

# Bayesian inference for chemical synthesis planning

Thesis by  
Zhongliang Guo

In Partial Fulfillment of the Requirements for the  
Degree of  
Doctor of Philosophy

The Graduate University for Advanced Studies, SOKENDAI

March 2021

© March 2021

Zhongliang Guo

ORCID: [orcid.org/0000-0003-1484-7926](https://orcid.org/0000-0003-1484-7926)

All rights reserved except where otherwise noted

## ACKNOWLEDGEMENTS

iii

I would like to express my gratitude to my supervisor Ryo Yoshida, whose advice and suggestions were immensely valuable throughout the course of my study at The Graduate University for Advanced Studies (SOKENDAI) and The Institute of Statistical Mathematics (ISM). In spite of his busy schedule, he generously gave me his time and encouragement and helped me to accomplish this study.

I am indebted to the members of my dissertation committee, Prof. Hideitsu Hino, Prof. Daichi Mochihashi, Prof. Ichigaku Takigawa, and Prof. Stephen Wu, who provided considerable and diverse feedback on my doctoral thesis.

I am deeply grateful to the staff at the ISM and SOKENDAI, who provided support and encouragement at various times in this study.

Finally, I would like to show my greatest appreciation to all my colleagues at SOKENDAI and ISM, people with whom I have spent so many joyful times throughout my life. My deepest gratitude goes to my family for their understanding while I pursued a Ph.D.

In organic chemistry, predicting the products from the reactants is called reaction prediction, while the design of synthetic routes in the opposite direction from the final products, which are the target molecules, is called chemical synthesis planning. Reaction prediction and chemical synthesis planning have been studied for more than 50 years. In recent years, advances in machine learning have significantly improved the accuracy of reaction prediction and chemical synthesis planning. However, most of prior researches have studied chemical synthesis planning as a separate problem from reaction prediction. Machine-learning models were designed to predict the reactants from a given product directly. As the products of a reaction consist of the target compound and other side products, in most cases predicting the reactants from the target compound in reverse without the information of the side products is an ill-posed problem. This ill-posed nature of backward prediction induced a limited predictive power of backward prediction models. Compared to the prediction accuracy of over 90% for forward prediction models that predict the products from the reactants, previously reported accuracy for backward prediction models ranged from 37% to 52%. In addition, most candidate reactants simulated from such backward prediction models are rarely contained within a given set of purchasable compounds that span the feasible solution space. For example, if a synthetic target is decomposed into A and B by a backward reaction prediction model, both the reactants will typically be non-purchasable. In such a case, further identification of synthesis routes to both A and B will be necessary.

In this thesis, we define the problem of chemical synthesis planning as a combinatorial optimization task with the solution space subject to the combinatorial complexity of all possible pairs of purchasable reactants. We propose a two-stage approach consisting of forward and backward predictions to solve the chemical synthesis planning. A trained forward model with high predictability defines the mapping  $Y = f(S)$  from a set of reactants  $S$  to their product  $Y$ . By solving the inverse mapping  $S = f^{-1}(Y^*)$  with a synthetic target  $Y^*$  with respect to possible combinations  $S$  of com-

mercially available reactants, we could obtain an algorithm for chemical synthesis planning that has high synthetic accessibility. As machine-learning models are not perfect prediction models, it is possible that all candidate reactants will never reach the target compound with the given forward model. Furthermore, if the model is incorrect, true reactants are expected to be close to the optimal solution. Considering that the purpose of chemical synthesis planning tools is to enumerate a wide variety of reaction routes and to facilitate the creativity of chemists, we addressed the problem of reaction mining within the framework of Bayesian inference:

$$p(S|Y = y^*) \propto p(Y = y^*, S) = p(Y = y^*|S)p(S). \quad (1)$$

The posterior is a discrete probability distribution that defines the probability of reactants  $S$  that can synthesize a given target compound  $y^*$ . The posterior probability is proportional to the joint distribution, which is formed by the forward prediction model. The support of the posterior consists of all possible combinations of reactants involved in a synthetic route. As exact computation across all candidates is unfeasible, the primary objective in the Bayesian computation is to identify a reduced set of reactant combinations with large joint probability, while those with ignorable probability are effectively eliminated. A diverse candidate set can help chemists find appropriate reaction routes to synthesize the target compound.

To enhance the search efficiency and exhaustively enumerate alternative pathways, a sequential Monte Carlo algorithm to sample in the discrete chemical space was developed. Cluster-level sampling was introduced to prevent particle impoverishment in the sampling step. In addition, a surrogate model was used to save on the cost of repeatedly evaluating the computationally expensive forward prediction model. Using a forward model with a prediction accuracy of approximately 87%, the Bayesian retrosynthesis algorithm successfully rediscovered 81.8% and 33.3% of known synthetic routes of one-step and two-step reactions, respectively, with top-10 accuracy. Remarkably, as the Monte Carlo algorithm is specifically designed to exhaustively explore highly probable reaction sequences ending with a given synthetic target, over 500 synthetic routes on average for each target were identified by the Bayesian retrosynthesis algorithm. In addition, we

investigated the potential applicability of such diverse candidates based on the expert knowledge of organic chemistry and revealed the influence of the publication bias in reaction datasets.

- [1] Zhongliang Guo, Stephen Wu, Mitsuru Ohno, and Ryo Yoshida. “Bayesian Algorithm for Retrosynthesis”. In: *J. Chem. Inf. Model.* 60 (2020), pp. 4474–4486. doi: 10.1021/acs.jcim.0c00320.  
Z.G. participated in the conception of the project, implemented the algorithm, and participated in the writing of the manuscript.

# TABLE OF CONTENTS

viii

Acknowledgements . . . . .	iii
Abstract . . . . .	iv
Published Content and Contributions . . . . .	vii
Table of Contents . . . . .	vii
List of Illustrations . . . . .	ix
List of Tables . . . . .	x
Chapter I: Introduction . . . . .	1
Chapter II: Machine learning in reaction prediction and chemical synthesis planning . . . . .	4
2.1 Descriptors of chemical compounds . . . . .	4
2.2 Chemical reaction and reaction prediction models . . . . .	9
2.3 Computer-assisted synthetic routes design . . . . .	16
2.4 Multistep synthetic route design . . . . .	20
2.5 Molecule design using reaction prediction models . . . . .	21
Chapter III: Mathematical formulation of synthetic route design . . . . .	24
3.1 Ill-posed nature of machine-learning-based retrosynthesis task . . . . .	24
3.2 A two-stage approach for the synthetic route design . . . . .	26
3.3 Mathematic formulation . . . . .	28
Chapter IV: Bayesian inference on retrosynthesis . . . . .	31
4.1 Difficulties for ordinary heuristic algorithms . . . . .	31
4.2 SMC accelerated by surrogate posterior distribution . . . . .	34
4.3 Ranking and prioritization . . . . .	40
4.4 Results . . . . .	43
4.5 Availability . . . . .	61
Chapter V: Conclusions . . . . .	64
Chapter VI: Future prospects . . . . .	66
Bibliography . . . . .	68



# LIST OF ILLUSTRATIONS

ix

<i>Number</i>		<i>Page</i>
1.1	Workflow of synthetic route design . . . . .	2
2.1	Descriptors of chemical compounds . . . . .	9
2.2	SMILES notation of reaction . . . . .	10
2.3	Rule-based reaction prediction . . . . .	11
2.4	Architecture of seq2seq model . . . . .	17
2.5	Tokenization for SMILES . . . . .	17
3.1	Workflow of Bayesian retrosynthesis algorithm . . . . .	27
4.1	Surrogate model for posterior probability . . . . .	35
4.2	Results of the surrogate-accelerated SMC for the one-step retrosynthesis . . . . .	51
4.3	Predicted energy for failed reactions . . . . .	55
4.4	Results of the surrogate-accelerated SMC for the two-step retrosynthesis . . . . .	62
4.5	An infeasible reaction in dataset and alternative synthetic routes proposed by the Bayesian retrosynthesis algorithm. . . . .	63

## LIST OF TABLES

x

<i>Number</i>		<i>Page</i>
2.1	Molecular fingerprint in RDKit . . . . .	7
3.1	Performances of existing methods on the prediction of synthetic reactions in forward and backward directions . . . . .	26
4.1	Parameters and experimental conditions for the simple SMC . . . . .	33
4.2	Hyperparameters used to train the surrogate model. . . . .	36
4.3	Parameters and experimental conditions for one-step retrosynthesis using the surrogate-assisted SMC algorithm. . . . .	39
4.4	Parameters and experimental conditions for two-step retrosynthesis using the surrogate-assisted SMC algorithm. . . . .	40
4.5	Summary of 10 reaction classes and number of recorded reactions in each reaction class. . . . .	42
4.6	Performance of the surrogate-accelerated SMC for the one-step retrosynthesis . . . .	45
4.7	Performance of various retrosynthetic prediction methods . . . . .	48
4.8	Top-10 accuracies (%) of five retrosynthetic prediction methods on 10 different reaction classes . . . . .	48
4.9	Number of times that the surrogate-accelerated SMC found the recorded synthetic route in 10 trials. . . . .	52
4.10	Reactions for which the surrogate-accelerated SMC failed to find the ground-truth synthetic route. . . . .	53
4.11	Ground-truth set consisting of two-step synthetic routes to 21 targets . . . . .	58

## INTRODUCTION

The synthesis of compounds is one of the most important topics in chemistry. In material and drug research, if a new compound is designed and expected to have desirable properties or efficacies, it must be synthesized and demonstrated. Therefore, the design of an effective synthetic route is an important issue. In recent years, with the development of machine learning, various new molecular design methods have been proposed since around 2010 (Ikebata et al., 2017; Jin, Barzilay, and Jaakkola, 2018; Cao and Kipf, 2018; You et al., 2018). These methods use existing datasets to train models, such as deep neural networks, to design molecules automatically, which satisfy the specified chemical properties. Compared to traditional methods, such as expert design and molecular screening, machine-learning methods can design a large number of new molecules in a shorter period; however, the accuracy of these methods depend on the training data and models.

To validate the chemical properties of novel, automatically designed molecules, an automated synthetic route design, synthesis, and instrumentation system is necessary. Although the planning of synthetic pathways has traditionally been conducted by experts, automatic synthesis planning has been studied for more than 50 years. In 1969, Corey and Wipke developed the first computer-aided synthesis planning system, known as Organic Chemical Simulation of Synthesis (E. J. Corey and Wipke, 1969), which was followed by the LHASA(Pensak and E J Corey, 1977), SYNCHEM(Gelernter et al., 1977), WODCA(Gasteiger et al., 2000) and others. These early synthetic route design systems relied on hand-coded reaction rules or reaction rules extracted from reaction databases by algorithms. Knowledge-based algorithms and more recently, machine-learning algorithms have been used to determine which rules to select. The transformations of chemical bonds described in the rule are applied to the current molecules to derive structurally simpler precursors until the compounds are finally available for purchase (Figure 1.1)

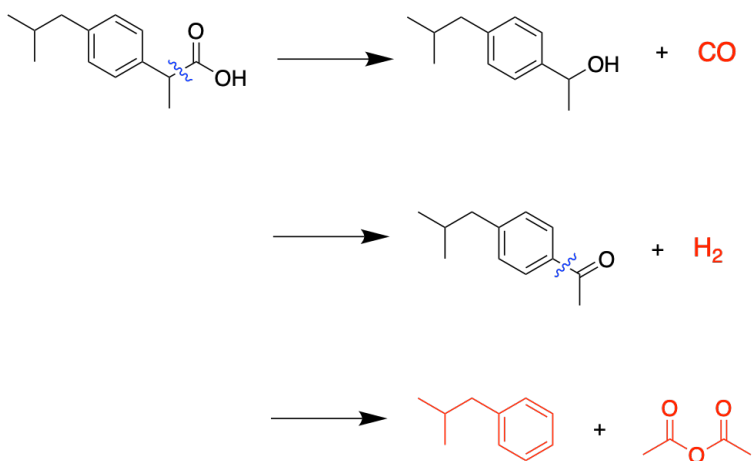


Figure 1.1: Workflow of synthetic route design. The synthetic target molecule is transformed into structurally simpler precursors based on the reaction rule. For non-purchasable precursors, further transformations are necessary until the compounds are available for purchase, which are shown in red.

A rule-based system can only support interpolated predictions. If the reaction mechanism is beyond the synthetic knowledge encoded in the rule set, the rule-based system is no longer applicable. In recent years, a variety of machine-learning techniques have been proposed for synthesis planning to cover a larger reaction space. However, most of the machine-learning methods take the target compound as input and predict the reactants that can synthesis the target compound. As will be described, the ill-posed nature of the inverse problem can induce a significant decline in the prediction accuracy.

This thesis is organized as follows: Chapter 2 summarizes machine-learning applications in reaction prediction and chemical synthesis planning. In reaction prediction, machine-learning models take the reactants as input and predict the products. In chemical synthesis planning, a general approach commences from the target compound and decomposes it into easy-to-synthesize precursor molecules, which is also called retrosynthesis. The machine-learning methods mimic the retrosynthesis approach by predicting the precursor molecules from the target compound. The reaction prediction and retrosynthesis prediction are referred to as forward prediction and backward prediction in this thesis, respectively, according to the directions of actual chemical reactions. In chapter

3, the mathematical formulation of chemical synthesis planning is given. In a chemical reaction, the products often consist of the target product and other by-products. Without the by-products, predicting reactants from the target compound can be an ill-posed problem. We define chemical synthesis planning as a combinatorial optimization problem, with the solution space subject to the combinatorial complexity of all possible combinations of purchasable reactants. As the forward prediction methods have achieved substantially high accuracy, the framework of the Bayesian inference is used to invert the forward model to the backward predicting system. In chapter 4, we propose a sequential Monte Carlo algorithm to sample in the discrete chemical space efficiently. Due to the large computational cost of the forward prediction model, a surrogate model is also introduced in the algorithm to filter synthetic routes with low probabilities and to accelerate the searching. The efficiency of the Bayesian algorithm is demonstrated by using a benchmark dataset and is compared with other methods. Chapters 5 and 6 conclude the proposed algorithm and discuss prospects for chemical synthesis planning. In contrast to other methods, the Bayesian algorithm for chemical synthesis planning can locate a diverse set of reaction routes to the synthetic target. By investigating the identified candidates based on expert knowledge, we determined that there are many false discoveries owing to no or exceedingly low reactivity. The paucity of negative data prevent the machine-learning model from being capable of determining the presence or absence of reactivity for the synthetic route. A comprehensive dataset is necessary to design evaluation methods to identify practical synthetic routes.

## MACHINE LEARNING IN REACTION PREDICTION AND CHEMICAL SYNTHESIS PLANNING

This chapter describes the machine-learning applications in reaction prediction and synthetic route design.

### 2.1 Descriptors of chemical compounds

For most machine-learning models, the input is numerical data, such as a matrix and the output is also numerical data. For language, image and graph data, the raw data is transformed to numerical data and models are designed to extract the information from the data efficiently to perform predictions and other tasks. To apply machine-learning methods to reaction prediction and synthetic route design, descriptors of chemical compounds are selected based on the models and perform important roles in the prediction.

#### Molecular graph

The most natural representation for a chemical compound is the labeled undirected graph (Figure 2.1a). A molecular graph  $G = (V, E, L_V, L_E)$  consists of a set of vertices  $V$  and a set of edges  $E$ . Each vertex of the graph corresponds to an atom in the compound and has an attribute representing its elemental symbol ( $L_V = \{C, O, N, F, \dots\}$ ); each edge in the graph corresponds to a bond in the compound and has an attribute representing its bond type (e.g.,  $L_E = 1, 1.5, 2, \dots$ ).

#### SMILES specification

SMILES (Simplified Molecular Input Line Entry System) (Weininger, 1988) describes chemical structures in the form of a line notation using ASCII strings (Figure 2.1b). Atoms are represented by elemental symbols and various grammar rules are specified to rigorously describe ring structures, branches, bond orders, isotopes, chiral centers, etc. All chemical structures can be converted into

SMILES strings.

For example, the structural formula of ibuprofen ( $C_{13}H_{18}O_2$ ) shown in Figure 2.1a is written as CC(C)Cc1ccc(cc1)C(C)C(=O)O. The ring structure is labeled with the same number (“1” in CC(C)Cc1ccc(cc1)C(C)C(=O)O) after the atoms at the beginning and end of the ring. Parentheses indicate a branch (side chain). The equal sign “=” denotes a double bond. Since atoms in an aromatic ring are denoted by lowercase letters, the ring’s constituent carbons are denoted by the lowercase “c” and all other carbons are denoted by the uppercase “C”. Hydrogen atoms are omitted in SMILES specifications, except in special cases.

The main grammar rules of SMILES are summarized below.

- (1) The elemental symbol is enclosed in square brackets to represent an atom (e.g., [Au]). However, square brackets may be omitted for B, C, N, O, P, S, F, Cl, Br, and I.
- (2) In general, the hydrogen atom H is omitted.
- (3) The same number is added after atoms at the beginning and end of the ring. For example, cyclohexane ( $C_6H_{12}$ ) is C1CCCCC1.
- (4) The branches are shown in parentheses.
- (5) The atoms constituting the aromatic ring are written in lower case.
- (6) Single bond, double bond and triple bond are denoted by -, =, #, respectively. Except in special cases, the symbol - for single bonds are omitted.
- (7) Add @ or @@ after the chiral center to indicate that the following neighbors are listed anticlockwise or clockwise, respectively.
- (8) Disconnected structures are separated by a period ..
- (9) Add +, - after the charged atoms (e.g., [Fe + 2]).

- (10) The mass number is given before the elemental symbol to express the isotope explicitly (e.g., carbon-14 is [14C]).
- (11) / and \ are used to specify the configuration of double-bond cis-trans isomer. For example,  $C/C = C \backslash C$ ,  $C/C = C/C$ .
- (12) The wildcard \* represents any atom.

A SMILES specification can be understood intuitively and allows for a linear representation of the chemical structures with minimal byte length. Given a SMILES string, the structural formula of the molecule is uniquely determined. These advantages have resulted in SMILES being the most widely used format in chemical informatics.

In contrast, there are multiple SMILES notations for one structural formula. For example, acetic acid can be written in various ways, such as CC(=O)O or O(=O)CC, by substituting the initiating atom. Therefore, it is not possible to determine the identity of chemical structures from strings alone. A canonicalization method using the Morgan algorithm (Morgan, 1965) has been developed to define the priorities of atoms uniquely and generate canonical SMILES from any generic SMILES.

In addition, the SMARTS notation, which is a modification of the SMILES notation, is widely used for structural searches in databases. For example, if written as C~N, it represents all the bond types (either single, double or triple bonds) that connect carbon C and nitrogen N. If written as X3, it represents an atom with a bond degree of 3. The [C, N] denotes an OR search of C and N. Thus, SMARTS notation is a special form of grammar introduced into SMILES for structural search.

### **Molecular fingerprint**

Molecular fingerprint is the most basic descriptor of chemical compounds. Given a set of substructures (fragments)  $\mathcal{F}$ , the element  $i$  of a binary fingerprint is set to 1 if the fragment  $f_i$  is in the molecule, otherwise, it is set to 0 (Figure 2.1c). The counting fingerprint records the number of  $f_i$  fragments in the molecule. Usually, the length of the fingerprint is  $O(10^2)$  to  $O(10^4)$ .



Table 2.1: Molecular fingerprint in RDKit

Molecular Fingerprint	Algorithm
RDK	Daylight-like fingerprinting
Layered	Daylight-like fingerprinting
Atom-pairs	Carhart et al. JCICS (1985) (Carhart, Smith, and Venkataraghavan, 1985)
Morgan	Similar to ECFP/FCFP
MACCSkeys	166 bits MDL MACCS keys
TopologicalTorsion	Topological torsion fingerprint (Nilakantan et al., 1987)
Pattern	Pre-defined structural pattern
E-state	Hall et al. JCICS (1995) (Hall and Kier, 1995)

Several algorithms to generate molecular fingerprints have been developed thus far. Table 2.1 depicts a list of fingerprints implemented in the Python chemoinformatics library RDKit (G. Landrum et al., 2006). The difference between these algorithms is how the fragment set is organized. There are two types of fingerprints: 1) A set of fragments is predefined (predefined type) and 2) The set of fragments is an automatic enumeration of fragments from a set of input compounds (enumeration type).

A predefined-type fingerprint uses a set of fragments that are predefined for a task. For example, the 4,860 fragments of the Klekota-Roth fingerprint were selected based on the pharmacological activity of drug molecules, which are constituted of substructures frequently occurring in compounds with high levels of activity (Klekota and Roth, 2008). The PubChem fingerprint is an 881-dimensional binary fingerprint, (Wang et al., 2017). PubChem is a database of chemical compounds owned by the U.S. National Institutes of Health (NIH). PubChem fingerprint can be used for similarity searches in a web interface. Another renowned predefined-type fingerprint is MACCS Keys (166 fragments), which was developed by MDL (Durant et al., 2002).

The fragment set of an enumeration-type fingerprint consists of all the fragments of less than or equal to a certain size in the input set of compounds. A binary fingerprint encodes the presence or absence of each fragment in the molecule. Whereas, a counting fingerprint encodes the number of each fragment in the molecule. Morgan fingerprints (aka circular fingerprint), ECFP (extended connectivity fingerprint) and FCFP (functional-class fingerprint) (D. Rogers and Hahn, 2010) are

classified as enumeration type. The basic concept and method of a fragment enumeration algorithm are derived from the Morgan algorithm (Morgan, 1965), as described in Algorithm 1. The objective is to calculate the atom-specific attribute values of atom  $1, \dots, N$ , which condense the information of the surrounding environment. After assigning an initial attribute value, such as an atomic number to each atom, the attribute values are updated by calculating the summations of the attribute values of the neighboring atoms. By repeating this operation  $R$  times (as a radius parameter), we can obtain the attribute values reflecting the bonding pattern of the neighboring atoms up to the  $R$  proximity. Such operations on molecular graphs are inherited by ECFP. In the algorithm of an ECFP calculation, each atom is assigned a vector of attribute values and the attribute vectors of the neighboring atoms are propagated to each atom. A hash function is applied to obtain a unique integer value, which is a numerical representation of the substructure around the target atom. Dividing the hash value by the length  $B$  of the fingerprint vector, the address of the remainder is set to one to indicate the presence of a substructure. The remainder operation can cause multiple substructures to be assigned to a single element of a vector, which is called as a “bit collision problem”. To evade this problem, researches on positive definite kernels for graphs (graph kernels) were actively conducted in the 2000s (Yamashita, Higuchi, and Yoshida, 2014).

---

**Algorithm 1** Morgan Algorithm

---

**Input:** chemical compound  $S$ , radius  $R$ **Output:** attribute values  $\{a_1, \dots, a_N\}$  of atom  $1, \dots, N$ .

- 1: **Initialize:** Assign an attribute value to each atom (e.g., atomic number).
- 2: **for**  $r \in \{1, \dots, R\}$  **do**
- 3:     **for**  $n \in \{1, \dots, N\}$  **do**
- 4:         Takes the sum of the attribute values of the neighboring atoms  $\mathcal{A}_n$  and updates its own attribute value.

$$a_n \leftarrow a_n = \sum_{i \in \mathcal{A}_n} a_i$$

- 5:     **end for**
  - 6: **end for**
-

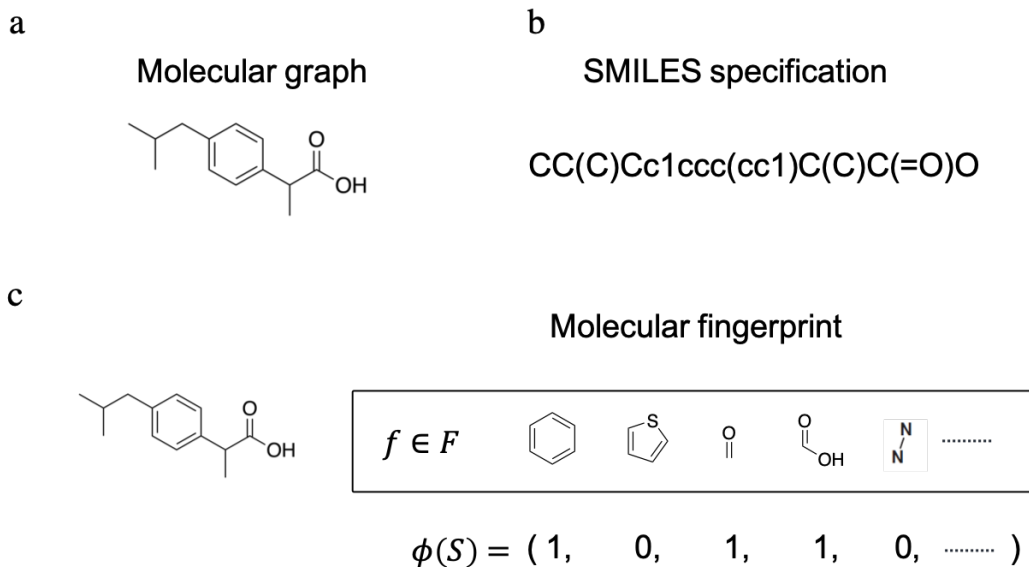


Figure 2.1: Descriptors of chemical compounds. **a.** The structure formula of ibuprofen. **b.** The SMILES string of ibuprofen. **c.** The fingerprint of ibuprofen.

## 2.2 Chemical reaction and reaction prediction models

A chemical reaction can be expressed by the reaction equation  $A + B \xrightarrow{C} D + E$ . This signifies that reactants A and B produce product D and E under the reaction condition C. Chemical reactions are essentially the interaction between molecular orbitals, resulting in the breakage of bonds and formation of new bonds between atoms. Thus, by focusing on the functional groups present in the molecule, experts can predict how the reaction will proceed. Rule-based reaction prediction systems use similar techniques, such as graph matching, to locate the functional groups present in the reactants and compare them to a rule set to predict how they will interact within and between molecules. When multiple rules in a rule set are applicable, the scoring function is used to compute the score of each rule and to determine which rule should be applied. The scoring function for traditional rule-based prediction systems is determined by experts using rules of thumb; however, as the size of the rule set increases, it becomes more challenging to select the correct rules. Therefore, machine-learning methods have been proposed to learn the scoring function from an existing reaction dataset. Furthermore, deep neural network models that do not rely on rule sets and

learn the mechanisms of reactions from data have emerged with the development of deep neural networks.

### Machine learning in rule-based reaction prediction model

The rule-based response prediction model consists of a rule set and a scoring function. Rule sets are often defined by experts, however, recently, heuristic methods have been established to extract them from the data (Coley, Barzilay, et al., 2017). There are databases for chemical reactions, such as Reaxys (Reaxys 2021) and Scifinder (SciFinder 2021), however, since they are proprietary and lack datasets for machine learning, the open source United States Patent and Trademark Office (USPTO) datasets is used in most cases. This dataset was extracted from the reactions published in the US patent and describes the reactants and products of each reaction, as well as the temperature, catalyst, solvent, etc. (Daniel Mark Lowe, 2012). Each reaction is described in SMILES notation (Figure 2.2), the reactants, the reagents and the products are separated by “>” signs. SMILES notations of molecules and reactions are supported by major cheminformatics software, making it useful for machine-learning models of chemical reactions.

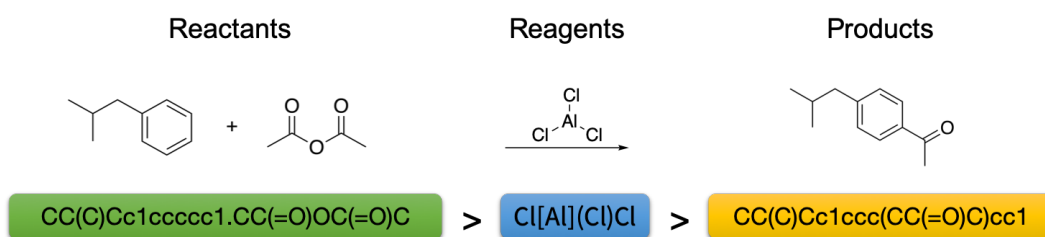


Figure 2.2: SMILES notation of reaction. Isobutylbenzen is acetylated in the Friedel-Crafts reaction using acetic anhydride and aluminum chloride. In the reaction SMILES, reactants (isobutylbenzen and acetic anhydride), reagent (aluminum chloride), and product (4'-Isobutylacetophenone) are separated by “>” signs.

The first step of the rule extraction is Atom-mapping for the reactants and products of the reaction (Figure 2.3). The type and number of atoms in a chemical reaction do not change before and after the reaction, although the bonds between the atoms change. Toolkits, such as Indigo, (Pavlov

et al., 2011) compare reactants and products, determine the type of reaction and describe the correspondence between atoms in reactants and the product in terms of atom-mapping. However, in some cases, more than one product exists for some reactions and only the target compound is listed in the reaction dataset, but not the by-products. Therefore, all the atoms of the product are in reactants, but some of the atoms of the reactants are not in the product. In such cases, the cheminformatics toolkit maps each atom in the product to the atoms in the reactant from which they are derived.

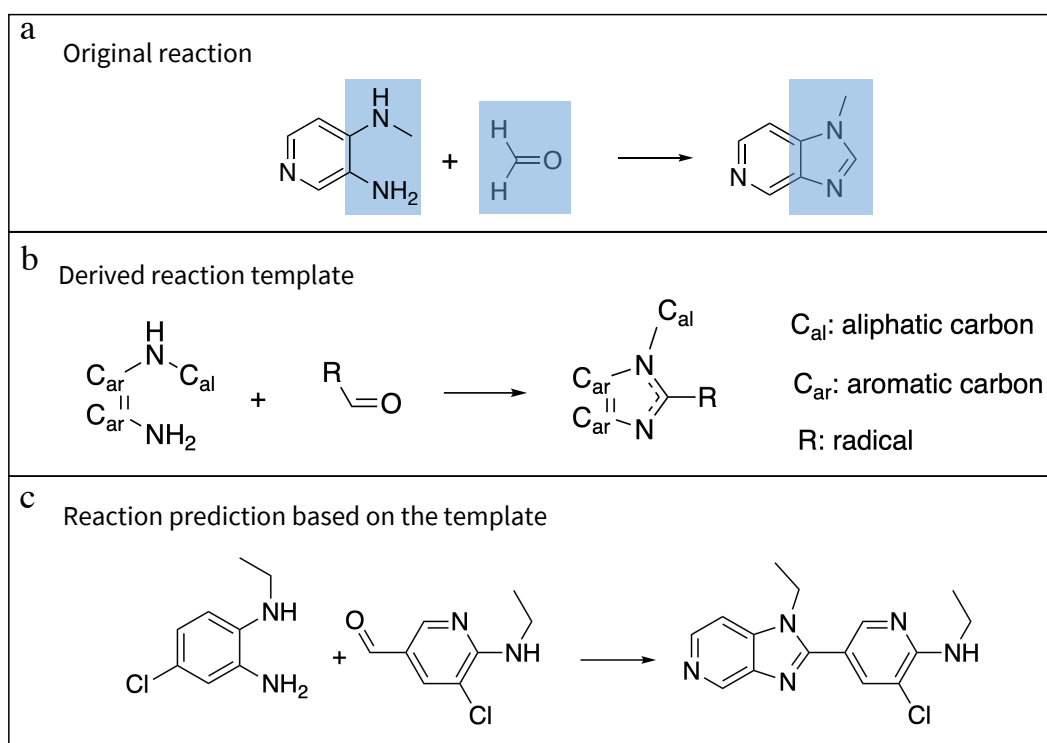


Figure 2.3: Rule-based reaction prediction. **a.** The original reaction in the dataset. **b.** The reaction template is extracted by comparing the bond charge before and after the reaction. **c.** The extracted reaction template is used for reaction prediction.

Based on the results of atom-mapping, the atoms that change their bonding before and after the reaction are defined as the reaction centers and the atoms in and around the centers are defined as the

rules of the changes before and after the reaction. Changes can be extracted as rules in SMARTS notation, which is originally an extension of SMILES notation for searching substructures in databases. Since the reaction centers are substructures of molecules, SMARTS notations can be used to define the rules for reactions.

To predict reactions using the rule set extracted from the dataset, rules that are applicable to a given reactant are identified from the rule set. When a SMILES notation is provided, it is possible to determine whether the reaction center of each rule is in the reactants by performing a substructure search by using the SMARTS notation. If there is more than one rule that can be applied, each rule is applied to predict the product. The reactions are scored against the given reactants and predicted products by using a scoring function to determine the most likely reactions and their products.

When building scoring functions using machine-learning models, the choice of descriptors and models is important: Segler and Waller used the Extended-Connectivity Fingerprint (D. Rogers and Hahn, 2010) as a descriptor and compared the accuracy of expert systems, logistic regression and two neural networks using as a scoring function (Segler and Waller, 2017). For a rule set size of 103, the accuracy of the expert system was 0.07 as compared to 0.86 for the logistic regression and 0.92 for both types of neural networks. Furthermore, when the size of the rule set increased to 8720, the accuracy of the expert system was 0.02 versus 0.41 for the logistic regression and 0.78 and 0.77 for the two types of neural networks. Compared to expert systems that use expert-designed scoring functions, machine-learning models achieved higher prediction accuracy. In addition, for problems with large rule sets, neural networks have indicated a high predictive capability. Coley et al. designed a descriptor based on the bond formation or breakage at the reaction center and constructed a scoring function using neural networks (Coley, Barzilay, et al., 2017). For a rule set size of 1689, the accuracy of the neural network was 0.685.

By introducing machine learning into the rule-based reaction prediction model, not only was the accuracy of the scoring function improved, but the difficulty of expert designs for large rule set sizes was also solved, allowing us to cover a wide range of responses.

## Reaction prediction model based on deep neural network

The coverage of reactions that can be predicted by a rule-based reaction prediction model is determined by the size of the rule set. As the reaction data increases, the size of the rule set increases and the design of the scoring function becomes more difficult. In addition, as the size of the rule set increases, a problem arises: The time for substructure retrieval and product prediction increases. However, limiting the size of the rule set reduces the predictive ability of the model. Thus, rule-based models for reaction predictions have an inherent scalability problem. To solve this problem, reaction prediction models based on deep neural networks have been proposed.

Jin et al. considered the molecule as a labelled graph and formulated the reaction prediction as a graph transformation problem (Jin, Coley, et al., 2017). Rather than using a rule set, they used a graph neural network named Weisfeiler-Lehman Network (WLN) to locate the reaction centers of the reactants and predict the products. The molecular graph  $G = (V, E)$  is represented as a set of  $\{(c_u^{(L)}, b_{uv}, c_v^{(L)}) | (u, v) \in E\}$ , where  $c_v^{(L)}$  is the label of atom  $v$  and  $b_{uv}$  is the bond type between  $u$  and  $v$ . For a node  $v \in G$  with neighbor nodes  $N(v)$ , node features  $f_v$ , and edge features  $f_{uv}$ , the relabeling function is defined as

$$r(v) = \tau(\mathbf{U}_1 \mathbf{f}_v + \mathbf{U}_2 \sum_{u \in N(v)} \tau(\mathbf{V}[\mathbf{f}_u, \mathbf{f}_{uv}])) \quad (2.1)$$

where  $\tau(\cdot)$  could be any non-linear function. The relabeling operation is applied iteratively to obtain context-dependent atom vectors

$$\mathbf{h}_v^{(l)} = \tau(\mathbf{U}_1 \mathbf{h}_v^{(l-1)} + \mathbf{U}_2 \sum_{u \in N(v)} \tau(\mathbf{V}[\mathbf{h}_u^{(l-1)}, \mathbf{f}_{uv}])) \quad (1 \leq l \leq L) \quad (2.2)$$

where  $\mathbf{h}_v^{(0)} = \mathbf{f}_v$  and  $\mathbf{U}_1, \mathbf{U}_2, \mathbf{V}$  are shared across layers. The final atom representations is

$$\mathbf{c}_v = \sum_{u \in N(v)} \mathbf{W}^{(0)} \mathbf{h}_u^{(L)} \odot \mathbf{W}^{(1)} \mathbf{f}_{uv} \odot \mathbf{W}^{(2)} \mathbf{h}_v^{(L)} \quad (2.3)$$

The representation of the whole graph  $G$  is defined as  $\mathbf{c}_G = \sum_v \mathbf{c}_v$

For two atoms  $u, v$ , the reactivity score of  $(u, v)$  is predicted using another neural network:

$$s_{uv} = \sigma(\mathbf{u}^T \tau(\mathbf{M}_a \mathbf{c}_u + \mathbf{M}_a \mathbf{c}_v + \mathbf{M}_b \mathbf{b}_{uv})) \quad (2.4)$$

where  $\sigma(\cdot)$  is the sigmoid function,  $\mathbf{c}_u, \mathbf{c}_v$  are the atom representation for atoms  $u, v$ , and  $\mathbf{b}_{uv}$  is an additional feature vector encoding auxiliary information.

The model is trained to minimize the following loss function:

$$\mathcal{L}(\mathcal{T}) = - \sum_{R \in \mathcal{T}} \sum_{u \neq v \in R} y_{uv} \log(s_{uv}) + (1 - y_{uv}) \log(1 - s_{uv}) \quad (2.5)$$

Specifically, the WLN was trained to identify the reaction centers in the reactants at first. The products were predicted by breaking the bonds between the atoms in the reaction centers or creating new bonds between the atoms in different molecules. This corresponds to increasing or decreasing the number of edges between nodes when considering the molecule as a graph. When there are several atoms in the reaction center, bond formation or breakage is conducted for  $K$  atoms with a high probability. As  $K$  increases, the number of new graphs that can be created increases combinatorically; however, the chemical constraints can be used to exclude molecules that cannot exist. Conventional rule-based reaction prediction models use substructure searches to identify the reaction centers, while graph neural network-based methods allow the model to predict the reaction centers. This method generates multiple predicted products based on different graph transformations. However, similar to rule-based methods, a scoring function was used to rank the predicted products and determine the most plausible one. The method of using graph neural networks can predict products with an accuracy of 85.6% in the USPTO dataset. It is a natural development of rule-based response prediction models to overcome the scalability problem derived from rule sets by using graph neural networks instead of rule sets. However, enumerating the possible graphical transformations is a time-consuming task, requiring an estimated 0.5 seconds for each response. For the virtual screening of molecules using a reaction prediction model and other applications, a more efficient prediction model is desired.



Focusing on the SMILES notation of molecules, a study that utilized seq2seq model to predict chemical reactions was published around the same time as the study of graph neural networks (Schwaller, Gaudin, et al., 2018). The seq2seq model is a deep neural networks model used in machine translation. The architecture of seq2seq is largely divided into two components, namely, the encoder and decoder (Sutskever, Vinyals, and Le, 2014) (Figure 2.4). For example, when translating from Japanese to English, the encoder a Japanese sentence as the input and encodes its meaning into an embedding vector. The decoder takes the embedding vector and decodes it into an English sentence. In the seq2seq model, the long short-term memory (LSTM) was used in the two RNN. An LSTM consists of units that process the input data sequentially. At time step  $t$ , the unit processes input  $x_t$  and the previous hidden state  $h_{t-1}$  to obtain the output and hidden state transition

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \quad (2.6)$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \quad (2.7)$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \quad (2.8)$$

$$c_t = f_t \otimes c_{t-1} + i_t \otimes \tanh(W_c x_t + U_c h_{t-1} + b_c) \quad (2.9)$$

$$h_t = o_t \otimes \tanh(c_{t-1}), \quad (2.10)$$

where  $i_t$ ,  $f_t$  and  $o_t$  are the input, forget, output gates;  $c$  is the cell state vector;  $W$ ,  $U$  and  $b$  are model parameters; and  $\sigma$  is the sigmoid function.

In addition, an attention mechanism is used to find out important word in the input for generating particular word in the output sequence. The attention weights  $\alpha$  is:

$$\alpha_{it} = \frac{\exp(s_i^T W_\alpha h_t)}{\sum_{t'=0}^T \exp(s_i^T W_\alpha h_{t'})} \quad (2.11)$$

$$c_i = \sum_{t=0}^T \alpha_{it} h_t, \quad (2.12)$$

where  $s_i$ ,  $h_t$  are the hidden state of decoder and encoder, respectively.

The attention vector is defined by

$$a_i = \tanh(W_a \{c_i; s_i\}) \quad (2.13)$$

Both  $W_\alpha$  and  $W_a$  are learned weights. The probability for a single token follows:

$$p(y_i|\{y_1, \dots, y_{i-1}, c_i\}) = \text{softmax}(W_p a_i) \quad (2.14)$$

where  $W_p$  are also trainable.

Formally, the neural network of machine translation takes a Japanese string and predicts the corresponding English string as output. Since SMILES notations of molecules are ASCII strings, it is possible to predict product strings from the reactant strings if the neural network of the machine translation can be trained. In machine translation, a pre-processing process called tokenization, which includes word separation, is necessary for a sentence with no words separated by spaces, such as Japanese. Since a SMILES notation is a string of non-separated characters, Schwaller et al. proposed a tokenization method (Figure 2.5) which takes each atom as a single word. In conjunction with the tokenization, the neural networks of machine translation can predict reactions with a precision of 90.4% in the USPTO dataset (Schwaller, Laino, et al., 2019). Compared to graph neural networks, machine translation neural networks do not enumerate and rank the possible products, thus allowing for high throughput predictions. The average prediction time for one response is about 30ms when using Transformer (Vaswani et al., 2017), which is a type of neural network for machine translation. By using the beam search, the decoder can deliver multiple predictions with a high probability. However, since the neural network of machine translation treats a SMILES notation as a string, it sometimes predicts strings that do not conform to the SMILES grammar. Verification of a reaction prediction model using the Transformer indicated that 0.5% of the output was an invalid SMILES. In addition, because the model does not explicitly estimate the reaction center or mechanism of reaction progression, the interpretability of predictions is lower than that of rule-based prediction models and models using graph neural networks.

### 2.3 Computer-assisted synthetic routes design

Given a target molecule, the synthetic routes have been designed by experts. The general strategy is to convert the target molecule into a simpler precursor molecule. When all the precursor molecules

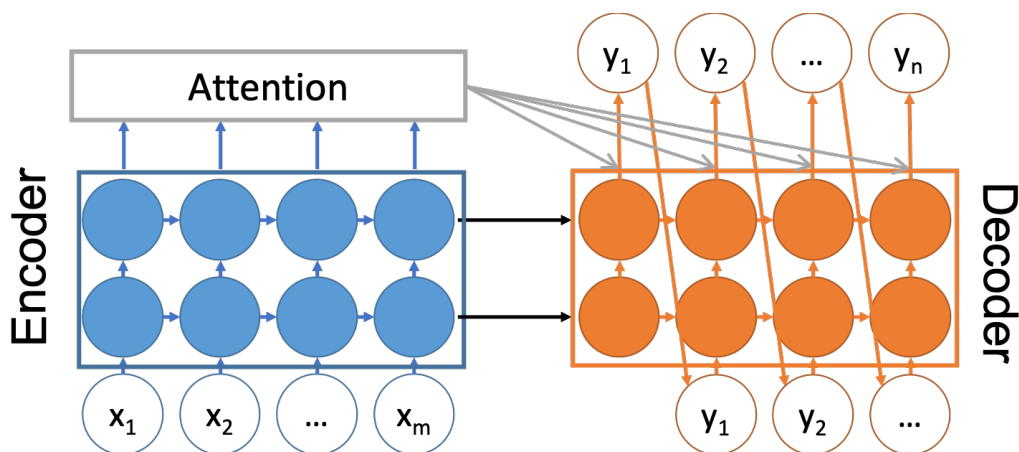


Figure 2.4: Architecture of seq2seq model. The encoder encodes the input to an embedding vector and the decoder decodes the embedding vector to the output. The attention mechanism determines important words in the input to generate a particular word in the output sequence.

Original Smiles Cc1cc(Cl)n(C)n1.O=[N+](O)O>>Cc1nn(C)c(Cl)c1[N+](1/4O)[O-]

Tokenized Smiles C c 1 c c ( C l ) n ( C ) n 1 . O = [ N + ] ( [ O - ] ) O >> C c 1 n n ( C ) c ( C l ) c 1 [ N + ] ( = O ) [ O - ]

Figure 2.5: Tokenization for SMILES

are available for purchase, the design of the synthetic route is completed. If unpurchaseable molecules appeared in the intermediate step, it is further converted into a simple precursor molecule. This method is also named retrosynthesis. The rule-based retrosynthesis software selects rules from the rule set that can synthesize the target compound and predict the reactants, which will serve as precursor molecules. When there are several applicable rules, a rational rule is selected by a scoring function, as in the reaction prediction model. In recent years, scoring functions can achieve higher prediction accuracy by using machine-learning models than those designed by experts. In addition, a rule-set-independent method for predicting precursor molecules directly from a target compound was also proposed, which considers a synthetic route design as a prediction problem. However, there are multiple synthetic pathways to synthesize a single compound. In most cases, the choice of the synthetic pathway is ultimately left to the synthetic scientist, depending on the

available compounds and the experimental environment. Therefore, methods of proposing multiple synthetic routes for experimental implementation are also an important research target.

### **Rule-based synthetic route design and machine learning**

The rule-based synthetic route design uses a rule set similar to the rule-based reaction prediction model. The rules extracted from the reaction data describe the atoms that changed their bonds and other properties before and after the reaction. In the reaction prediction, the given reactants are compared with the structure of the reaction center on the left hand of the reaction equation, which is the structure before the reaction, and the applicable rules are identified. In contrast, in synthetic route design, the target products are compared with the structure of the reaction center on the right hand of the reaction equation, which is the structure post the reaction, and the reactants are inferred using the applicable rules. If there is more than one rule that can be applied, the scoring function is used for ranking. If products with a high degree of similarity are present in the reaction data, Coley et al. proposed a scoring function based on the degree of similarity (Coley, L. Rogers, et al., 2017). Based on the Tanimoto similarity (Tanimoto, 1958) of the target compound and the products in the reaction data, as well as the Tanimoto similarity of the predicted reactants and the reactants in the reaction data, the similarity to the existing reactions was calculated. This was used as a score of the rule for application to a target compound. Such a simple scoring function was able to find reactions that matched the data with an accuracy of 37.3%.

$$\text{Tanimoto}(\mathbf{x}, \mathbf{y}) = \frac{\sum x_i y_i}{\sum x_i^2 + \sum y_i^2 - \sum x_i y_i} \quad (2.15)$$

where  $\mathbf{x}, \mathbf{y}$  are the fingerprint of two molecules.

Dai et al. proposed a graph neural network model-based scoring function (Dai et al., 2019). For the applicable rules, the score of each atom in the reaction center of the target compound was calculated by the neural network and the rule with the highest overall score was selected. The scoring function using the machine-learning model was able to find the reactions to synthesize the target compounds that matched the data with a 52.5% accuracy.

All the above results have the accuracy of models trained on 50,000 reactions extracted from USPTO data (USPTO-50k), of which 80% were used as training data, 10% were used as validation data and 10% were used as test data. However, the USPTO contains more than one million reaction data and as the size of the dataset increases, the size of the rule set is expected to increase, making the design of the scoring function more burdensome and increasing the computation time. To overcome the problems caused by the rule set, we require a rule-set-independent method.

### **End-to-end retrosynthesis**

Retrosynthesis is a method to design a synthetic pathway to a target compound by commencing from a target compound and converting it into simple molecules, which are finally purchasable. The conversion from a molecule that cannot be purchased into a simple precursor molecule can be viewed as a task to predict a reactant from a product. Henceforth, predicting the products from the reactants is a forward prediction in the same direction as the reaction proceeds and predicting the reactants from the products is a backward prediction. The deep neural network model used in forward reaction prediction has also been implemented in retrosynthesis. In particular, a number of end-to-end retrosynthesis methods have been proposed that use the models of machine translation with SMILES. Liu et al. first attempted to predict the SMILES of the reactants from the SMILES of the products by using the model of seq2seq (B. Liu et al., 2017). In the USPTO-50k dataset, the seq2seq model was able to achieve an accuracy of 37.4%. However, since the predictions are using the information of reaction type in addition to the reactants, they cannot be directly compared to other methods. Furthermore, more than 10% of the output string were not conformed to the SMILES grammar. This is a specific challenge of machine translation models. Zheng et al. attempted to improve the prediction accuracy by adding a self-correction feature to another machine translation model, Transformer (Zheng et al., 2020). Without the self-correction function, the prediction accuracy on the USPTO-50k dataset was 43.3%, while the accuracy was improved to 43.7% by correcting the SMILES using a corrector. In addition, another study compared the accuracy by using different SMILES tokenization (Lin et al., 2020).

## 2.4 Multistep synthetic route design

For a multistep synthesis route design, Segler et al. proposed a method combining the deep neural networks and Monte Carlo tree search (MCTS) (Segler, Preuss, and Waller, 2018). Since a target molecule can be synthesized by using different precursor molecules, finding synthetic routes connecting the target molecule and the given purchasable compound set is a pathfinding problem. Tree search algorithms, such as a heuristic best first search, require hand-design heuristic functions to determine the position values in the search tree, which is difficult to design in synthesis planning. For example, since protecting group can increase the complexity, the premise that simpler precursors are better choice is not true. In addition, the position value is dependent on the purchasable molecule set. To evade the problem of the position value function, Segler et al. used MCTS with three neural networks to plan the synthetic routes. For each node in the search tree, the position value was determined by the rollout policy network, which performed a random search without branching until a solution had been found or the maximum depth was reached. The expansion policy network was used to predict the best transformation for the given product. Over 300 thousand rules constituted the reaction rule collection used in the expansion policy network. The top 50 transformations were performed to generate the child nodes. For each child node, the rollout policy network estimates the position values. Seventeen thousand rules constituted the reaction rule collection for the rollout policy network. The target molecule was deconstructed recursively and was called solved if all the building blocks were in the set of available chemical compounds. In addition, the in-scope filter network was used to decide whether the proposed reaction is feasible. Since the predicted reactant by expansion policy sometimes failed to react as predicted, the in-scope filter network was trained to predict whether the reaction corresponding to the transformations selected by policy networks are actually feasible. Since there were no failed reactions recorded in the reaction dataset, a set of negative data were generated by applying the reaction rules to the reactants of reported reactions and the products not matching the reported product were considered as a failed product. The three neural networks were trained by using 12.4 million single-step reactions from Reaxys. In a double AB test, a chemist assessed the synthetic

routes generated by a computer as an equivalent to the recorded synthetic routes.

For planning the syntheses of complex natural products, Mikulak-Klucznik et al. developed a hybrid expert-AI system (Mikulak-Klucznik et al., 2020), which is based on a commercial software named Chemitica (Szymkuć et al., 2016). In contrast to the pure machine-learning approach, a >100,000 hand-coded reaction rule set was used in Chemitica. In the planning of the synthesis of natural products, the stereochemical control is necessary, which is complicated to capture by the machine-learning approach. In Chemitica, the applicability of the hand-coded rules were evaluated by the machine-learning or quantum-chemistry methods, which reviewed the site- or regio-selectivity. Decisions regarding subsequent reaction moves were made by a heuristic or best-in-class neural-network scoring function. A multiple beam-like searches algorithm was used to explore the synthetic space. Several queues were used to keep a given number of encountered nodes with the lowest score at each search depth. Typically, two queues were used, one with a score function that preferred wide searches and another with a scoring function that preferred to go deeply. The first queue discovered a promising beginning of routes, which could be quickly completed with the assistance of the second queue. The total chemical space being evaluated in a search was up to  $O(10^7)$ . Most of the CPU time was not spent on performing the reaction and search operations but on enforcing the proper stereochemistry of the reaction and evaluating the molecule by using the scoring function. In a Turing test administered to synthesis experts, the routes designed by such a program were largely indistinguishable from those designed by humans. Three computer-designed syntheses of natural products were validated in the laboratory.

## **2.5 Molecule design using reaction prediction models**

Machine-learning methods for chemical synthetic planning are used in de novo drug design. Most machine-learning generative approaches cannot warrant the synthesizability of a proposed molecule nor provide the synthesis routes of a proposed molecule. Gottipati et al. proposed a method based on reinforcement learning (Gottipati et al., 2020). In their research, a molecule was considered as a sequence of unimolecular or bimolecular reactions applied to an initial molecule. The state

of the system at each step was a product molecule and the rewards were computed according to the properties of the product. The actions of synthetic step were decomposed in two sub-actions. A reaction template was first selected and was followed by the selection of a reactant compatible with it. The agent consisted of three neural networks:  $f$ ,  $\pi$  and  $Q$ . The  $f$  network predicted the best reaction template  $T_t$  given the current molecule  $R_t$ . The  $\pi$  neural network computed the action  $a_t$  based on the template  $T_t$  and  $R_t$ . The  $Q$  network estimated the Q-value of the state-action pair. The environment took the current molecule  $R_t$ , best reaction template  $T_t$  and action  $a_t$  as the input to compute the reward  $r_t$ , next state  $s_{t+1}$  and a Boolean to determine whether the episode had ended. Since the actions must be reactants in the predefined set,  $k$  reactants from the set that were closest to the predicted action were selected and then passed through a reaction predictor to obtain the corresponding  $k$  products. The reward associated with the product was computed using a scoring function. The reward and product corresponding to the maximum reward were returned. The episode terminated when either the maximum number of reaction steps was reached or when the next state had no valid templates. Gottipati et al. demonstrated this reinforcement learning approach achieved state-of-the-art performance in generating a molecule with several benchmark properties and HIV targets.

Liu et al. also discussed the challenge of unfeasible molecules, which appeared in the de novo molecule generation (C.-H. Liu et al., 2020). They trained deep graph neural networks to approximate the synthetic accessibility score provided by retrosynthesis planning software and used it to bias the search process. To find synthesizable and drug-like antibiotics, they defined three scores: SynthScore, as an estimate of synthesizability; QEDScore, as an estimate of drug-likeness and AntibioticScore, reflecting the probability of being an antibiotic. The score function was defined as

$$\text{score}(x) = \text{AntibioticScore}(x) \cdot \text{QEDScore}(x) \cdot \text{SynthScore}(x) \quad (2.16)$$

The search was performed in a graph edit search space, which included actions such as adding an atom, mutating an atom/bond and adding an aliphatic ring or an aromatic ring. For each



optimization step for the antibiotics, the value of each action was estimated by evaluating the score of the resulting molecules. One of the actions was sampled based on the score. Since the evaluation of the synthetic accessibility score was time-consuming, this approach accelerated the de novo molecule design by filtering out molecules that were difficult to synthesize.

## MATHEMATICAL FORMULATION OF SYNTHETIC ROUTE DESIGN

### 3.1 Ill-posed nature of machine-learning-based retrosynthesis task

Using the backward prediction models, the simulated candidate reactants are rarely contained within a given set of purchasable compounds that span the feasible solution space. Many non-purchasable reactants will likely appear in the intermediate steps of a proposed reaction sequence that are obtained by decomposing a target product in the reverse several times. For example, if a synthetic product is decomposed into A and B by a retrosynthetic prediction model, both the reactants will typically be non-purchasable. In such a case, further identification of the synthesis routes to both A and B will be necessary. However, by restricting one or both reactants to be purchasable, we can make the proposed reaction simpler to implement experimentally. This study aimed to improve the synthetic accessibility of proposed reactions by explicitly restricting the search space to a set of available reactants.

Another shortcoming arises from the limited predictive power of backward reaction models due to the ill-posed nature of the machine-learning-based retrosynthesis task. One reason for this is the loss of information in the reaction data as the side products of synthetic reactions are often unrecorded in the dataset. For example, consider a reaction  $A + B \rightarrow C + D$ . The reaction database often omits the byproduct D, in other words, it records only  $A + B \rightarrow C$ . In a conventional scenario, where the missing structure of D is included in A and B, the prediction of A and B from a given C alone is an ill-posed problem. As summarized in Table 3.1, the previously reported accuracy in the retrosynthetic prediction of known reactants ranged from 37% to 52% in the top-1 accuracy. The ill-posed nature of the retrosynthesis problem is one of the reasons for the limited performance.

---

This chapter is a modified version of Guo et al. Bayesian Algorithm for Retrosynthesis. *J. Chem. Inf. Model.* 2020, 60, 10, 4474–4486 (doi.org/10.1021/acs.jcim.0c00320) published in *Journal of Chemical Information and Modeling*. Copyright 2020 American Chemical Society.

In contrast, the forward prediction of reaction outcomes has achieved substantially better accuracy than retrosynthetic prediction for known reactions (Coley, Barzilay, et al., 2017; Jin, Coley, et al., 2017; Schwaller, Laino, et al., 2019). As summarized in Table 3.1, the reported prediction accuracy is up to 90.4% for the Molecular Transformer, which is almost twice the accuracy of the backward prediction. The retrosynthetic mapping is essentially one-to-many, as there inherently exists multiple synthetic routes to the same product. Therefore, it is inevitable that the accuracy of the retrosynthetic prediction will be much lower than that of the forward prediction. In addition, as mentioned above, the ill-posed nature of the retrosynthetic problem could be causing the significant decline in the prediction accuracy that cannot be improved without reformulating the problem. Here, it should be remarked that it is not meaningful to compare the accuracy of predictions in the forward and reverse directions as they have different purposes. The objective of retrosynthesis is to propose multiple diverse synthetic routes of the target product. However, a higher accuracy of retrosynthetic prediction indicates more bias based on the prevalence of the reaction data, which can result in limiting the diversity of the retrosynthetic suggestion.

Task	Model	Top-1	Top-3	Top-5	Top-10
Backward	similarity (Coley, L. Rogers, et al., 2017) <sup>a</sup>	37.3	54.7	63.3	74.1
	SCROP (Zheng et al., 2020) <sup>b</sup>	43.7	60.0	65.2	68.7
	Lin et al. 2019 (Lin et al., 2020) <sup>b</sup>	43.1	64.6	71.8	78.7
	GLN (Dai et al., 2019) <sup>a</sup>	52.5	69.0	75.6	83.7
Forward	template-based (Coley, Barzilay, et al., 2017) <sup>c</sup>	71.8	86.7	90.8	94.6
	WLDN (Jin, Coley, et al., 2017) <sup>d</sup>	79.6	87.7	89.2	-
	modified WLDN (Coley, Jin, et al., 2019) <sup>d</sup>	85.6	90.5	92.8	93.4
	Molecular Transformer (Schwaller, Laino, et al., 2019) <sup>e</sup>	90.4	94.6	95.3	-

**Table 3.1:** Performances of existing methods on the prediction of synthetic reactions in forward and backward directions. The top-1, top-3, top-5 and top-10 accuracies in [%] are presented for each. Note that different datasets were used in the training and performance evaluations of these models. Symbols in the second column represent references to the source datasets. a: 50k reactions extracted from USPTO granted patents, where the atom mapping for each reaction was kept; b: 50k reactions extracted from USPTO granted patents without atom mappings; c: 15k reactions extracted from USPTO granted patents, where the stereochemical information was kept; d: 500k reactions from USPTO granted patents without stereochemical information; e: 1M reactions from USPTO granted patents, where the stereochemical information was kept.

### 3.2 A two-stage approach for the synthetic route design

Considering the above, we propose a two-stage approach consisting of forward and backward predictions (Figure 3.1). The key concept of this study is quite simple. A trained forward model with high predictability defines the mapping  $Y = f(S)$  from a set of reactants  $S$  to their product  $Y$ . By solving the inverse mapping  $S = f^{-1}(Y^*)$  with a synthetic target  $Y^*$  with respect to possible combinations  $\mathcal{S}$  of commercially available reactants, we could obtain a retrosynthesis algorithm that has a high synthetic accessibility. The problem to be solved is a combinatorial optimization

problem with the solution space subject to the combinatorial complexity of all possible pairs of purchasable reactants in the catalog. The complexity increases exponentially with the number of candidate reactants as well as the number of the reaction steps considered. In this study, we addressed the task of reaction mining within the framework of the Bayesian inference, namely, the Bayesian retrosynthesis, which provides a principled means of inverting any given forward model into a retrosynthetic prediction system. To enhance the search efficiency and exhaustively enumerate alternative pathways, we developed a sequential Monte Carlo (SMC) algorithm using a surrogate model accelerator.

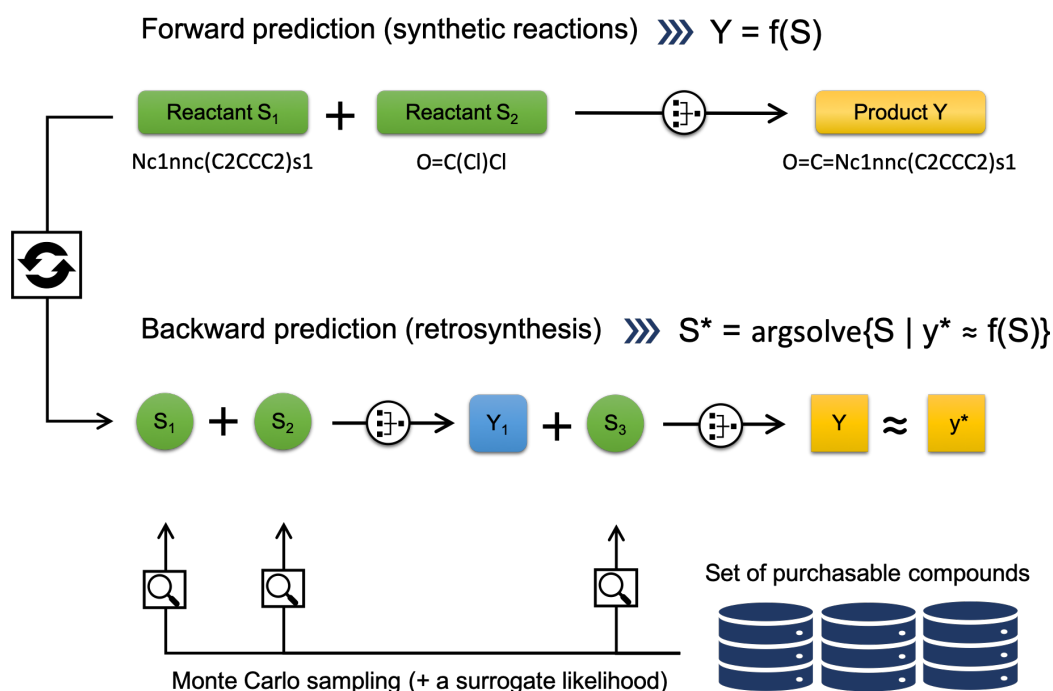
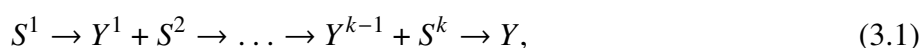


Figure 3.1: Workflow of Bayesian retrosynthesis algorithm. A synthetic reaction model  $Y = f(S)$  that forwardly predicts a product  $Y$  of any given reactants  $S$  is inverted into the backward model using Bayes' law of conditional probability. Monte Carlo sampling from the posterior distribution conditioned by a desired product  $Y = y^*$  provides a diverse set of  $M$  highly probable reactant pairs,  $\mathcal{P} = \{S_1, \dots, S_M\}$  that meet the requirement  $y^* \approx f(S)$ .

### 3.3 Mathematic formulation

In a single-step reaction prediction model  $f(S)$ , a product  $Y$  is described as a function of a set of reactants  $S$ , where  $S$  is composed of  $p$  reactants  $\{m_1, \dots, m_p\}$ . Using such a model, a single-step reaction for any  $S$  can be simulated ( $S \rightarrow Y$ ). Herein, the function  $f$  is treated as deterministic. Solvents, reagents and catalysts can be augmented into the input variable as required.

A  $k$ -step reaction sequence ending with a final product  $Y$  can be generated by convoluting the single-step model  $f(S)$   $k$  times with randomly selected reactant sets  $S^1, \dots, S^k$  at each step:



where  $Y^i = f(Y^{i-1}, S^i)$  denotes an intermediate product from the product  $Y^{i-1}$  at the previous step and the currently selected reactants  $S^i$ . For clarity, the multi-step model is denoted as  $Y = f(S) := f(S^1, \dots, S^k)$ , where  $f(S)$  is a composite function that convolves the single-step reaction  $k$  times and all reactants are concatenated into a sequence  $S = \text{concat}([S^1, \dots, S^k]) = [m_1^1, \dots, m_{p^1}^1, \dots, m_1^k, \dots, m_{p^k}^k]$ . Hereafter, we assume that a fixed-length sequence  $S$  contains  $r = \sum_{i=1}^k p^i$  reactants involved in a  $k$ -step reaction.

The ultimate goal of retrosynthetic prediction is to enumerate all possible  $S \in \mathcal{S}$  satisfying  $y^* = f(S)$  for a given synthetic target  $Y = y^*$ , or equivalently, to solve the inverse mapping of the forward model  $S = f^{-1}(y^*)$ . The solution space  $\mathcal{S}$  is composed of all combinations of the  $r$  reactants that are purchasable. The number of the candidate reactants is typically  $O(10^6)$ , resulting in the cardinality of the solution space expanding to  $|\mathcal{S}| = O(10^{6 \times r})$  because  $r$  reactants are involved in the synthetic route planning.

In certain cases, we may want to identify an ensemble of  $S$  that approximately meets the requirement, i.e.,  $y^* \simeq f(S)$ , rather than obtaining a strict solution. First, it is possible that all candidate reactants will never reach the target product with the given forward model  $f(S)$ . Furthermore, if the model is incorrect, true reactants are expected to be close to the optimal solutions  $y^* \simeq f(S)$ . Technically, all models of machine learning are incorrect, and therefore, the strict solution of the forward model

is not always true in the real world. Retrosynthetic prediction models are essentially tools to facilitate the creativity of synthetic chemists. After a wide variety of candidates are identified, the final decision should be left to synthetic chemists. In such scenarios, it is more beneficial to investigate the distribution of  $S$  as approximately satisfying  $y^* \simeq f(S)$  than to obtain only an strict solution. This is the underlying concept for addressing a retrosynthetic analysis within the Bayesian framework.

Bayesian retrosynthesis relies on Bayes' law of conditional probability:

$$p(S|Y = y^*) \propto p(Y = y^*, S) = p(Y = y^*|S)p(S). \quad (3.2)$$

This law states that the posterior probability distribution  $p(S|Y = y^*)$  is proportional to the joint distribution  $p(Y = y^*, S)$  that consists of the product of the likelihood  $p(Y = y^*|S)$  and prior  $p(S)$ . The forward prediction model forms the joint probability distribution, which is given by the Boltzmann distribution with an inverse temperature  $\beta$ , as follows:

$$p(Y = y^*, S) \propto \exp\left(-\beta E(y^*, f(S))\right). \quad (3.3)$$

The energy function  $E$  is provided by the Euclidean distance or Tanimoto distance (Tanimoto, 1958) between the chemical structures of the target  $Y = y^*$  and the predicted product  $f(S)$ . The distance is calculated with the extended-connectivity fingerprint (ECFP) (D. Rogers and Hahn, 2010) with a diameter of four using the open-source cheminformatics toolkit RDKit in Python. The diversity of the identified solutions can be increased or decreased by setting lower or higher values to  $\beta$ , respectively. In our implementation, we applied an annealing schedule using a non-decreasing sequence of  $\beta$  in the iterative algorithm.

The posterior is a discrete probability distribution  $p(S|Y = y^*) \propto \sum_{i=1}^{|\mathcal{S}|} p(Y = y^*, S) \delta_{s_i}(S)$ , where  $\delta_{s_i}(S)$  denotes an indicator function that takes the value of 1 if  $S = s_i$  and 0, otherwise. The support of this discrete measure consists of all possible combinations of  $r$  reactants involved in a synthetic route. As exact computation across  $|\mathcal{S}|$  candidates is generally infeasible, we explore the

approximated form  $\hat{p}(S|Y = y^*)$  as follows.

$$\hat{p}(S|Y = y^*) \propto \sum_{i=1}^n p(Y = y^*, S) \delta_{s_i}(S). \quad (3.4)$$

The primary objective in the Bayesian computation is to identify the reduced set of  $n$  reactant pairs,  $\{s_i\}_{i=1}^n$ , possibly with  $n \ll |\mathcal{S}|$ . Candidates with greater  $p(Y = y^*, S = s_i)$  should have a better chance of surviving, while those that are ignorable  $s_i$  with low  $p(Y = y^*, S = s_i)$  values are effectively eliminated.



## BAYESIAN INFERENCE ON RETROSYNTHESIS

## 4.1 Difficulties for ordinary heuristic algorithms

Sampling from a posterior distribution over a huge discrete space is an extremely difficult task. Conventional heuristic algorithms cannot be applied to solve this task. To demonstrate this, we first present the simplest form of the SMC algorithm (Moral, Doucet, and Jasra, 2006). The SMC has the same structure as that of genetic algorithms, which have been applied to various machine-learning problems and molecular design tasks (e.g., an inverse analysis of quantitative structure–property relationships.) (Ikebata et al., 2017; Venkatasubramanian, Chan, and Caruthers, 1995; Kawai, Nagata, and Takahashi, 2014; Miyao, Kaneko, and Funatsu, 2016; Douguet, Thoreau, and Grassy, 2000; Lameijer et al., 2006). It begins with an arbitrary set of  $p$  particles,  $\mathcal{P}_0 = \{s_i^0\}_{i=1}^p$ , which represent an ensemble of candidate reactant pairs, followed by the iteration of two operations, hereafter referred to as *sampling* and *resampling*.

---

**Algorithm 2** Simple SMC

---

- 1: Initialize a set of  $p$  particles  $\mathcal{P}_0 = \{s_i^0\}_{i=1}^p$ .
- 2: **for**  $t = 1, \dots, T$  **do**
- 3:     **for**  $i = 1, \dots, p$  **do**
- 4:         Generate a new particle  $s_i^* \sim g(s_i^* | s_i^{t-1})$  according to a proposal distribution  $g$ .
- 5:         Evaluate the importance weight as

$$w_i = \frac{p(Y = y^*, S = s_i^*)}{g(s_i^* | s_i^{t-1})}.$$

- 6:     **end for**
  - 7:     Resample  $\{s_i^*\}_{i=1}^p$  with the probability proportional to  $\{w_i\}_{i=1}^p$  to obtain a new set  $\mathcal{P}_t = \{s_i^t\}_{i=1}^p$ .
  - 8: **end for**
- 

This chapter is a modified version of Guo et al. Bayesian Algorithm for Retrosynthesis. J. Chem. Inf. Model. 2020, 60, 10, 4474–4486 (doi.org/10.1021/acs.jcim.0c00320) published in Journal of Chemical Information and Modeling. Copyright 2020 American Chemical Society.

To reveal the associated technical difficulties, we consider an elementary version of the SMC algorithm, as shown in Algorithm 2. For a given particle set  $\mathcal{P}_{t-1}$  at step  $t - 1$ , a particle  $s_i^{t-1}$  is tentatively replaced with a new  $s_i^*$  according to a prescribed proposal distribution  $g(s_i^*|s_i^{t-1})$ . Specifically, we sample  $s_i^*$  with equal probability from the  $k$ -nearest neighbors of  $s_i^{t-1}$  in the candidate set  $\mathcal{S}$  of available reactant pairs. For the neighbor search, all candidates in  $\mathcal{S}$  and reactants in  $s_i^*$  are fingerprinted by ECFP with a diameter of four, followed by calculation of the Tanimoto distance (Tanimoto, 1958). For each candidate particle, the goodness-of-fit  $w_i$  to the synthetic target, which is referred to as the importance weight, is evaluated by simulating the product using the Molecular Transformer. It should be noted that the importance weight is reduced to  $w_i \propto p(Y = y^*, S = s_i^*)$ , when using the  $k$ -nearest neighbor proposal  $g(s_i^*|s_i^{t-1}) = 1/k$ .  $\{s_i^*\}_{i=1}^p$  is then resampled based on the selection probability proportional to  $\{w_i\}_{i=1}^p$ , which yields a new particle set  $\mathcal{P}_t = \{s_i^t\}_{i=1}^p$ . For a given  $s_i^t$ , as the predicted product moves closer to the given synthetic target, its reactant set has a greater chance of surviving in the resampling step. By repeating the sampling and resampling  $T$  times, we obtain  $n = p \times T$  samples in  $\mathcal{P}_1, \dots, \mathcal{P}_T$  with  $p(Y = y^*, S = s_i)$ , forming the approximated posterior (Eq. 3.4).

The experimental parameters and conditions are summarized in Table 4.1. To evaluate the performance of the simple SMC, we randomly sampled 40 reactions. Each selected reaction consisted of two reactants and one product. We verified that all the 40 reactions were successfully predicted by a pre-trained forward prediction model, called the Weisfeiler–Lehman Difference Network (WLDN) (Jin, Coley, et al., 2017). This model was the state-of-the-art forward prediction model for chemical reactions before the emergence of the Molecular Transformer (Schwaller, Laino, et al., 2019). To further simplify the analysis, one of the two reactants in a reaction was assumed to be known and the simple SMC was applied to search for only one unknown reactant.

This simple SMC performed very poorly as it failed to discover approximately 50% of known one-step reactions in the performance test. This is due to two common drawbacks in conventional heuristic search methods, including genetic algorithms. The first drawback is the underlying

Description	Parameter
Forward model	WLDN (Jin, Coley, et al., 2017)
Particles	One reactant in each particle
Number of reactants $ \mathcal{S} $	325,995
Size of population $p$	100
Number of iterations $T$	1,000
Energy function $E(y^*, f(\mathcal{S}))$	Tanimoto distance
Importance weight $w$	$(1 - E^{\eta_t}) \times \alpha_{\text{WLDN}}$ with $\eta_t = 1 + 9/t$
Number of neighbors $k$	100
Random sampling ratio	0.05

Table 4.1: Parameters and experimental conditions for the simple SMC.  $\alpha_{\text{WLDN}}$  denotes the output probability of the WLDN model for a predicted product.

diversity of the solutions. Our analysis demonstrated that a large number of synthetic routes ended with the same product in the search space. Using an extended algorithm (presented later), an average of over 500 different routes were identified for a synthetic target. In general, it is difficult to identify a diverse set of highly probable solutions comprehensively using ordinary heuristic methods. In the SMC workflow, the repeated resampling induces a rapid loss of diversity among the particle sets, which is known as particle impoverishment (Stavropoulos and Titterton, 2001). The other drawback is the cost of repeated calculations of a reaction prediction model. For example, on average, a single-step reaction prediction using the Molecular Transformer required 30 to 40 ms on a Linux server with an NVIDIA V100 or P100 GPU, thereby leading to a total execution time of over seven hours for  $p = 1,000$  and  $T = 600$ . In summary, a surrogate model-assisted Monte Carlo algorithm was required to save the cost of repeatedly evaluating the computationally expensive reaction prediction model, while maintaining the diversity of particles, including highly probable solutions.

## 4.2 SMC accelerated by surrogate posterior distribution

### Surrogate posterior based on gradient boosting

A key concept behind our strategy is the use of a computationally inexpensive surrogate model, such as a gradient boosting regression (Schapire and Freund, 2012), to approximately evaluate the posterior probability in Eq. 3.3 of the Molecular Transformer for any given reactants with high accuracy. For  $m$  randomly selected instances  $\{s_i\}_{i=1}^m$  of reactant pairs, their posterior probability up to the normalizing constant,  $w_i = p(Y = y^*, S = s_i)$  ( $i = 1, \dots, m$ ), can be calculated using the top-1 proposed product of a forward model or an observed product for any  $s_i$  with a given target  $y^*$ . Using this dataset, we trained a gradient boosting regression tree  $w_{\text{GBM}}(S)$ , which can be used to predict  $w_i$  of  $S = s_i$  without passing through the Molecular Transformer (Figure 4.1a). The ECFPs in the RDKit package were used as inputs for the gradient boosting regression based on the LightGBM package (Ke et al., 2017) in Python. Details on the parameter settings of the gradient boosting regression in the numerical experiments described later are presented in Table 4.2. As the logarithmic posterior probability is proportional to the Euclidean distance between the fingerprinted target product and predicted product of the Molecular Transformer, the gradient boosting surrogate was trained to predict the distance directly from the input reactions. Figure 4.1b illustrates the performance of the surrogate models for two different target molecules. The true reactants are scored as the most likely for each target molecule among the test data. The strong correlation between the predicted and true  $w_i$  made the surrogate posterior distribution a reliable alternative to the true one, which is expensive to calculate.

### Modified SMC with cluster-level sampling

The proposed retrosynthesis algorithm is described in Algorithm 3. It relies on the evolutionary strategy involving the sampling and resampling operations, as in Algorithm 2. Sampling from the proposal distribution is modified in three ways to reduce the computational cost and maintain the diversity of the final candidates. First, the currently provided top- $l$  particles are expanded into  $l \times m$  particles by taking the  $m$ -nearest neighbors of each particle (line 12). This operation aims

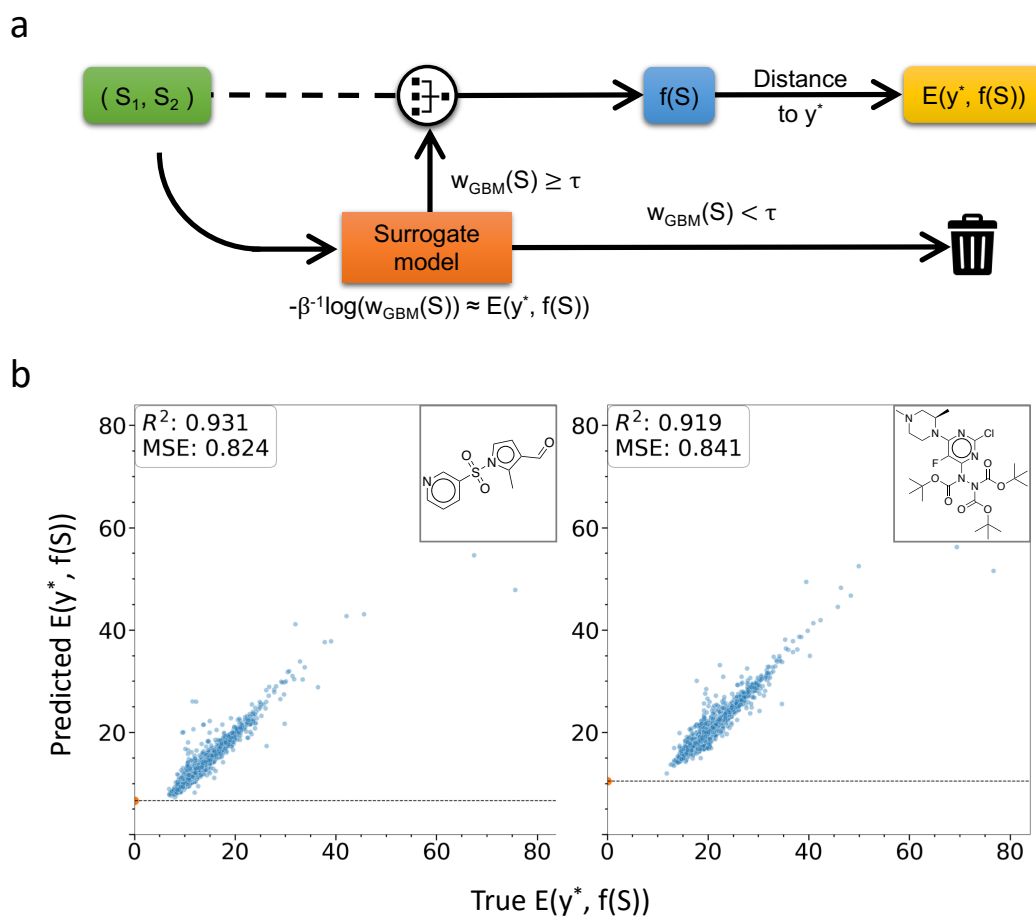


Figure 4.1: **a**. The energies (proportional to the negative logarithmic posterior probability) predicted by a surrogate model are used to prioritize the promising reactants before conducting expensive reaction prediction using the Molecular Transformer. **b**. For two synthetic targets, the predicted energies of surrogate models trained on 40,000 instances of randomly selected reactants from a set of candidate reactants are displayed against the true energies of 5,000 unseen reactants. The orange points represent the ground-truth reactants. The dashed lines indicate the predicted values of the ground-truth reactants. The target molecules are displayed in the top right corners.

Description	Parameter
Model	Gradient boosting machine (LightGBM (Ke et al., 2017))
Boosting type	gradient boosting decision tree
Objective	regression
Metric	l1 and l2
Number of boost rounds	500
Number of leaves	63
Learning rate	0.1
Early stopping rounds	5
Minimum child samples	10
Bagging fraction	1
Feature fraction	0.8

Table 4.2: Hyperparameters used to train the surrogate model.

to increase the heterogeneity of the sample population in the following generation. The fast-to-evaluate surrogate model can efficiently prioritize increasingly heterogeneous particles according to the surrogate posterior (line 13). Moreover, the  $l \times m$  particles are clustered into  $K$  subgroups according to the fingerprints of the chemical structures (ECFPs with a diameter of four; line 14). From each cluster, one particle is sampled to maintain the population diversity (line 15). The surrogate posterior probabilities are transformed into cluster-level goodness-of-fit scores by taking the within-cluster average (line 16). As a result, a new set of particles is created by performing the sampling of the  $l \times m$  particles such that the number of particles in each cluster is proportional to the within-cluster posterior probability (line 17). This sampling improves the efficiency in searching  $\{S \in \mathcal{S} | f(S) \simeq y^*\}$  as giving the priority to reactant pairs with a high surrogate posterior probability. Finally, the set of randomly selected candidates is augmented with the new set of particles (line 19). The sampled particles are excluded from the candidate set to avoid repeated calculation of the Molecular Transformer.

---

**Algorithm 3** Surrogate-accelerated SMC

---

- 1: Initialize a set of  $p$  particles:  $\mathcal{P}_0 = \{s_i^0\}_{i=1}^p$ .
  - 2: Initialize the active candidate set  $\mathcal{A}_1 = \mathcal{S} \setminus \mathcal{P}_0$ , which consists of all reactant pairs in  $\mathcal{S}$  that have not been sampled.
  - 3: Generate a training set  $\mathcal{D} = \{s_i, w_i\}_{i=1}^n$  to construct a surrogate posterior probability function, where  $w_i$  are calculated using the top-1 predicted product of  $s_i$  or an observed product in dataset for each  $s_i$ .
  - 4: Build the surrogate model  $w_{\text{GBM}}(S)$  (gradient boosting) on  $\mathcal{D}$ .
  - 5: Compute the pairwise similarity of all candidate reactants.
  - 6: Set an annealing schedule as  $\beta_0 \leq \beta_1 \leq \dots \leq \beta_T$ .
  - 7: **for**  $t = 1, \dots, T$  **do**
  - 8:     **Resampling based on importance weights**
  - 9:         Calculate  $\mathcal{W}_{t-1} = \{w_i^{t-1}\}_{i=1}^p$  of all entries in  $\mathcal{P}_{t-1}$  with  $\beta_{t-1}$ .
  - 10:         Select the top  $l$  of  $p$  particles from  $\mathcal{P}_{t-1}$  in terms of the importance weights in  $\mathcal{W}_{t-1}$ .
  - 11:     **Sampling  $\mathcal{P}_t$  from a proposal distribution**
  - 12:         Generate  $l \times m$  particles  $\mathcal{P}^* = \{s_i^*\}_{i=1}^{l \times m}$  by taking the  $m$ -nearest neighbors of each  $s_i^{t-1}$  ( $i = 1, \dots, l$ ) selected from  $\mathcal{A}_t$ .
  - 13:         Calculate  $w_{\text{GBM}}(s_i^*)$  of  $s_i^* \in \mathcal{P}^*$ .
  - 14:         By performing  $K$ -means clustering with the fingerprints of  $\mathcal{P}^*$ , the  $l \times m$  temporally proposed particles are grouped into  $K$  non-overlapping clusters  $C_1, \dots, C_K$ .
  - 15:         From each cluster, sample one particle randomly to generate a set  $\mathcal{S}_1^*$ .
  - 16:         Calculate the cluster-level surrogate importance weight as  $W_{\text{GBM}}(C_k) = (1/|C_k|) \sum_{i \in C_k} w_{\text{GBM}}(s_i^*)$ .
  - 17:         Sample  $p/2$  particles from  $\mathcal{P}^*$ , denoted by  $\mathcal{S}_2^*$ , such that the number of particles in each  $C_k$  is proportional to  $W_{\text{GBM}}(C_k)$ . Within each cluster, particles are sampled with probabilities proportional to  $w_{\text{GBM}}(s_i^*)$ .
  - 18:         Generate the remaining  $(p/2 - K)$  particles,  $\mathcal{S}_3^*$ , with equal probabilities from  $\mathcal{A}_t$ .
  - 19:         Set the new particles as  $\mathcal{P}_t = \mathcal{S}_1^* + \mathcal{S}_2^* + \mathcal{S}_3^*$ .
  - 20:     Set  $\mathcal{A}_{t+1} = \mathcal{A}_t \setminus \mathcal{P}_t$
  - 21: **end for**
  - 22: Calculate  $\mathcal{W}_T = \{w_i^T\}_{i=1}^p$  of all entries in  $\mathcal{P}_T$  with  $\beta_T$ .
  - 23: Return  $\mathcal{P}_t$  and  $\mathcal{W}_t$  ( $t = 1, \dots, T$ ) to calculate the approximate posterior with Eq. 3.4.
-

The sampling process in the surrogate-accelerated SMC involved three operations: Sampling one particle from each cluster, sampling based on the surrogate importance weight and random sampling from the candidate set. For each operation, one particle is sampled from a specific proposal, which is determined by the surrogate importance weights and the result of the clustering. The proposal of the sampling process can be approximately represented as a mixture of the three proposals of the three operations described above. It should be noted that the composition method, which is typically used for sampling from a mixture proposal (Rubinstein and Kroese, 2007), uses an auxiliary discrete random variable to determine the component to sample from. In the surrogate-accelerated SMC, each operation in the sampling process is performed a predetermined number of times: Once for sampling a particle from each cluster,  $p/2$  times for the surrogate importance weight-based sampling and  $p/2 - K$  times for the random sampling. Using the predetermined number of times as the mixture weights for the three components, the proposal of the surrogate-accelerated SMC can be approximated as

$$p(s_*^t | \mathcal{P}_{t-1}) \simeq \begin{cases} \frac{1}{p} \frac{1}{|C(s_*^t)|} + \frac{1}{2} \frac{1}{\sum_{k=1}^K W_{\text{GBM}}(C_k)} \frac{w_{\text{GBM}}(s_*^t)}{|C(s_*^t)|} + \frac{p/2-K}{p} \frac{1}{|\mathcal{A}_t|} & \text{if } s_*^t \in \Omega(\mathcal{P}_{t-1}) \\ \frac{p/2-K}{p} \frac{1}{|\mathcal{A}_t|} & \text{if } s_*^t \in \mathcal{A}_t \setminus \Omega(\mathcal{P}_{t-1}) \\ 0 & \text{otherwise} \end{cases} \quad (4.1)$$

where  $\Omega(\mathcal{P}_{t-1}) = \bigcup_{i=1}^l NN(s_i^{t-1}) \cap \mathcal{A}_t$  is the nearest neighbors of  $s_i^{t-1}$  ( $i = 1, \dots, l$ ) that have the top  $l$  importance weights in  $\mathcal{P}_{t-1}$ , and  $C(s_*^t)$  represents the set of particles belonging to the same cluster as  $s_*^t$ . The probabilities of the particles in the nearest neighbors of the previous generation consist of the three terms (the first equation in E.q 4.1). The first term denotes the probability of sampling  $s_*^t$  as the representative from cluster  $C(s_*^t)$ . It is inversely proportional to the number of particles in the same cluster. Therefore, the particles in minor clusters have a greater chance to survive in surrogate-accelerate SMC. The second term denotes the probability of sampling based on surrogate importance weights. Considering the summation over the cluster  $C(s_*^t)$ ,  $C(s_*^t)$  with large  $W_{\text{GBM}}(C(s_*^t)) = \frac{\sum_{s_i \in C(s_*^t)} w_{\text{GBM}}(s_i)}{|C(s_*^t)|}$  will have more particles to be sampled. In other words, the



surrogate-accelerated SMC gives the priority to sampling from the area in the reaction space with high surrogate importance weights. For all particles in the candidate set  $\mathcal{A}_t$ , random sampling is performed with equal probabilities, which is the third term in the first equation and appears in the second equation for  $s_i^t \in \mathcal{A}_t \setminus \Omega(\mathcal{P}_{t-1})$ .

The experimental parameters and conditions for one-step and two-step retrosynthesis are summarized in Table 4.3 and Table 4.4, respectively. Note that the pairwise similarity of all given candidate reactants is computed before running the SMC (line 5). While the SMC is running, it only reads the already calculated similarity values to select the neighboring particles. In our implementation (see the Results section for details), a GPU was used to compute the pairwise similarity. It took approximately three hours to compute the pairwise similarity of 600k reactants. It should be noted that in practice, it is not necessary to include all the millions of reactants in the candidate list (Nicolaou, Watson, Hu, et al., 2016; Yu and Bakken, 2009). Based on the chemical substructure of a given target product, we can exclude seemingly unrelated reactants from the candidate list in advance. Similarly, we can rule out blacklisted reactants.

Description	Parameter
Forward model	Molecular Transformer (Schwaller, Laino, et al., 2019)
Particles	One reactant in each particle for $t < 100$ Two reactants in each particle for $100 \leq t < 600$
Number of reactants $ \mathcal{S} $	637,637
Size of population $p$	1,000
Number of iterations $T$	600
Energy function $E(y^*, f(S))$	Euclidean distance
Importance weight $w$	$\exp\{-\beta_t E\}$ with $\beta_t = 1/\max\{1, 20/(t \bmod 100 + 0.1)\}$
Number of selected particles $l$	100
Number of neighbors $m$	100
Random sampling ratio	0.5

Table 4.3: Parameters and experimental conditions for one-step retrosynthesis using the surrogate-assisted SMC algorithm.

Description	Parameter
Forward model	Molecular Transformer (Schwaller, Laino, et al., 2019)
Particles	Three reactants in each particle
Number of reactants $ \mathcal{S} $	637,637
Size of population $p$	2,000
Number of iterations $T$	1,000
Energy function $E(y^*, f(S))$	Euclidean distance
Importance weight $w$	$\exp\{-\beta_t E\}$ with $\beta_t = 1/\max(1, 20/(t \bmod 200 + 0.1))$
Number of selected particles $l$	200
Number of neighbors $m$	100
Random sampling ratio	0.5

Table 4.4: Parameters and experimental conditions for two-step retrosynthesis using the surrogate-assisted SMC algorithm.

### 4.3 Ranking and prioritization

In the result section, we demonstrate that the surrogate-accelerated SMC algorithm generally discovers an excessive number of hypothetical routes to a given product — several hundred in many cases for one-step reactions. The majority of these candidates are chemically unrealistic and are false discoveries, which may result from the unavailability of failed reactions or data of low-yielding reactions in the machine-learning workflow. Indeed, in the following applications, our expert chemists determined that approximately 65% of the proposed solutions were irrational due to exceedingly low or no reactivity in the identified reaction routes. Here, we consider two means of prioritizing more promising candidates by using a heuristic ranking method or reaction-type grouping.

The ranking method scores a given sequence  $S$  of reactants as

$$\gamma(S) = \alpha(S) \max\{p(\{Y_S, S\} \in C_1), \dots, p(\{Y_S, S\} \in C_{10})\}, \quad (4.2)$$

where  $\alpha(S)$  represents the output probability of the Molecular Transformer with respect to a candidate  $S$ ,  $p(\{Y_S, S\} \in C_i)$  indicates the probability of  $\{Y_S, S\}$  belonging to a prescribed known

reaction class  $C_i$ . In this study, we considered 10 reaction classes defined by Schneider et al. (Schneider, Stiefl, and G. A. Landrum, 2016), as described in Table 4.5. A total of 50k reactions from the United States Patent and Trademark Office (USPTO) dataset (Daniel Mark Lowe, 2012) were categorized into the 10 reaction classes. A sparse logistic regression model (Friedman, Hastie, and Tibshirani, 2010) was trained on a randomly selected 80% of the dataset, where the ECFPs of  $Y_S$  and  $S$  were calculated with diameter 4 and concatenated to obtain a descriptor. The prediction accuracy reached 97.3% on the test set. The trained model was used to evaluate  $p(\{Y_S, S\} \in C_i)$  and the most probable reaction class was selected to define the score. The underlying strategy of the hand-designed heuristic score is that candidate reactions that are highly discriminable into one of the known reaction classes are expected to be reliable. This ranking method aims to assign a high priority to candidates that are close to the already known response types. A similar idea was presented in a recently published paper (Nicolaou, Watson, LeMasters, et al., 2020) where they used building block collections of the available reaction data to develop a model to recommend synthetic routes matching a precedent-derived template. It should be noted that the purpose of the ranking is only to assign priority when examining a large number of candidates under limited resources in practice. However, in some cases, it would be unnecessary to prioritize higher-ranked ones. We should examine the distribution of all the proposed candidates to see the various potential routes to the desired products.

Another method to select promising candidates is based on the grouping of the reaction patterns. To visualize the distribution of the identified synthetic routes in a low-dimensional space, t-distributed Stochastic Neighbor Embedding (t-SNE) (Maaten and Hinton, 2008) was performed on the augmented fingerprints of all given  $S$  which was projected onto a two-dimensional space. In addition, X-means clustering (Dau Pelleg, 2000) has been performed on t-SNE projections to automate the determination of the number of clusters and the grouping of reaction patterns, as shown in Figure 4.2. For each cluster, we selected a representative reaction that exhibited the best within-cluster score. In this manner, we could infer the number of different types of synthetic routes,

which potentially existed or were feasible to design with a given set of purchasable compounds.

Reaction class	Reaction name	Number of reactions
1	heteroatom alkylation and arylation	15,122
2	acylation and related processes	11,913
3	C-C bond formation	5,639
4	heterocycle formation	900
5	protections	650
6	deprotections	8,353
7	reductions	4,585
8	oxidations	814
9	functional group interconversion (FGI)	1,834
10	functional group addition (FGA)	227

Table 4.5: Summary of 10 reaction classes and number of recorded reactions in each reaction class (B. Liu et al., 2017).

## 4.4 Results

### Data

We used a collection of 50k single-step reactions (Schneider, Stiefl, and G. A. Landrum, 2016) extracted from nine million patent applications and issued patents from 1976 to 2016. This dataset has served as a benchmark set in previous studies to evaluate retrosynthesis methods (B. Liu et al., 2017; Coley, L. Rogers, et al., 2017; Zheng et al., 2020; Lin et al., 2020). In this study, the dataset was used to train a forward model and to create ground-truth sets of one-step and two-step reactions, as described below. Following a previous study (B. Liu et al., 2017), we divided the dataset into 80% for training, 10% for validation and 10% for testing. Note that the recorded reactions were classified into 10 reaction classes according to an expert system (Schneider, Stiefl, and G. A. Landrum, 2016) (Table 4.5). While most existing methods have employed one of the pre-defined reaction classes as the input for a retrosynthetic prediction system to reduce the solution space, we applied the Bayesian algorithm with no such prior knowledge. For a candidate set of reactants, we used approximately 600k reactants from a dataset called USPTO\_STEREO, which was cleaned by Schwaller et al. (Schwaller, Laino, et al., 2019) and used to train the original Molecular Transformer. The solution space  $\mathcal{S}$  was spanned by all possible combinations of the 600k reactants.

### Forward prediction model

We used a pre-trained Molecular Transformer (Schwaller, Laino, et al., 2019), which was trained on a dataset from USPTO. This attention-based neural translation model defines the translation between the SMILES strings of reactants and their products. For simplicity, the reagents were removed from the input. Any number of reactants could be an input and they were separated by “.” symbols. All the reactions were canonicalized using RDKit. The inputs were tokenized with the regular expression according to a previous study (Schwaller, Laino, et al., 2019). With direct application to our test set, the distributed pre-trained model achieved a top-1 accuracy of 70.7% and a top-5 accuracy of 86.4%. However, with a fine-tuned model using the training and validation data, the top-1 and top-5 accuracies reached 86.9% and 95.5%, respectively. The prediction accuracy

was slightly lower than the best benchmark performance reported thus far (top-1 accuracy of 90.4% and top-5 accuracy of 95.3% for the Molecular Transformer).

It should be noted that there was a risk of data leakage in the fine-tuned forward model since our test set included a part of the training set for the pretrained Molecular Transformer. To stabilize the training of the Molecular Transformer, the public pretrained model was used as a starting point for training and we performed early stopping based on the accuracy on the validation set. In addition, the USPTO reactants were used to form  $S$  for demonstration but some of these reactants were also included in the training data of the forward model. Therefore, the prediction accuracy of this forward model should have an upward bias for the reactants in the USPTO dataset. In the interpretation of the performance evaluation presented below, it is important to note that the reported accuracy on the ground-truth sets represent the performance of the backward prediction method under the condition that a forward prediction model is given with an accuracy of approximately 87%. One of the objectives here is to confirm that the backward prediction can reach almost the same level of accuracy on the ground-truth sets as in the forward direction.

## **Implementation**

The surrogate-accelerated SMC was implemented in Python (version 3.6.8), coupled with RDKit and scikit-learn. The Molecular Transformer was developed using PyTorch was plugged into the forward model (Schwaller, Laino, et al., 2019). All experiments were run on the AI Bridging Infrastructure (ABCI) at National Institute of Advanced Industrial Science and Technology. The computation was performed with 25 nodes and 100 NVIDIA Tesla V100 devices. We used the supercomputer system to conduct repeated experiments under different initial conditions for performance evaluation across many target products. In terms of a single run on the one-step or two-step retrosynthetic prediction of one product, the average computation times were approximately six and thirty hours, respectively. Most of the computation time was spent on hundreds of thousands of calculations of the Molecular Transformer. Detailed experimental conditions, such as the number of particles and number of steps in the SMC, are presented below.

	Number of reactions	Detection of reactants ending with target product [%]	Inclusion of ground-truth reactants [%]
Random1000	1,000	98.0	92.0
MT-predictable	855	99.9	98.1

Table 4.6: Performance of the surrogate-accelerated SMC on 1,000 randomly selected ground-truth reactions (“Random1000”). “MT-predictable” denotes the subset of 855 reactions that the Molecular Transformer could forwardly predict their products as top-1 candidates. The average success rate for detecting one or more synthetic routes ending with each target product is indicated in the third column. The fourth column indicates the percentage of target products whose ground-truth reactants were detected out of the 1,000 target products or 855 products predictable by the Molecular Transformer.

### One-step retrosynthesis

For the one-step retrosynthesis prediction, we randomly selected 1,000 test reactions from the test set consisting of one or two reactants. Among the 1,000 reactions, the Molecular Transformer could predict the true products as top-1 candidates for 855 reactions. The number of particles was set to 1,000. For the surrogate-accelerated SMC, we selected the 1,000 initial particles completely at random from  $\mathcal{S}$ . However, in practice, the candidates of the initial particles can be narrowed down substantially based on the structural similarity to the given synthetic targets. In the first 100 steps of the SMC, each particle consisted of one reactant to explore the one-reactant reaction space. The reaction space of two reactants was explored in the subsequent 500 steps, in which a combination of two reactants constituted a particle. The surrogate-accelerated SMC was used to perform a total of 600,000 ( $= 600 \times 1,000$ ) evaluations of the forward reaction models in each test reaction, corresponding with approximately 0.0001% of the entire search space  $|\mathcal{S}| = O(10^{11})$ . For 98.0% of the 1,000 target molecules, the Bayesian retrosynthesis algorithm proposed one or more solutions exactly ending with each target. The ground-truth reactants were found in the identified solutions for 92.0% of the 1,000 targets. Focusing on the 855 reactions for which the Molecular Transformer could forwardly predict their products as top-1 candidates, the ground-truth reactants were found with a 98.1% success rate (Table 4.6).

In this experiment, the Bayesian retrosynthesis algorithm revealed an average of over 500 synthetic routes for each design target. Figure 4.2 shows the diversity of the detected reaction routes on two

synthetic targets. The ECFPs  $\phi(S_1), \dots, \phi(S_m)$  of  $m$  reactants in  $S$  were reduced to an augmented descriptor as  $\phi(S) = \sum_{i=1}^m \phi(S_i)$  and these were embedded in the two-dimensional subspace for visualization using the t-SNE algorithm. The distribution of the projected reactants indicates that there are multiple motifs of synthetic routes to the same product. A comprehensive detection of the candidate synthetic routes and their visualization-based summary may help facilitate chemists' investigations and decision-making in synthetic route design.

In some cases, it would be difficult to investigate the validity of all such candidate routes. Here, we demonstrate the use of the ranking method to narrow them down to a small number of prioritized candidates. For each of the 1,000 ground-truth reactions, we obtained a ranked list of the top- $N$  most probable candidates based on their ranking and investigated whether to include the ground-truth reactants. As summarized in Table 4.7, the top- $N$  accuracy of the Bayesian retrosynthesis was slightly lower than or comparable to the previously reported accuracy of the Conditional Graph Logic Network (GLN) (Dai et al., 2019). This is one of the current best performing models and it outperformed other models for any  $N \in \{1, 3, 5, 10\}$ . In particular, 75.2% and 81.8% of the top-5 and top-10 accuracies were achieved by our method, respectively. Within the 855 ground-truth reactions for which the Molecular Transformer successfully predicted the products in the forward manner, the top-5 and top-10 accuracies reached 84.1% and 90.9%, respectively. For the GLN, we conducted an additional experiment: The model was trained from scratch using the same dataset as for the Bayesian retrosynthesis and we evaluated the accuracy for the same 1,000 ground-truth reactions. The top-5 and top-10 accuracies were 74.9% and 81.9%, respectively, which were similar to those of our method.

Furthermore, to investigate the sensitivity to different reaction classes, we experimented with an additional test set on the one-step retrosynthesis. Since the number of reactions attributed to each reaction class was highly unbalanced, we randomly selected 50 reactions from each class to form the test datasets to avoid bias towards a certain class. Class 10 had only 23 reactions from the 5,000 test data, so all 23 reactions were selected for this test. With this test set of 473 reactions,



the top-10 accuracies of the proposed method for each of the 10 classes were evaluated. The results are summarized in Table 4.8, along with those of the four existing algorithms. Table 4.8 also provides the forward prediction accuracy of the Molecular Transformer (Schneider, Stiefl, and G. A. Landrum, 2016) as a reference. We observed a positive correlation across the reaction classes between the retrosynthetic prediction performance of the proposed method and forward prediction performance of the Molecular Transformer, which is implemented in our method as the forward model.

According to the comparison presented in Table 4.7, our method also outperformed other existing methods using the reaction class explicitly as an extra input in the retrosynthetic prediction. With a given reaction class, such prediction models can reduce the reaction space to a focused region to enhance the prediction performance. However, such prior knowledge is rarely available in real applications. As a reference, we performed modified Bayesian retrosynthesis. Given a true reaction class  $C_i$  and its class probability  $p(\{Y_S, S\} \in C_i)$ , we calculated the ranking score as  $\gamma(S) = \alpha(S)p(\{Y_S, S\} \in C_i)$  instead of taking the maximum with respect to the classification probabilities, as discussed in the Methods section. The top-5 and top-10 accuracies were further improved to 80.3% and 85.6%, respectively, for the 1,000 ground-truth reactions (Table 4.7).

When considering the results of the comparative experiments, it is important to note that the Bayesian retrosynthesis made its prediction in a predefined search space composed of combinations of the given reactants. However, the search spaces of the other methods were much larger, and the prediction task can be made more or less difficult. The primary objective of this study was to provide a diverse set of synthetic routes using randomly selected candidate reactants to improve the synthetic feasibility rather than finding the so-called ground-truth reactions.

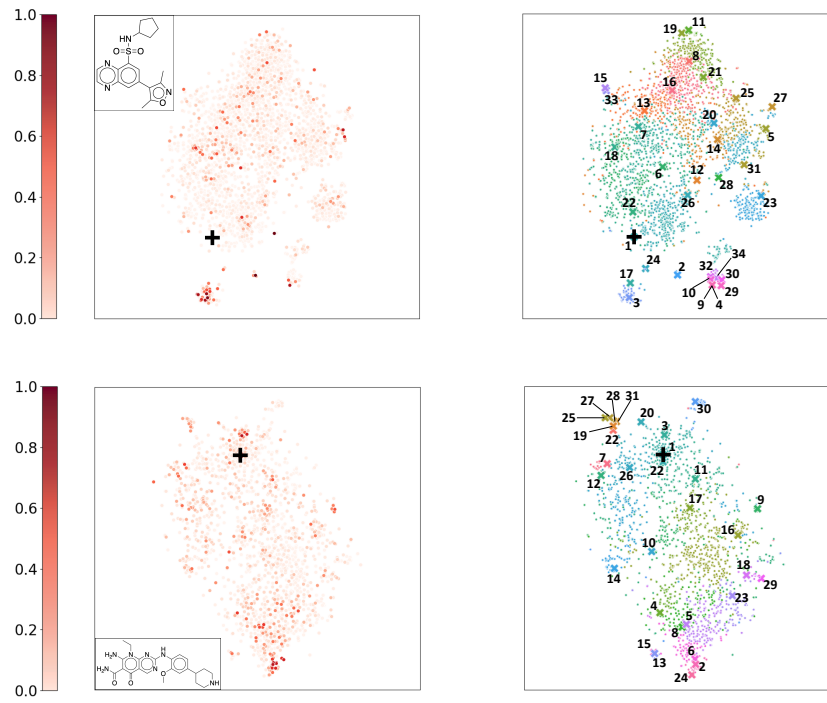
Data	Model	Top-1	Top-3	Top-5	Top-10
Without reaction class	similarity (Coley, L. Rogers, et al., 2017)	37.3	54.7	63.3	74.1
	SCROP (Zheng et al., 2020)	43.7	60.0	65.2	68.7
	Lin et al. 2020 (Lin et al., 2020)	43.1	64.6	71.8	78.7
	GLN (Dai et al., 2019)	<b>52.5</b>	<b>69.0</b>	<b>75.6</b>	<b>83.7</b>
	Bayesian-Retro	47.5 (0.92)	68.1 (0.87)	75.2 (1.10)	81.8 (1.00)
	GLN-Random1000	52.7	67.4	74.9	81.9
	Bayesian-Retro (MT-predictable)	53.8 (1.06)	76.5 (0.90)	84.1 (1.40)	90.9 (1.16)
With reaction class	baseline (B. Liu et al., 2017)	35.4	52.3	59.1	65.1
	seq2seq (B. Liu et al., 2017)	37.4	52.4	57.0	61.7
	similarity (Coley, L. Rogers, et al., 2017)	52.9	73.8	81.2	88.1
	SCROP (Zheng et al., 2020)	59.0	74.8	78.1	81.1
	Lin et al. 2020 (Lin et al., 2020)	54.6	74.8	80.2	84.9
	GLN (Dai et al., 2019)	<b>64.2</b>	<b>79.1</b>	<b>85.2</b>	<b>90.0</b>
	Bayesian-Retro	55.1 (1.08)	74.9 (1.04)	80.3 (1.02)	85.6 (0.92)
GLN-Random1000	62.9	78.1	82.7	87.7	
Bayesian-Retro (MT-predictable)	62.1 (1.17)	83.4 (1.05)	88.8 (1.17)	94.3 (1.06)	

Table 4.7: performance of various retrosynthetic prediction methods with or without reaction class labels. “Bayesian-Retro” denotes the proposed method; “MT-predictable” denotes the performance on the 855 ground-truth reactions with their products forwardly predictable by the Molecular Transformer. The top-1, top-3, top-5 and top-10 accuracies in [%] are indicated for each case. Previously reported performances from the literature are shown here, except for that of “GLN-Random1000”, where a retrained GLN was used to predict the same 1,000 ground-truth reactions in the present study. Moreover, we repeatedly calculated the accuracy of the proposed method over 100 reactions, which were taken from the 1,000 ground-truths 10 times. The corresponding standard deviations are shown in parentheses.

Model	1	2	3	4	5	6	7	8	9	10
baseline (B. Liu et al., 2017)	77.2	84.9	53.4	54.4	6.2	26.9	74.7	68.4	46.7	73.9
seq2seq (B. Liu et al., 2017)	57.5	74.6	46.1	27.8	80.0	62.8	67.8	69.1	47.3	56.5
similarity (Coley, L. Rogers, et al., 2017)	86.7	94.2	74.6	67.0	<b>97.1</b>	<b>95.5</b>	<b>88.3</b>	<b>98.8</b>	71.2	<b>91.3</b>
Lin et al. 2020 (Lin et al., 2020)	83.1	90.4	76.2	60.0	92.3	88.6	88.2	86.4	<b>73.9</b>	82.6
Bayesian-Retro	<b>90.0</b>	<b>96.0</b>	<b>82.0</b>	<b>76.0</b>	90.0	86.0	68.0	38.0	70.0	65.2
Molecular Transformer (top-1)	88.0	98.0	94.0	84.0	94.0	88.0	68.0	76.0	78.0	78.3

Table 4.8: Top-10 accuracies (%) of five retrosynthetic prediction methods on 10 different reaction classes. The last row shows the top-1 forward prediction accuracy of the Molecular Transformer on each of the 10 classes.

**a**



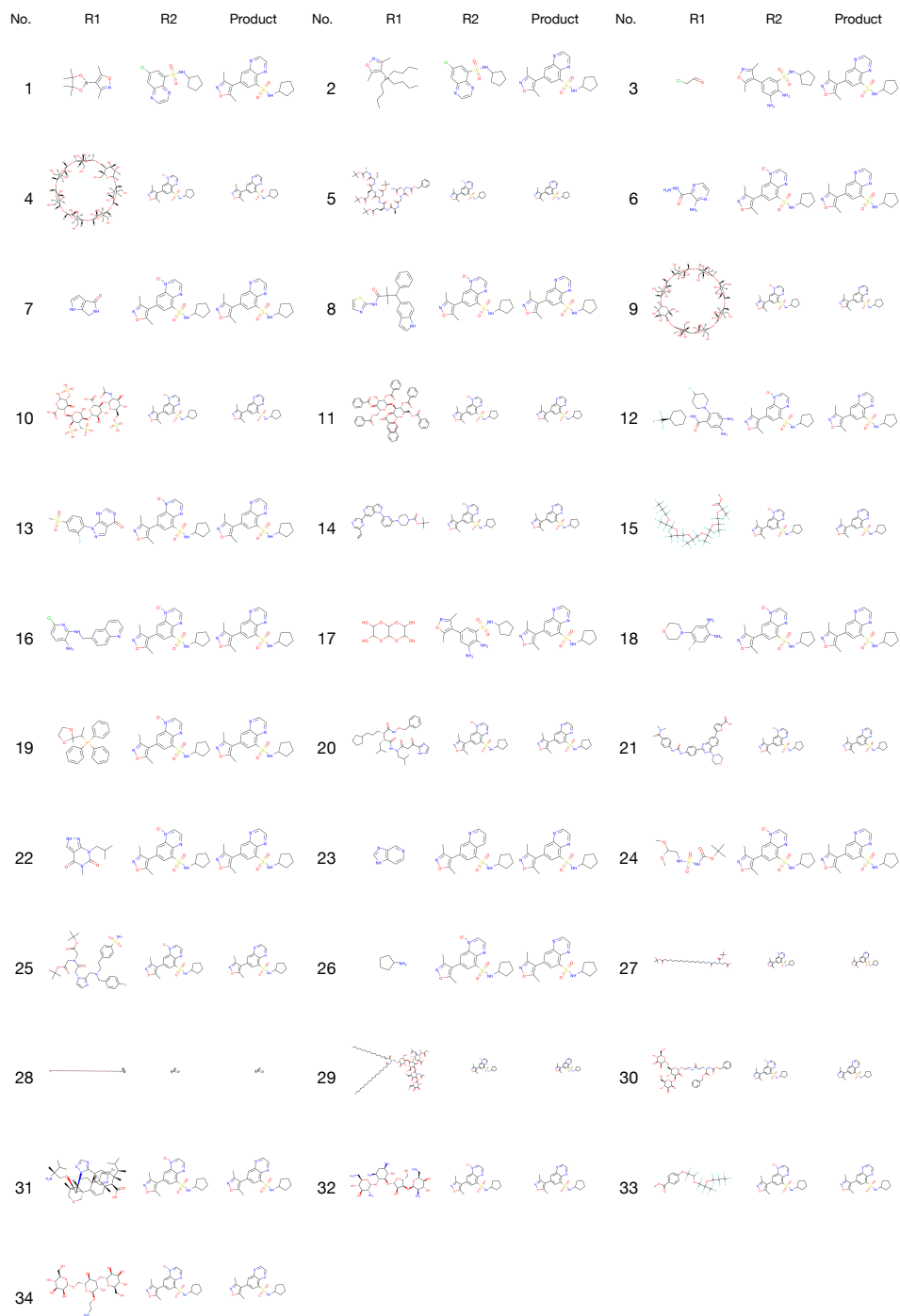
**b**



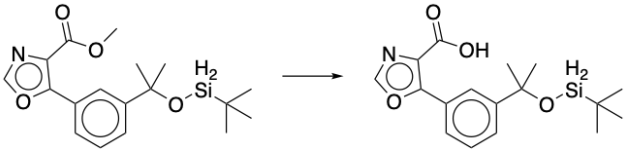
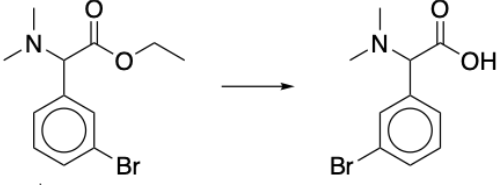
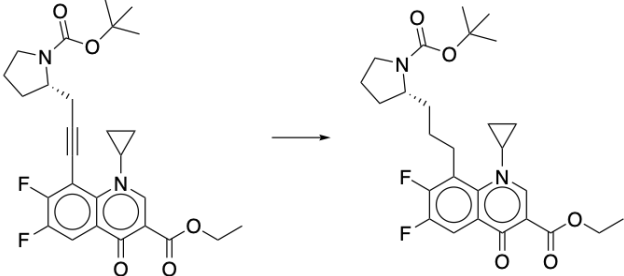
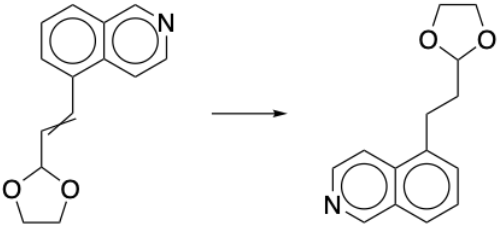
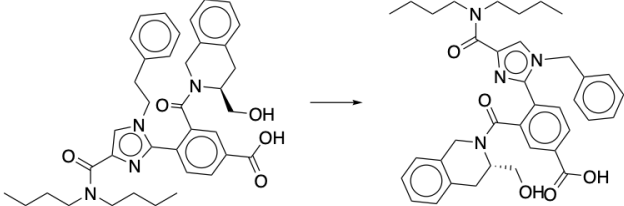
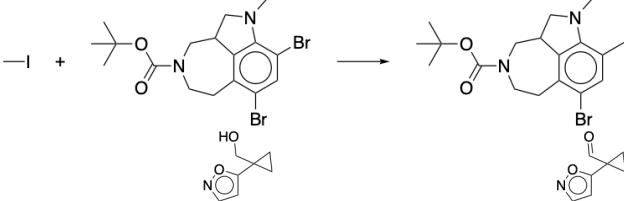
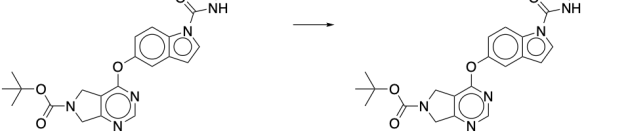
Figure 4.2: Results of the surrogate-accelerated SMC for the two synthetic targets. **a.** t-SNE projection and X-means clustering of the candidate reactions. + denotes the ground-truth reactions. In the left panel, the data points are color-coded according to the ranking scores, normalized to [0, 1]. In the right panel, the clusters identified by X-means are indicated in different colors. × denotes the candidate reactions with the highest score in each cluster. The corresponding reactions are shown in **b** and **c**. **b.** 34 candidate reactions with highest scores, which correspond to the 34 clusters in the top panel of **a**. **c.** 32 candidate reactions with highest scores, which correspond to the 32 clusters in the bottom panel of **a**.

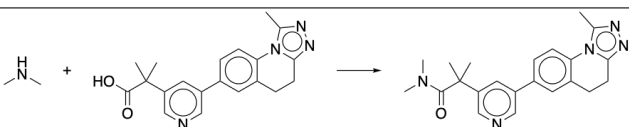
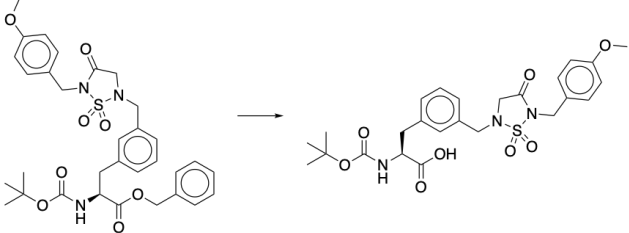
To investigate when the surrogate-accelerated SMC fails in finding the ground-truth synthetic routes, we randomly selected 100 reactions and applied the Bayesian retrosynthesis algorithm 10 times with different initial points for each reaction. Table 4.9 shows that the surrogate-accelerated SMC succeeded in most reactions regardless of the initial points. For the nine reactions which the surrogate-accelerated SMC failed, the Molecular Transformer failed to predict the true product for all except one (Table 4.10).

Num. of successful trials	0	1	2	3	4	5	6	7	8	9	10
Num. of reactions	9	0	1	1	1	0	0	1	1	1	85

Table 4.9: Number of times that the surrogate-accelerated SMC found the recorded synthetic route in 10 trials.

Table 4.10: Reactions for which the surrogate-accelerated SMC failed to find the ground-truth synthetic route. “MT-predictable” denotes if the product can be predicted by the Molecular Transformer or not.

No.	Reaction	MT-predictable
1		False
2		False
3		False
4		False
5		False
6		False
7		False

No.	Reaction	MT-predictable
8		True
9		False

To elucidate the failure of the Bayesian retrosynthesis algorithm in reaction No. 8, we plotted the prediction of the surrogate model for reaction No. 8 and another successful reaction (Figure 4.3). For all other 99 reactions, the predicted energies of the ground-truth reactants were lower than most other reactant pairs, which do not generate the target product. For reaction No. 8, the predicted energy of the ground-truth reactants was located near the mode of the predicted energies of randomly selected reactant pairs. In the surrogate-accelerated SMC, the surrogate model modified the proposal distribution to the sample reactants with a high surrogate posterior efficiently, and the Molecular Transformer was used to predict products of sampled reactants. The posterior probability were calculated based on the predicted product. Therefore, the surrogate-accelerated SMC was able to find the ground-truth reactants in limited steps if the surrogate posterior probability of the ground-truth is a high amount of the generated reactant pairs. If the surrogate posterior probability of the ground-truth is lower than many other reactant pairs, the surrogate model worked as a filter to omit reactant pairs with a low surrogate posterior probability, which may include the ground-truth reactants. In other words, it is indispensable for the surrogate-accelerate SMC to find the true reactants in a few steps that the predicted posterior of the ground-truth by the surrogate model is higher than most random generate reactant pairs.



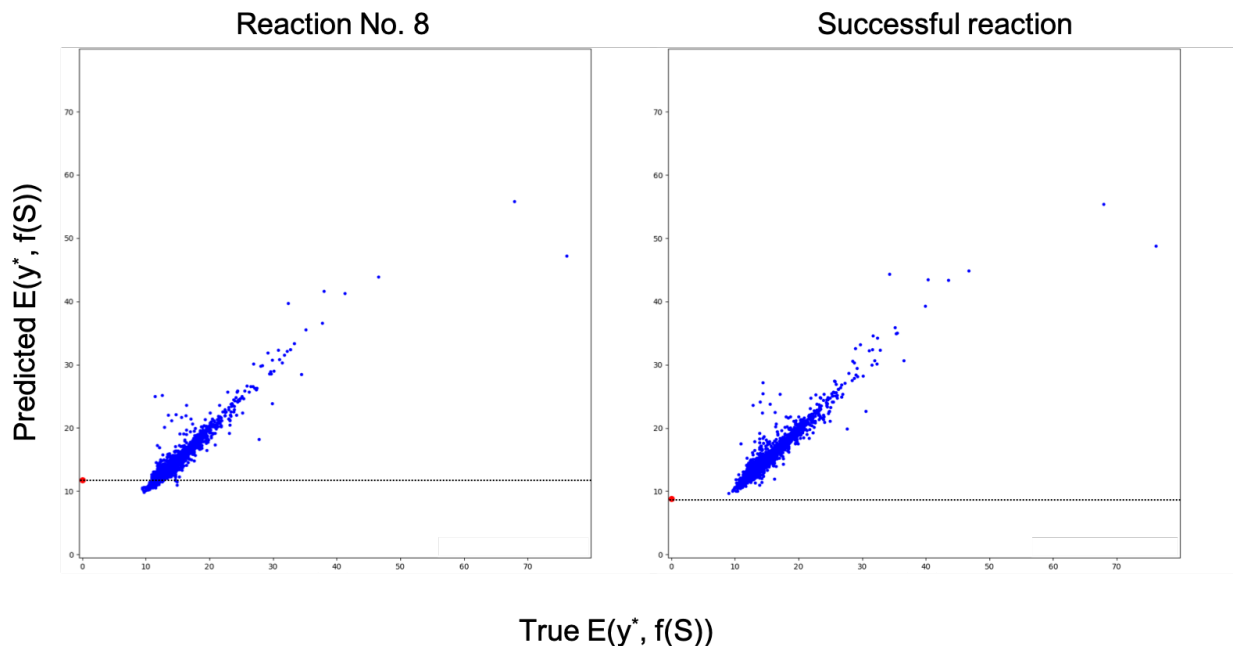


Figure 4.3: The predicted energy of surrogate models for reaction No. 8 in Table 4.10 and another successful reaction. The orange points represent the ground-truth reactants. The dash lines indicate the predicted values of the ground-truth reactants.

### Multi-step retrosynthesis

The surrogate-accelerated SMC was applied to the design of the two-step synthetic routes. Here, we considered the two-step reactions of the form  $S^1 + S^2 \rightarrow X + S^3 \rightarrow Y$ . Here,  $X$  is the intermediate product made by the first step and  $Y$  represents the desired final product. The three reactants  $S = \text{concat}([S^1, S^2, S^3])$  were treated as unknown. By combining the single-step reactions that were predictable with the Molecular Transformer in the test reaction set and USPTO\_STEREO reaction set, we generated a ground-truth set of the two-step reactions as follows: If a product of a recorded reaction appeared in a different reaction as a reactant, the two reactions were connected to form a two-step synthetic route. The product of the former reaction was involved as a reactant of the second-step reaction as an intermediate product. A total of 21 two-step reactions were constructed. Since the database may contain incorrect reactions or reactions with omitted intermediate steps, the extracted reaction sequences were verified by our expert chemists, as summarized in Table 4.11.

According to their evaluations, in which unrecorded reagents and reaction conditions were inferred based on experts' knowledge, reactions 3, 19 and 21 were determined to be chemically unrealistic. However, instead of excluding them from the ground-truth set, we tested whether the Bayesian retrosynthesis could determine alternative synthetic routes to the target product.

The Bayesian retrosynthesis algorithm was performed 10 times for each reaction. The number of particles was set to 2,000 and each particle consisted of two and one reactants in the first and second reactions, respectively. We randomly selected the initial particles from  $\mathcal{S}$ . The number of steps in the SMC was set to 1,000, representing a total of  $2 \times 10^6$  searches in each test case. This number corresponds to approximately  $10^{-11}$  of the entire search space. In 16 of the 21 reactions, the recorded reactants could be successfully identified at least once among the 10 repeated tests. An average of over 16,000 candidate routes were identified for each target product. Hence, we aggregated all the candidate synthetic routes and performed the ranking procedure. The recorded reactions were ranked as top-10 candidates in six cases and as top-100 in 13 cases (Table 4.11).

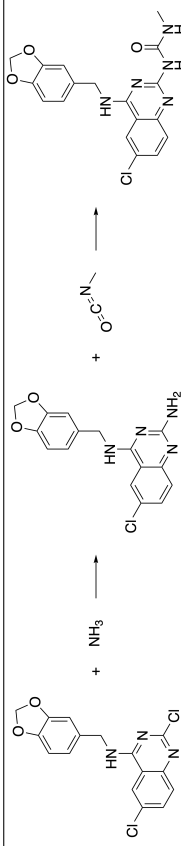
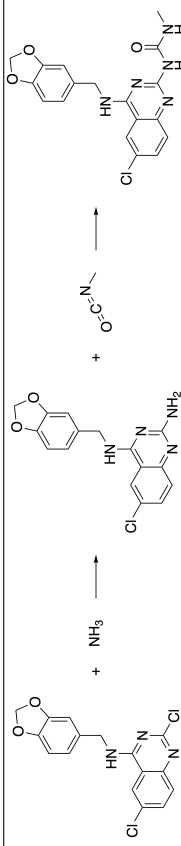
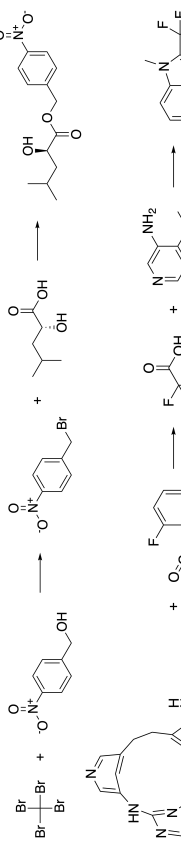
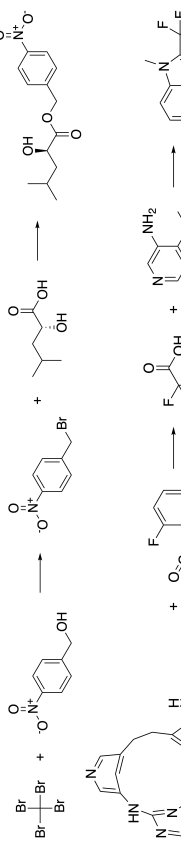
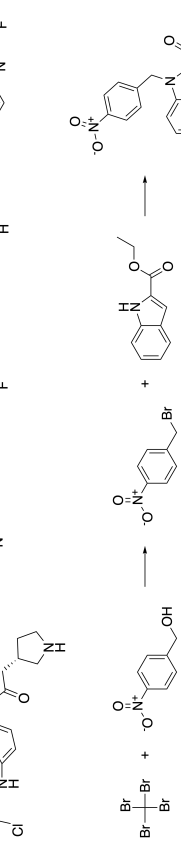
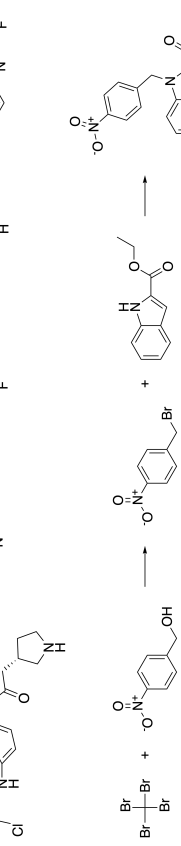
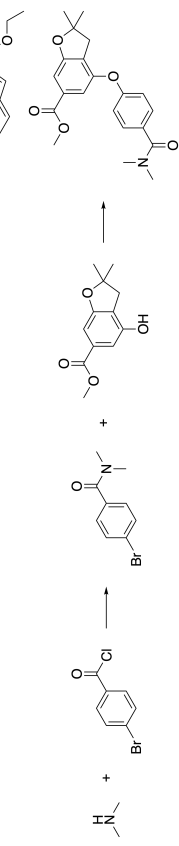
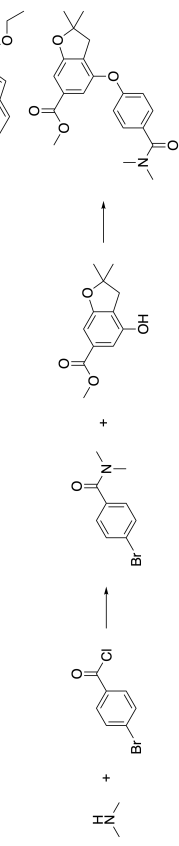


To investigate the kinds of reaction routes captured by the two-step retrosynthetic predictions, we selected the most representative candidate synthetic route from each cluster in all 21 test cases. Specifically, after performing the X-means clustering, the highest ranked candidate in each cluster was selected and the top 10 candidates were selected from among them. According to the assessment made by the expert chemists, 35% of the identified synthetic routes were determined to be valid.

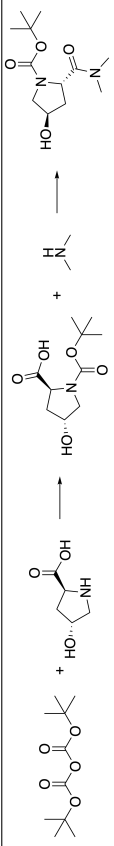
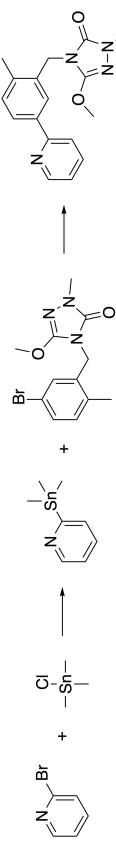
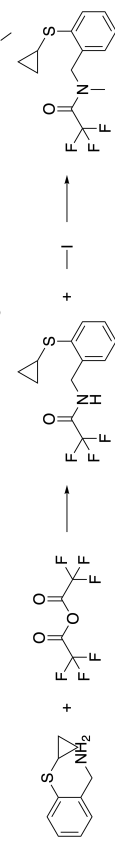
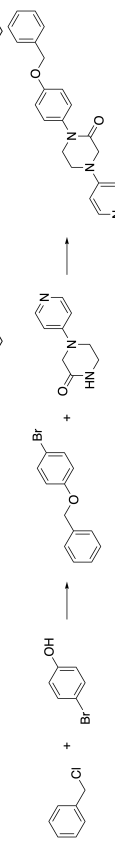
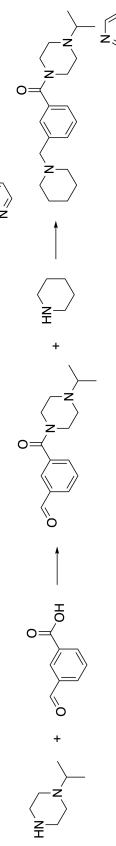
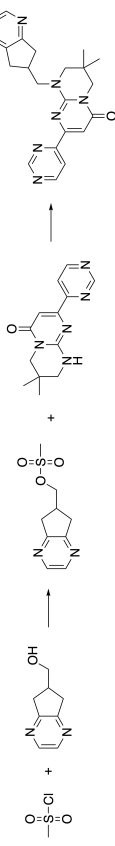
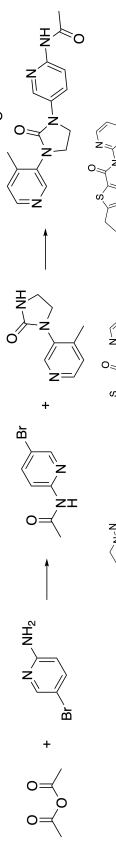
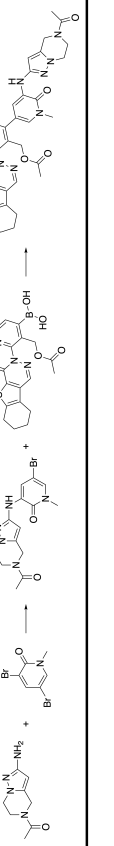
To observe the distribution of the candidate synthetic routes, the t-SNE projection of the detected reactants for reaction nine is presented in Figure 4.4a. Candidate reactants closer to the recorded reactants exhibited higher scores. To identify the different motifs in the candidate synthetic routes, the X-means clustering was applied to the ECFPs of the reactants, which were grouped into 98 clusters. We investigated the synthesis feasibility of the 10 proposed reactions exhibiting the highest score in each of the 10 clusters (Figure 4.4a). According to evaluations by expert chemists, seven of the 10 proposed routes would be chemically reactive and synthesizable. In candidate routes one to three, the first and second steps are known as Williamson ether synthesis (Williamson, 1852)

and palladium-catalyzed coupling reaction (Miyaura and Suzuki, 1995), respectively. In candidate route nine, the second step involves ether synthesis. It should be emphasized that ranking only by the score does not always reveal such promising synthetic routes. It is important to extract a diverse set of candidates based on the clustering procedure to enhance the investigations of the chemists.

For ground-truth reaction three, which the chemists determined to be chemically infeasible, the Bayesian retrosynthesis algorithm identified 2,280 alternative synthetic routes to the target molecule (Figure 4.5). Based on our ranking and clustering procedures, 10 synthetic routes were selected and two reactions with different side-chain modifications were determined to be reactive and synthesizable based on the chemists' evaluations (Figure 4.5b). A ring-forming synthesis was also proposed by the algorithm (route indicated at the top of Figure 4.5c). Although the orthoformic acid monoester used in the first step is chemically unstable, it is expected that the proposed synthetic route will be beneficial for chemists when considering alternative synthetic routes. In fact, a different ring-forming synthetic route was manually designed using formaldehyde instead of the orthoformic acid monoester.

**Table 4.11:** Ground-truth set consisting of two-step synthetic routes to 21 targets. The third and fourth columns show the results of the chemists' validation for each reaction step. The validity was classified into one of {1, 2, 3} as follows: 1: feasible, 2: contentious, or 3: infeasible. The Bayesian retrosynthesis was performed 10 times for each reaction. The fifth column shows the total number of candidate routes ending with the target molecule in the 10 trials. T and F indicate the presence or absence, respectively, of the recorded reaction in the detected synthetic routes. The final column indicates the rank of the ground-truth predicted by the ranking algorithm.

No.	Reaction		Step 1	Step 2	Num. of candidates	Ground truth	Rank
1			1	1	784	F	-
2			1	1	1,326	T	2
3			3	1	2,280	F	-
4			1	1	18,197	T	5
5			1	1	28,388	T	94

No.	Reaction	Step 1	Step 2	Num. of candidates	Ground Truth	Rank
6		1	1	48,601	T	3
7		1	1	7,622	T	45
8		1	1	2,167	T	17
9		1	1	6,613	T	6
10		1	1	3,122	T	23
11		1	1	4,838	T	8
12		1	1	7,273	T	13
13		1	1	35,195	F	-

No.	Reaction	Step 1	Step 2	Num. of candidates	Ground Truth	Rank
14		1	1	4,445	T	82
15		1	1	25	F	-
16		1	1	7,222	T	61
17		1	1	447	T	6
18		1	1	17,003	T	2067
19		1	3	19,333	T	2276
20		1	1	125,322	T	321
21		1	3	3,313	F	-

## 4.5 Availability

A Python implementation of the Bayesian retrosynthesis algorithm is available at GitHub (Guo et al., 2020). All results shown in the Results section can be reproduced by following the instructions. The program is applicable to any numbers of reaction steps and reactants.

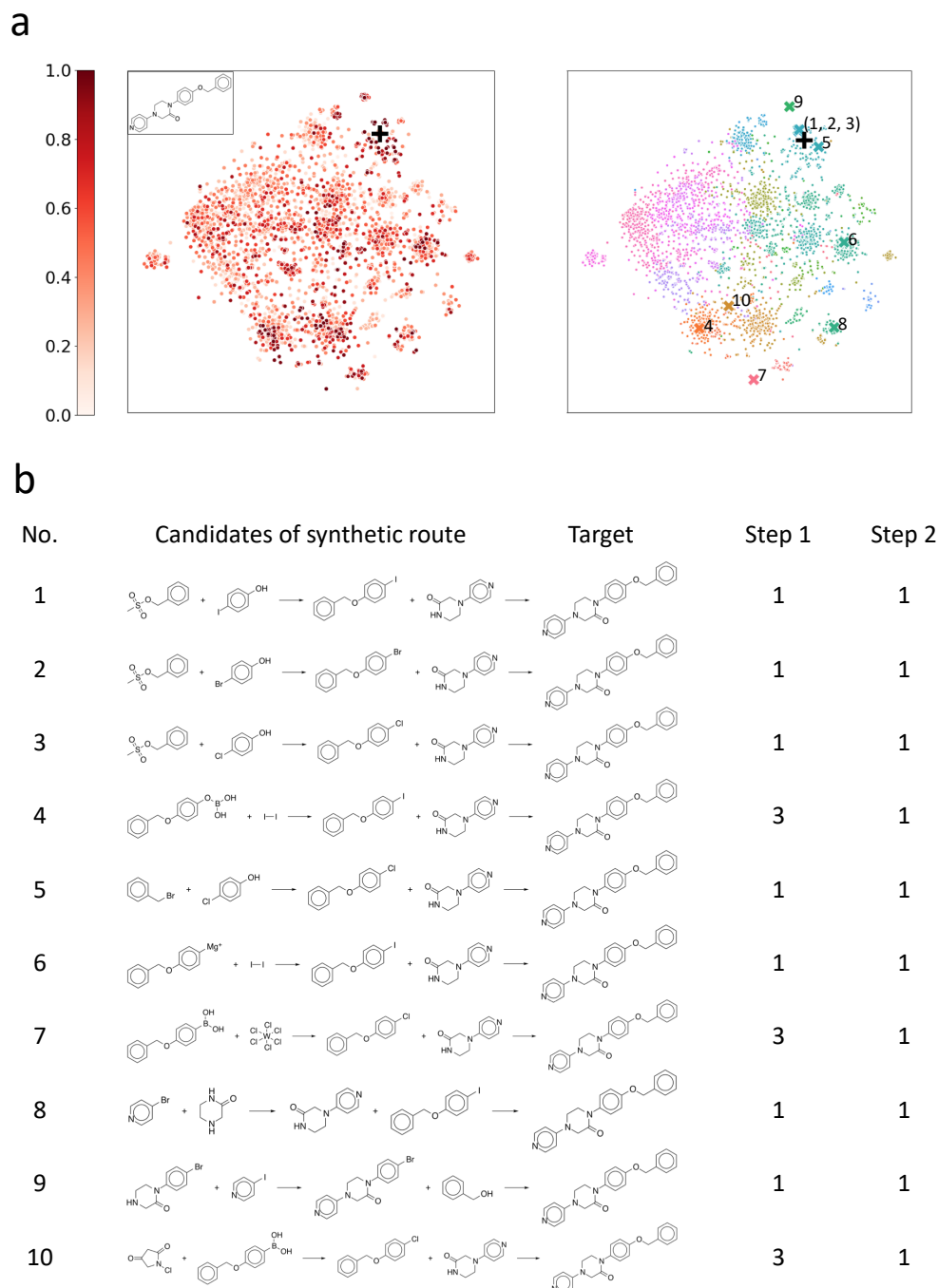


Figure 4.4: **a.** t-SNE projection of 6,613 candidates for two-step synthetic routes to target product in reaction 9, where + denotes the ground-truth reaction route. In the left panel, the data points are color-coded according to the scores, normalized to [0, 1]. The right panel shows that the X-means clustering classified the 6,613 candidate routes into 98 groups, which are mapped onto the t-SNE plot. The identified clusters are indicated in different colors. × denotes the 10 candidate routes presented in **b.** **b.** 10 candidate routes belonging to different clusters. A score from {1, 2, 3} is assigned to each reaction step, indicating 1: feasible, 2: contentious, or 3: infeasible, as judged by expert chemists.



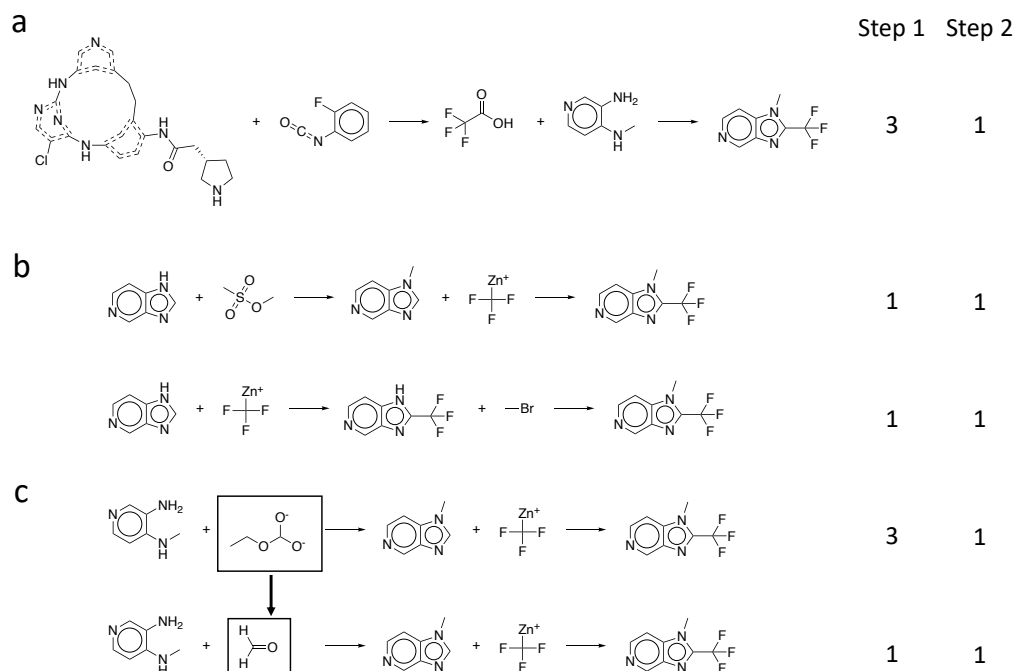


Figure 4.5: **a**. Two-step reaction 3 in test dataset. The scores {1, 2, 3} represent the same as in the previous figure. **b**. Two alternative synthetic routes proposed by the Bayesian retrosynthesis algorithm. **c**. The top reaction is a ring-forming synthetic route proposed by the Bayesian retrosynthesis algorithm. The first step is infeasible because the orthoformic acid monoester is chemically unstable. The bottom reaction is a ring-forming synthetic route suggested by chemists based on the proposed synthetic route indicated at the top.

## CONCLUSIONS

In this thesis, we developed a Bayesian retrosynthesis algorithm. Most previous studies have focused on the direct backward prediction of reaction inputs from a target product. In general, backward prediction is significantly more difficult than the forward prediction as the model must reconstruct several building blocks of reactant molecules that are generally missing in the target product. Moreover, most reactants resulting from such backward predictions are non-purchasable, making the reactants themselves synthetic targets. To overcome these obstacles, we first re-posed the problem of retrosynthetic prediction. We obtained a forward reaction model to define the mapping from the reactants to the products, achieving a high level of accuracy. Thereafter, a retrosynthetic prediction was performed by exploring the inverse mapping from a target product to a pair of reactants in the given forward model, where all possible combinations of purchasable compounds spanned the feasible solution space.

The most important contribution of this study is the development of a methodology that can improve the prediction capability of retrosynthesis by improving the performance of forward prediction models. Thus far, the problems of forward and backward prediction on synthetic reactions have been independently studied. In this paper, we demonstrated Bayesian retrosynthesis using the Molecular Transformer; however, any forward model can be applied. In the near future, a forward prediction model exhibiting higher accuracy is expected to emerge. In addition, the computation speed of forward prediction models, which is a bottleneck in the computation of the Bayesian retrosynthesis, may be improved. This will enable us to apply the proposed method to the design of more multi-step reaction pathways.

The Bayesian retrosynthesis algorithm revealed the presence of numerous alternative routes to the target product. These routes were programmed in the trained reaction prediction model. The

identification of such diverse candidate routes may help to stimulate the researchers' ideas in organic synthesis. However, our expert chemists concluded that almost 65% of the proposed two-step reactions would be false discoveries owing to no or exceedingly low reactivity. The prediction of the presence or absence of reactivity is currently beyond the capability of any synthetic prediction models since these models are trained only on highly reactive instances which are extracted from published data. The lack of negative data on failed and low-yield reactions prevents us from obtaining effective machine-learning models capable of determining the presence or absence of reactivity in a candidate synthetic route. Several previous studies generated artificial negative examples by perturbing and shuffling reported known reactions. In this study, we introduced a heuristic rule for the ranking and prioritization of the candidate reaction routes. However, none of these methods can solve the problem on a fundamental level. Eventually, a comprehensive dataset of negative reactions must be created from experimental observations in laboratory synthesis, the literature, chemists' hand-coded heuristics and/or high-throughput quantum chemistry calculations.

## FUTURE PROSPECTS

In the Bayesian retrosynthesis algorithm, the computational cost of the forward model Molecular Transformer is high; therefore, we developed an efficient search method by introducing the surrogate model. As the average computing times for one-step or two-step synthetic route design of one target molecule were approximately six and thirty hours, respectively, a low-cost forward prediction model is necessary for its practical use. In particular, as the number of the reaction steps increase, the number of reactants increase and the search space increases exponentially. The time required is also expected to increase exponentially.

In this study, we defined the problem of chemical synthesis planning as a discrete optimization problem and proposed an approach in the Bayesian framework; however, different approaches also exist. One commonly used approach is beginning from the target compound and converting it to a simple precursor molecule recursively to determine the synthetic route, based on the concept of retrosynthesis. In this approach, dynamic optimization can be used, however, it is expected that the algorithm will have a lengthy duration as the number of steps increases since there are an enormous amount of combinations of compounds for chemical reactions and the prediction model is time-consuming. In addition, due to the difficulty in defining the evaluation function of the state in chemical synthesis planning, algorithms, such as MCTS and beam search have been proposed to overcome these problems (section 2.4, Segler, Preuss, and Waller, 2018; Gottipati et al., 2020).

The Bayesian retrosynthesis algorithm proposed multiple synthetic routes, which can assist experts in chemical synthesis planning. However, some of the proposed routes are not valid. In this study, we developed a ranking method based on the similarity to existing reactions and used clustering to determine reactions of the different types. Since it is necessary to consider the availability of the raw materials, cost and reaction efficiency for practical use, a comprehensive evaluation method

including other data is essential to identify the effective synthetic route.

Finally, we hope that we can conduct demonstration experiments on useful synthetic routes discovered by the Bayesian algorithm for retrosynthesis. It will have a huge impact on chemistry and machine learning if new reaction routes can be identified by computer-assisted synthetic planning.

- [1] Nicola De Cao and Thomas Kipf. “MolGAN: An implicit generative model for small molecular graphs”. In: CoRR abs/1805.11973 (2018). URL: <http://arxiv.org/abs/1805.11973>.
- [2] Raymond E. Carhart, Dennis H. Smith, and R. Venkataraghavan. “Atom pairs as molecular features in structure-activity studies: definition and applications”. In: J. Chem. Inf. Comput. Sci. 25 (1985), pp. 64–73. DOI: 10.1021/ci00046a002.
- [3] Connor W. Coley, Regina Barzilay, et al. “Prediction of organic reaction outcomes using machine learning”. In: ACS Cent. Sci. 3 (2017), pp. 434–443. DOI: 10.1021/acscentsci.7b00064.
- [4] Connor W. Coley, Wengong Jin, et al. “A graph-convolutional neural network model for the prediction of chemical reactivity”. In: Chem. Sci. 10 (2019), pp. 370–377. DOI: 10.1039/C8SC04228D.
- [5] Connor W. Coley, Luke Rogers, et al. “Computer-assisted retrosynthesis based on molecular similarity”. In: ACS Cent. Sci. 3 (2017), pp. 1237–1245. DOI: 10.1021/acscentsci.7b00355.
- [6] E. J. Corey and W. Todd Wipke. “Computer-assisted design of complex organic syntheses”. In: Science 166 (1969), pp. 178–192. DOI: 10.1126/science.166.3902.178.
- [7] Hanjun Dai et al. “Retrosynthesis Prediction with Conditional Graph Logic Network”. In: Adv. Neural Inf. Process Syst. (2019), pp. 8872–8882. URL: <http://papers.nips.cc/paper/9090-retrosynthesis-prediction-with-conditional-graph-logic-network.pdf>.
- [8] Daniel Mark Lowe. “Extraction of chemical structures and reactions from the literature”. PhD thesis. 2012. DOI: 10.17863/CAM.16293.
- [9] Andrew Moore Dau Pelleg. “X-means: Extending k-means with efficient estimation of the number of clusters”. In: Proceedings of the 17th International Conference on Machine Learning. 2000, pp. 727–734. URL: <https://www.cs.cmu.edu/~dpelleg/download/xmeans.pdf>.
- [10] Dominique Douguet, Etienne Thoreau, and Gérard Grassy. “A genetic algorithm for the automated generation of small organic molecules: Drug design using an evolutionary algorithm”. In: J. Comput.-Aided Mol. Des. 14 (2000), pp. 449–466. DOI: 10.1023/A:1008108423895.
- [11] Joseph L. Durant et al. “Reoptimization of MDL Keys for Use in Drug Discovery”. In: J. Chem. Inf. Comput. Sci. 42 (2002), pp. 1273–1280. DOI: 10.1021/ci010132r.
- [12] Jerome H. Friedman, Trevor Hastie, and Rob Tibshirani. “Regularization Paths for Generalized Linear Models via Coordinate Descent”. In: J. Stat. Softw. 33 (2010), pp. 1–22. DOI: 10.18637/jss.v033.i01.

- [13] Johann Gasteiger et al. “Computer-assisted synthesis and reaction planning in combinatorial chemistry”. In: Perspect. Drug Discovery Des. 20 (2000), pp. 245–264. doi: 10.1023/A:1008745509593.
- [14] H. L. Gelernter et al. “Empirical explorations of SYNCHEM”. In: Science 197 (1977), pp. 1041–1049. doi: 10.1126/science.197.4308.1041.
- [15] Sai Krishna Gottipati et al. “Learning to Navigate The Synthetically Accessible Chemical Space Using Reinforcement Learning”. In: vol. 119. Proceedings of Machine Learning Research. 2020, pp. 3668–3679. URL: <http://proceedings.mlr.press/v119/gottipati20a.html>.
- [16] Zhongliang Guo et al. A Bayesian algorithm for retrosynthesis. 2020. URL: [https://github.com/zguo235/bayesian\\_retro](https://github.com/zguo235/bayesian_retro).
- [17] Lowell H. Hall and Lemont B. Kier. “Electrotopological State Indices for Atom Types: A Novel Combination of Electronic, Topological, and Valence State Information”. In: J. Chem. Inf. Comput. Sci. 35 (1995). Publisher: American Chemical Society, pp. 1039–1045. doi: 10.1021/ci00028a014.
- [18] Hisaki Ikebata et al. “Bayesian molecular design with a chemical language model”. In: J. Comput.-Aided Mol. Des. 31 (2017), pp. 379–391. doi: 10.1007/s10822-016-0008-z.
- [19] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. “Junction Tree Variational Autoencoder for Molecular Graph Generation”. In: vol. 80. Proceedings of Machine Learning Research. 2018, pp. 2323–2332. URL: <http://proceedings.mlr.press/v80/jin18a.html>.
- [20] Wengong Jin, Connor W. Coley, et al. “Predicting organic reaction outcomes with Weisfeiler-Lehman network”. In: Adv. Neural Inf. Process Syst. (2017), pp. 2608–2617. URL: <https://papers.nips.cc/paper/6854-predicting-organic-reaction-outcomes-with-weisfeiler-lehman-network>.
- [21] Kentaro Kawai, Naoya Nagata, and Yoshimasa Takahashi. “De Novo Design of Drug-Like Molecules by a Fragment-Based Molecular Evolutionary Approach”. In: J. Chem. Inf. Model. 54 (2014), pp. 49–56. doi: 10.1021/ci400418c.
- [22] Guolin Ke et al. “LightGBM: A highly efficient gradient boosting decision tree”. In: Adv. Neural Inf. Process Syst. (2017), pp. 3147–3155. URL: <https://papers.nips.cc/paper/6907-lightgbm-a-highly-efficient-gradient-boosting-decision-tree>.
- [23] Justin Klekota and Frederick P Roth. “Chemical substructures that enrich for biological activity”. In: Bioinformatics 24 (2008), pp. 2518–2525. doi: 10.1093/bioinformatics/btn479.
- [24] Eric-Wubbo Lameijer et al. “The Molecule Evuator. An Interactive Evolutionary Algorithm for the Design of Drug-Like Molecules”. In: J. Chem. Inf. Model. 46 (2006), pp. 545–552. doi: 10.1021/ci050369d.

- [25] Greg Landrum et al. RDKit: Open-source cheminformatics. 2006. URL: <https://www.rdkit.org>.
- [26] Kangjie Lin et al. “Automatic Retrosynthetic Route Planning Using Template-Free Models”. In: Chem. Sci. (2020), pp. 3355–3364. DOI: 10.1039/C9SC03666K.
- [27] Bowen Liu et al. “Retrosynthetic reaction prediction using neural sequence-to-sequence models”. In: ACS Cent. Sci. 3 (2017), pp. 1103–1113. DOI: 10.1021/acscentsci.7b00303.
- [28] Cheng-Hao Liu et al. “RetroGNN: Approximating Retrosynthesis by Graph Neural Networks for De Novo Drug Design.” In: CoRR abs/2011.13042 (2020). URL: <https://arxiv.org/abs/2011.13042>.
- [29] Laurens van der Maaten and Geoffrey Hinton. “Visualizing data using t-SNE”. In: J. Mach. Learn. Res. 9 (2008), pp. 2579–2605. URL: <http://www.jmlr.org/papers/v9/vandermaaten08a.html>.
- [30] Barbara Mikulak-Klucznik et al. “Computational planning of the synthesis of complex natural products”. In: Nature (2020). DOI: 10.1038/s41586-020-2855-y.
- [31] Tomoyuki Miyao, Hiromasa Kaneko, and Kimito Funatsu. “Inverse QSPR/QSAR Analysis for Chemical Structure Generation (from y to x)”. In: J. Chem. Inf. Model. 56 (2016), pp. 286–299. DOI: 10.1021/acs.jcim.5b00628.
- [32] Norio Miyaura and Akira Suzuki. “Palladium-catalyzed cross-coupling reactions of organoboron compounds”. In: Chem. Rev. 95 (1995), pp. 2457–2483. DOI: 10.1021/cr00039a007.
- [33] Pierre Del Moral, Arnaud Doucet, and Ajay Jasra. “Sequential Monte Carlo samplers”. In: J. R. Stat. Soc. B 68 (2006), pp. 411–436. DOI: 10.1111/j.1467-9868.2006.00553.x.
- [34] H. L. Morgan. “The Generation of a Unique Machine Description for Chemical Structures-A Technique Developed at Chemical Abstracts Service.” In: J. Chem. Doc. 5 (1965), pp. 107–113. DOI: 10.1021/c160017a018.
- [35] Christos A. Nicolaou, Ian A. Watson, Hong Hu, et al. “The Proximal Lilly Collection: Mapping, Exploring and Exploiting Feasible Chemical Space”. In: J. Chem. Inf. Model. 56 (2016), pp. 1253–1266. DOI: 10.1021/acs.jcim.6b00173.
- [36] Christos A. Nicolaou, Ian A. Watson, Mark LeMasters, et al. “Context Aware Data-Driven Retrosynthetic Analysis”. In: J. Chem. Inf. Model. 60 (2020), pp. 2728–2738. DOI: 10.1021/acs.jcim.9b01141.
- [37] Ramaswamy Nilakantan et al. “Topological torsion: a new molecular descriptor for SAR applications. Comparison with other descriptors”. In: J. Chem. Inf. Comput. Sci. 27 (1987), pp. 82–85. DOI: 10.1021/ci00054a008.
- [38] D Pavlov et al. “Indigo: universal cheminformatics API”. In: J Cheminform 3 (2011), P4–P4. DOI: 10.1186/1758-2946-3-S1-P4.



- [39] David A Pensak and E J Corey. “LHASA—Logic and Heuristics Applied to Synthetic Analysis”. In: Computer-Assisted Organic Synthesis. 1977, pp. 1–32. DOI: 10.1021/bk-1977-0061.ch001.
- [40] Reaxys. URL: <https://www.elsevier.com/solutions/reaxys> (visited on 02/20/2021).
- [41] David Rogers and Mathew Hahn. “Extended-connectivity fingerprints”. In: J. Chem. Inf. Model. 50 (2010), pp. 742–754. DOI: 10.1021/ci100050t.
- [42] Reuven Y Rubinstein and Dirk P Kroese. Simulation and the Monte Carlo method. John Wiley & Sons, 2007. DOI: 10.1002/9780470230381.
- [43] Robert E. Schapire and Yoav. Freund. Boosting: Foundations and algorithms. London, England: MIT Press, 2012. URL: <https://mitpress.mit.edu/books/boosting>.
- [44] Nadine Schneider, Nikolaus Stiefl, and Gregory A. Landrum. “What’s what: The (nearly) definitive guide to reaction role assignment”. In: J. Chem. Inf. Model. 56 (2016), pp. 2336–2346. DOI: 10.1021/acs.jcim.6b00564.
- [45] Philippe Schwaller, Théophile Gaudin, et al. ““Found in Translation”: predicting outcomes of complex organic chemistry reactions using neural sequence-to-sequence models”. In: Chem. Sci. 9 (2018), pp. 6091–6098. DOI: 10.1039/C8SC02339E.
- [46] Philippe Schwaller, Teodoro Laino, et al. “Molecular Transformer: A model for uncertainty-calibrated chemical reaction prediction”. In: ACS Cent. Sci. 5 (2019), pp. 1572–1583. DOI: 10.1021/acscentsci.9b00576.
- [47] SciFinder. URL: <https://www.cas.org/products/scifinder> (visited on 02/20/2021).
- [48] Marwin H. S. Segler, Mike Preuss, and Mark P. Waller. “Planning chemical syntheses with deep neural networks and symbolic AI”. In: Nature 555 (2018), pp. 604–610. DOI: 10.1038/nature25978.
- [49] Marwin H. S. Segler and Mark P. Waller. “Neural-symbolic machine learning for retrosynthesis and reaction prediction”. In: Chem. - Eur. J. 23 (2017), pp. 5966–5971. DOI: 10.1002/chem.201605499.
- [50] Photis Stavropoulos and D. M. Titterton. “Improved particle filters and smoothing”. In: Sequential Monte Carlo methods in practice. New York, NY: Springer New York, 2001, pp. 295–317. DOI: 10.1007/978-1-4757-3437-9\_14.
- [51] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. “Sequence to Sequence Learning with Neural Networks”. In: Adv. Neural Inf. Process Syst. (2014), pp. 3104–3112. URL: <http://papers.nips.cc/paper/5346-sequence-to-sequence-learning-with-neural-networks.pdf>.
- [52] Sara Szymkuć et al. “Computer-Assisted Synthetic Planning: The End of the Beginning”. In: Angewandte Chemie International Edition 55 (2016), pp. 5904–5937. DOI: <https://doi.org/10.1002/anie.201506101>.

- [53] T T Tanimoto. An elementary mathematical theory of classification and prediction. International Business Machines Corporation, 1958. URL: <https://books.google.co.jp/books?id=y34HAAACAAJ>.
- [54] Ashish Vaswani et al. “Attention is All you Need”. In: Adv. Neural Inf. Process Syst. (2017), pp. 5998–6008. URL: <http://papers.nips.cc/paper/7181-attention-is-all-you-need.pdf>.
- [55] Venkat Venkatasubramanian, King Chan, and James M. Caruthers. “Evolutionary Design of Molecules with Desired Properties Using the Genetic Algorithm”. In: J. Chem. Inf. Model. 35 (1995), pp. 188–195. DOI: 10.1021/ci00024a003.
- [56] Yanli Wang et al. “PubChem BioAssay: 2017 update”. In: Nucleic Acids Res 45 (2017), pp. D955–D963. DOI: 10.1093/nar/gkw1118.
- [57] David Weininger. “SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules”. In: J. Chem. Inf. Model. 28 (1988), p. 31. DOI: 10.1021/ci00057a005.
- [58] Alexander W. Williamson. “XXII.—On etherification”. In: Q. J. Chem. Soc. 4 (1852), pp. 229–239. DOI: 10.1039/QJ8520400229.
- [59] Hiroshi Yamashita, Tomoyuki Higuchi, and Ryo Yoshida. “Atom Environment Kernels on Molecules”. In: J. Chem. Inf. Model. 54 (2014), pp. 1289–1300. DOI: 10.1021/ci400403w.
- [60] Jiaxuan You et al. “Graph Convolutional Policy Network for Goal-Directed Molecular Graph Generation”. In: Adv. Neural Inf. Process Syst. (2018), pp. 6410–6421. URL: <http://papers.nips.cc/paper/7877-graph-convolutional-policy-network-for-goal-directed-molecular-graph-generation.pdf>.
- [61] Ning Yu and Gregory A. Bakken. “Efficient Exploration of Large Combinatorial Chemistry Spaces by Monomer-Based Similarity Searching”. In: J. Chem. Inf. Model. 49 (2009), pp. 745–755. DOI: 10.1021/ci800392z.
- [62] Shuangjia Zheng et al. “Predicting retrosynthetic reactions using self-corrected Transformer neural networks”. In: J. Chem. Inf. Model. 60 (2020), pp. 47–55. DOI: 10.1021/acs.jcim.9b00949.

