RetCL: A Selection-based Approach for Retrosynthesis via Contrastive Learning

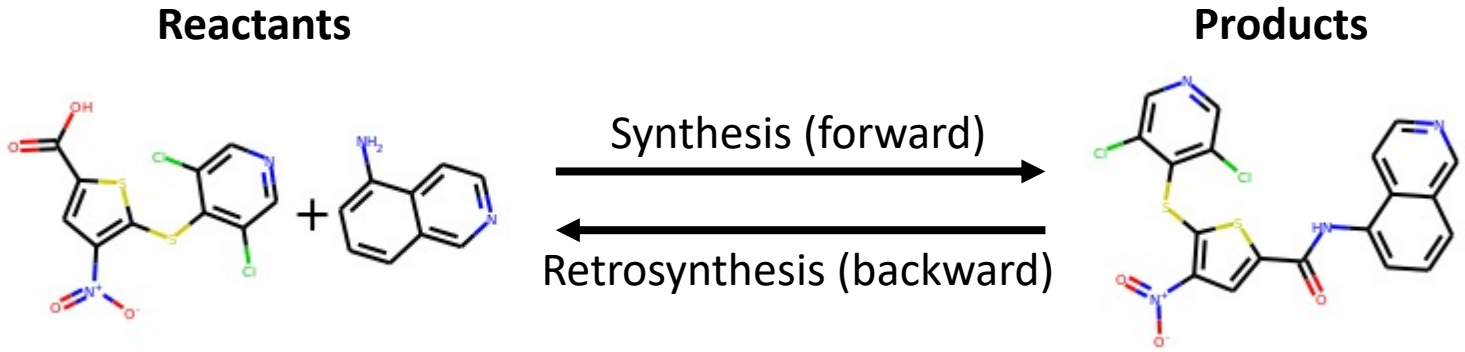
August, 2021.

Hankook Lee[1], Sungsoo Ahn[2], Seung-Woo Seo[3], You Young Song[4], Eunho Yang[15], Sung-Ju Hwang[15], Jinwoo Shin[1]

[1]Korea Advanced Institute of Science and Technology
[2]Mohamed bin Zaeyed University of Artificial Intelligence
[3]Standigm
[4]Samsung Electronics
[5]AITRICS

What Is Retrosynthesis?

- **Retrosynthesis** aims at finding **a synthetic route starting from commercially available reactants to synthesize a target product**

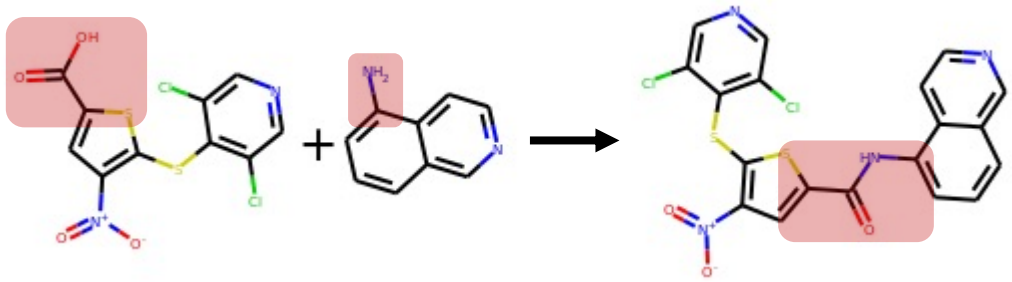


  - It plays an essential role in practical applications by finding a new synthetic path, which can be more cost-effective or avoid patent infringement

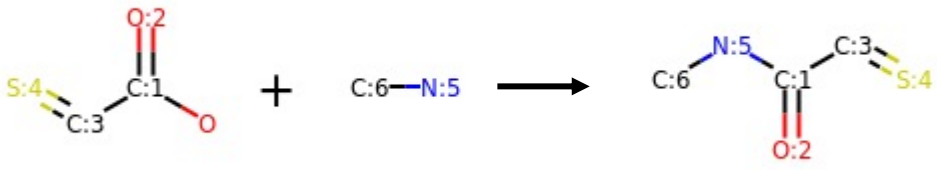
- **Challenges:**
  - One molecule could be synthesized by different combinations of reactants
  - Some complex compounds require more than 100 synthesis steps
  - The number of reaction types (or rules) is very huge
  - Hence, the **search space is too vast**

Existing Approaches for Retrosynthesis

1. Template-based approaches [1-3]
   - A reaction template describes how the chemical reaction occurs among reactants
     - Reaction templates can be extracted from a reaction database automatically or encoded by experts



Reaction



Reaction template

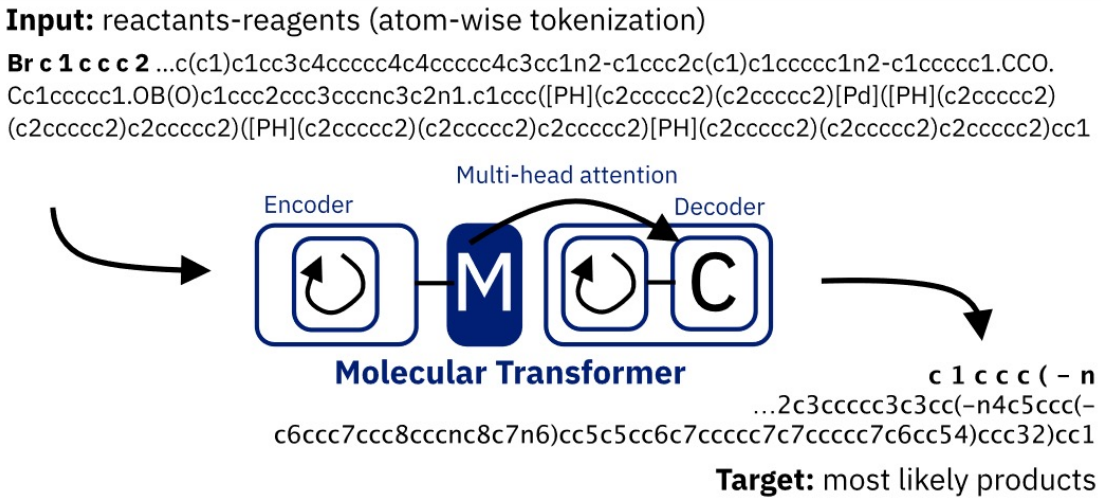
   - How do template-based approaches perform retrosynthesis?
     1. Construct a set of templates $\mathcal{T} = \{T_1, T_2, \ldots, T_{|\mathcal{T}|}\}$ by automatic tools or experts
     2. Given a product molecule P, find a well-matched template $T \in \mathcal{T}$
     3. Obtain a set of reactants by applying the template T to the product P

   - **Limitation:** they limit the search space to known templates and **cannot discover novel synthetic routes**

[1] Coley et al., Computer-assisted retrosynthesis based on molecular similarity. ACS central science, 3(12):1237–1245, 2017.
[2] Segler & Waller, Neural-symbolic machine learning for retrosynthesis and reaction prediction. Chemistry–A European Journal, 23(25):5966–5971, 2017.
[3] Dai et al., Retrosynthesis prediction with conditional graph logic network, NeurIPS, 2019.

Existing Approaches for Retrosynthesis

2. Template-free approaches [1-4]
   - They generate the reactants from scratch without knowledge of reaction templates
     - In other words, they consider retrosynthesis as a conditional generation problem such as **machine translation**
     - **Note.** Molecules can be encoded by graph or string format (SMILES)



   - **Limitation:** they require to search the entire molecular space, and **their predictions could be either unstable or commercially unavailable**

[1] Liu et al., Retrosynthetic reaction prediction using neural sequence-to-sequence models. ACS central science, 3(10): 1103–1113, 2017
[2] Karpov et al., A transformer model for retrosynthesis. In International Conference on Artificial Neural Networks, pp. 817–830. Springer, 2019.
[3] Zheng et al., Predicting retrosynthetic reactions using self-corrected transformer neural networks. Journal of Chemical Information and Modeling, 2019.
[4] A graph to graphs framework for retrosynthesis prediction. ICML, 2020.

Proposed Method: Selection-based Approach

- **Recall.** Existing approaches have fundamental limitations:
  - Template-based ones cannot generalize to unseen templates
  - Template-free ones does not consider the availability of reactants

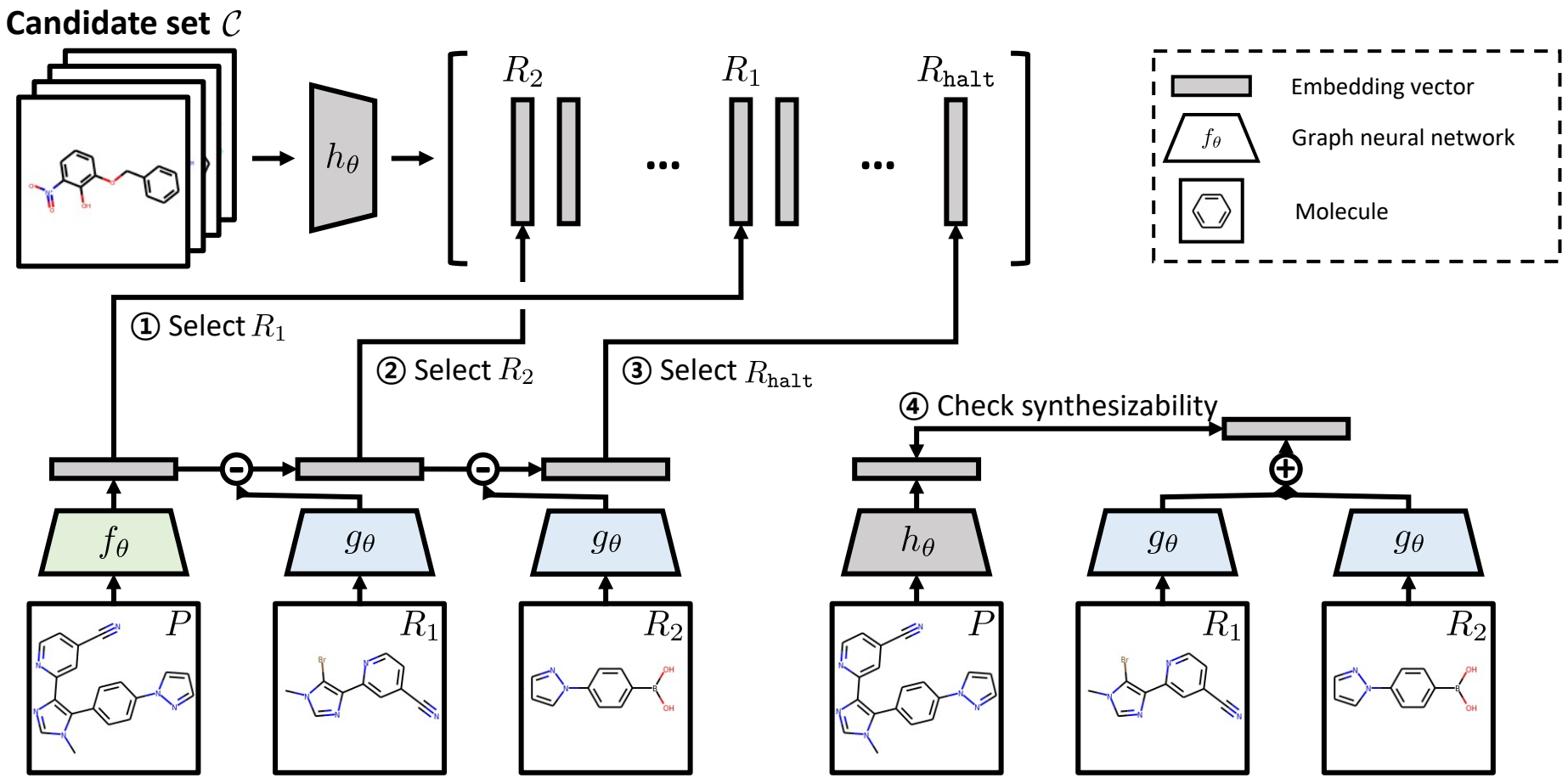
- We propose a new **selection-based** approach **considering the availability of reactants**
  - **Assumption:** we have a candidate set of commercially available reactants $\mathcal{C}$
  - We reformulate retrosynthesis as the following selection problem:

> Given a target product P, our goal is to select a set of reactants $\mathcal{R} = \{R_1, \dots, R_{|\mathcal{R}|}\}$ from the candidate set $\mathcal{C}$ (i.e., $\mathcal{R} \subset \mathcal{C}$) for synthesizing the product P

Benefits over the existing approaches:

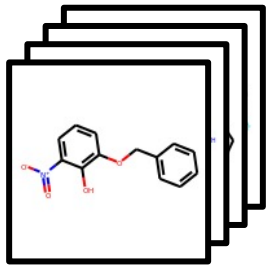
- It guarantees the **commercial availability** of the selected reactants
- It can **generalize to unseen reaction templates** and find novel synthetic routes

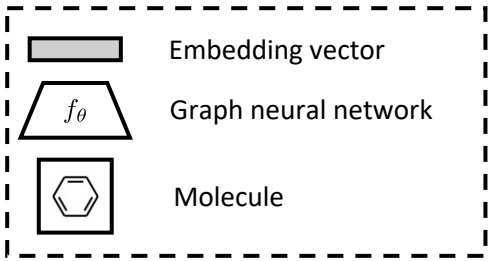
RetCL: Retrosynthesis via Contrastive Learning



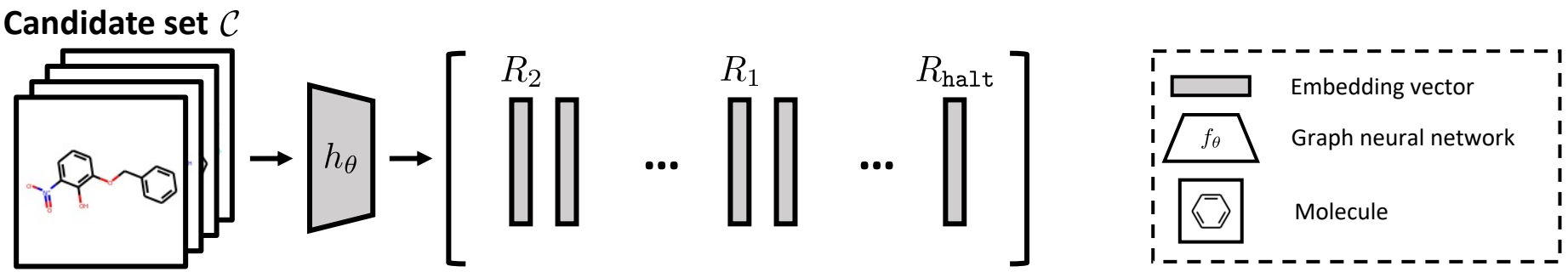
RetCL: Retrosynthesis via Contrastive Learning

Candidate set $\mathcal{C}$

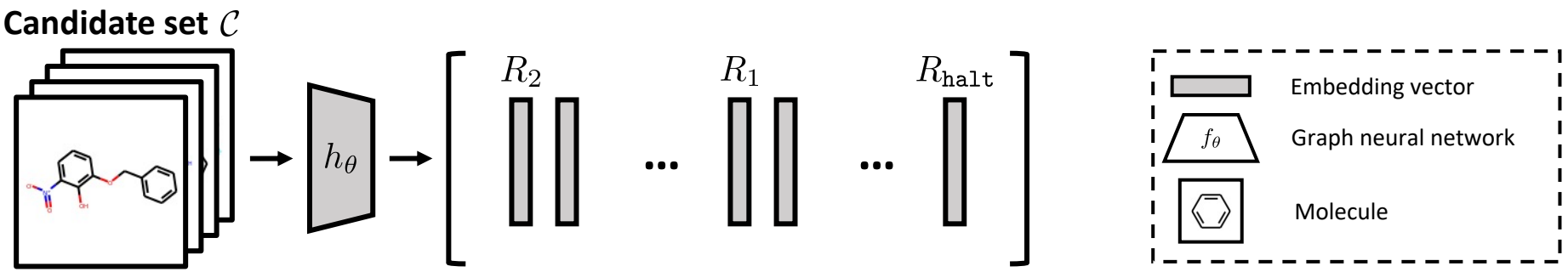


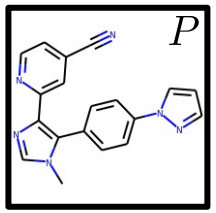


RetCL: Retrosynthesis via Contrastive Learning

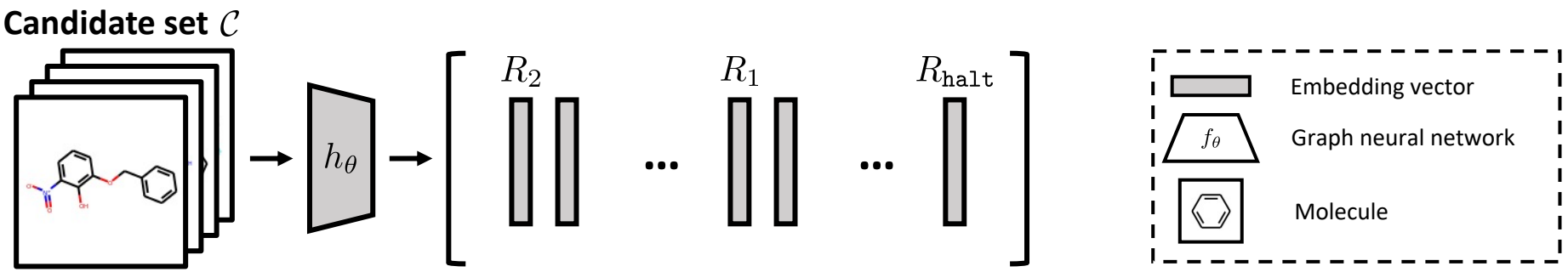


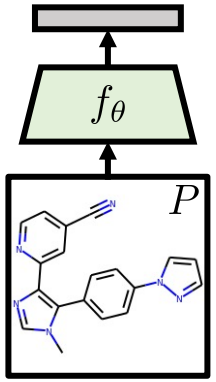
RetCL: Retrosynthesis via Contrastive Learning





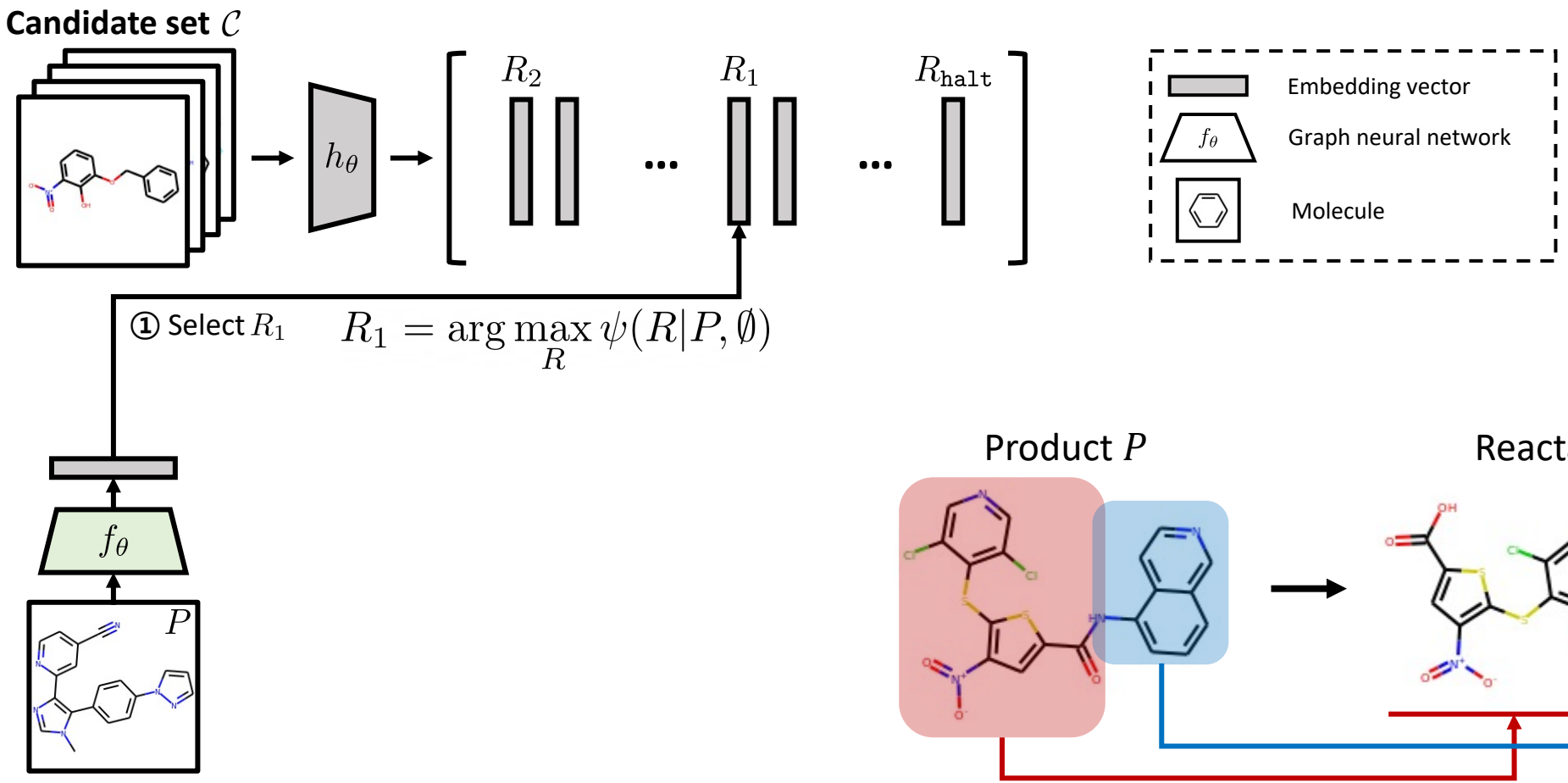
RetCL: Retrosynthesis via Contrastive Learning





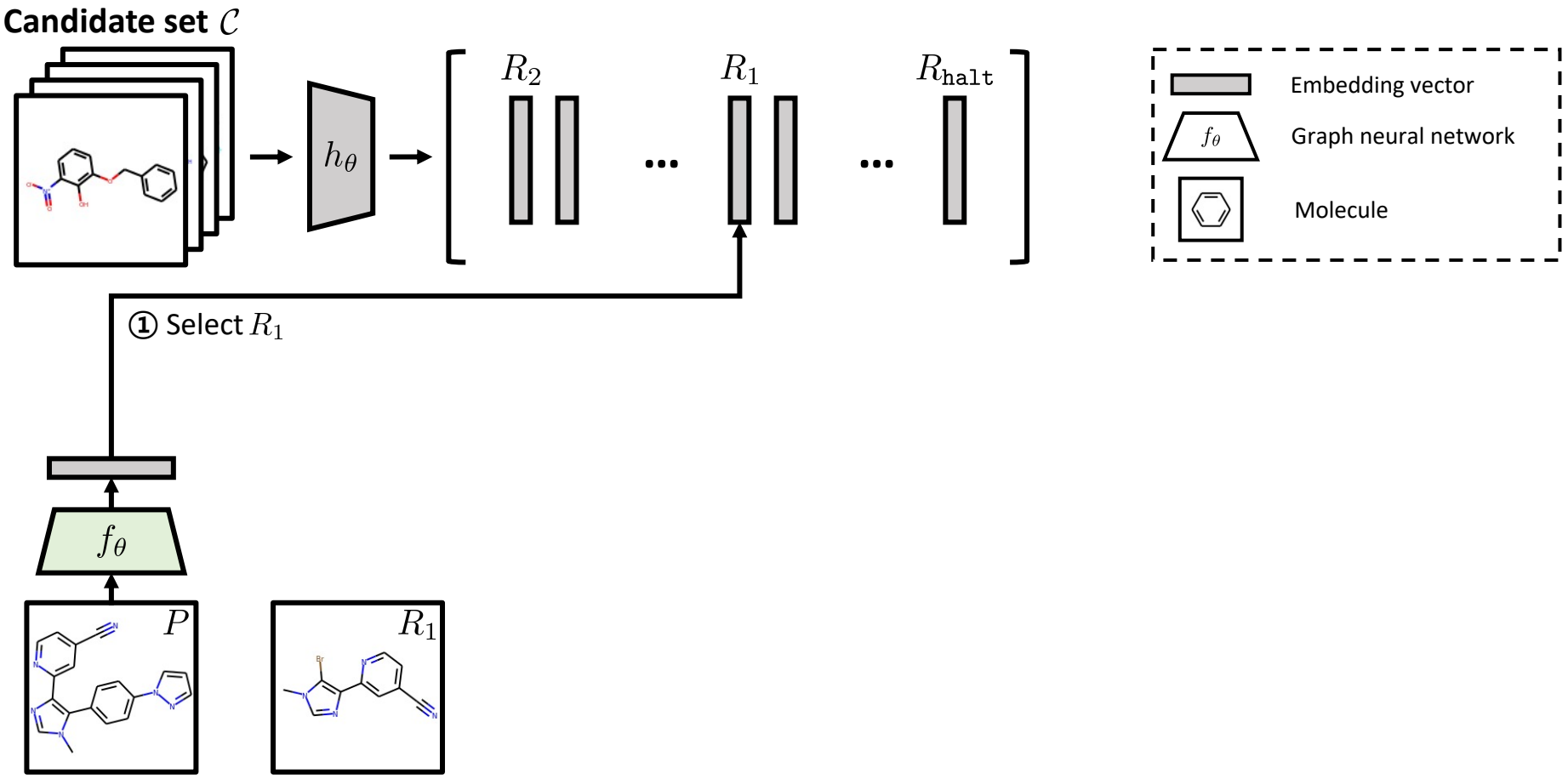
RetCL: Retrosynthesis via Contrastive Learning

Backward Selection Score: $\psi(R|P, \mathcal{R}_{\texttt{given}}) = \texttt{CosSim}\left(f_\theta(P) - \sum_{S \in \mathcal{R}_{\texttt{given}}} g_\theta(S),\ h_\theta(R)\right)$



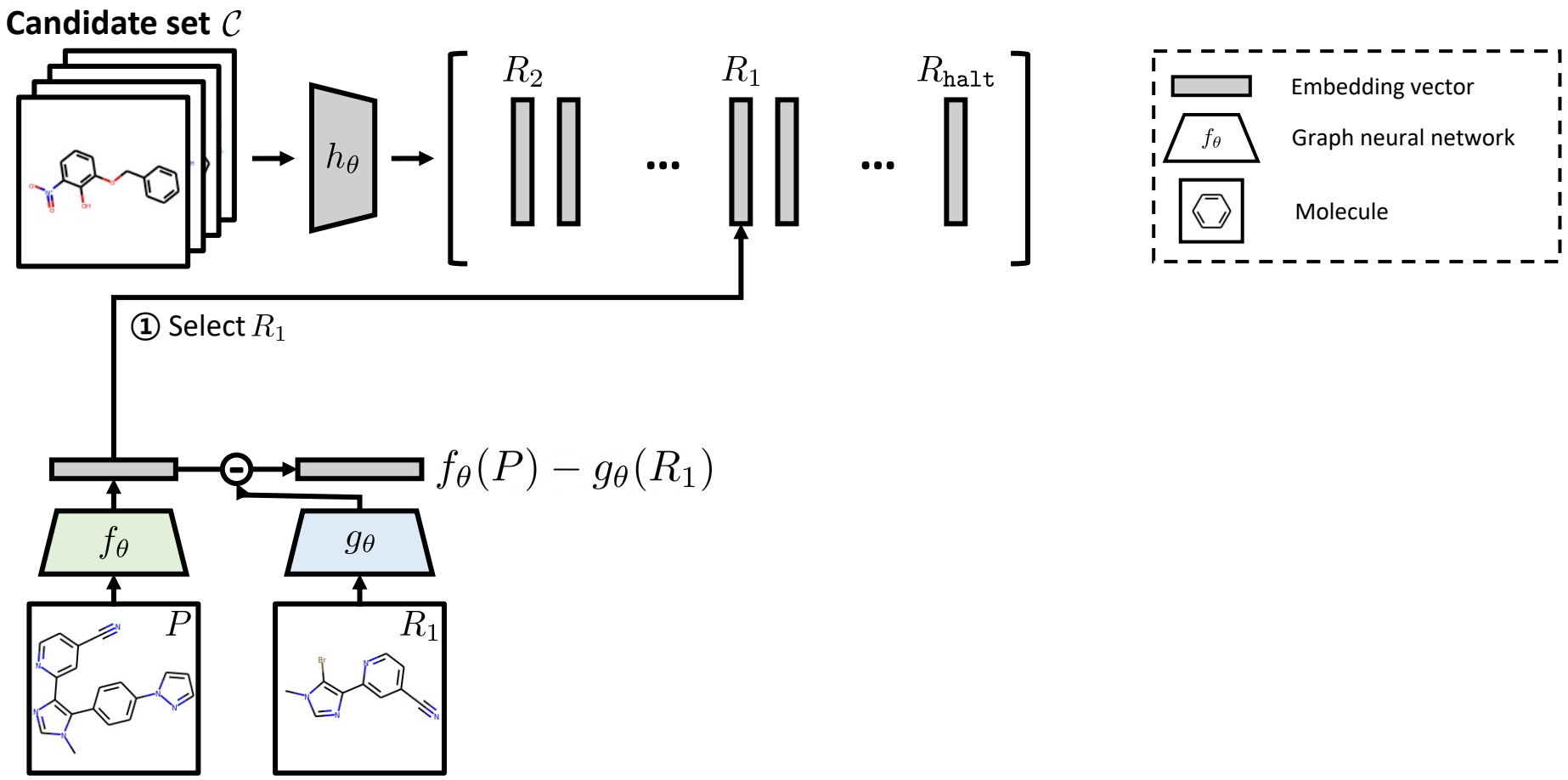
RetCL: Retrosynthesis via Contrastive Learning

Backward Selection Score: $\psi(R|P, \mathcal{R}_{\texttt{given}}) = \texttt{CosSim}\left(f_\theta(P) - \sum_{S \in \mathcal{R}_{\texttt{given}}} g_\theta(S),\ h_\theta(R)\right)$



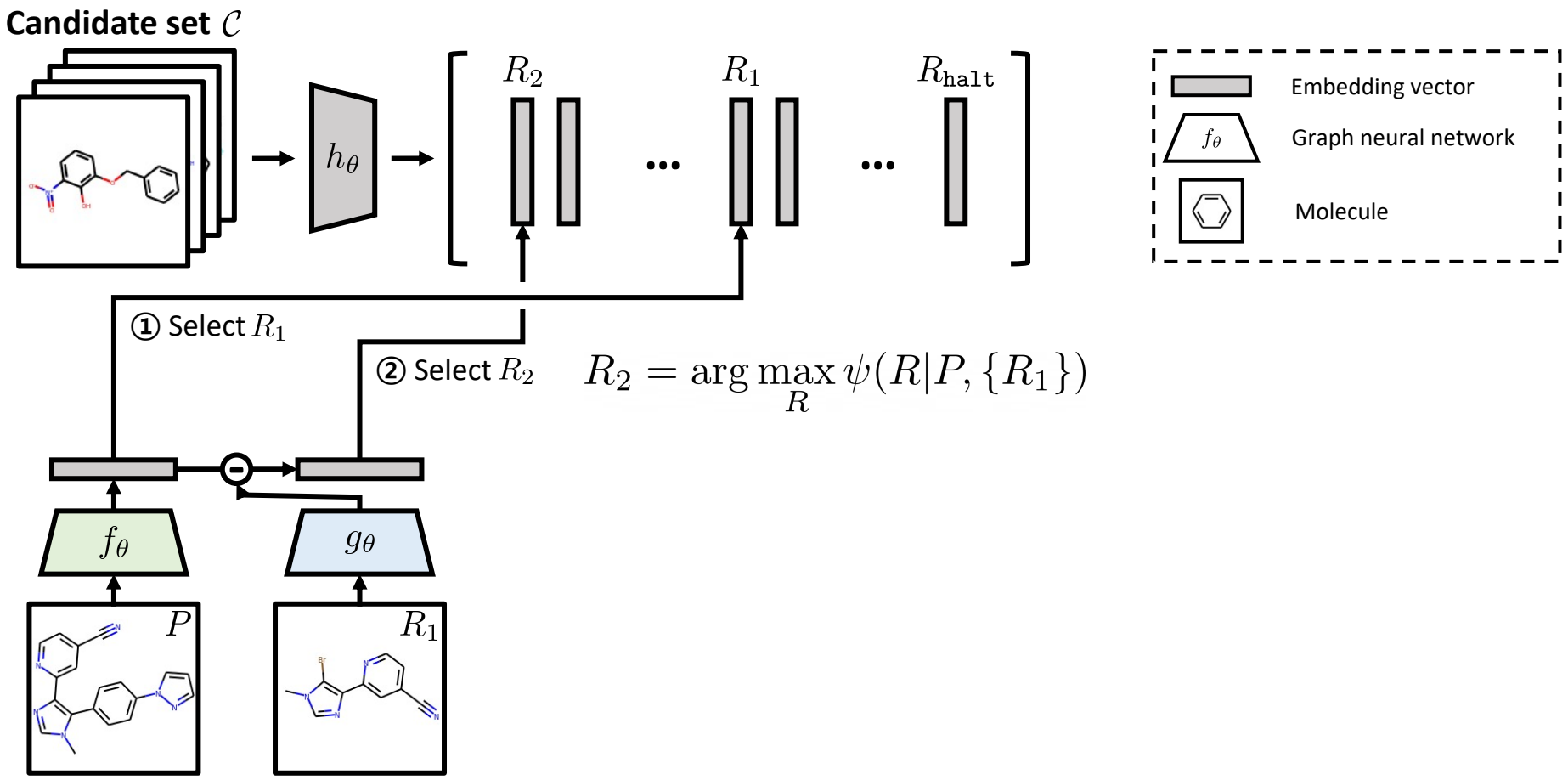
RetCL: Retrosynthesis via Contrastive Learning

Backward Selection Score: $\psi(R|P, \mathcal{R}_{\texttt{given}}) = \texttt{CosSim}\left(f_\theta(P) - \sum_{S \in \mathcal{R}_{\texttt{given}}} g_\theta(S),\ h_\theta(R)\right)$



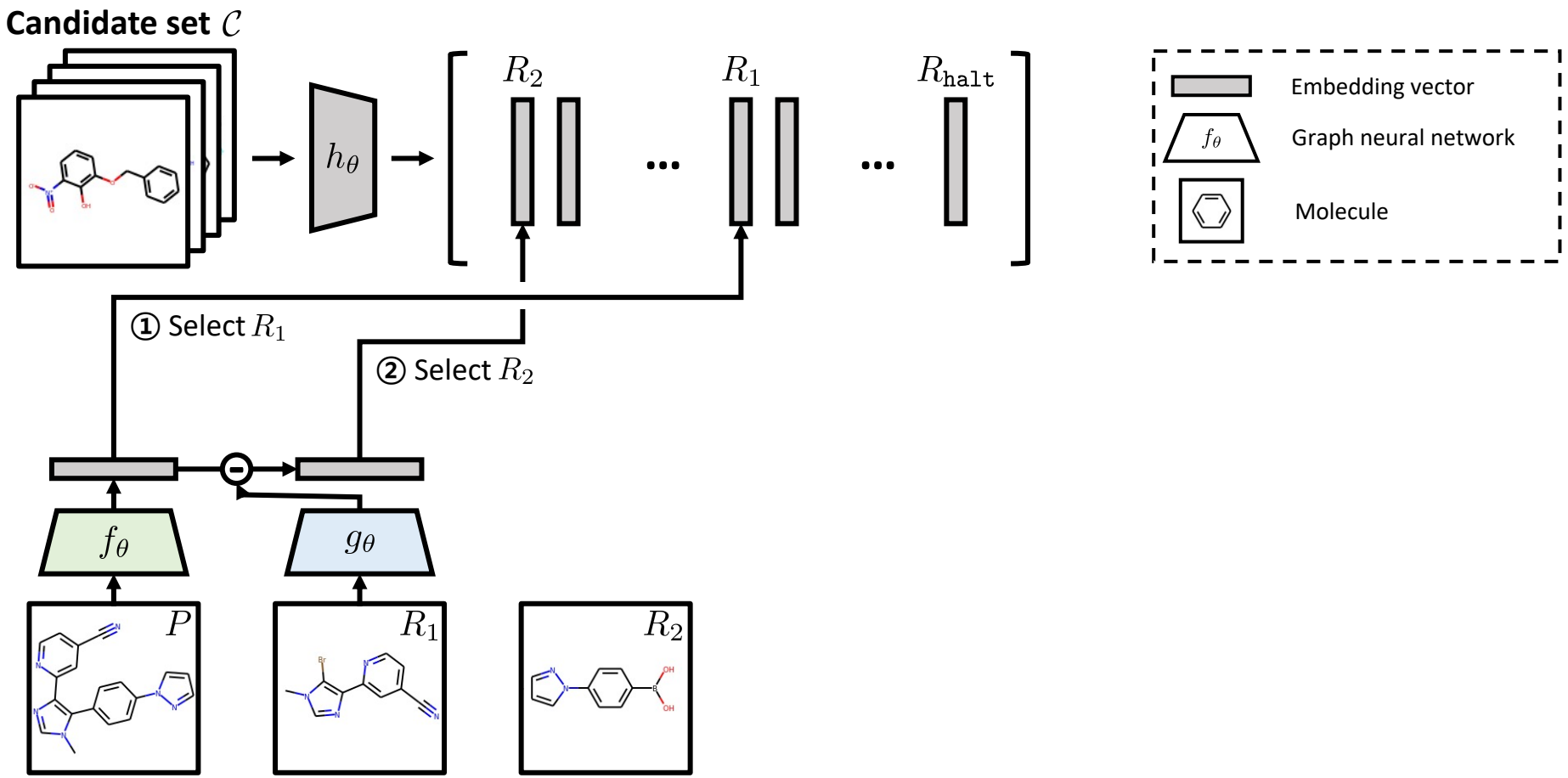
RetCL: Retrosynthesis via Contrastive Learning

Backward Selection Score: $\psi(R|P, \mathcal{R}_{\texttt{given}}) = \texttt{CosSim}\left(f_\theta(P) - \sum_{S \in \mathcal{R}_{\texttt{given}}} g_\theta(S),\ h_\theta(R)\right)$



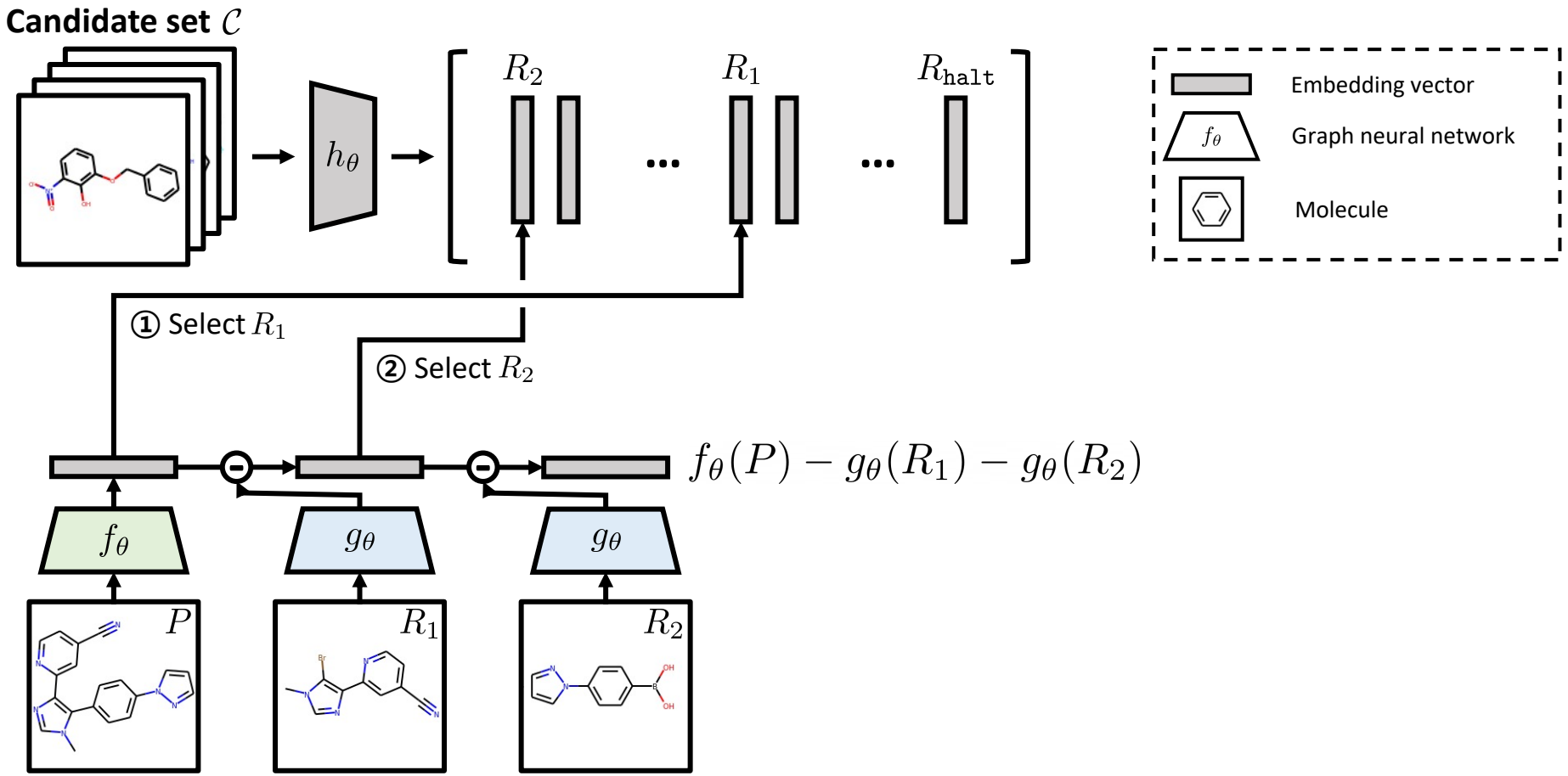
RetCL: Retrosynthesis via Contrastive Learning

Backward Selection Score: $\psi(R|P, \mathcal{R}_{\texttt{given}}) = \texttt{CosSim}\left(f_\theta(P) - \sum_{S \in \mathcal{R}_{\texttt{given}}} g_\theta(S),\ h_\theta(R)\right)$



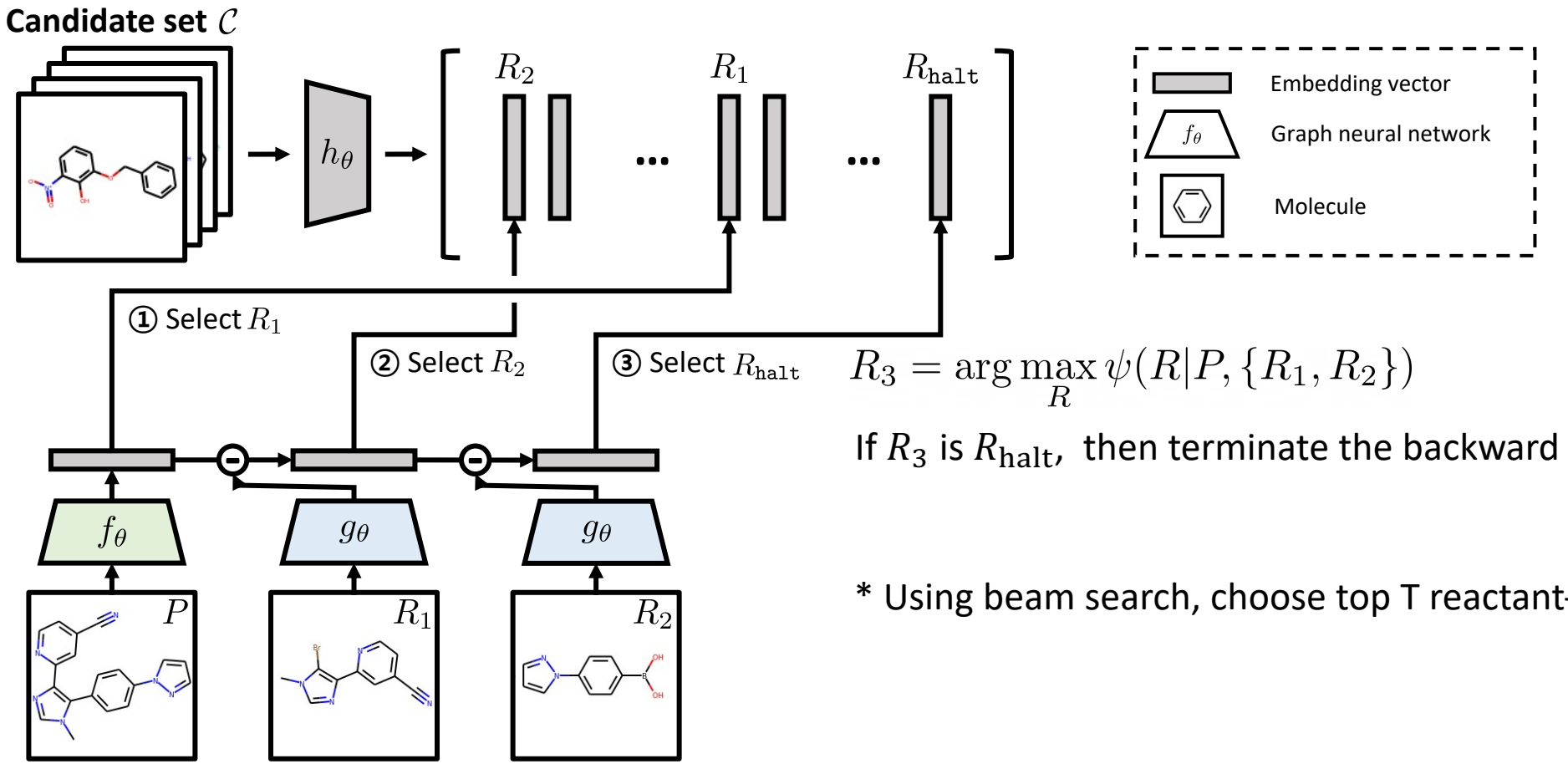
RetCL: Retrosynthesis via Contrastive Learning

Backward Selection Score: $\psi(R|P, \mathcal{R}_{\texttt{given}}) = \texttt{CosSim}\left(f_\theta(P) - \sum_{S \in \mathcal{R}_{\texttt{given}}} g_\theta(S),\ h_\theta(R)\right)$



RetCL: Retrosynthesis via Contrastive Learning

Backward Selection Score: $\psi(R|P, \mathcal{R}_{\texttt{given}}) = \texttt{CosSim}\left(f_\theta(P) - \sum_{S \in \mathcal{R}_{\texttt{given}}} g_\theta(S),\ h_\theta(R)\right)$



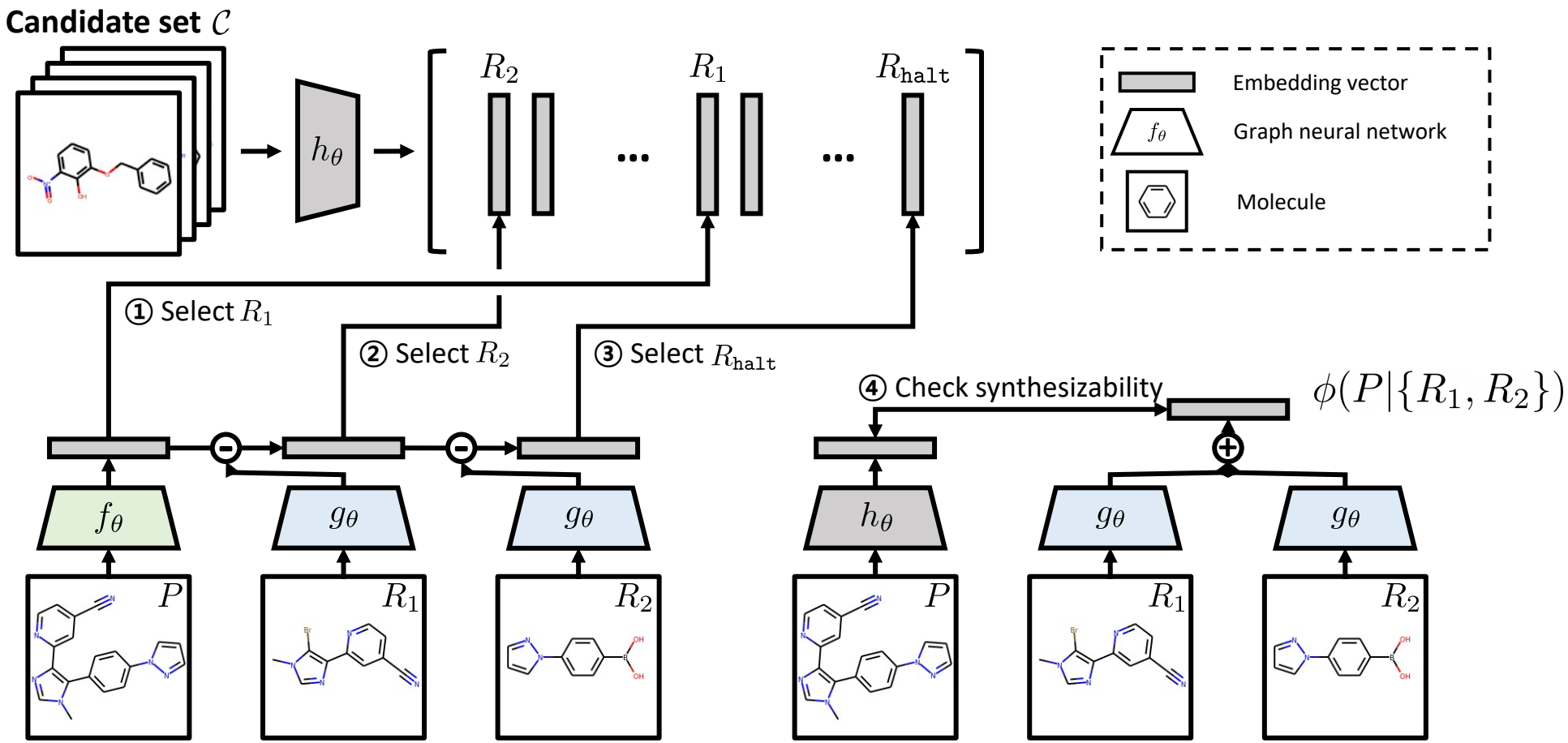
$R_3 = \arg\max_R \psi(R|P, \{R_1, R_2\})$

If R_3 is $R_{\texttt{halt}}$, then terminate the backward selection

* Using beam search, choose top T reactant-sets $\mathcal{R}_1, \ldots, \mathcal{R}_T$

RetCL: Retrosynthesis via Contrastive Learning

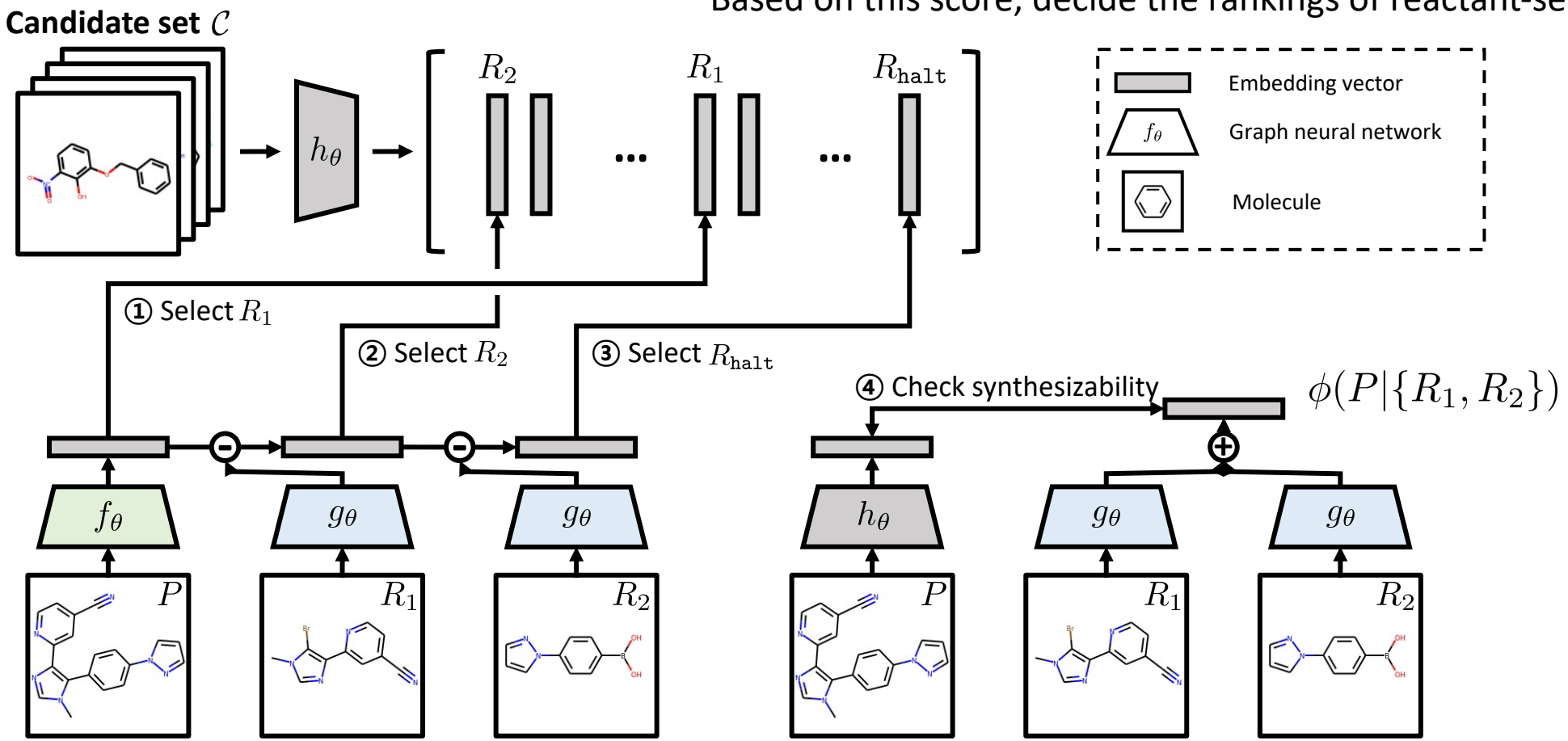
Forward Selection Score: $\phi(P|\mathcal{R}) = \texttt{CosSim}\left(\sum_{R\in\mathcal{R}} g_\theta(R),\ h_\theta(P)\right)$



RetCL: Retrosynthesis via Contrastive Learning

Overall Score: $\texttt{score}(P, \mathcal{R}) = \frac{1}{n+2}\left(\max_{\pi \in \Pi} \sum_{i=1}^{n+1} \psi(R_{\pi(i)} | P, \{R_{\pi(1)}, \ldots, R_{\pi(i-1)}\}) + \phi(P|\mathcal{R})\right)$

* Based on this score, decide the rankings of reactant-sets $\mathcal{R}_1, \ldots, \mathcal{R}_T$



RetCL: Retrosynthesis via Contrastive Learning

- **Recall.** We design two selection scores:
  - $\psi\left(R\middle|P,\mathcal{R}_{given}\right)$: score of a reactant R given a product P and a set of previously selected reactants $\mathcal{R}_{\text{given}}$
  - $\phi(P|\mathcal{R})$: score of a product P given a set of reactants $\mathcal{R}$

- How to learn the scores?
  - The score functions resemble **the classification scores** of selecting a reactant or a product
  - Given a reaction $(\mathcal{R}, P)$ in a database, we consider two **classification tasks:**

Backward $P \rightarrow \mathcal{R}$

$$p(R|P,\mathcal{R}_{\texttt{given}},\mathcal{C}) = \frac{\exp(\psi(R|P,\mathcal{R}_{\texttt{given}})/\tau)}{\sum_{R'\in\mathcal{C}\setminus\{P\}}\exp(\psi(R'|P,\mathcal{R}_{\texttt{given}})/\tau)}$$

$$\mathcal{L}_{\texttt{backward}}(P,\mathcal{R}|\theta,\mathcal{C}) = -\max_{\pi\in\Pi}\sum_{i=1}^{n+1}\log p(R_{\pi(i)}|P,\mathcal{R}^{\pi}_{<i},\mathcal{C})$$

Forward $\mathcal{R} \rightarrow P$

$$q(P|\mathcal{R},\mathcal{C}) = \frac{\exp(\phi(P|\mathcal{R})/\tau)}{\sum_{P'\in\mathcal{C}\setminus\mathcal{R}}\exp(\phi(P'|\mathcal{R})/\tau)}$$

$$\mathcal{L}_{\texttt{forward}}(P,\mathcal{R}|\theta,\mathcal{C}) = -\log q(P|\mathcal{R},\mathcal{C})$$

$$\mathcal{L}_{\texttt{total}} = \mathcal{L}_{\texttt{backward}}(P,\mathcal{R}|\theta,\mathcal{C}) + \mathcal{L}_{\texttt{forward}}(P,\mathcal{R}|\theta,\mathcal{C})$$

RetCL: Retrosynthesis via Contrastive Learning

- How to learn the scores? (Cont.)
  - The optimization is intractable since $\mathcal{C}$ contains a large number of candidate molecules
  - To resolve this, we approximate $\mathcal{C}$ with the set of molecules in a mini-batch $\mathcal{B}$

$$\mathcal{C}_{\mathcal{B}} = \{M \mid \exists (\mathcal{R}, P) \in \mathcal{B} \text{ such that } M = P \text{ or } M \in \mathcal{R}\}$$

$$\mathcal{L}(\theta|\mathcal{C}_{\mathcal{B}}) = \frac{1}{|\mathcal{B}|} \sum_{(\mathcal{R},P)\in\mathcal{B}} \mathcal{L}_{\texttt{backward}}(P, \mathcal{R}|\theta, \mathcal{C}_{\mathcal{B}}) + \mathcal{L}_{\texttt{forward}}(P, \mathcal{R}|\theta, \mathcal{C}_{\mathcal{B}})$$

  - To further improve approximation, we add hard negatives (i.e., nearest neighbors) into the candidate set

$$\widetilde{\mathcal{C}}_{\mathcal{B}} = \mathcal{C}_{\mathcal{B}} \cup \bigcup\nolimits_{M\in\mathcal{C}_{\mathcal{B}}} \{\text{Top-}K \text{ NN of } M \text{ from } \mathcal{C}\}$$

    - The nearest neighbors (NN) are defined with respect to the cosine similarity on $\{h_\theta(M)\}_{M\in\mathcal{C}}$
    - Since computing all embeddings for every iteration is time-consuming, we update the information periodically
    - We found that **this hard negative mining significantly improves the performance** of RetCL

Experiments

- Experimental setup
  - Our models are evaluated on **USPTO-50k**, which is a standard benchmark for retrosynthesis
  - For the candidate set $\mathcal{C}$, we use **all reactants in the entire USPTO database (671k molecules)**
  - For molecule encoders $f_\theta, g_\theta, h_\theta$, we use a single shared structure2vec [1] and separate residual layers
  - For evaluation, we use top-k exact match accuracy, which is widely used in the retrosynthesis literature

Experiments

- Experimental setup
  - Our models are evaluated on **USPTO-50k**, which is a standard benchmark for retrosynthesis
  - For the candidate set $\mathcal{C}$, we use **all reactants in the entire USPTO database (671k molecules)**
  - For molecule encoders $f_\theta, g_\theta, h_\theta$, we use a single shared structure2vec [1] and separate residual layers
  - For evaluation, we use top-k exact match accuracy, which is widely used in the retrosynthesis literature

- Effects of components

$\phi(P\|\mathcal{R})$	K	sum	Top-1	Top-10
✓			59.5	79.8
✓	1		69.6	92.2
✓	2		70.9	92.7
✓	4		71.1	92.9
	4		69.8	90.3
✓	4	✓	71.3	94.1

Experiments

- Experimental setup
  - Our models are evaluated on **USPTO-50k**, which is a standard benchmark for retrosynthesis
  - For the candidate set $\mathcal{C}$, we use **all reactants in the entire USPTO database (671k molecules)**
  - For molecule encoders $f_\theta, g_\theta, h_\theta$, we use a single shared structure2vec [1] and separate residual layers
  - For evaluation, we use top-k exact match accuracy, which is widely used in the retrosynthesis literature

- Effects of components
  - Hard negative mining is crucial in contrastive learning

$\phi(P\|\mathcal{R})$	K	sum	Top-1	Top-10
✓			59.5	79.8
✓	1		69.6	92.2
✓	2		70.9	92.7
✓	4		71.1	92.9
	4		69.8	90.3
✓	4	✓	71.3	94.1

Experiments

- Experimental setup
  - Our models are evaluated on **USPTO-50k**, which is a standard benchmark for retrosynthesis
  - For the candidate set $\mathcal{C}$, we use **all reactants in the entire USPTO database (671k molecules)**
  - For molecule encoders $f_\theta, g_\theta, h_\theta$, we use a single shared structure2vec [1] and separate residual layers
  - For evaluation, we use top-k exact match accuracy, which is widely used in the retrosynthesis literature

- Effects of components
  - Hard negative mining is crucial in contrastive learning
  - Considering the forward direction is important in retrosynthesis

$\phi(P\|\mathcal{R})$	K	sum	Top-1	Top-10
✓			59.5	79.8
✓	1		69.6	92.2
✓	2		70.9	92.7
✓	4		71.1	92.9
	4		69.8	90.3
✓	4	✓	71.3	94.1

Experiments

- Experimental setup
  - Our models are evaluated on **USPTO-50k**, which is a standard benchmark for retrosynthesis
  - For the candidate set $\mathcal{C}$, we use **all reactants in the entire USPTO database (671k molecules)**
  - For molecule encoders $f_\theta, g_\theta, h_\theta$, we use a single shared structure2vec [1] and separate residual layers
  - For evaluation, we use top-k exact match accuracy, which is widely used in the retrosynthesis literature

- Effects of components
  - Hard negative mining is crucial in contrastive learning
  - Considering the forward direction is important in retrosynthesis
  - Sum-pooling is more effective than mean-pooling

$\phi(P\|\mathcal{R})$	K	sum	Top-1	Top-10
✓			59.5	79.8
✓	1		69.6	92.2
✓	2		70.9	92.7
✓	4		71.1	92.9
	4		69.8	90.3
✓	4	✓	71.3	94.1

Experiments

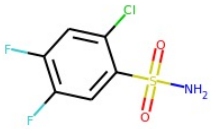
- Experimental setup
  - Our models are evaluated on **USPTO-50k**, which is a standard benchmark for retrosynthesis
  - For the candidate set $\mathcal{C}$, we use **all reactants in the entire USPTO database (671k molecules)**
  - For molecule encoders $f_\theta, g_\theta, h_\theta$, we use a single shared structure2vec [1] and separate residual layers
  - For evaluation, we use top-k exact match accuracy, which is widely used in the retrosynthesis literature

- Effects of components
  - Hard negative mining is crucial in contrastive learning
  - Considering the forward direction is important in retrosynthesis
  - Sum-pooling is more effective than mean-pooling

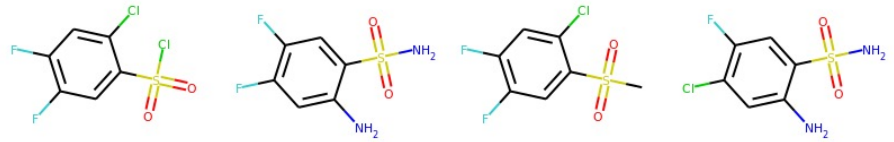
$\phi(P\|\mathcal{R})$	K	sum	Top-1	Top-10
✓			59.5	79.8
✓	1		69.6	92.2
✓	2		70.9	92.7
✓	4		71.1	92.9
	4		69.8	90.3
✓	4	✓	71.3	94.1

- Nearest neighbors based on molecule embeddings $h_\theta(M)$

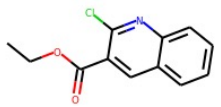
Example A



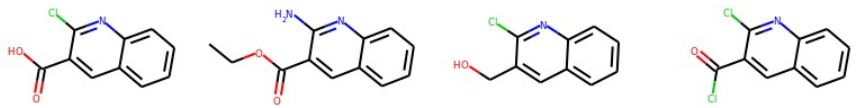
Top 4 nearest neighbors of A



Example B



Top 4 nearest neighbors of B



Experiments

- Single-step retrosynthesis
  - **Note.** Different categories = different assumptions about prior knowledge

Category	Method	Top-1	Top-3	Top-5	Top-10	Top-20	Top-50
Reaction type is unknown							
Template-free	Transformer [Karpov *et al.*, 2019]	37.9	57.3	62.7	-	-	-
	SCROP [Zheng *et al.*, 2019]	43.7	60.0	65.2	68.7	-	-
	Transformer [Chen *et al.*, 2019]	44.8	62.6	67.7	71.1	-	-
	G2Gs [Shi *et al.*, 2020]	**48.9**	**67.6**	**72.5**	**75.5**	-	-
Template-based	retrosim [Coley *et al.*, 2017]	37.3	54.7	63.3	74.1	82.0	85.3
	neuralsym [Segler and Waller, 2017]	44.4	65.3	72.4	78.9	82.2	83.1
	GLN [Dai *et al.*, 2019]	**52.5**	**69.0**	**75.6**	**83.7**	**89.0**	**92.4**
Selection-based	Bayesian-Retro [Guo *et al.*, 2020]	47.5	67.2	77.0	80.3	-	-
	RETCL (Ours)	**71.3**	**86.4**	**92.0**	**94.1**	**95.0**	**96.4**
Reaction type is given as prior							
Template-free	seq2seq [Liu *et al.*, 2017]	37.4	52.4	57.0	61.7	65.9	70.7
	Transformer† [Chen *et al.*, 2019]	54.1	70.0	74.2	77.8	80.4	83.3
	SCROP [Zheng *et al.*, 2019]	59.0	74.8	78.1	81.1	-	-
	G2Gs [Shi *et al.*, 2020]	**61.0**	**81.3**	**86.0**	**88.7**	-	-
Template-based	retrosim [Coley *et al.*, 2017]	52.9	73.8	81.2	88.1	91.8	92.9
	neuralsym [Segler and Waller, 2017]	55.3	76.0	81.4	85.1	86.5	86.9
	GLN [Dai *et al.*, 2019]	**64.2**	**79.1**	**85.2**	**90.0**	**92.3**	**93.2**
Selection-based	Bayesian-Retro [Guo *et al.*, 2020]	55.2	74.1	81.4	83.5	-	-
	RETCL (Ours)	**78.9**	**90.4**	**93.9**	**95.2**	**95.8**	**96.7**

Experiments

- Single-step retrosynthesis
  - **Note.** Different categories = different assumptions about prior knowledge
  - It is hard to fairly compare between methods operating under different assumptions

  - To alleviate such a concern, we incorporate our prior knowledge of candidates C into the baselines
  - **How?** we simply filter out reactants outside the candidates C from the predictions made by the baselines

Prior knowledge

Candidates $\mathcal{C}$

templates $\mathcal{T}$ + Candidates $\mathcal{C}$

Category	Method	Top-1	Top-5	Top-10	Top-50	Top-100	Top-200
Reaction type is unknown							
Template-free	Transformer [Chen *et al.*, 2019]	59.6	74.3	77.0	79.4	79.5	79.6
	RETCL (Ours)	71.3	**92.0**	**94.1**	**96.4**	**96.7**	**97.1**
Template-based	GLN [Dai *et al.*, 2019]	**77.3**	90.0	92.5	93.3	93.3	93.3
Reaction type is given as prior							
Template-free	Transformer [Chen *et al.*, 2019]	68.4	82.4	84.3	85.9	86.0	86.1
	RETCL (Ours)	78.9	**93.9**	**95.2**	**96.7**	**97.1**	**97.5**
Template-based	GLN [Dai *et al.*, 2019]	**82.0**	91.7	92.9	93.3	93.3	93.3

coverage of known templates, i.e.,
upper bound of template-based approaches

Conclusion

- We propose a selection-based approach considering the commercial availability of reactants
  - We reformulate the task of retrosynthesis as a problem where reactants are selected from a candidate set of available molecules
  - We design two effective selection scores in synthetic and retrosynthetic manners using graph neural networks
  - We propose a novel contrastive learning scheme with hard negative mining to overcome a scalability issue while handling a large-scale candidate set
  - We demonstrate the effectiveness of our framework in various single- and multi-step retrosynthesis experiments based on the USPTO database

Thank you for your listening!