

COSC 426LA F24 Lab 3

Introduction

In this lab you will be writing grammars that can generate a subset of sentences that are grammatical in Standard American English (SAE), while ensuring that these grammars do not generate sentences that are ungrammatical in SAE.

By completing this lab, you will demonstrate that you:

- Understand how context free grammars work
- Can use trees to describe why sentences are ambiguous.
- Can reason about the underlying grammar that might have generated a set of grammatical sentences, and failed to generate a set of ungrammatical sentences.

Provided files

- Lab3.py
- grammar1.txt
- [A google doc template](#) to write responses

What to submit

- grammar2.txt for the grammar in Part 2
- grammar3.txt for the grammar in Part 3
- grammar4.txt for the grammar in Part 4
- Lab3.py with the added functions and tests for each grammar.

Part 0

Before starting each lab, get the latest version of the NLP Scholar repo by first navigating to the folder on terminal and then executing:

```
git pull
```

Additionally, a package is missing that you need for today. With the nlp environment activated run:

```
conda install nltk
```

Part 1

In this part familiarize yourself with the provided grammar in **grammar1.txt**. The main function in **Lab3.py** generates 10 random sentences from the grammar, and also generates the parses for two sentences.

Answer the following questions in the google doc template.

1. How many words (or “terminals”) are in this grammar?
2. What are the non-terminals in this grammar?
3. What is the shortest sentence that this grammar can generate?
4. What is the longest sentence that this grammar can generate?
5. When you run **Lab3.py** you will see that the sentence “the panda saw my friend in her pajamas” has two different parses. Describe the difference between these parses.
6. You will also see that the grammar generates parses for seemingly nonsensical sentences like “the pajamas ate my friend in the panda”. Why does this happen? Is this a problem?

Next, complete the function **is_grammatical** in **Lab3.py**, which determines if a given sentence is grammatical under a given grammar.

Part 2

Create a new file **grammar2.txt** which is a copy of **grammar1.txt**. Add the following verbs to the grammar: **existed**, **gave**, **died**, **vanished**, **lent**, **sent**.

Confirm whether the following sentences are deemed grammatical by your grammar:

- The panda existed the friend
- The panda gave the friend
- The panda died the sandwich
- The panda vanished the pajamas
- The panda lent the sandwich
- The panda sent the pajamas

Note, all of these sentences are actually ungrammatical in SAE. In considering this, answer the following question in the google doc template:

1. Are all of these sentences ungrammatical for the same reasons? If not, can you divide the sentences into categories such that all sentences in a category are ungrammatical for the same reason?

Finally, update your grammar in `grammar2.txt` so that you don't generate these sentences (by generate sentences with those verbs that are grammatical). Write test cases for your new grammar in the `test_grammar2` function in `Lab3.py`.

Part 3

Make a copy of your `grammar2.txt` called `grammar3.txt`. You will improve this grammar by adding in adverbs and adjectives. Here are some examples of sentences that this grammar should be able to generate (these are also grammatical in SAE):

- the panda vanished peacefully
- the ponderous pajamas peacefully died
- my very very ponderous friend existed very very very peacefully.
- the friend ate the excited delicious sandwich joyfully
- the delicious panda joyfully saw the excited sandwich
- the panda took the delicious delicious delicious pajamas in the sandwich in my very ponderous friend.
- the pajamas joyfully gave my friend the ponderous delicious sandwich in the panda.

Here are examples of sentences that should be ungrammatical under your grammar (these are also ungrammatical in SAE):

- the panda ponderous vanished peacefully.
- the ponderous panda excited vanished peacefully.
- the panda gave the delicious very delicious pajamas to my friend peacefully
- the panda joyfully peacefully gave the delicious pajamas to my friend.

Start by answering the following questions in the google doc template:

1. How does adjective modification in SAE work?
2. How does adverb modification in SAE work?

You should modify `grammar3.txt` to capture these processes. You should minimally include the following adjectives and adverbs:

- Adjectives: excited, delicious, ponderous
- Adverbs: joyfully, peacefully

Write test cases for your new grammar in the `test_grammar3` function in `Lab3.py`.

Part 4

Make a copy of `grammar3.txt` called `grammar4.txt`. You will add another structure to your grammar: fronted sentential complements. Here are some examples of sentences that this grammar should be able to generate (these are also grammatical in SAE):

- that the panda existed very peacefully pleased my friend
- that the friend that ate my excited delicious sandwich vanished perplexed the ponderous pajamas.

- that the sandwich in my panda gave the friend that ate the pajamas my sandwich perplexed the friend in her pajamas.

Start by answering the following question in the google doc template:

1. How does fronted sentential complements in SAE work?

Next, create a new file **grammar4.txt** and modify the grammar from the previous step to accept these new sentences. Write test cases for your new grammar in the **test_grammar4** function in **Lab3.py**.