Chemical Reaction Path Generation Using State-Space Search Algorithms

Sai Srujan Chinta

Columbia University, New York, NY, United States

Abstract

Computer-assisted synthesis design has been in existence since over 40 years now. However, optimum reaction path generators have not been effective in gathering widespread approval by the research community. We propose a completely customizable tool to generate the optimum reaction path given a reactant and a product. Our tool leverages the use of traditional reaction templates combined with the computational complexity afforded by state-space search algorithms. Using reaction templates generated from extensive reaction databases from granted United States patents, out tool automatically generates a tree-like state space structure and searches for the required product intelligently. We report promising results with clearly discernable patterns of behaviour even when the tool fails to find the optimum path. Along with the path, our tool also returns the reaction templates which were used at each step of the reaction, thereby adding more interpretability to the process.

Keywords: A* Search, Reaction Path, SMARTS

1. Introduction

Synthesis planning is a task typically performed by highly qualified chemists with decades of experience. In order to assist chemists, retrosynthetic software was developed nearly 40 years ago. Nearly all approaches to automated retrosynthesis involve some use of reaction templates.

In Corey's [1] paper, he proposed a method to extract SMARTS templates from a chemical database of SMILES reaction strings. Specifically, the source of reaction templates is the set of USPTO patents granted between

1976 and 2013, preparsed by Lowe [2]. However, in this data source, contextual information and environmental conditions during each experiment are only sporadically present. Due to the lack of consistent data, all contextual information and environmental conditions (temperature, reagents, catalysts etc.) are dropped and only the reactants and products are considered.

2. Methodology

The overall methodology of this project is shown in Figure 1. Each individual component is explained in-depth in the following sub-sections.

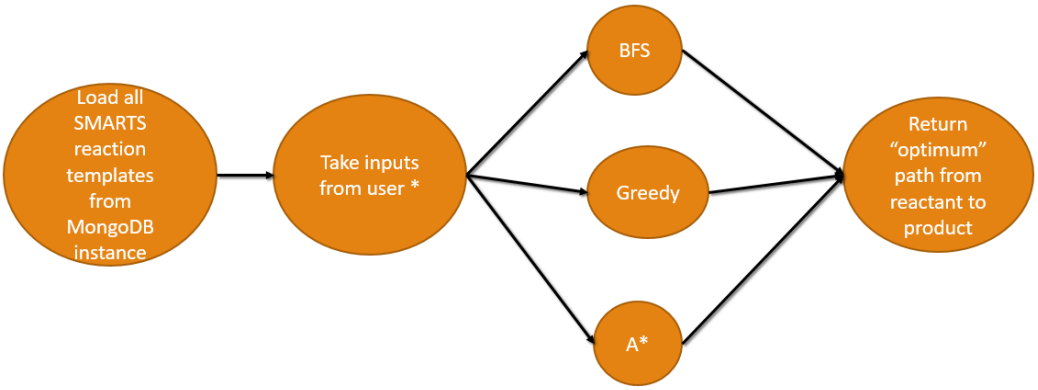

Figure 1: Overall Methodology

2.1. Generating SMARTS templates

From the 1,122,662 unique reaction SMILES strings, 140,284 unique SMARTS templates are generated [1]. In essence, these reaction SMILES strings are generalised by including wildcard characters which can be replaced by any atom. The process followed for generalising the SMILES strings is as follows:

1. Identify which atoms have changed from the reactants to the products
2. Generate SMARTS label for each changed atom and make the bond structure and number of Hydrogen atoms explicit
3. Replace the atom-mapped neighbours of the changed atoms with wildcard characters

However, out of the 140,284 unique SMARTS templates generated, the rank versus popularity decays exponentially. This information is clearly encapsulated in Figure 2. As we will discuss in the coming sections, our tool allows the end user to customize the number of templates. However, it is in the user's best interest to consider the top few percentage of SMARTS templates because the increased computational complexity does not necessarily yield proportionate results.

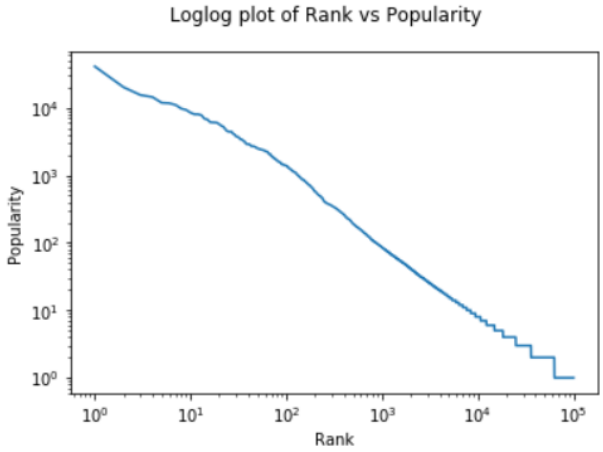

Figure 2: Loglog plot of rank vs popularity of SMARTS templates

2.2. Inputs from user

The user first needs to choose among the three state-space search algorithms: Breadth First Search, Greedy Search and A* search. Inputs common to all the three state-space search algorithms are:

- Reactant and Product
- Maximum depth of the search tree
- Number of SMARTS reaction templates to be used

For both greedy search and A* search, we need a definition of molecular similarity. This definition will be used as a heuristic to gauge the distance between the current molecule and the required product. In this project, we present a choice between radius-2 Morgan circular fingerprints and the default definition of molecular similarity provided by RDKit. Additionally,

for A* search, we also need the definition of backwards cost. Typically in A* search, backward cost is usually the depth of the current node. However, it is very uncommon for reactions to exceed more than 3 steps. Therefore, using this definition does not make sense because A* search would boil down to BFS with this definition. During our experiments, defining the backward cost as the molecular similarity between the reactant and the current molecule provided the best results in terms of number of nodes expanded and accuracy of results.

2.3. Find Optimal Reaction Path

The final step is to apply the state-space search algorithm with the inputs given by the user. Once the product is found, we return the optimal path, along with all the SMARTS templates at each step of the reaction. If the product is not found before the maximum depth of the search tree (given by the user), then we reverse all the SMARTS templates. Essentially, in each template, the reactants become the products and vice versa. The product is now passed in as the reactant and the reactant is passed as the product. This works because we are still accomplishing the same goal: tracing a chemical route between the reactant and the product. While not all of these reactions may be chemically viable in the opposite direction, this exercise is definitely helpful in gaining valuable insights about the chemical signatures of the reactant and product. In the results section, we explore examples of reactions which initially failed in the forward reaction but returned a path when the templates were inverted and the reactant and product switched.

3. Results

In this section, we will discuss a few sample chemical reactions in which our tool was able to find the optimal path and a few reactions where it was unable to find the optimal path.

3.1. Found correct optimal path

Let us first examine a chemical reaction wherein the reactant and product are both aromatic molecules. The reaction that we considered was the conversion from 4-Aminophenol to Benzene, as shown in Figure 3.

Upon feeding in 4-Aminophenol to our tool, it was able to find the optimal path two-step path to Benzene as shown in Figure 4.

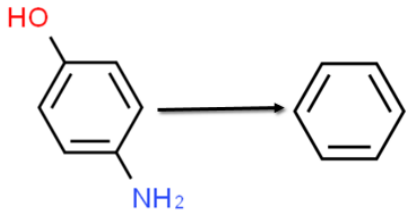

Figure 3: 4-Aminophenol to Benzene

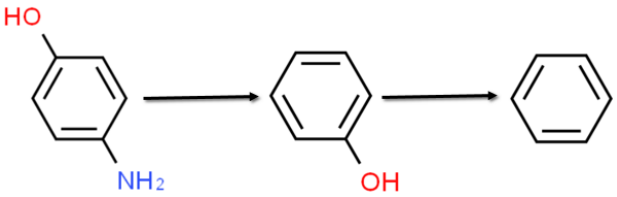

Figure 4: 4-Aminophenol to Benzene Path

Even though our tool does not explicitly return all the molecules (other reactants, reagents, catalysts etc) needed for the above two-step reaction to occur, it does return the SMARTS template corresponding to each of the two steps in the above reaction. In this example, the exact SMARTS templates being used are shown in Figure 5 and Figure 6.

```
([#7]-[c;H0;+0:1](:[*:2]):[*:3])>>([*:2]:[cH;+0:1]:[*:3])
```

Figure 5: Template in 4-Aminophenol to Phenol Reaction

]Among aliphatic molecules, we show the example of a standard reaction: hydrolysis of esters to give acids and alcohols. Specifically, we show the hydrolysis of propane butanoate in Figure 7

Our tool also accepts multiple reactants. We show the opposite reaction (Esterification) here as an example. Specifically, the reaction shown in Figure 8 is the formation of propane butanoate.

3.2. Did not find correct optimal path

We will now discuss a couple of examples where our tool was unable to find the correct optimal path. The first example is the creation of Toluene from Benzene as shown in Figure 9. The main reason that this reaction fails is that there is no reaction template which the reactant conforms to which

```
([#8]-[c;H0;+0:1](:[*:2]):[*:3])>>([*:2]:[cH;+0:1]:[*:3])
```

Figure 6: Template in Phenol to Benzene Reaction

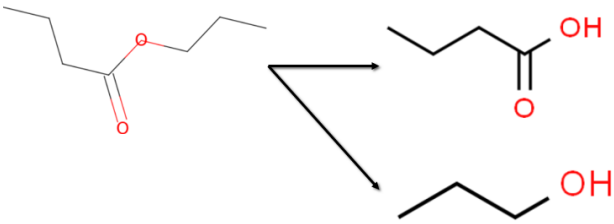

Figure 7: Propane Butanoate to Butyric acid and propanol

will enable it to add a methane molecule to the benzene ring. However, whenever the forward path fails, we reverse the SMARTS templates and try to find a path. In this case, we were able to go from Toluene to Benzene in a three step reaction.

The second example is the creation of aniline from benzoic acid as shown in Figure 10 and Figure 11. However, the interesting thing about this reaction is that our tool is able to generate the intermediary product. Essentially, our tool is able to reproduce the reaction from Benzoic Acid to Benzamide. However, it fails to finish off the reaction and find Aniline from Benzamide. This is likely because there is no template matching the catalyst (Bromine) in our set of SMARTS reaction templates.

3.3. Observations

A* search is more efficient than BFS and Greedy search. This is because as the depth of the search tree increases, the number of nodes to expand increases exponentially. Therefore, the order of expansion of nodes is crucial to the smooth functioning of this tool. With BFS, the nodes are expanded (molecules are passed through the given number of SMARTS templates) in the order in which they exist in the search tree. Since this is a brute force approach, it is computationally very expensive (sometimes infeasible) to run BFS on a large number of SMARTS templates and even more so to run it for a decent depth. Within A* search, better results are achieved when molecular similarity is used as a heuristic for backward cost instead of the traditional definition of backwards cost (number of steps taken thus far). The

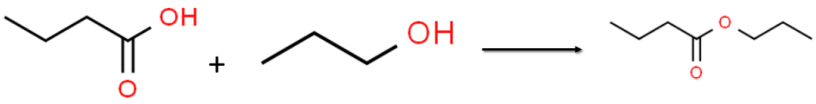

Figure 8: Butyric acid and propanol to Propane Butanoate

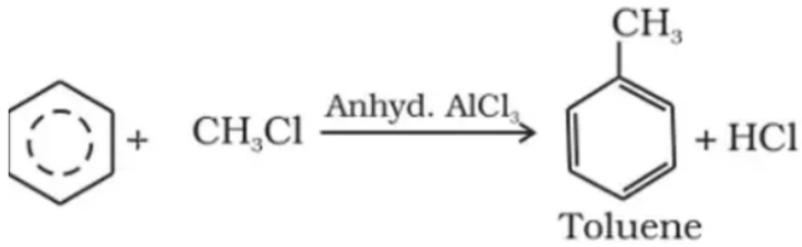

Figure 9: Toluene from Benzene

tool tends to perform well for reactions where the product is "simpler" than the reactant chemically, i.e breakdown of chemicals as opposed to creation of complex chemicals. When reactions require a catalyst, the catalyst needs to be added as one of the reactants in the reaction. In general, the tool works better for chained molecules than cyclic molecules.

4. Conclusion

In this project, we have built a completely customizable chemical reaction path generator for organic chemical reactions. The user can decide how many SMARTS templates to use, which state-space search algorithm to use, specific properties for each search algorithm etc. We leverage the power of search-space algorithms to explore relevant nodes (molecules) first, thereby allowing us to cleverly navigate the search space quickly and efficiently. The results indicate that our tool is successful in finding the optimal path in most cases. Even when it fails to find the path in the forward direction, there is a provision to reverse the templates and find the path from the product to the reactant. In the future, we could add more SMARTS templates at the elementary level to give the resultant path more intuition. Moreover, environmental conditions were not considered while generating the SMARTS templates. Adding this data will add more authenticity to the paths generated by the tool.

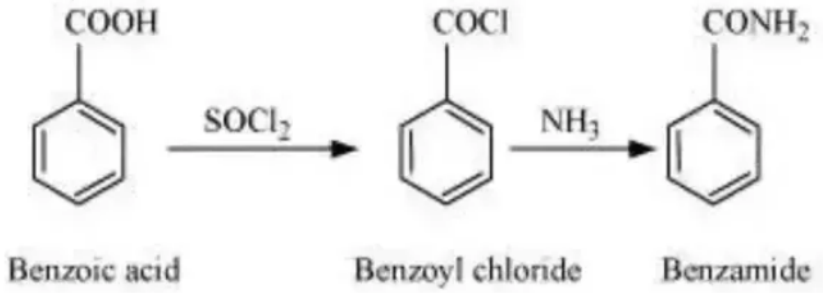

Figure 10: Aniline from Benzoic Acid

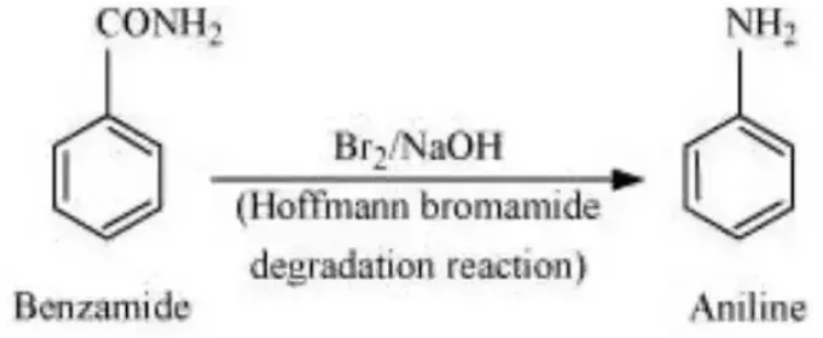

Figure 11: Benzamide to Aniline

References

[1] Coley, C. W., Barzilay, R., Jaakkola, T. S., Green, W. H., Jensen, K. F. (2017). Prediction of organic reaction outcomes using machine learning. ACS central science, 3(5), 434-443.

[2] (15) Lowe, D. M. Extraction of chemical structures and reactions from the literature. Thesis, University of Cambridge, 2012.