Machine Learning the Language of Chemistry As A Guide for Retrosynthesis

Matthew P. Wampler-Doty† Andrea Cadeddu† Bartosz A. Grzybowski†

† Department of Chemistry and Department of Chemical Engineering, Northwestern University, 2145 Sheridan Road, Evanston, Illinois 60208, United States

Abstract

This paper introduces data science techniques to computer assisted chemical retrosynthesis. We develop an unsupervised machine learning algorithm for *segmenting* a compound into recognizable submolecules and motifs. The approach adapts techniques from natural language processing to chemistry, establishing that the subjects may be understood in a unified framework.

Outline

Introduction :

Automated retrosynthesis is an old, outstanding problem of chemical informatics. Traditionally the problem has been tackled strictly with tree search, which has lead it to be likened to chess AI programming (2005). However the analogy between the two activities is strained - for example while chemistry researchers have lauded Samuel's α–β pruning (1959) as an examplar heuristic tree search algorithm, the Samuel's basic game-theoretic justification makes little sense for chemistry, where there are no opponents. It is challenging to develop heuristics for guiding automated retrosynthesis. As a result, researchers often opted for *exhaustive* search, exploring all retrosynthetic pathways from the target molecule. This quickly gets out of hand for molecules with molecular weight exceeding 200 g/mol.

An early proposed heuristic for simplifying retrosynthetic search is the *Localised Matching Unit* (LMU), introduced by Corey in his LHASA program (1980). The idea behind LMUs is to break the target molecule up into reasonable submolecules, guiding retrosynthesis to first combine those submolecules to construct the target, then recursively synthesizing those submolecules. Corey's LHASA contained a small number of hand coded LMUs, covering just a small number of cases.

In this paper we propose an unsupervised machine learning system for the robust, algorithmic identification of LMUs. Our algorithm is dirived from the observation that the distribution of submolecules in organic chemistry follow the same distributions as sentence fragments in natural language. This suggests that statistical techniques in one domain can inform problems in the other. We provide an algorithm for identifying boundaries of LMUs, based on linguistic TF-IDF scores of learned submolecules.

Chemistry As A Language

Informally, there is a clear analogy between a chemist's understanding of a compound and a Chinese speaker's understanding of a sentence. In either case, the trained mind effortlessly identifies part-whole relationships, despite a lack of demarcations. The analogy breaks down, however, since chemicals have no orientation, nor can their graphs be represented by simple lines as in sentences.

Despite the obvious differences, chemicals can be seen as conforming to generalization of language, where

sentences are akin to simple polymers. The two can be observed to obey similar part-whole statistics, as the following exercise demonstrates:

1. Acquire from a database a sample of chemical compounds or (English) sentences.
    a. Using NIH's CIR database, we acquired 10000 random chemicals.
    b. Using the English Wikipedia, we acquired 10000 random sentences. The spaces were removed, to make them similar to chemicals without demarcations.
2. Construct a library of common subfragments.
    a. For chemicals, this is done by taking roughly 200 compounds, pairing them up, and computing the *most common subfragment*, as seen in Fig. 1

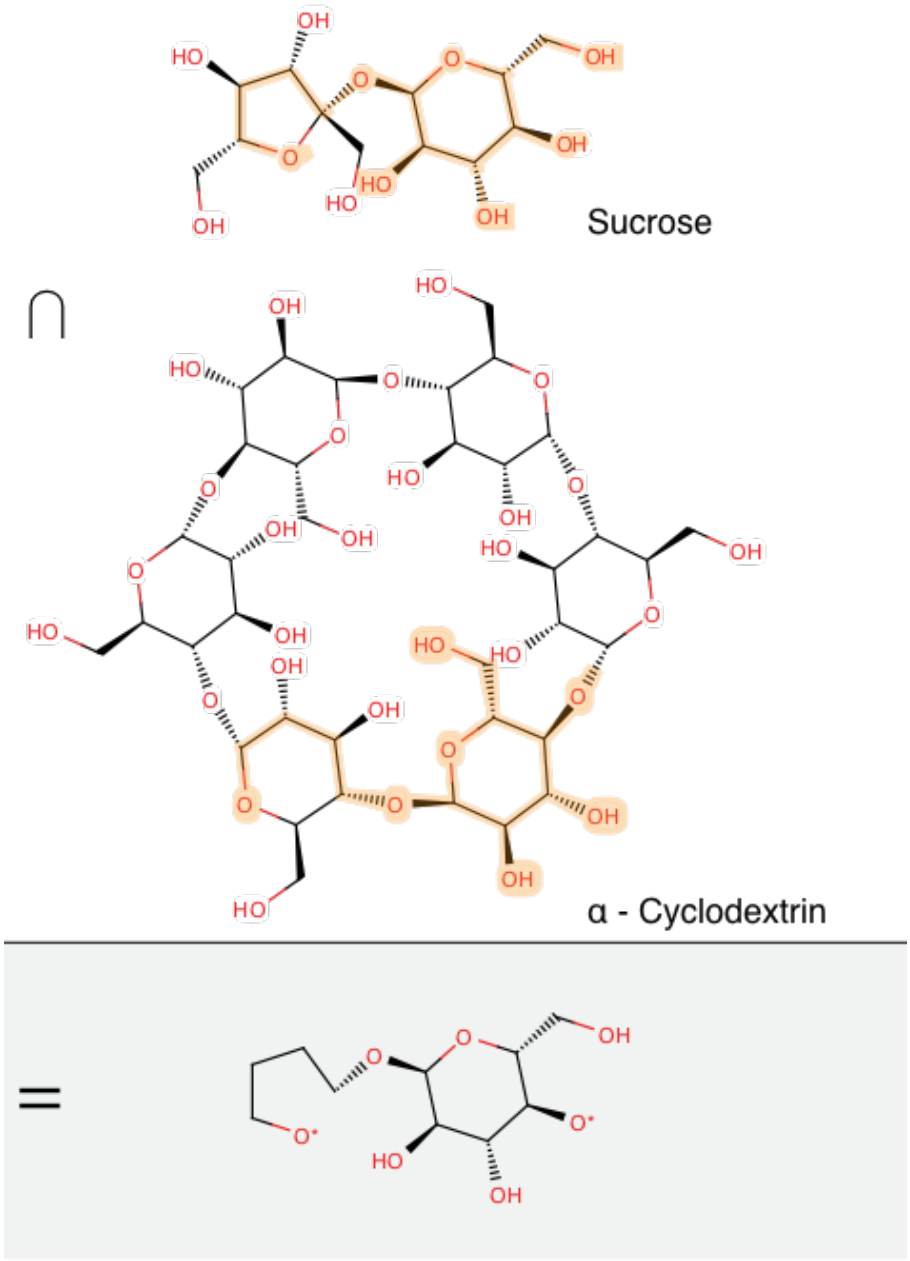


Figure 1. Most Common Subfragment

   a. For language, this is done by taking roughly 200 sentences, and similarly pairing them up and finding the *largest common substring*. To mimic the lack of orientation in chemistry, we ignore the order of sentences in this calculation.
3. Compute the frequencies of the common fragments in the sample, and rank the most frequent as 1, the second most frequent as 2, etc.

The results of this exercise are shown in Fig. 2. In both cases, the distributions can be fit to an *upper-*

truncated power law, which may be explained by our small sample size [5]. A larger sample would exibit a proper power-law relationship. Fits to both distributions are given in Fig. 3.

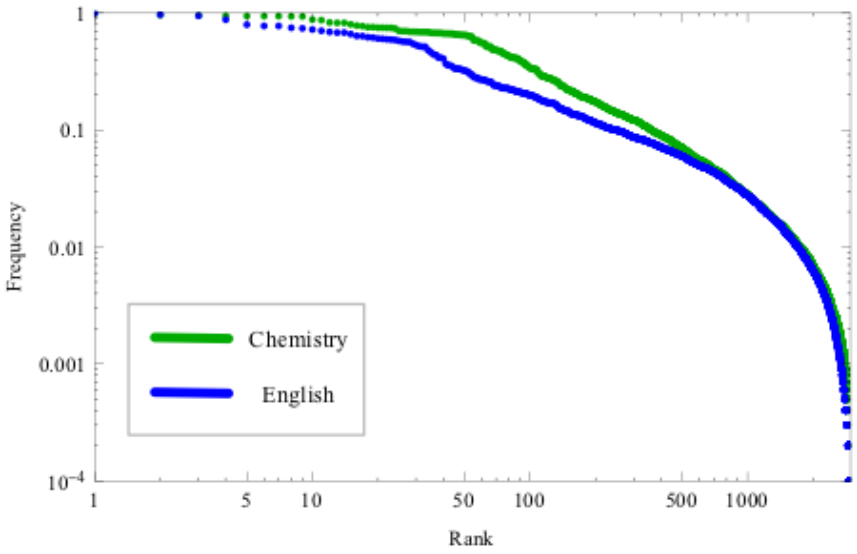
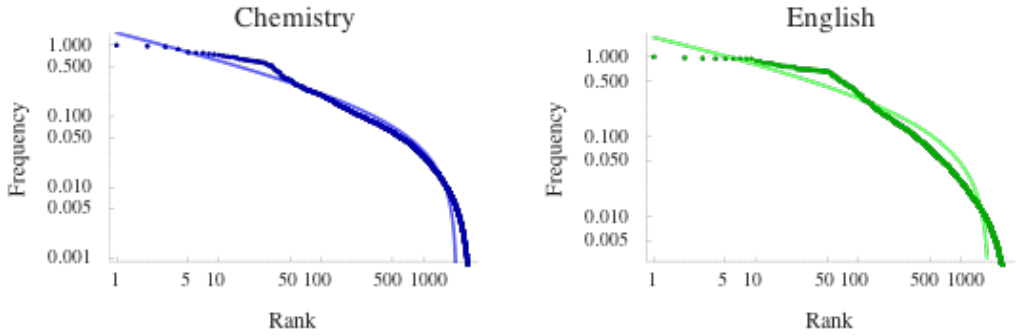


Figure 2. Rank vs. Frequency for chemical fragments and English sentence fragments



In the case of linguistics, the observed power-law distribution is known as *Zipf's Law* (1935). As far as we know, this simple statistical observation has never been observed in chemistry before. It immediately suggests that techniques for dealing with the analysis of language may fruitfully be applied to the analysis of chemical compounds.

Algorithm and Analysis

TF-IDF

From the statistical observations of chemical fragments we have made, we develop a heuristic, unsupervised learning algorithm for finding the boundaries of submolecules.

In linguistics, Zipf's law has provided the basis for the automated recognition of *keywords* in a document. This is done by ranking words according to their *TF-IDF* score, developed by Jones (1972). Given a dataset

D, the TF-IDF for a term *t* in a document *d* , is defined as:

$$\text{tfidf}(t, f) = (\text{Frequency of } t \text{ per term in } d) \times -log(\text{Frequency of } d' \text{ in } D \text{ with } t \text{ in } d')$$

TF-IDF scores are commonly used to generate *word clouds* - Fig. 4 shows a word cloud for the Wikimedia Foundation's 2009-2010 annual report. In a word cloud, the word's size is proportional to its TF-IDF score.

Figure 4. Word cloud for Wikimedia Foundation's 2009-2010 annual report

TF-IDF naturally gives low weight to extremely common words like "a" and "the", as well as extremely uncommon words such as "sesquipedalian". In chemistry, using the statistics we have computed, fragments like *methylene* will also get low weight.

Ranking

In order to guide retrosynthesis, our algorithm identifies atoms as likely sites for retrosynthesis. It *ranks* an atom `A` in molecule `M`, using a library `L` of submolecules with frequencies given by `F`, according to the following algorithm:

```
def rank(A,M,L)
    submolecules <- Find all submolecules of M in library L
    entropy <- sum the TF-IDF scores of the submolecules for which A is a member
    return entropy
```

Results

The following shows the results of our algorithm on a few molecules:

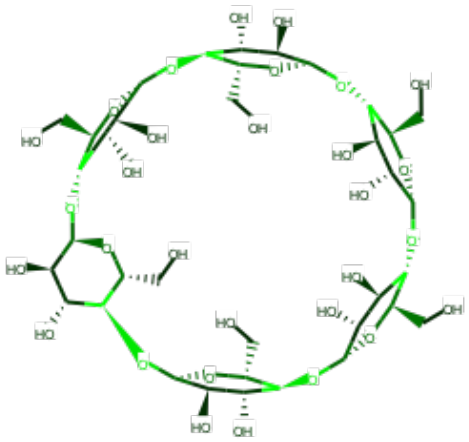


Figure 5. α-cyclodextrin

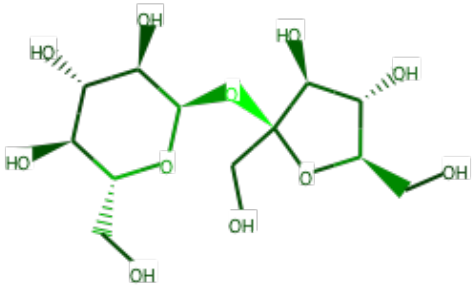


Figure 6. Sucrose

Figure 7. (HO)2-PO-S-C11-EG6-OH

Future Work

Conclusions

Supporting Material

- Zipf's Law:
- Performance Evaluation
    - F Scores
- Suggested Retrosynthetic "*cuts*" of 100 compounds

References

1. Jones, K. S. A Statistical Interpretation Of Term Specifity And Its Application In Retrieval. Journal of Documentation 28, 11–21 (1972).
2. Todd, M. H. Computer-aided organic synthesis. Chemical Society Reviews 34, 247–266 (2005).
3. Corey, E. J., Johnson, A. P. & Long, A. K. Computer-assisted synthetic analysis. Techniques for efficient long-range retrosynthetic searches applied to the Robinson annulation process. J. Org. Chem. 45, 2051–2057 (1980).
4. Samuel, A. L. Some studies in machine learning using the game of checkers. IBM J. Res. Dev. 3, 210–229 (1959).
5. Zipf, G. K. The psycho-biology of language. ix, (Houghton, Mifflin, 1935).
6. Burroughs, S. M. & Tebbens, S. F. Upper-Truncated Power Law Distributions. Fractals 09, 209–222

(2001).