Task 2

1. The sentence we chose is: John ate salad with mushrooms with a fork. The drawings of this sentence are showed in Figure 1.

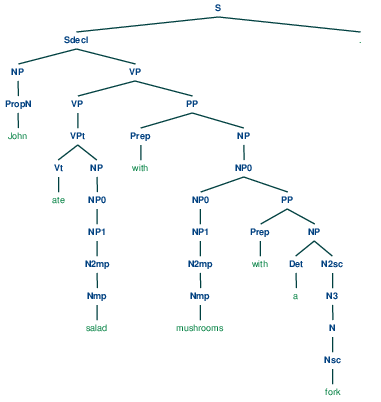
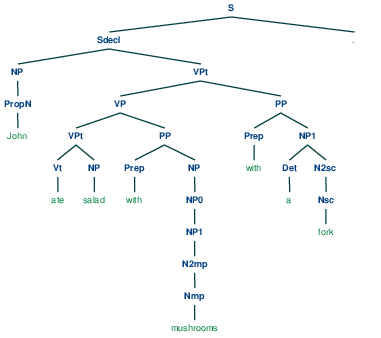


Figure 1: left one is Tree A, right one is Tree B.

1. Tree A attaches “a fork” to the end of “ate salad with mushrooms”, which means using a fork to eat salad and mushrooms. It is reasonable. However, Tree B treats “a fork” as a component of “mushrooms”, which means “a fork” is not a tool here, but a part of mushrooms. It is inappropriate.

We could replace “with a fork” with “in the soup”, then the sentence is like: John ate salad with mushrooms in the soup. Now, Tree A means salad and mushrooms are in the soup, which is weird. While Tree B means eat salad and the mushrooms soup, which is more reasonable than Tree A.

Task 5

Implementation details:

In order to construct the tree, we need to know where each point is from, which means from the top to the bottom of the tree, we need to know each point’s children and children’s position. So we store the information of position and children in the Label class. Each label is stored in a matrix. Then, we can use the information in this matrix to construct NLTK tree recursively.

We have made several changes in this Task:

1. Edit the class Label, add attributes: children (Label or tuple of Labels) and location(tuple). “children” is the children of this label, “location” is where this label lies in the chart.
2. Edit method “unaryFill” and “maybeBuild” in CKY class, fill the chart with Label instead of string or Noneterminal.
3. Edit method “unaryUpdate” in Cell class, fill the chart with Label instead of string or Noneterminal.
4. Add method “firstTree” to construct a nltk tree.
5. Add method “createTree”, call itself recursively to form a tree string.