Accepted to J. Chem. Phys. 10.1063/5.0076749

Generating reaction trees with cascaded variational autoencoders

Dai Hai Nguyen[1] and Koji Tsuda[1, 2, 3]
[1)]*Graduate School of Frontier Sciences, The University of Tokyo, 5-1-5 Kashiwa-no-ha, Kashiwa, Chiba 277-8561, Japan*
[2)]*RIKEN Center for Advanced Intelligence Project, 1-4-1 Nihonbashi, Chuo-ku, Tokyo 103-0027, Japan*
[3)]*Research and Services Division of Materials Data and Integrated System, National Institute for Materials Science, Tsukuba, Ibaraki 305-0047, Japan*

(*Electronic mail: tsuda@k.u-tokyo.ac.jp)

(Dated: 10 January 2022)

To develop useful drugs and materials, chemists synthesize diverse molecules by trying various reactants and reaction routes. Toward automating this process, we propose a deep generative model, called cascaded variational autoencoder (casVAE), for synthesizable molecular design. It generates a reaction tree, where the reactants are chosen from commercially available compounds and the synthesis route is constructed as a tree of reaction templates. The first part of casVAE is designed to generate a molecule called surrogate product, while the second part constructs a reaction tree that synthesizes it. In benchmarking, casVAE showed its ability to generate reaction trees that yield high quality and synthesizable molecules. An implementation of casVAE is publicly available at `https://github.com/tsudalab/rxngenerator`.

I. INTRODUCTION

To discover highly functional drugs and organic materials, chemists relentlessly synthesize various compounds by choosing reactants and synthetic routes. Based on their rich knowledge about chemical reactions, they design a new molecule and its synthetic route altogether. In contrast, most de novo molecular generators based on deep generative models[1–4] or other algorithmic techniques[5,6] do not use knowledge about synthesis and tend to generate non-synthesizable molecules. Recently, synthesizability-aware molecule generators, including MoleculeChef[7] and DoG-Gen[8] have been proposed. These two methods avoid creating unrealistic molecules by using a series of actions such as atom addition and bond formation.

In this paper, we propose a deep generative model that generates a reaction tree (Figure 1a) and use it for creating synthesizable molecules. A reaction tree consists of three parts: reactants, template tree and product. Leaves of a reaction tree corresponds to the reactants selected from commercially available compounds. Template tree consists of several reaction templates, each of which defines a single-step reaction[9]. Reaction templates can be extracted from databases such as Reaxys[10], Pistachio[11] and USPTO[12]. Finally, the root node represents the product of the whole reaction.

Our generative model, called cascade variational autoencoder (casVAE), consists of two variational autoencoders. The graphical model representation of casVAE is shown in Figure 2. In the first VAE, called product VAE, a decoder maps a latent vector to a molecule called *surrogate product*, while the second VAE, called reaction VAE, maps another latent vector to a reaction tree. Training of this graphical model is difficult due to its complexity, hence variational Bayesian learning[13] is employed, where a parametric model called *encoder* that maps a data point back to the latent space is trained together with the graphical model. In variational learning, the evidence lower bound (ELBO) is optimized instead of the evidence (i.e., the probability of observing the data). The encoder and decoder of product VAE as well as the encoder of reaction VAE are implemented using the components of Junction Tree VAE[3]. The decoder of reaction VAE is a top-down tree construction algorithm aiming to obtain a reaction tree that synthesizes the surrogate product. Note that casVAE generates two molecules at once, a surrogate product and the product of a generated reaction tree (i.e., tree product). They are not identical in general. We regard the tree product as the only outcome of casVAE, because of enhanced synthesizability.

Once our generative model is trained from data, Bayesian optimization[14] can be applied to optimize the reaction tree with respect to a target property of its product (Figure 1b). A joint latent vector consisting of the latent vectors of both VAEs is mapped to a reaction tree and the target property such as bioactivity or photochemical quality is obtained via simulation or a prediction model. A Bayesian probabilistic model defined in the latent space is updated using the new information and a next point in the latent space is recommended. By repeating this step, the reaction tree is gradually optimized towards generation of optimal molecules.

CasVAE was evaluated with a set of reaction trees obtained from USPTO database. In comparison to synthesizability-aware molecule generators, MoleculeChef[7] and DoG-Gen[8], the molecules generated by casVAE achieved comparable quality and superior synthesizability.

II. METHOD

In this section, we describe an overview of our method. See supplementary material for details. CasVAE consists of two parts, product VAE and reaction VAE. Product VAE is implemented as a junction tree VAE. It uses a molecular representation called a junction tree. The decoder maps the latent vector z_p to a junction tree x called *surrogate product*, while the encoder uses a tree message passing network (TMPN).

Reaction VAE generates a reaction tree from the set of reaction templates T and the set of available molecules M. The

encoder of reaction VAE is implemented as TMPN, where a template and a molecule are represented as a $|T|$-dimensional and $|M|$-dimensional one-hot vector, respectively. The decoder of reaction VAE generates a reaction tree from latent vector z_y and surrogate product x. First, x is mapped to the set of embedded vectors H_x obtained via the encoder of product VAE. The reaction tree is constructed from root to leaves (Figure 3). Each node retains a vector, called hidden state, that is computed progressively. A template node has one of the three states, empty, open and closed. An empty node implies that the template corresponding to the node is not yet specified. In an open node, the template is chosen but at least one of their reactants is not yet specified. A closed node has all reactants ready. The construction proceeds as follows.

1. Create the root node and an empty template node connected to it. Compute the hidden state of the root node via a neural network (NN) from H_x and z_y.

2. Pick up an empty template node. Choose its template from T subject to the probability distribution computed from parent's hidden state and z_y. Compute its hidden state via a NN from parent's hidden state and z_y.

3. Pick an open template node and generate child nodes. For each child node, pick a molecule from M or an empty template node subject to the probability distribution computed from parent's hidden state and z_y.

4. Repeat 2,3 until no empty or open template nodes are left.

The number of child nodes of a template node is variable according to the number of reactants of the corresponding reaction. Given a set of reaction trees as the training set, all parameters of both encoders and decoders are simultaneously determined by maximizing the evidence lower bound (ELBO)[13], where the prior distribution in the latent spaces is set to the standard Gaussian distribution $\mathcal{N}(0,\mathbf{I})$.

III. RESULTS

In comparion to existing deep generative models, JT-VAE, MoleculeChef[7] and DoG-Gen[8], we evaluate the reaction trees generated by casVAE in terms of their products. MoleculeChef and DoG-Gen have mechanisms to make molecules synthesizable, while JT-VAE is a pure molecule generator. For use in casVAE, we extracted reaction templates from USPTO reaction data set[12]. The templates are given as reaction SMILES strings, so the atom mapping is already done. We note that the quality of atom mapping in USPTO may not be optimal and can still be improved[15]. The reaction SMILES strings are converted to SMARTS patterns by RDChiral[16] for use in `RunReactants` function of Rdkit[17]. As done by Chen et al.[18], the molecule set M is designated as those appearing at least five times in USPTO dataset. Similarly, the reaction templates appearing at least five times are included in the template set T. As a result, we obtained 9766 molecules and 5567 templates.

TABLE I. Comparison of molecule generation methods in terms of FCD and compound quality. FCD stands for Frechet ChemNet Distance (FCD). For FCD, lower values are desirable. For compound quality, higher values are desirable.

Method	FCD	Compound Quality
JT-VAE	1.05	96.2
MoleculeChef	0.73	95.3
DoG-Gen	0.45	101.6
casVAE	0.66	95.8

TABLE II. Comparison of molecule generation methods in terms of synthesizability scores. For SAscore and SCscore, lower values are desirable. For RAscore and *Retro**-score, higher values are desirable.

Method	SAscore	SCscore	RAscore	*Retro**-score
JT-VAE	2.72	3.41	0.91	50.5
MoleculeChef	2.39	2.66	0.96	65.6
DoG-Gen	2.41	2.54	0.95	67.1
casVAE	2.38	2.64	0.93	96.7

As the training data, the reaction trees of the molecules in M were constructed by *Retro**[18], a neural-based A^*-like algorithm to find a template-based synthetic route. The dimensionality of the latent spaces of casVAE was set to 50. CasVAE was trained with mini-batch gradient descent with Adam, learning rate of 0.001 and batch size of 32. Training was performed on a NVIDIA TESLA V100 SXM3-32GB with 100 epochs. For each method, 10,000 molecules were generated from the prior distribution, i.e., the standard multivariate Gaussian distribution in its latent space. The success rate of casVAE creating a valid molecule from a sampled latent vector was 64.5%. Note that a reaction tree whose depth is more than seven was regarded as a failure, because it is unlikely to be implemented in practice. It took around 17 minutes for casVAE to create 10,000 molecules. Generated molecules were evaluated with three criteria, Frechet ChemNet Distance (FCD)[19], compound quality[20] and synthesizability. FCD is a distance measure between the distributions of molecules in the descriptor space derived from a neural network called ChemNet. In our experiment, the molecules obtained from a molecule generator are compared with those in the training set. MoleculeChef, Do-Gen and casVAE were trained on USPTO, while JT-VAE was trained on MOSES. Small FCD implies high statistical fidelity, i.e., generated molecules maintain statistical features of real molecules. The compound quality indicates the fraction of molecules passing a set of rule-based filters developed in the field of medicinal chemistry. It is normalized such that the compound quality of training set is 100. For evaluating synthesizability, we used the following scores: SAscore[21], SCscore[22] and RAscore[23]. We also defined *Retro**-score as the fraction of molecules that *Retro** is able to retrosynthesize.

All of the synthesizability-aware methods, MoleculeChef, DoG-Gen and casVAE, attained small FCD and high com-

pound quality (see Table I), indicating that these methods can generate realistic molecules. Table II shows the comparison in terms of synthesizability. In SAscore, SCscore and RAscore, casVAE was comparable to the other methods, while casVAE achieved exceptionally high *Retro**-score. In template-free methods such as MoleculeChef and DoG-Gen, a reaction route is represented as a set of *actions*, which may or may not correspond to known chemical reactions. Due to lack of templates, the product of a reaction step cannot be deduced deterministically, hence a machine learning model called Molecular Transformer[24] is used instead. Their relatively poor synthesizability by template-based *Retro** indicates the difficulty of grounding action-based reaction routes in known reactions. While free exploration of the chemical space may be limited to a certain degree, use of templates in casVAE contributes in making reaction trees interpretable and realistic. See Figure S2 in supplementary material for examples of reaction trees generated by casVAE.

Figure 4 demonstrates how casVAE is combined with Bayesian optimization (BO). For benchmarking, we optimized commonly-used molecular properties, quantitative estimate of drug-likeness (QED), and octanol-water partition coefficients penalized by the synthetic accessibility score and the number of long cycles (penalized logP). Using a BO implementation by Kusner et al.,[2], five batches of 50 molecules were generated by applying Bayesian optimization to the latent space. In both properties, the optimized molecules were substantially enhanced from those generated from the prior distribution. BO is available in our implementation.

IV. CONCLUSION

We presented a template-based deep generative model, casVAE, and demonstrated its ability to generate high-quality molecules. To be useful as a chemists' desktop tool, however, improvements from multiple aspects are necessary. Currently, reaction templates are applied by simple pattern matching, hence it may not be realizable by real experiments[9]. The construction of a reaction tree ignores important aspects such as yield, toxicity of products and the cost of reactants. It would also be useful if casVAE could be controlled to search under certain constraints such as a specific scaffold. In future work, we use casVAE as a footstep for developing more elaborate and specialized generative models that meet the practical demands of chemists. Reaction tree generation is a newly introduced problem and there is plenty of room for methodological improvement. In concurrent with our study, Gao et al. proposed another generative model based on a Markov decision process[25]. A variety of ideas would further be required towards the goal of creating a truly useful method.

SUPPLEMENTARY MATERIAL

See supplementary material for the algorithmic details and the examples of reaction trees.

ACKNOWLEDGMENTS

Tis work is supported by AMED JP20nk0101111, NEDO P15009, SIP (Technologies for Smart Bio-industry and Agriculture) and JST ERATO JPMJER1903.

DATA AVAILABILITY STATEMENT

USPTO dataset can be downloaded from `https://figshare.com/articles/dataset/Chemical_reactions_from_US_patents_1976-Sep2016_/5104873`. Implementation of casVAE is available at `https://github.com/tsudalab/rxngenerator`.

REFERENCES

[1] R. Gómez-Bombarelli, J. N. Wei, D. Duvenaud, J. M. Hernández-Lobato, B. Sánchez-Lengeling, D. Sheberla, J. Aguilera-Iparraguirre, T. D. Hirzel, R. P. Adams, and A. Aspuru-Guzik, "Automatic chemical design using a data-driven continuous representation of molecules," ACS Cent. Sci. **4**, 268–276 (2018).
[2] M. J. Kusner, B. Paige, and J. M. Hernández-Lobato, "Grammar variational autoencoder," in *International Conference on Machine Learning* (2017) pp. 1945–1954.
[3] W. Jin, R. Barzilay, and T. Jaakkola, "Junction tree variational autoencoder for molecular graph generation," in *International Conference on Machine Learning* (2018) pp. 2323–2332.
[4] X. Yang, J. Zhang, K. Yoshizoe, K. Terayama, and K. Tsuda, "ChemTS: an efficient python library for de novo molecular generation," Sci. Tech. Adv. Mater. **18**, 972–976 (2017).
[5] J. H. Jensen, "A graph-based genetic algorithm and generative model/monte carlo tree search for the exploration of chemical space," Chem. Sci. **10**, 3567–3572 (2019).
[6] N. Yoshikawa, K. Terayama, M. Sumita, T. Homma, K. Oono, and K. Tsuda, "Population-based de novo molecule generation, using grammatical evolution," Chem. Lett. **47**, 1431–1434 (2018).
[7] J. Bradshaw, B. Paige, M. J. Kusner, M. H. Segler, and J. M. Hernández-Lobato, "A model to search for synthesizable molecules," in *Advances in Neural Information Processing Systems*, Vol. 32 (2019).
[8] J. Bradshaw, B. Paige, M. J. Kusner, M. H. Segler, and J. M. Hernández-Lobato, "Barking up the right tree: an approach to search over molecule synthesis dags," in *Advances in Neural Information Processing Systems*, Vol. 33 (2020).
[9] R. Shibukawa, S. Ishida, K. Yoshizoe, K. Wasa, K. Takasu, Y. Okuno, K. Terayama, and K. Tsuda, "CompRet: a comprehensive recommendation framework for chemical synthesis planning with algorithmic enumeration," J. Cheminformatics **12**, 52 (2020).
[10] "Reaxys," `https://www.reaxys.com/`, accessed on 2021-10-23.
[11] "Pistachio," `https://www.nextmovesoftware.com/pistachio.html`, accessed on 2021-10-23.
[12] D. M. Lowe, *Extraction of chemical structures and reactions from the literature*, Ph.D. thesis, University of Cambridge (2012).
[13] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," arXiv preprint arXiv:1312.6114 (2013).
[14] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," Proc. IEEE **104**, 148–175 (2015).
[15] P. Schwaller, B. Hoover, J.-L. Reymond, H. Strobelt, and T. Laino, "Extraction of organic chemistry grammar from unsupervised learning of chemical reactions," Sci. Adv. **7**, eabe4166 (2021).
[16] C. W. Coley, W. H. Green, and K. F. Jensen, "RDChiral: an RDKit wrapper for handling stereochemistry in retrosynthetic template extraction and application," J. Chem. Inf. Model. **59**, 2529–2537 (2019).

[17]"RDKit: open-source cheminformatics," http://www.rdkit.org/, accessed on 2021-10-23.
[18]B. Chen, C. Li, H. Dai, and L. Song, "Retro*: learning retrosynthetic planning with neural guided A* search," in *International Conference on Machine Learning* (2020) pp. 1608–1616.
[19]K. Preuer, P. Renz, T. Unterthiner, S. Hochreiter, and G. Klambauer, "Fréchet chemnet distance: a metric for generative models for molecules in drug discovery," J. Chem. Inf. Model. **58**, 1736–1741 (2018).
[20]N. Brown, M. Fiscato, M. H. Segler, and A. C. Vaucher, "GuacaMol: benchmarking models for de novo molecular design," J. Chem. Inf. Model. **59**, 1096–1108 (2019).
[21]P. Ertl and A. Schuffenhauer, "Estimation of synthetic accessibility score of drug-like molecules based on molecular complexity and fragment contributions," J. Cheminformatics **1**, 8 (2009).
[22]C. W. Coley, L. Rogers, W. H. Green, and K. F. Jensen, "SCScore: synthetic complexity learned from a reaction corpus," J. Chem. Inf. Model. **58**, 252–261 (2018).
[23]A. Thakkar, V. Chadimova, E. J. Bjerrum, O. Engkvist, and J.-L. Reymond, "Retrosynthetic accessibility score (RAscore)–rapid machine learned synthesizability classification from AI driven retrosynthetic planning," Chem. Sci. **12**, 3339–3349 (2021).
[24]P. Schwaller, T. Laino, T. Gaudin, P. Bolgar, C. A. Hunter, C. Bekas, and A. A. Lee, "Molecular transformer: a model for uncertainty-calibrated chemical reaction prediction," ACS Cent. Sci. **5**, 1572–1583 (2019).
[25]W. Gao, R. Mercado, and C. W. Coley, "Amortized tree generation for bottom-up synthesis planning and synthesizable molecular design," arXiv preprint arXiv:2110.06389 (2021).

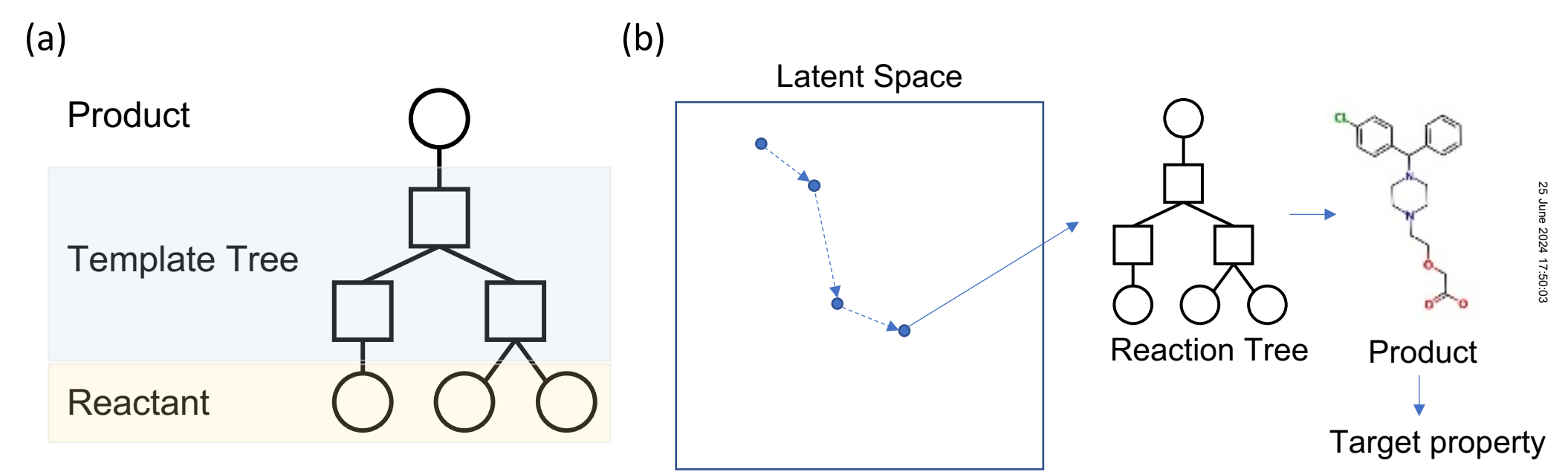
FIGURE CAPTIONS

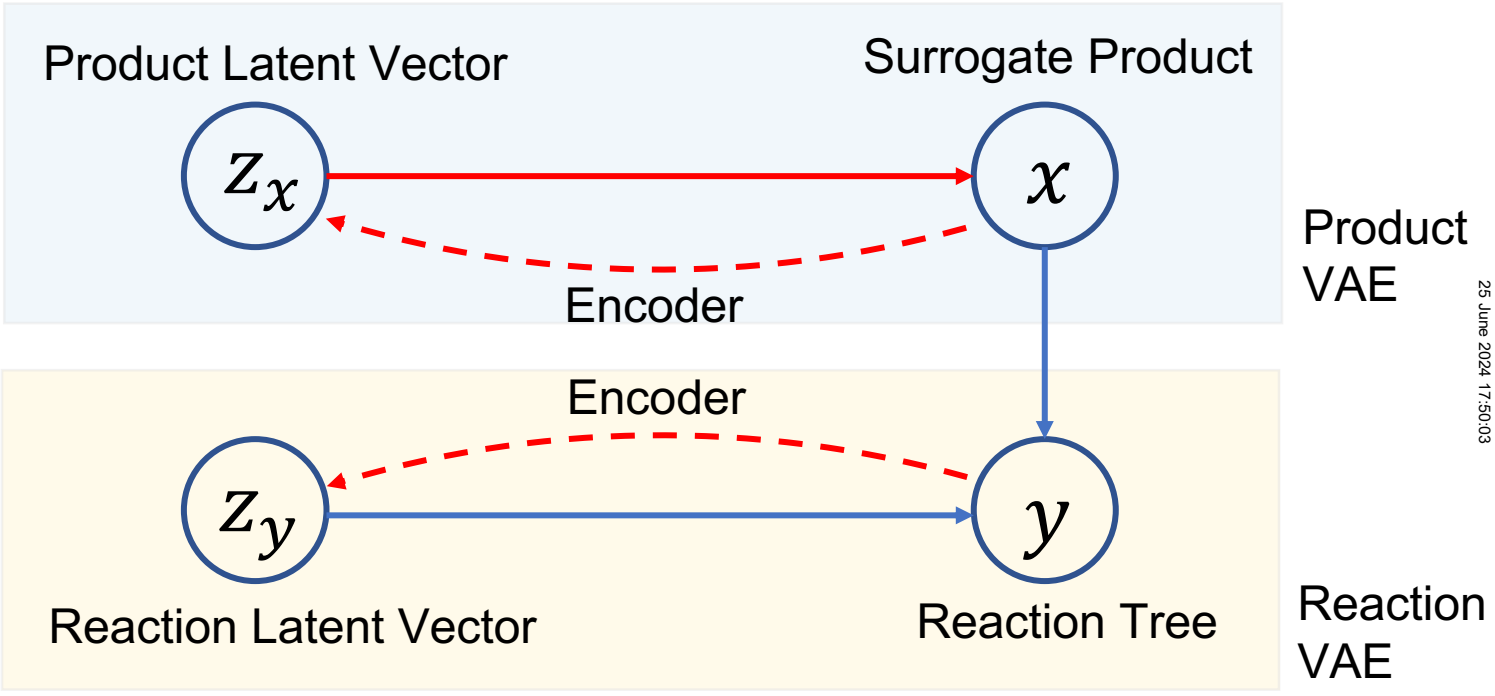
- Figure 1: Reaction tree and its optimization. (a) A reaction tree describes how the product is synthesized and consists of reactants, template tree and product. (b) Bayesian optimization is applied in the latent space to generate the molecules with an optimal target property.

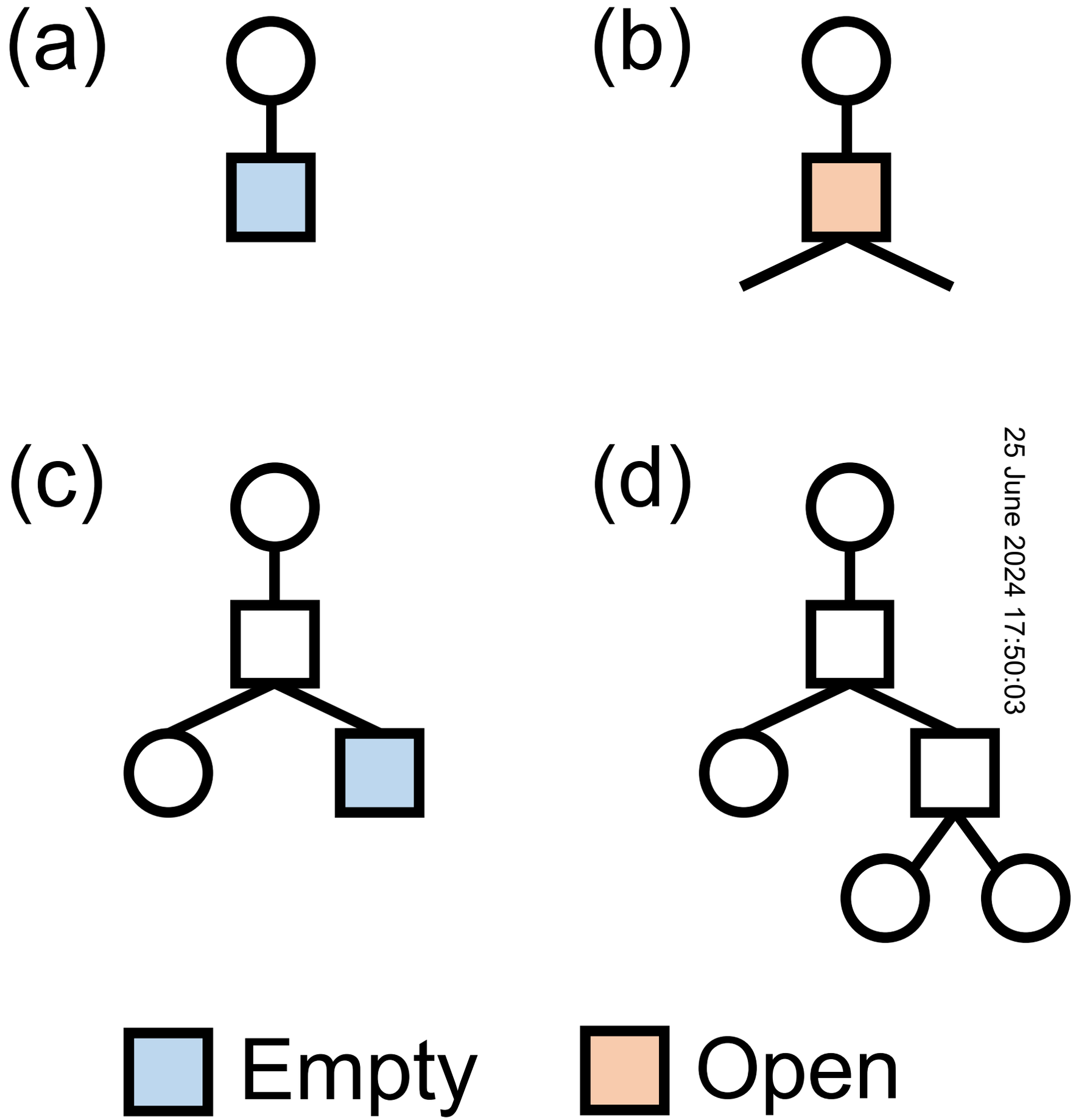
- Figure 2: Graphical model representation of casVAE. The components shown by red arrows are implemented using those of Junction Tree VAE (Jin et al.,2018).

- Figure 3: Construction of a reaction tree by the decoder of reaction VAE. Blue and red nodes indicate empty and open template nodes, respectively. (a) At first, the root node and an empty template node connected to it are created. (b) A reaction template for the template node is chosen and its state is updated to open. (c) Child nodes of the template node are sampled. They can either be a commercially available molecule or an empty template node. (d) Final reaction tree after all the empty and open template nodes are processed.

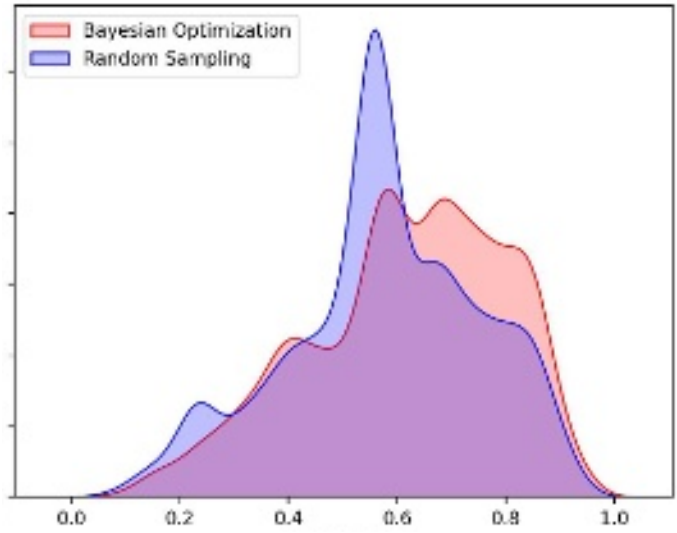
- Figure 4: Distributions of (a) QED and (b) penalized logp scores. The distributions of scores obtained by randomly sampling latent vectors from the prior distribution and Bayesian optimization (BO) are shown in blue and red, respectively.






(a)
(b)
(c)
(d)
25 June 2024 17:50:03
Empty
Open

(a) QED



(b) Penalized logP