CSCI 1710 Capstone: Grammar Modeling and Exploration through Forge

Abigail Marks
Faculty Sponsor: Tim Nelson

Background

In this project, I worked with Daniel Segel to model English grammar using Forge. The modeling language Forge allows programmers to model how different systems interact. This project involves modeling how different elements in English grammar interact. This project simplifies English into its phrase structure grammar, or rules on how different parts of speech combine with each other to become sentences. The basic rules say that a sentence is composed of a Noun Phrase (NP) followed by a Verb Phrase (VP). An NP breaks down into a noun or a noun and a determiner. Phrase structure grammar has rules for modeling different types of verbs, like regular verbs, transitive verbs, and ditransitive verbs. A regular verb, like "run," does not take an object. A transitive verb, like "likes," must combine with an NP to form a VP, and two NPs follow a ditransitive verb. In our project, we model sentence structure using phrase structure grammar. The model includes nouns, verbs, determiners, linkers, and prepositional phrases as the most basic unit of grammar. These elements can combine to form noun phrases, verb phrases, and sentences.

Model

Our model produces grammatically correct sentences that follow the phrase structure grammar. It allows for all the three types of verbs listed above, combining NPs and VPs with "and", and modeling prepositional phrases. For compute time and complexity's sake, we do not model any punctuation, adjectives, or relative clauses. The output of the model gives the structure of the sentence in the form of a tree. It shows which components break into smaller units, how they combine, and the order in which they are combined. We also create a word order to demonstrate the order of the words read in a sentence. The word order is different from the order of the components as it only contains the words that are said aloud, not the internal structural nodes.

Figure 1.

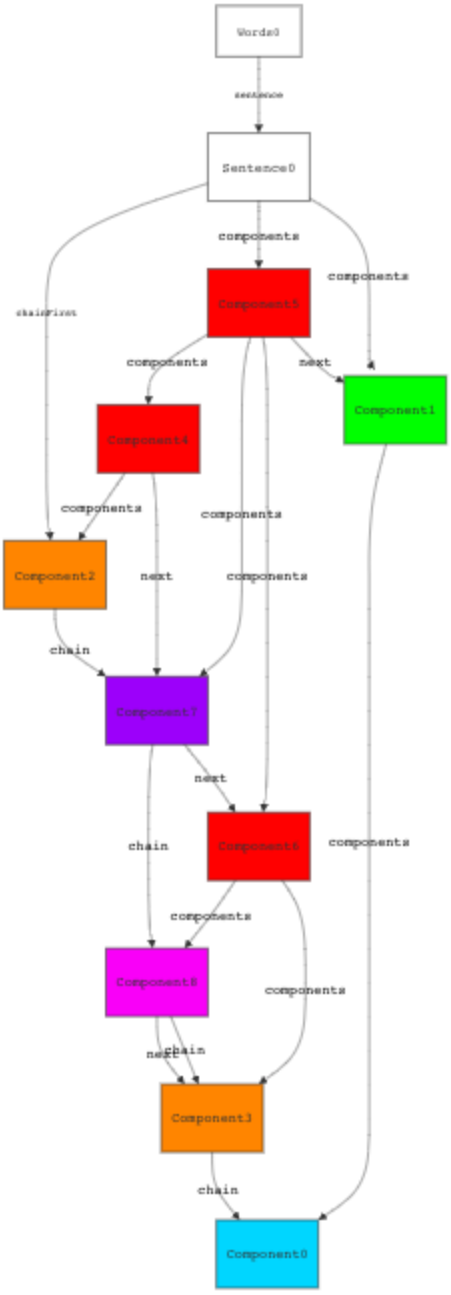

Figure 2

Figure 1 above shows a themed version of the default output. This example shows NP in red, VP in green, determiners in pink, nouns in orange, and verbs in blue. The chain arrow depicts the order of the words in the sentence, connecting the determiner, to the noun, to the verb. The next arrow shows the order of elements on the same structural level, showing that the NP comes before the VP. Figure 2 depicts a more complicated instance using the same color scheme.

Given that the instances and default visualization are difficult to digest and contextualize, we also created a custom visualization to provide example sentences for the instance created by the model. The visualizer creates sentences from a fixed bank of words and synthesizes them with a color key for the part of speech. Figures 3 and 4 show the custom visualizations in Figures 1 and 2, respectively.

Figure 3, Figure 4

Findings

Using the model we were able to explore different properties of sentence structure. For example, if there is no “and” in the sentence, the number of nouns in a sentence will always be greater than or equal to the number of verbs. We were also able to reason about sentence structure. We found that sentences can have the same word order but have different sentence structures that produce this word order. We were interested in looking at possible word orders. We found that two NPs are only next to each other when there is a ditransitive verb in the sentence. On the other hand, sentences with two verbs next to each other are impossible to create using our model. We also discovered no restrictions for when prepositional phrases are next to each other. Modeling grammar using Forge allowed us to explore edge cases and properties and learn more about the system we use to communicate.