%

# Part 1

1. Design

For grammar1, since it's very simple, I just implemented the entire programming based on the given grammar rules and vocabulary.

By carefully comparing grammar1 and grammar2, we can see that grammar2 is a simplified version of grammar1 in terms of grammar rules, which means we need to find the difference between them to make each vocabulary bring its own features. The main difference lies in the singular and plural forms of nouns and the singular and plural forms of verbs as well as proper nouns. I carefully designed the features and finally realized the function.

2. Limitation

Since the grammar rules of grammar1 and grammar2 are very simple, there are bound to be many restrictions, for example it can't parse different tense (like "Fido fed the dog with biscuits"). Also, it has some overgenerations. Because my grammar is context-free, it may parse some meaningless sentences correctly. (like “The dog feeds the dog with the dog”).

# Part 2

1. Design

According to the requirements, I design six features for each verb, including vform, agent, theme, ben:, exp, gap\_type. Vform refers to the tense of the verb, ben refers to Beneficiary, exp refers to Experiencer, and gap\_type refers to whether sleep belongs to subject or object. At the same time, I also design two feature (gap and rec) to save and transmit subject and object. I design one vp rule for every sentence without adding extra nonterminals.

2. Limitation

First of all, because I didn't do too much processing on nouns, my grammar can’t parse the noun phase like “students” or “teachers”. Also, it can’t parse the present tense or other tense of verb like “the student try to sleep”. As the same, due to the context-free feature, my grammar may parse some meaningless sentences (like “The teacher promised the teacher to sleep.”).