Project1

Introduction

This project involves parsing multiple Phrase Trees produced from Penn Tree Bank to find the following

- Count of Sentences
- Count of Noun Phrases
- Count of Verb Phrases
- Count of Ditransitive Verb Phrases
- Count of Intransitive Verb Phrases

Approach

Every node with in the phrase tree structure along with its daughters is encapsulated in parenthesis. The beginning of the string in parenthesis provides the parent. Nested parenthesis with in the string represent daughters. The understanding of this structure allows us to recursively build parent nodes and corresponding child nodes.   The provided file project1.py loops through each input file and parses each character of the provided PTB phrase structure. If the character '(' is encountered, the character is pushed down a stack along with its index. If the character ')' is encountered, the last index having the character '(' is popped from the stack and the substring is formed. The string is split by a space and the first word is taken as the parent tag.

Example:

(S (NP-SBJ (NP Your Oct. 2 article)))

On parsing the string using the above approach, the phrase (NP Your Oct. 2 article) is identified and the parent-child relationship is formed with 'NP' as the parent and 'Your Oct. 2 article' as the child. In the next iteration, the phrase (NP-SBJ (NP Your Oct. 2 article)) is identified with the parent as NP-SBJ and (NP Your Oct. 2 article) as the child.

By parsing the string, since we are able to collect parent tags, We are able to form a list of tuples with (Parent Tag, parent index, child index). Parent tags are made in the format-'Parent:VP'. The child index in one tuple would be the parent in another tuple, if the child node is a parent for another node. We loop through the generated list of tuples of (parent tag, parent index, child index) to form parent to child tuple relationship.

Post this step, We can count the number of tags to get the results for sentences, Noun Phrases, Verb Phrases, DVPs and IVPs.

DVPs:
The program checks that tag of the parent is ‘Parent:VP’ and the tag of every child ‘Parent:NP’, and that there are only 2 children.

IVPs:
The program checks that the tag of the parent is ‘Parent:VP’ and the child does not have a separate table with itself as the parent.

Results

On implementing the above approach, the following counts are collected

Type	Count
Sentence	4670
Noun Phrase	13221
Verb Phrase	7920
Ditransitive Verb Phrase	48
Intransitive Verb Phrase	123