccc1 → CC(C)c1cc(Br)ccc1OCc1ccccc1.CN(C)C=O

2. Randomized SMILES:

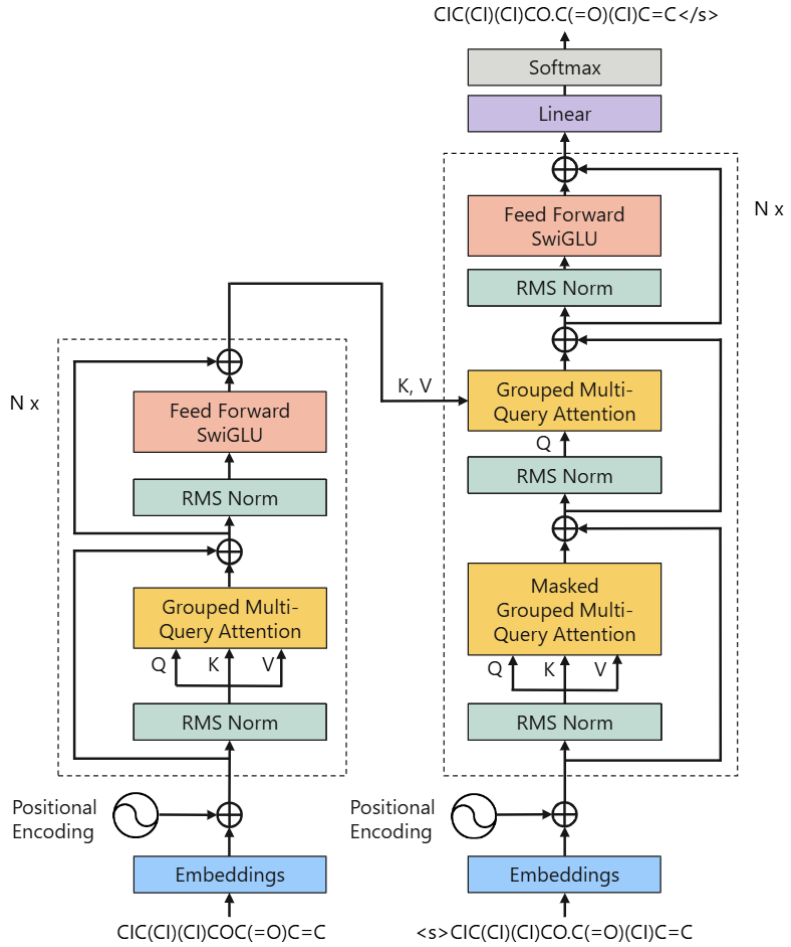
CC(C)c1cc(C=O)ccc1OCc1ccccc1 → CN(C)C=O.c1(C(C)C)c(OCc2ccccc2)ccc(Br)c1
c1ccccc1COc1c(C(C)C)cc(C=O)cc1 → CN(C)C=O.c1(COc2ccc(cc2C(C)C)Br)ccccc1
c1cccc(COc2c(C(C)C)cc(C=O)cc2)c1 → C(C)(C)c1cc(ccc1OCc1ccccc1)Br.CN(C=O)C

3. Root-aligned SMILES (plus N times augmentation):

CC(C)c1cc(C=O)ccc1OCc1ccccc1 → CC(C)c1cc(Br)ccc1OCc1ccccc1.C(=O)N(C)C
c1ccccc1COc1c(C(C)C)cc(C=O)cc1 → c1ccccc1COc1c(C(C)C)cc(Br)cc1.C(=O)N(C)C
c1cccc(COc2c(C(C)C)cc(C=O)cc2)c1 → c1cccc(COc2c(C(C)C)cc(Br)cc2)c1.C(=O)N(C)C

Leverage atom mapping information to reduce edit distance between two sides

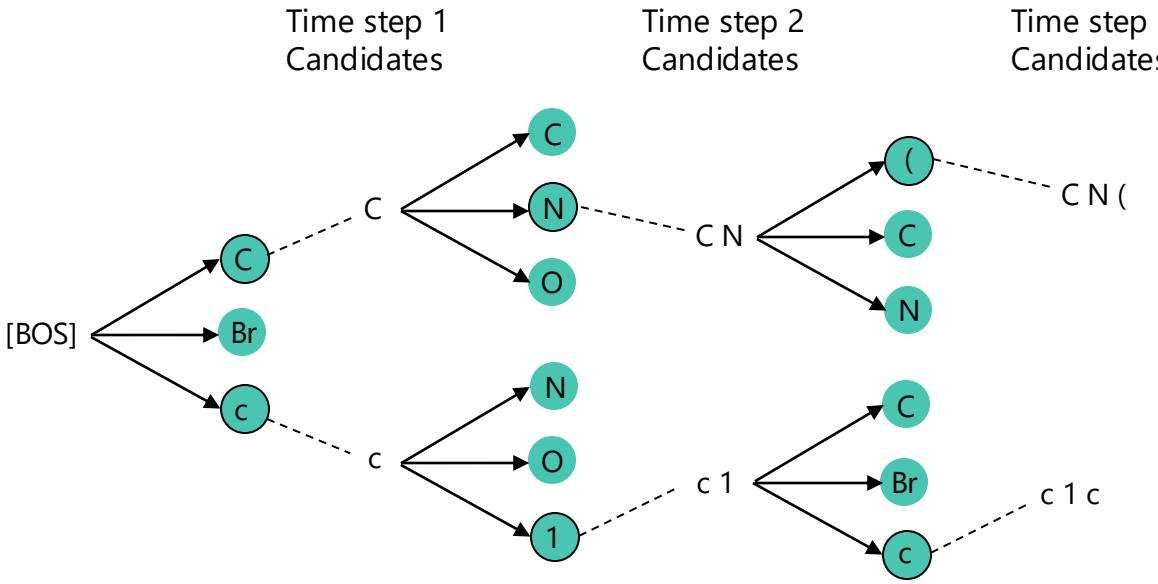
2. Model architecture



Approach #1: end-to-end Transformer

aka.ms/RetroChimeraNeurIPS

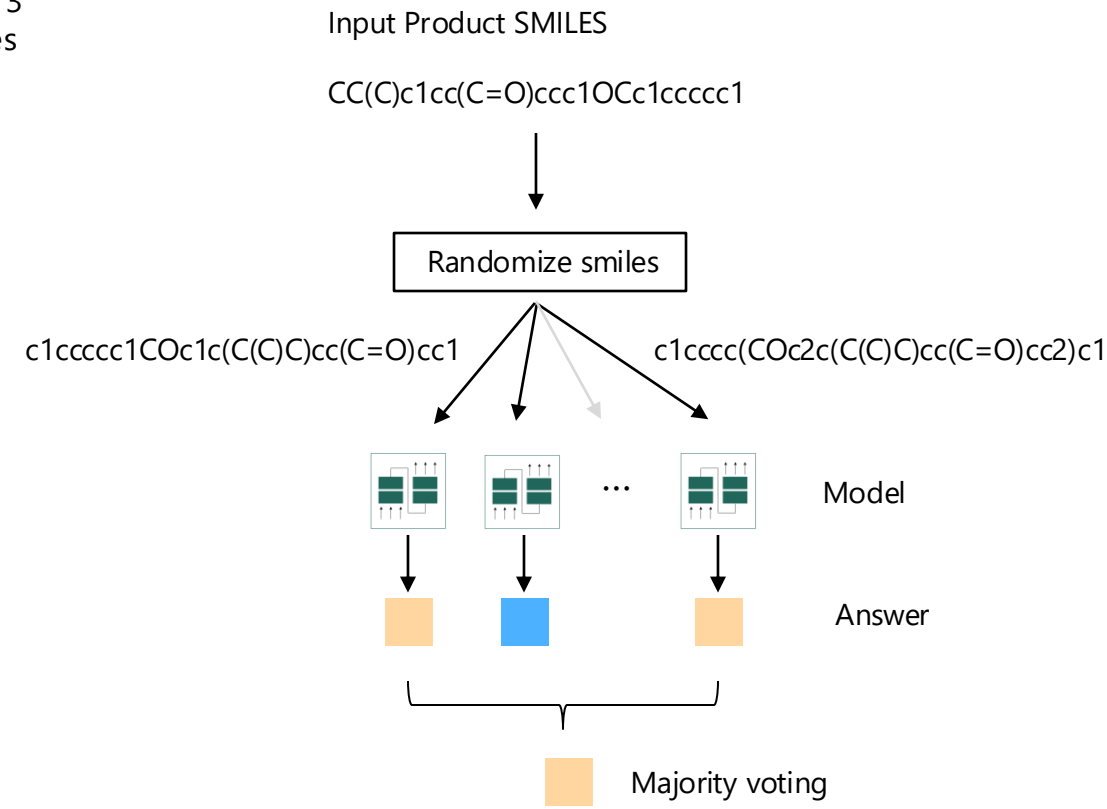
3. Optimized decoding for retrosynthesis prediction



Beam Search Strategy 1 (utilized by OpenNMT): Maintain a pool to save ended sequences during search, search are done until two conditions are met: 1) the pool size = beam size; 2) the top-rated sequence in the beam has lower probability than all candidates in the pool

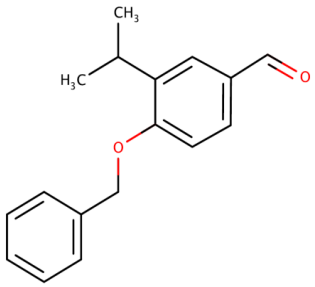
Beam Search Strategy 2: Keep ended sequences within the beam itself, search are done when each sequence in the beam finishes with the EOS token

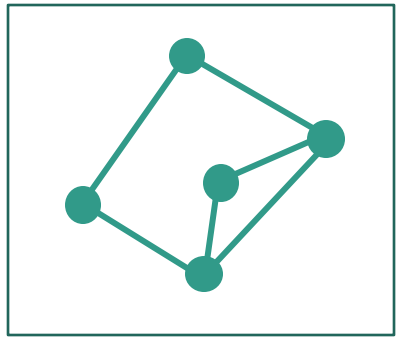
4. Test-time augmentation



Approach #2: template-based GNN

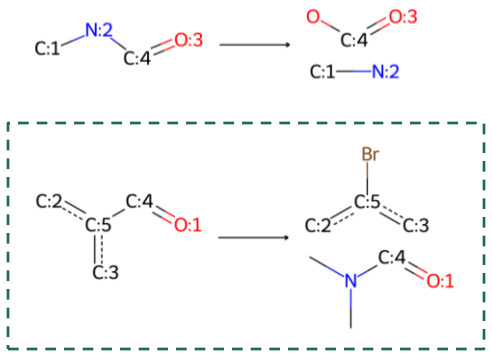
aka.ms/RetroChimeraNeurIPS

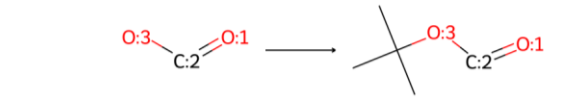


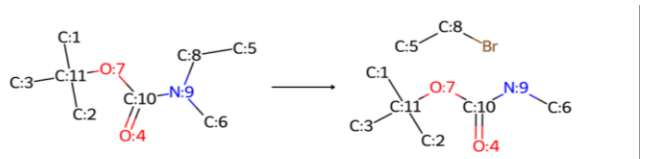


GNN Encoder

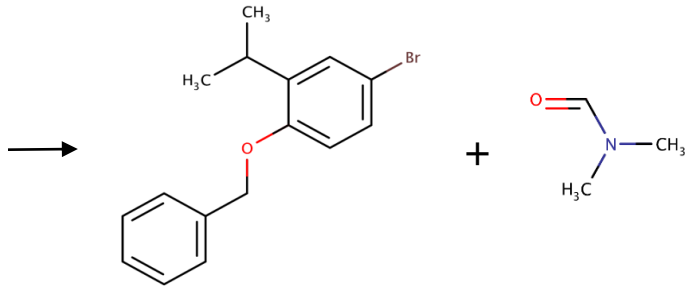
Template (graph rewrite) library





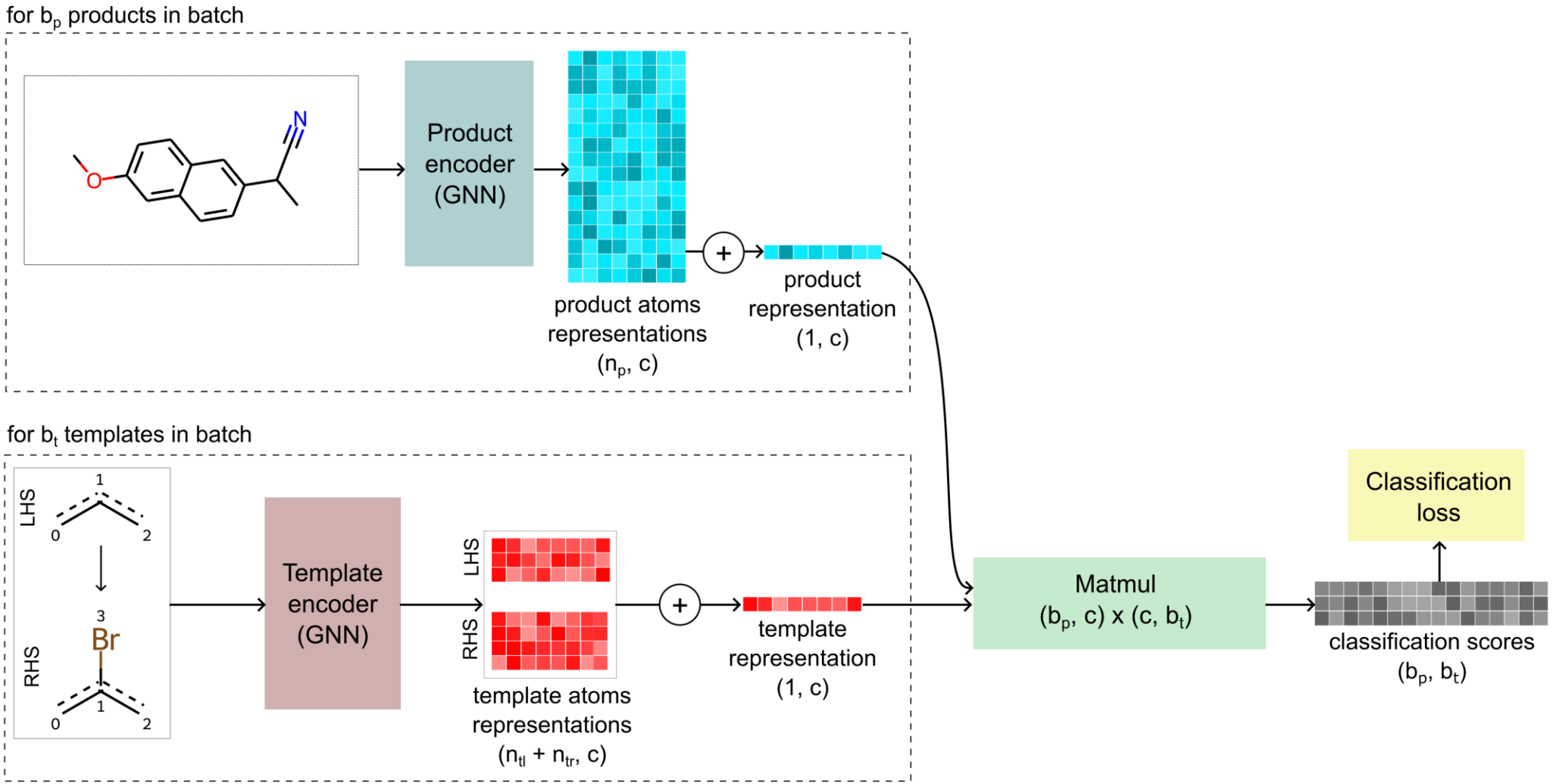


...



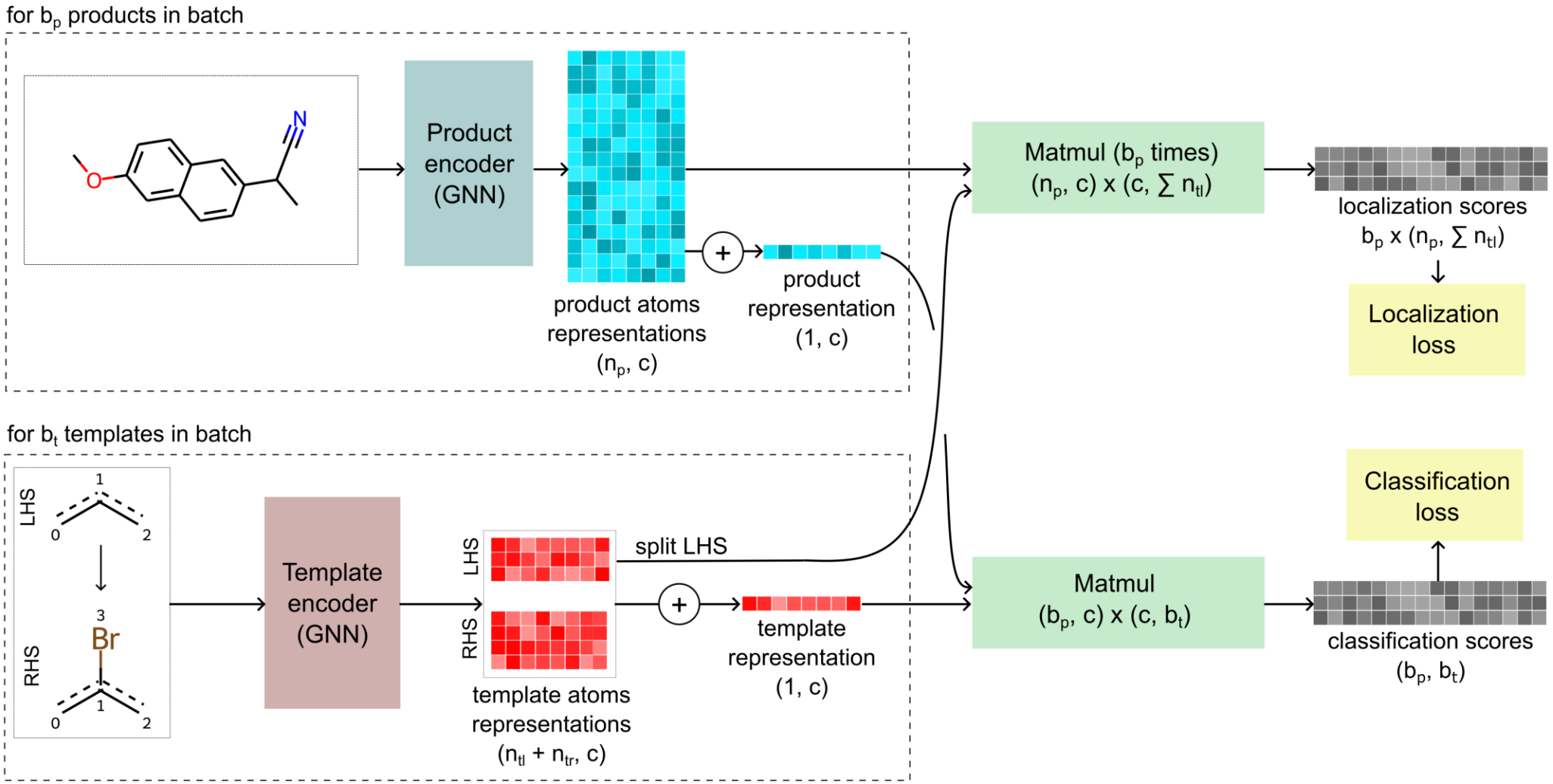
Approach #2: template-based GNN

aka.ms/RetroChimeraNeurIPS



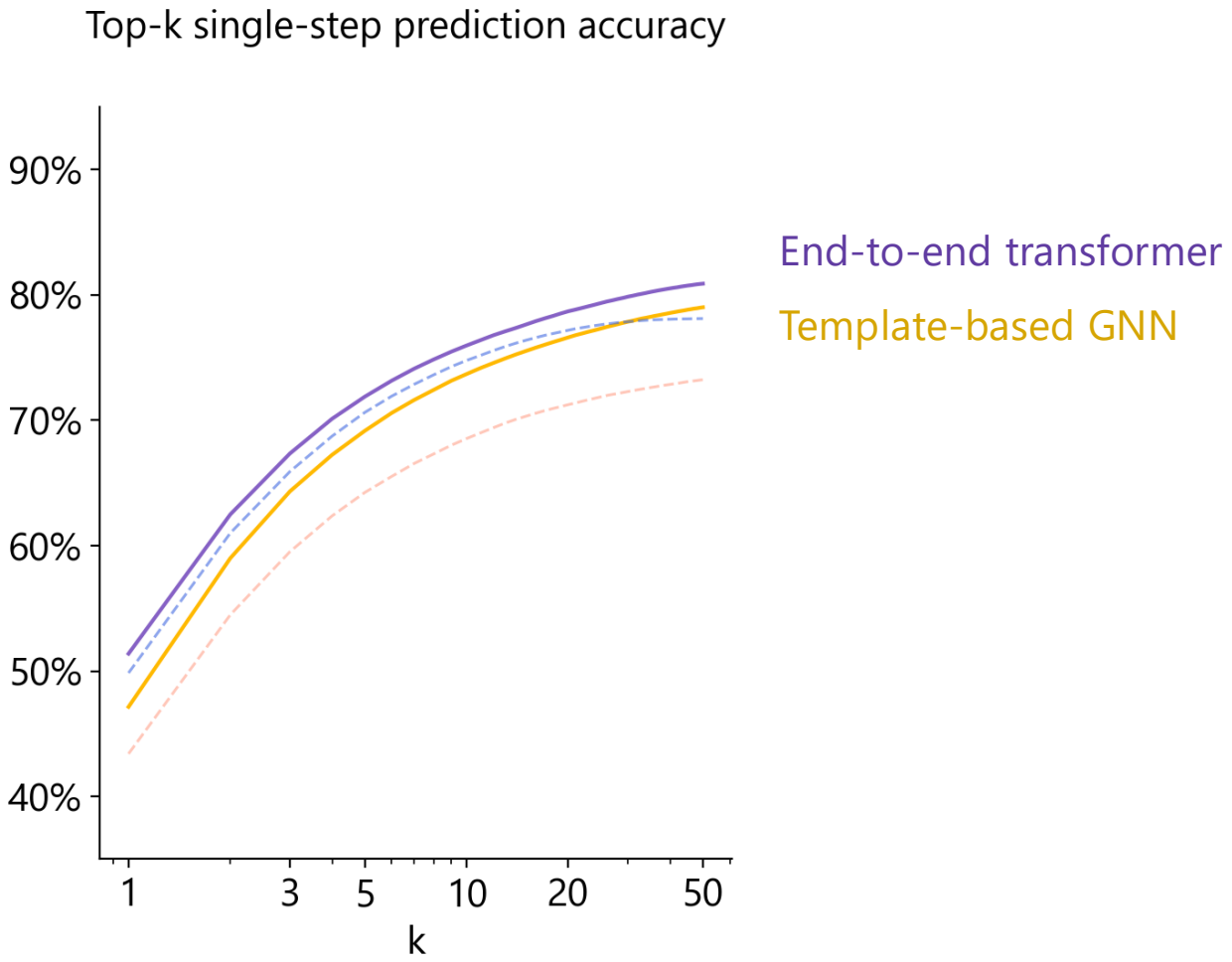
Approach #2: template-based GNN

aka.ms/RetroChimeraNeurIPS



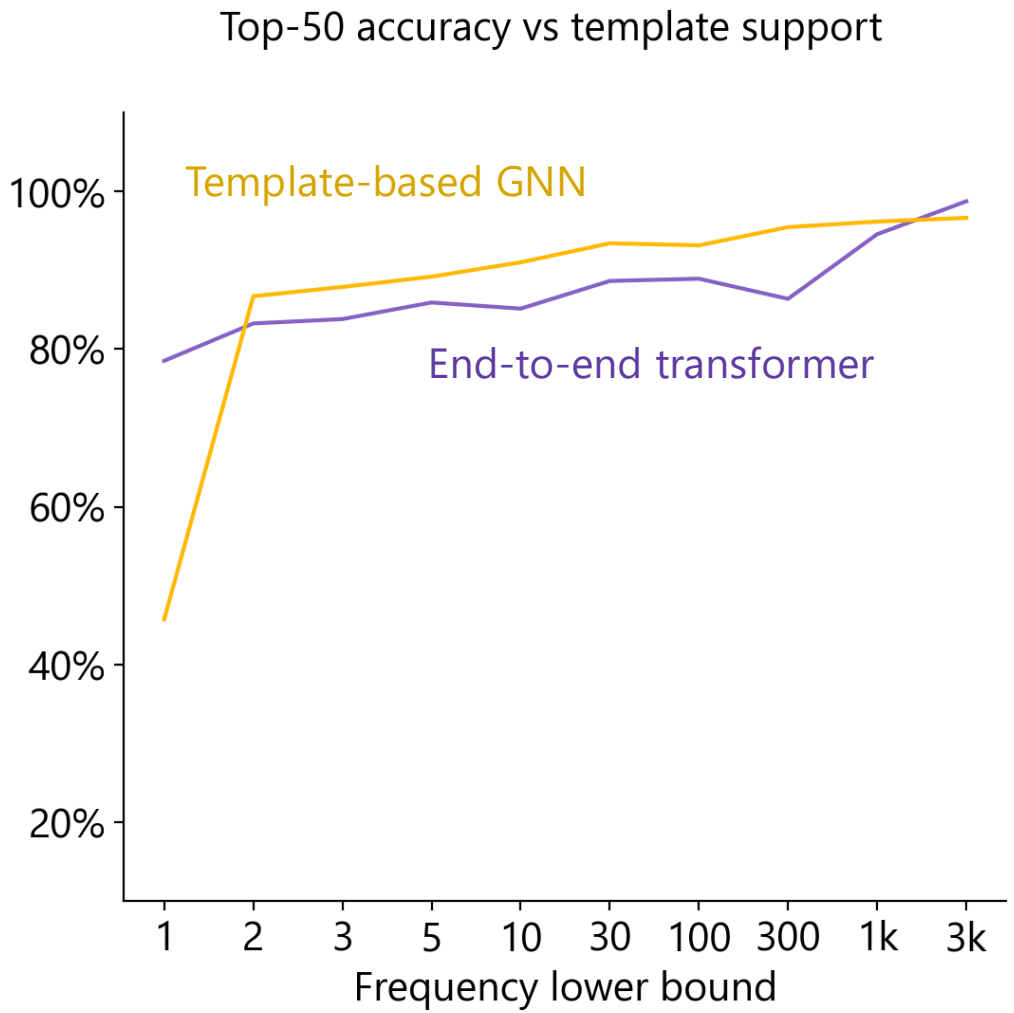
Comparison

aka.ms/RetroChimeraNeurIPS



Comparison: trade-offs

aka.ms/RetroChimeraNeurIPS



Microsoft

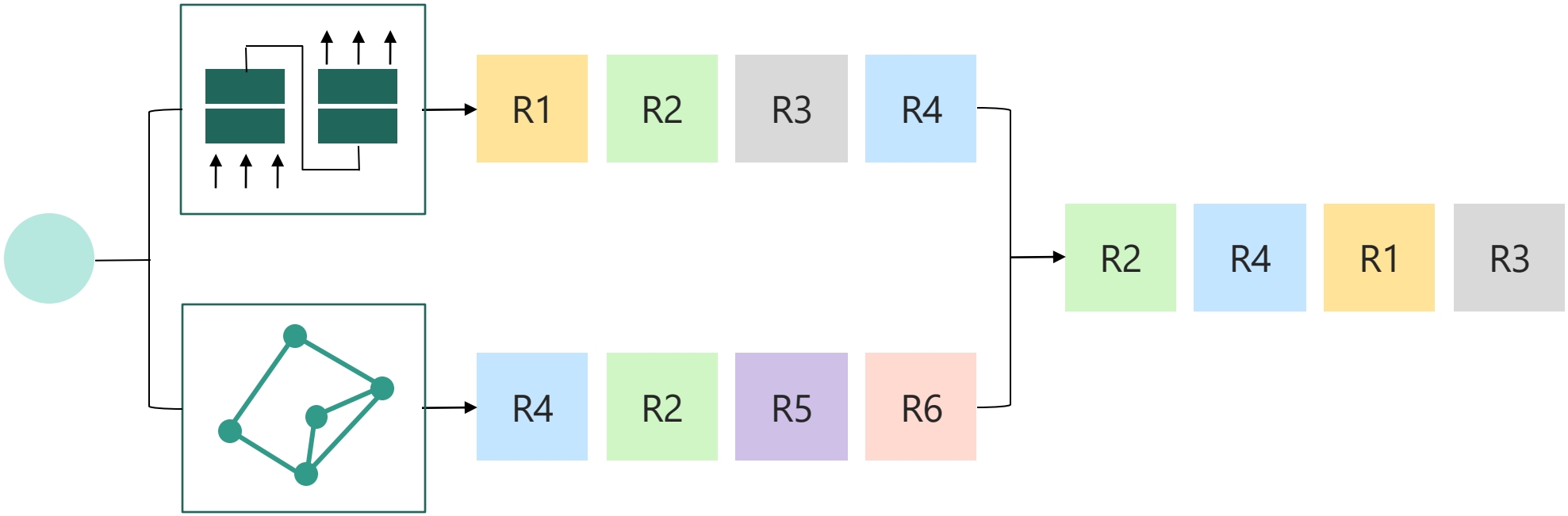
aka.ms/RetroChimeraNeurIPS

How does RetroChimera fuse results from its submodels?

Microsoft Research

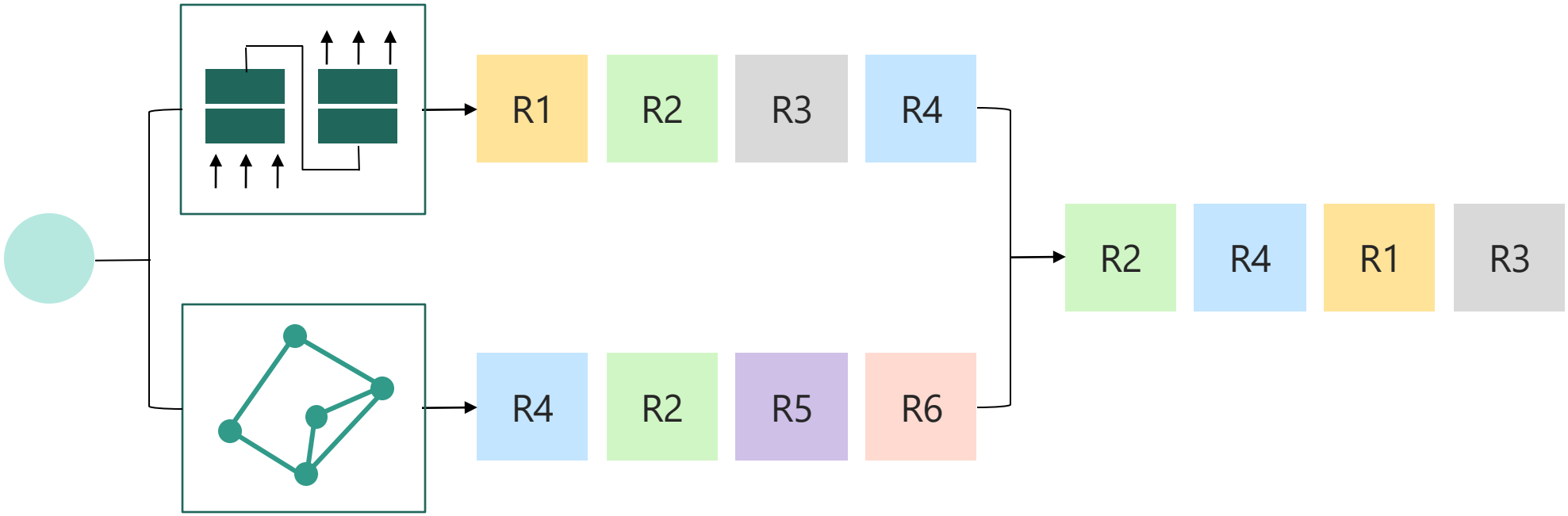
Learned ensembling

aka.ms/RetroChimeraNeurIPS



Learned ensembling

aka.ms/RetroChimeraNeurIPS



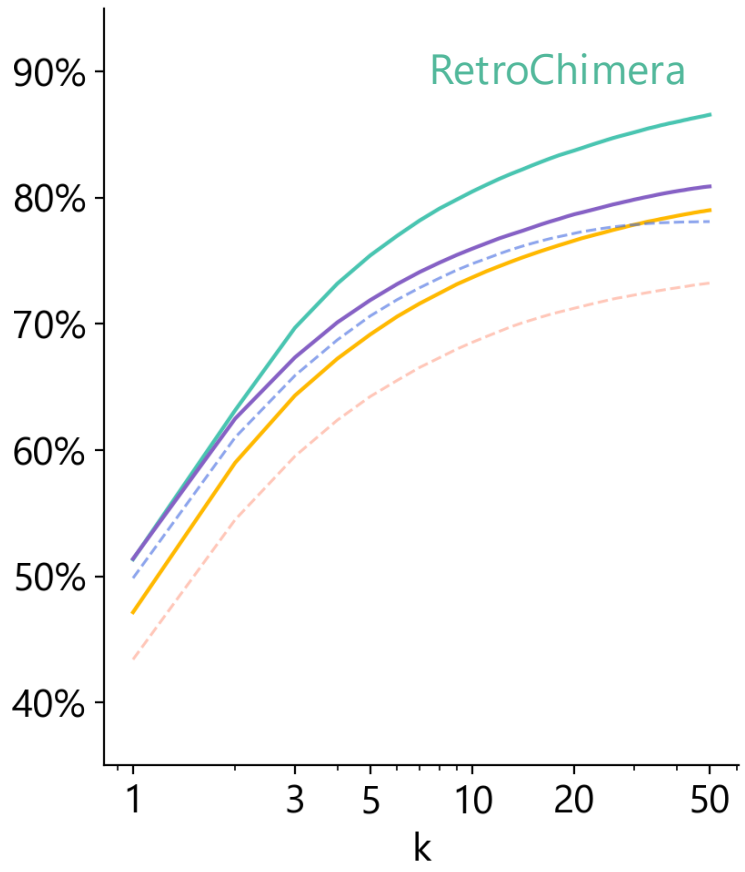
$$\texttt{score}(r) = \sum_{i=1}^{m} \sum_{k=1}^{k_{max}} \mathbb{1}[r = r_{i,k}] \cdot \theta_{i,k}$$

$$\mathcal{L}_{rank}(r^+, r^-) = \sigma\left(\frac{\texttt{score}(r^-) - \texttt{score}(r^+) + \epsilon}{T}\right)$$

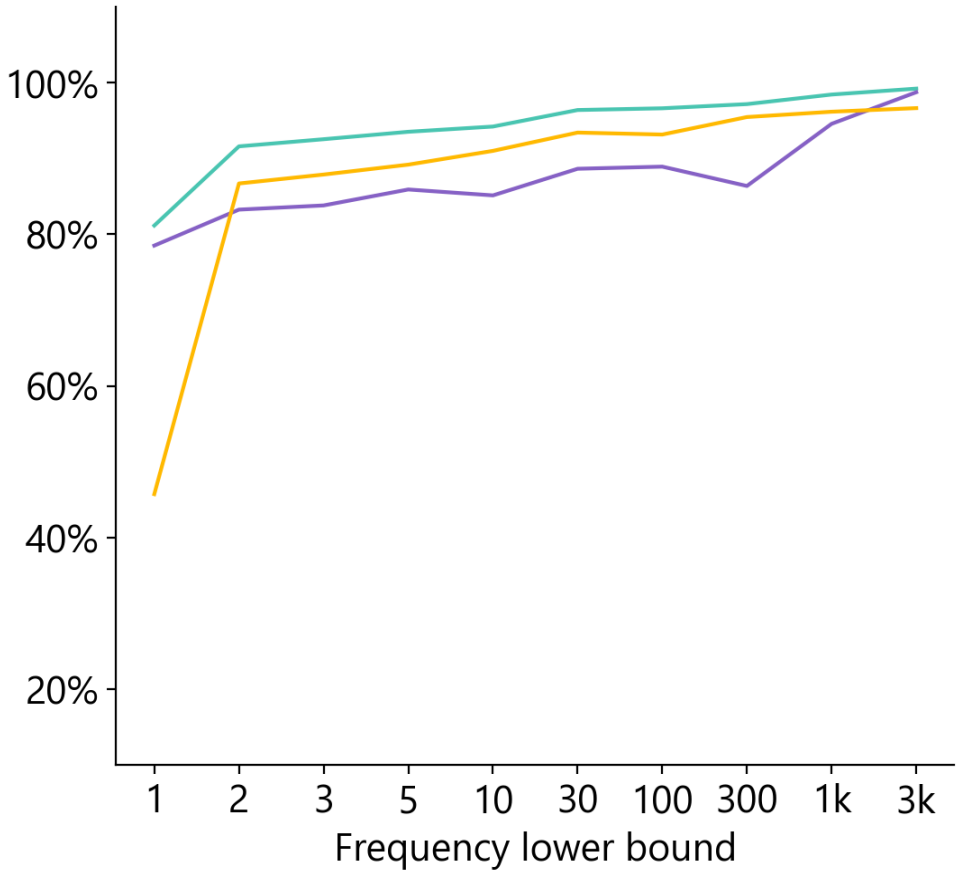
RetroChimera's performance

aka.ms/RetroChimeraNeurIPS

Top-k single-step prediction accuracy

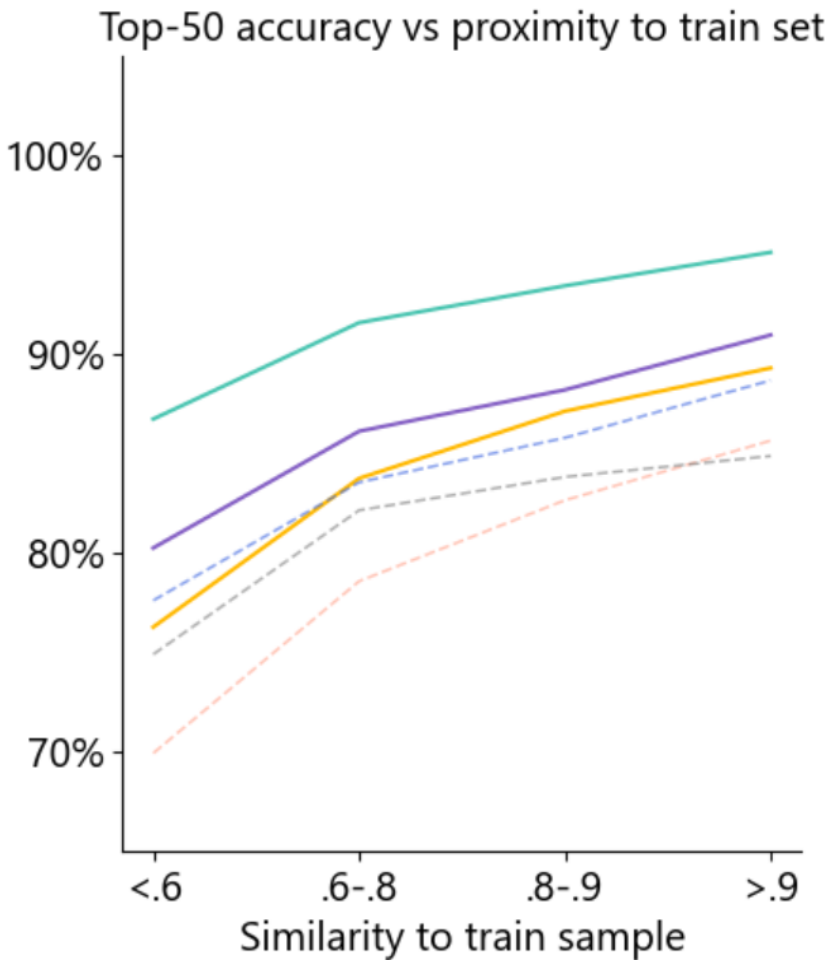


Top-50 accuracy vs template support



Performance is robust OOD

aka.ms/RetroChimeraNeurIPS



Microsoft

aka.ms/RetroChimeraNeurIPS

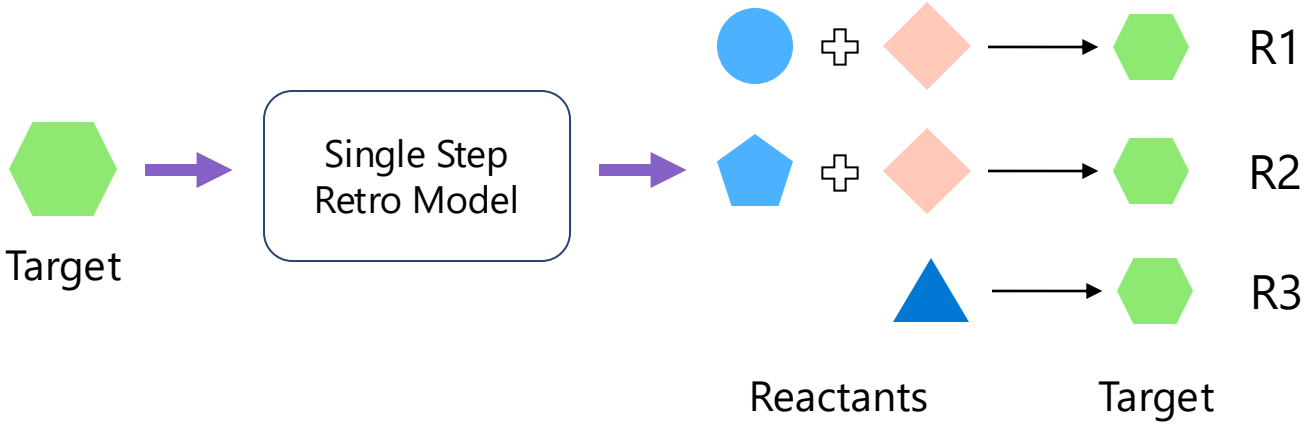
How to combine individual reactions into full plans?

Microsoft Research

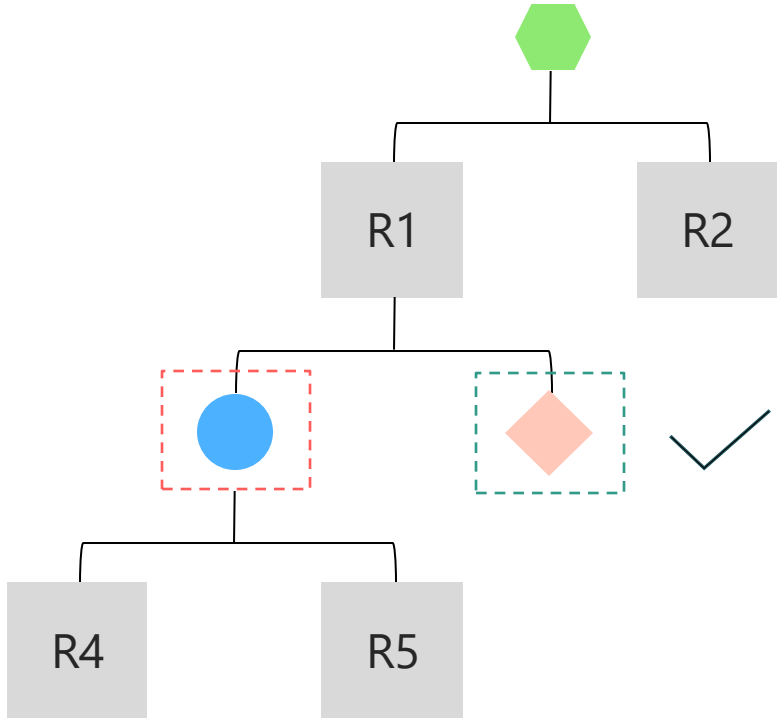
Synthesis prediction models and search

aka.ms/RetroChimeraNeurIPS

(1) Given target, predict different possible reactants



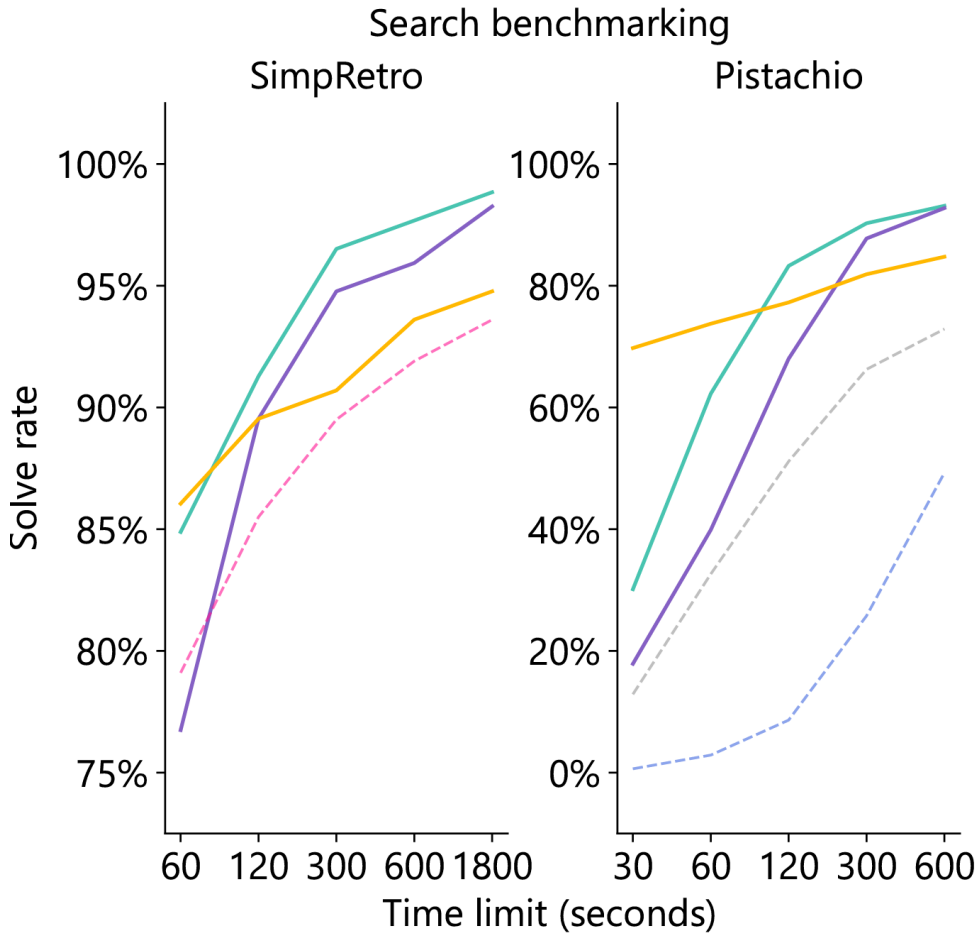
(2) Apply model recursively to obtain routes



(2) Maziarz et al, "Re-evaluating Retrosynthesis Algorithms with Syntheseus", Faraday Discuss. 2024
Liu et al, "Retrosynthetic Planning with Dual Value Networks", ICML 2023

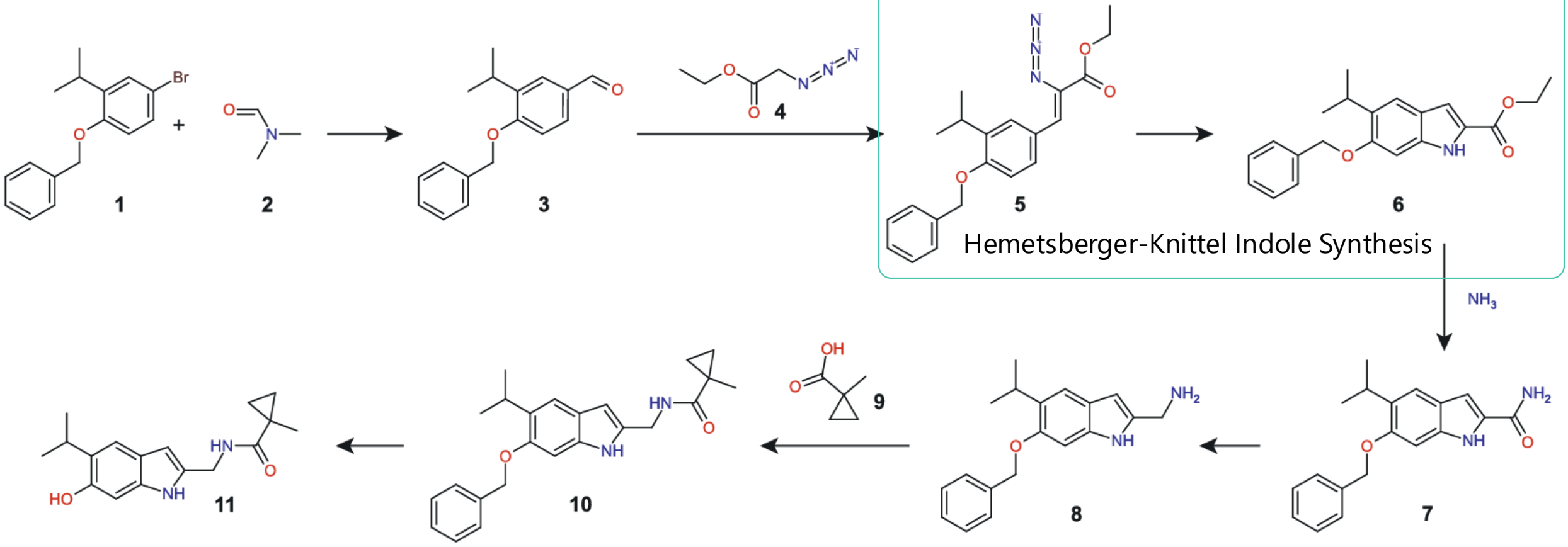
Search with RetroChimera

aka.ms/RetroChimeraNeurIPS



aka.ms/RetroChimeraNeurIPS

Example synthesis plan



RetroChimera can find non-trivial routes

Microsoft

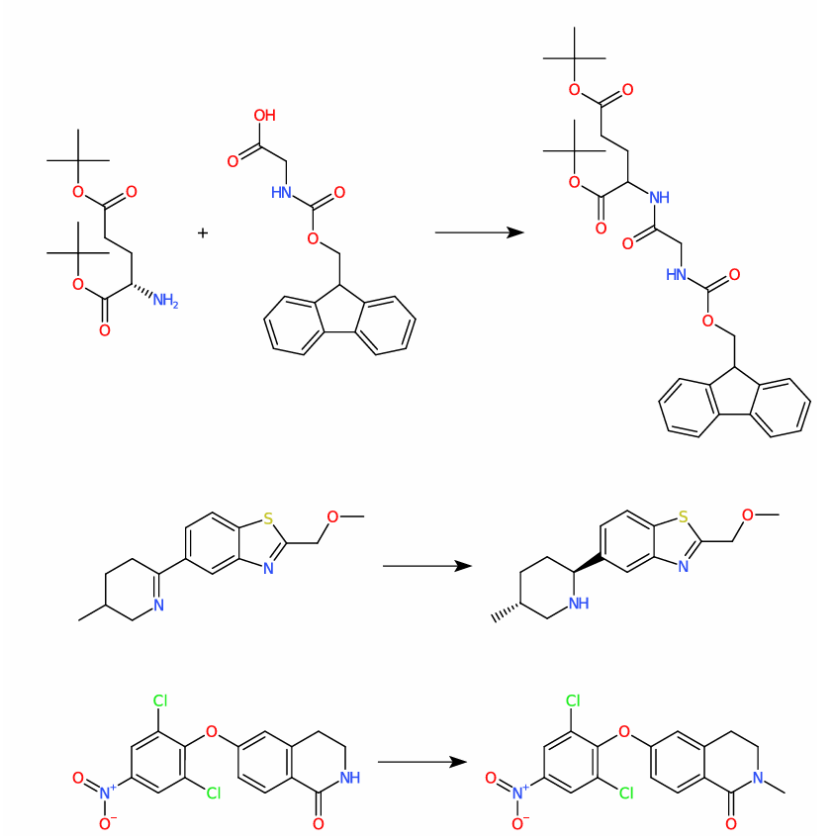
aka.ms/RetroChimeraNeurIPS

Exploring RetroChimera's predictions qualitatively

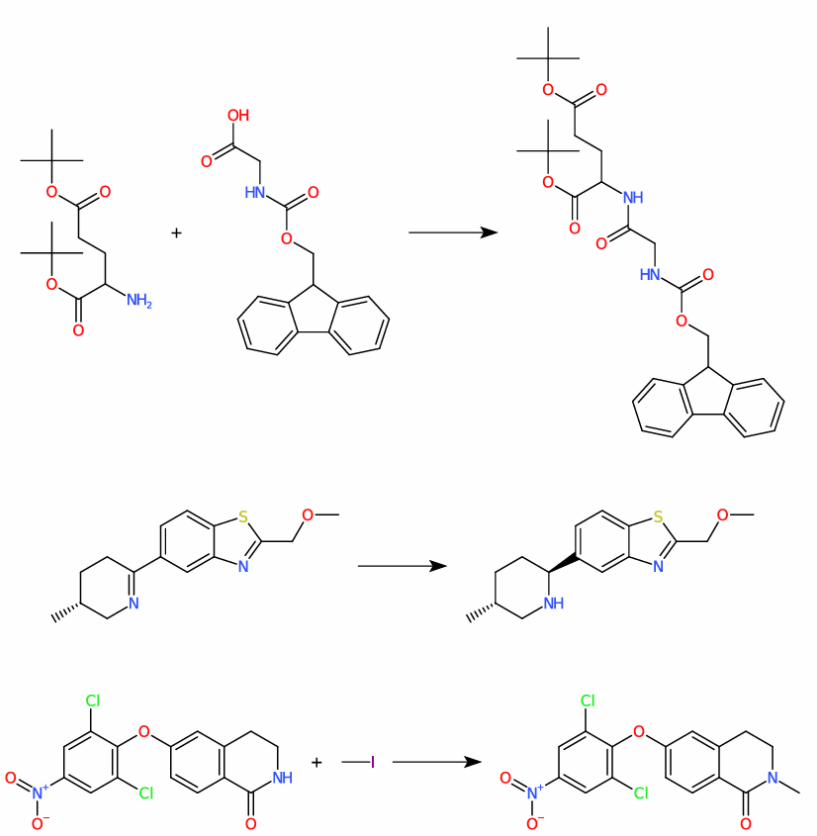
Microsoft Research

Denoising

aka.ms/RetroChimeraNeurIPS



original patent reaction



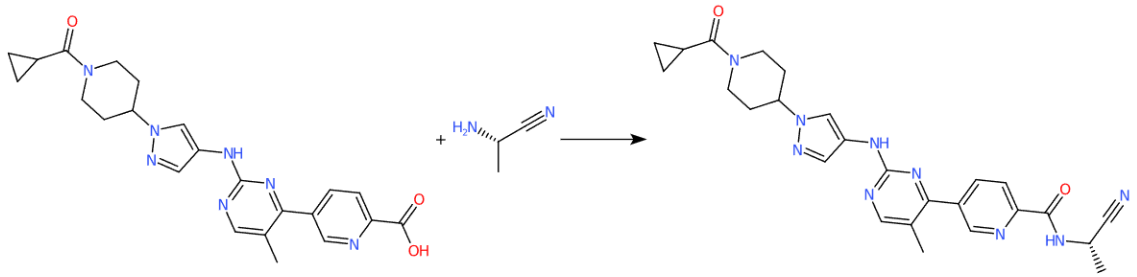
denoised by RetroChimera

Pairwise (blind) expert comparisons

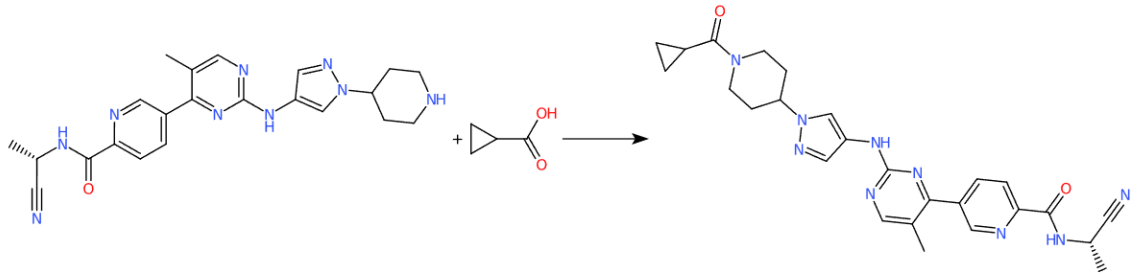
aka.ms/RetroChimeraNeurIPS

Please pick your preferred reaction.

Model 1:



Model 2:



Model 1 better

Model 2 better

Both good

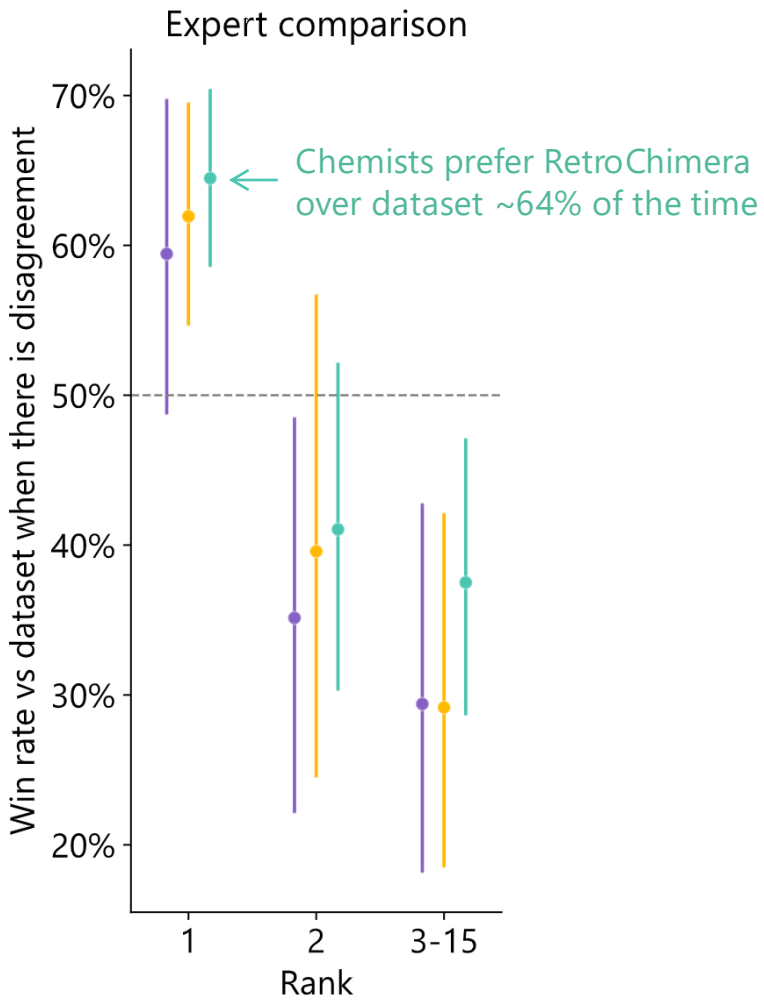
Both bad

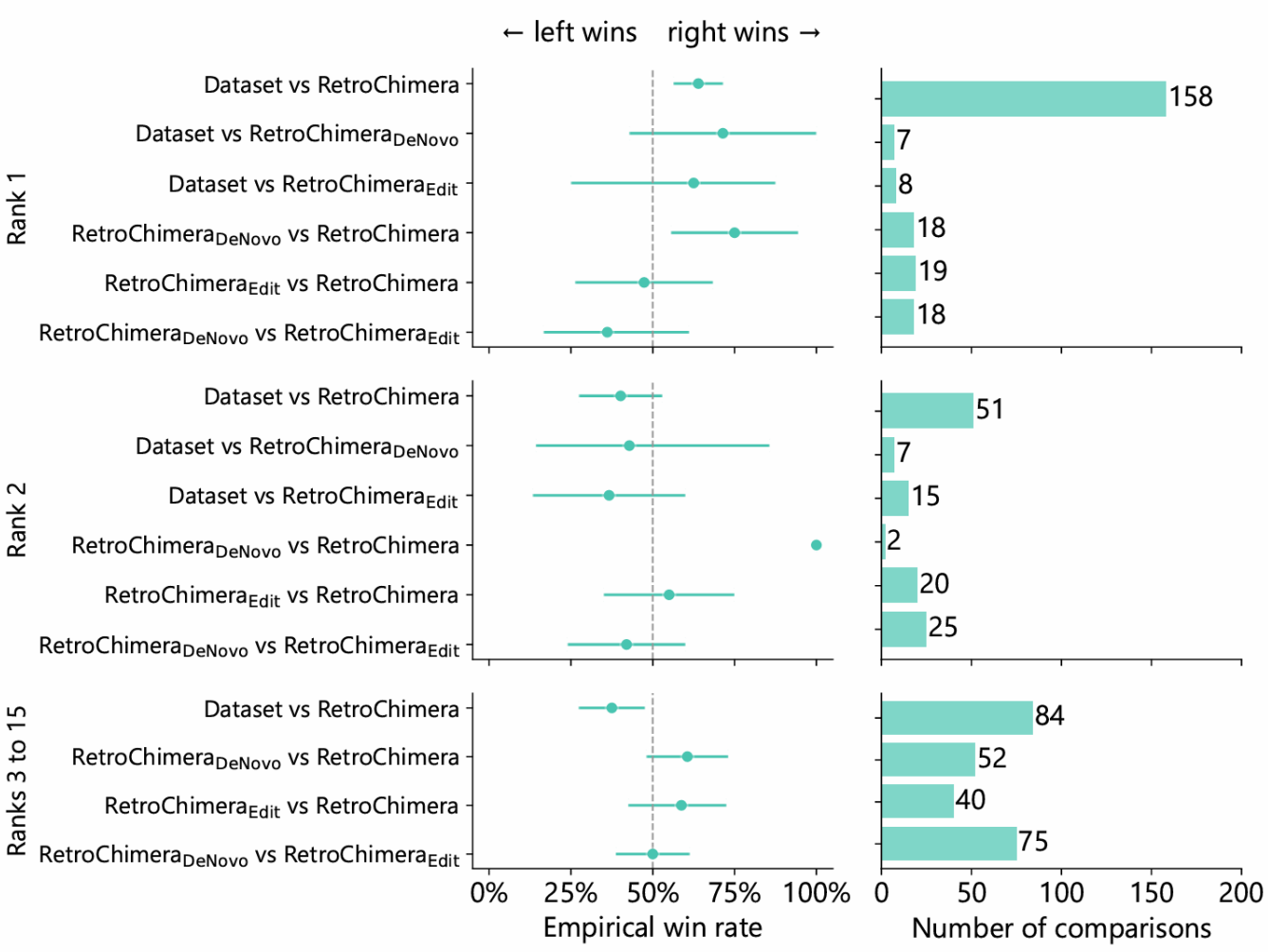
Sample ID: ed2e9b78-bed5-43a5-b399-928248f08c2f

Experiment ID: 12

Preferences in pairwise expert comparisons

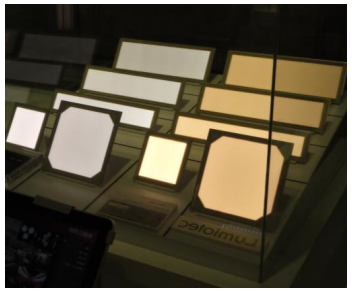
aka.ms/RetroChimeraNeurIPS





Questions

Our goal: Help chemists discover new essential molecules with predictive synthesis



aka.ms/RetroChimeraNeurIPS

`aka.ms/RetroChimeraPaper`

`aka.ms/RetroChimeraCode` (github.com/microsoft/retrochimera)

`aka.ms/CondPredPaper` (cond. pred. model QFANG, coming soon)

`github.com/microsoft/syntheseus`