Student: Michael A. d’Amore

Introduction

This assignment was about designing and testing a Context-Free Grammar (CFG) to parse sentences related to a Camelot-themed corpus. My task was to build a grammar in camelot_grammar.cfg that could parse a set of example sentences and identify the CFG's limitations when handling natural language. The assignment was split into two parts:

- Part 1: Develop a CFG and attempt to parse a given set of sentences, noting which sentences were successfully parsed and which were not.
- Part 2: Show examples of the CFG's limitations—one sentence that should parse correctly but doesn’t, and another that’s nonsensical but does parse.

Part 1: CFG Development and Parsing

CFG Rules and Grammar Design

For Part 1, I created rules in the camelot_grammar.cfg file to account for different sentence structures, verb phrases, noun phrases, prepositional phrases, and other constructs based on the Camelot sentence set. Here are some of the specific rules I implemented in addition to the starter rules that should have parsed the sentences provided:

1. Sentence Structure:

S1 -> NP VP Eos | Q

This rule defines basic sentence structure with a noun phrase (NP) followed by a verb phrase (VP) and end-of-sentence punctuation (Eos).

2. Noun Phrases (NP):

NP -> Det Adj Noun | Proper | Noun PP | NPB Conj NPB

NPB -> Det Noun | Proper | Noun | Det Adj Noun

Proper -> 'Arthur' | 'Guinevere' | 'Sir_Lancelot' | 'Holy_Grail'

   - This rule should handle phrases like “the Holy Grail” and “the sensational Holy Grail” by combining determiners, adjectives, and proper nouns.
   - Example sentence that should parse: "the sensational Holy Grail is a sacred chalice."

3. Verb Phrases (VP):

VP -> VerbT NP | Modal Aux Verb | Aux Verb | VP PP | VP Adv

VerbT -> 'is' | 'rides' | 'carries' | 'suggests' | 'knows'

Modal -> 'might' | 'could' | 'would'

Aux -> 'have' | 'been' | 'is' | 'was'

- This rule covers different verb combinations, including auxiliary and modal verbs, as well as transitive verbs like "carries" or "rides."
- Example sentence that should parse: "he suggests to grow fruit at home."

4. Prepositional Phrases (PP):

PP -> Prep NP

Prep -> 'near' | 'to' | 'with' | 'for' | 'at'

- This rule supports prepositional phrases to expand noun or verb phrases, such as "near the castle" or "for five weary nights."
- Example sentence that should parse: "Guinevere had been riding with Patsy for five weary nights."

5. Questions (Q):

Q -> QW Aux NP Verb | QW NP VP | QW Aux NP Verb | QW NP VP PP

QW -> 'do' | 'why' | 'what' | 'who'

- This structure aims to cover interrogative sentences, including wh-questions with auxiliary verbs.
- Example sentence that should parse: "why does England have a king?"

Challenges Encountered

Despite creating these CFG rules with all the necessary syntactic elements, some sentences failed to parse. The error outputs indicate that, while all intended rules were in place, certain word combinations or structures weren't parsed due to unknown limitations. This could be due to the structural complexity and non-context-free characteristics of natural language, which present challenges to CFGs.

Parsing Results

Here is a summary of the parsing results, categorized by parsed and unparsed sentences.

- Successfully Parsed Sentences:
    - Arthur is the king.
    - Arthur rides the horse near the castle.
    - Arthur rides the plodding horse near the castle.
    - Sir_Lancelot might have spoken.
- Failed to Parse:
    - the Holy_Grail is a chalice.
    - every coconut was carried to the hottest mountains.
    - sixty strangers are at the Round_Table.
    - Guinevere had been riding with Patsy for five weary nights.
    - he suggests to grow fruit at home.
    - do coconuts speak?
    - why does England have a king?

The failure to parse certain sentences, even with the necessary grammatical structures in place, shows CFG's limitations in fully capturing English syntax for complex sentences, especially for coordination, nested phrases, or complex semantics.

Part 2: Example Sentences Demonstrating CFG Limitations

In Part 2, I tested the CFG with two example sentences to demonstrate its limitations:

1. Unparsed Sentence: An example of a grammatically correct sentence that couldn’t be parsed by the CFG.
    - Sentence: The king with the brave knights.
    - Expected Parse: This sentence is grammatically correct, but it has nested prepositional phrases that the CFG wasn’t able to handle. The grammar lacks rules for parsing this nested structure.

    - Output: The parser fails to produce a parse tree for this sentence, with the error message: Error parsing sentence 'The king with the brave knights.': Grammar does not cover some of the input words: "'The', 'brave', 'knights.'"

2. Overgeneralized Sentence: A nonsensical sentence that the CFG did manage to parse.
    - Sentence: King castle knight ride.
    - Parse Tree: This nonsensical sentence was parsed successfully by the CFG. The grammar's overly flexible noun and verb rules allow it to form a tree structure that's technically valid but meaningless.
    - Output: The parser produced a parse tree due to lenient noun and verb rules, highlighting how the CFG tends to overgeneralize, allowing nonsensical combinations as long as they follow syntactic patterns.

Conclusion

In this assignment, I developed a CFG to parse Camelot-themed sentences and encountered challenges in capturing all the syntactic details of English. While the grammar was structured to cover basic rules, it lacked the precision needed to parse more complex nested structures accurately. Its broad structure also led to some nonsensical parses. These challenges illustrate the difficulty of using CFGs for natural language, as they can struggle to balance the nuance of English syntax without overgeneralizing or being too restrictive.