**IST-664**

**HW3**

**Pan Chen**

**Prof. Nancy McCracken**

**Part 1**

1. **Arthur is the king .**
   1. **Parse Tree**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Proper Arthur))

         (VP

            (VerbT is)

            (NP

               (Det the)

               (Nbar

                  (Noun king))))) .))

* 1. **The rules added**

None because its grammar rules and vocabulary types are already properly included in S1.gr and Vocab.gr, respectively

1. **Arthur rides the horse near the castle .**
   1. **Parse Tree**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Proper Arthur))

         (VP

            (VerbT rides)

            (NP

               (NP

                  (Det the)

                  (Nbar

                     (Noun horse)))

               (PP

                  (Prep near)

                  (NP

                     (Det the)

                     (Nbar

                        (Noun castle))))))) .))

* 1. **The rules added**

None because its grammar rules and vocabulary types are already properly included in S1.gr and Vocab.gr, respectively

1. **Arthur rides the plodding horse near the castle .**
   1. **The vocabulary rules added**
      1. **1 AdjN plodding**

“AdjN” denotes all non-comparative adjectives

* 1. **The grammar rules added**
     1. **15 NP NP PP**

This rule helps to parse a more complex noun phrase that contains a noun phrase and a prepositional phrase, so a noun phrase like “the plodding horse near the castle” can be correctly parsed as NP (“the plodding horse”) and PP (“near the castle”) after adding this rule.

* + 1. **15 NP Det Adj Nbar**

This rule helps to parse a noun phrase that contains a determiner, and an adjective and an inner level noun phrase, so a noun phrase like “**the plodding house**” can be correctly parsed as Det (“the”), Adj (“plodding”) and an inner noun phrase (“house”) after adding this rule.

* + 1. **1 Adj AdjN**

This rule adds a non-comparative adjective like “plodding” as an Adj

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Proper Arthur))

         (VP

            (VerbT rides)

            (NP

               (NP

                  (@Det-Adj

                     (Det the)

                     (Adj

                        (AdjN plodding)))

                  (Nbar

                     (Noun horse)))

               (PP

                  (Prep near)

                  (NP

                     (Det the)

                     (Nbar

                        (Noun castle))))))) .))

1. **the Holy Grail is a chalice .**
   1. **The vocabulary rules added**
      1. **1 ProperO Holy Grail**

ProperO to denote non-people proper nouns, such as “Holy Grail”

* 1. **The grammar rules added**
     1. **1 Nbar ProperO**

This rule helps Nbar to include ProperO, so that phrase like “Holy Grail” would be included one of the many types of nouns.

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Det the)

            (Nbar

               (ProperO Holy) Grail)))

         (VP

            (VerbT is)

            (NP

               (Det a)

               (Nbar

                  (Noun chalice))))) .))

1. **the sensational Holy Grail is a sacred chalice .**
   1. **The vocabulary rules added**
      1. **1 AdjN sensational**

“sensational” is a regular non-comparativeadjective.

* + 1. **AdjN sacred**

“sacred” is another regular non-comparativeadjective.

* 1. **The grammar rules added**

None

* 1. **Parse Tree after adding rules**

PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (@Det-Adj

               (Det the)

               (Adj sensational))

            (Nbar

               (ProperO Holy) Grail)))

         (VP

            (VerbT is)

            (NP

               (@Det-Adj

                  (Det a)

                  (Adj sacred))

               (Nbar

                  (Noun chalice))))) .))

1. **every coconut was carried to the hottest mountains .**
   1. **The vocabulary rules added**
      1. **1 VerbP was**

VerbP to denote past tense verbs, such as “was”

* + 1. **1 VerbP carried**

VerbP to denote past tense verbs, such as “carried”

* + 1. **1 AdjS hottest**

AdjS to denote the superlative adjectives, such as “hottest”

* + 1. **1 NounS mountains**

NounS to denote the plural nouns, such as “mountains”

* + 1. **1 To to**

“To” to denote “to” exclusively

* 1. **The grammar rules added**
     1. **1 VP VerbP VP**

This rule introduces past tense verbs to verb phrases, so that a verb phrase phrase like “was carried to the hottest mountain” would be parsed into “was” (VerbP) and “carried to the hottest mountain” (VP).

* + 1. **1 VP VerbP PP**

This rule further separates a verb phrase into a past tense verb and a prepositional phrase, so that a verb phrase like “carried to the hottest mountain” would be parsed into “carried” (VerbP) and “to the hottest mountain” (PP).

* + 1. **1 PP To NP**

This rule further separates a prepositional phrase into “to” and a noun phrase, so that a prepositional phrase like “to the hottest mountain” would be parsed into “to” (To) and “the hottest mountain” (NP).

* + 1. **1 Adj AdjS**

This rule introduces the superlative adjectives to Adj, so that a superlative adjective like “hottest” would be included in Adj.

* + 1. **1 Nbar NounS**

This rule introduces the plural nouns as an inner level of noun phrases, so that a plural noun like “mountains” would be included in Nbar.

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Det every)

            (Nbar

               (Noun coconut)))

         (VP

            (VerbP was)

            (VP

               (VerbP carried)

               (PP

                  (To to)

                  (NP

                     (@Det-Adj

                        (Det the)

                        (Adj

                           (AdjS hottest)))

                     (Nbar

                        (NounS mountains))))))) .))

1. **sixty strangers are at the Round Table .**
   1. **The vocabulary rules added**
      1. **1 NounS strangers**
      2. **1 Num sixty**

“Num” to denote numbers.

* + 1. **1 VerbTS are**

“VerbTS” to denote verbs that are present, plural and third person

* + 1. **1 ProperO Round Table**
  1. **The grammar rules added**
     1. **1 NP Num Nbar**

This rule separates a noun phrase into a number and a noun, so that a noun phrase like “Sixty strangers” would be parsed into “sixty” (Num) and “Strangers” (Nbar).

* + 1. **1 VP VerbTS PP**

This rule separates a verb phrase into a verb that are present, plural and third person and a prepositional phrase, so that a verb phrase like “are at the Round Table” would be parsed into “are” (VerbTS) and “at the round table” (PP).

* + 1. **1 NP Det Nbar**

This rule separates a noun phrase into a determiner and a noun, so that a noun phrase like “the Round Table” would be parsed into “the” (Det) and “Round Table” (Nbar).

* 1. **Parse Tree after adding rules**

(START

   (S1

      (@NP-VP

         (NP

            (Num sixty)

            (Nbar

               (NounS strangers)))

         (VP

            (VerbTS are)

            (PP

               (Prep at)

               (NP

                  (Det the)

                  (Nbar

                     (ProperO Round) Table)))))) .))

1. **Sir Lancelot might have spoken .**
   1. **The vocabulary rules added**
      1. **1 Mod might**

“Mod” to denote modals like “might”.

* + 1. **1 VerbN have**

“VerbN” to denote a verb in its normal form like “have”.

* + 1. **1 VerbPaP spoken**

“VerbPaP” to denote a verb in past participle form like “spoken”.

* 1. **The grammar rules added**
     1. **1 VP Mod VP**

This rule separates a verb phrase into a modal word and another verb phrase, so that a verb phrase like “might have spoken” would be parsed into “might” (Mod) and “have spoken” (VP).

* + 1. **1 VP VerbN VP**

This rule separates a verb phrase into a normal form verb and another verb phrase, so that a verb phrase like “have spoken” would be parsed into “have” (VerbN) and “spoken” (VP).

* + 1. **1 VP VerbPaP**

This rule adds a verb in past participle from to VP, because a past particle like “spoken” could act as a verb phrase.

* 1. **Parse Tree after adding rules**

(START

   (S1

      (@NP-VP

         (NP

            (Proper Sir) Lancelot))

         (VP

            (Mod might)

            (VP

               (VerbN have)

               (VP

                  (VerbPaP spoken))))) .))

1. **Guinevere had been riding with Patsy for five weary nights .**
   1. **The vocabulary rules added**
      1. **1 VerbP had**
      2. **1 VerbPaP been**
      3. **1 VerbPrP riding**

“VerbPrP” to denote a verb in present participle form like “riding”.

* + 1. **1 Num five**
    2. **1 AdjN weary**
    3. **1 NounS nights**
  1. **The grammar rules added**
     1. **1 VP VerbPaP VP**

This rule separates a verb phrase into a verb in past participle form and another verb phrase, so that a verb phrase like “been riding with Patsy” would be parsed into “been” (**VerbPaP**) and “riding with Patsy” (VP).

* + 1. **1 VP VerbPrP PP NP**

This rule separates a verb phrase into a verb in present participle form, prepositional phrase, and a noun phrase, so that a verb phrase like “riding with Patsy for five weary nights” would be parsed into “riding” (**VerbPrP**), “with Patsy for five” (PP) and “weary nights” (NP).

* + 1. **1 NP Proper PP**

This rule separates a noun phrase into a proper noun for person and a prepositional phrase, so a noun phrase like “Patsy for five” would be parsed into “Patsy” (Proper) and “for five” (PP).

* + 1. **1 NP Num**

This rule treats number as the only part of a noun phrase, so a noun phrase like “five” would be parsed into “five” (Num).

* + 1. **1 NP Adj Nbar**

This rule separates a noun phrase into an adjective and a noun, so a noun phrase like “weary nights” would be parsed into “weary” (Adj) and “nights” (Nbar)

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Proper Guinevere))

         (VP

            (VerbP had)

            (VP

               (VerbPaP been)

               (VP

                  (@VerbPrP-PP

                     (VerbPrP riding)

                     (PP

                        (Prep with)

                        (NP

                           (Proper Patsy)

                           (PP

                              (Prep for)

                              (NP

                                 (Num five))))))

                  (NP

                     (Adj

                        (AdjN weary))

                     (Nbar

                        (NounS nights))))))) .))

1. **Sir Bedevere might have been suggesting this quest .**
   1. **The vocabulary rules added**
      1. **1 VerbPrP suggesting**
   2. **The grammar rules added**
      1. **1 VP VerbPrP NP**

This rule separates a verb phrase into a verb in present participle form, and a noun phrase, so that a verb phrase like “suggesting this quest” would be parsed into “suggesting” (VerbPrP) and “this quest” (NP).

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Proper Sir) Bedevere))

         (VP

            (Mod might)

            (VP

               (VerbN have)

               (VP

                  (VerbPaP been)

                  (VP

                     (VerbPrP suggesting)

                     (NP

                        (Det this)

                        (Nbar

                           (Noun quest)))))))) .))

1. **the Britons migrate south frequently .**
   1. **The vocabulary rules added**
      1. **1 ProperOS Britons**

“ProperOS” to denote a proper noun in plural form like “Britons”.

* + 1. **1 VerbTS migrate**
    2. **1 Adv south**

“**Adv**” to denote an adverb like “south”.

* + 1. 1 Adv frequently
  1. **The grammar rules added**
     1. **1 NP Det ProperOS**

This rule separates a noun phrase into a determiner word and a proper noun in plural form, so a noun phrase like “the Britons” would be parsed into “the” (Det) and “Britons” (ProperOS)

* + 1. **1 VP VerbTS Advp**

This rule separates a verb phrase into an verbs that are present, plural and third person and an adverb phrase (Advp), so a verb phrase like “migrate south frequently” would be parsed into “migrate” (VerbTS) and “south frequently” (Advp)

* + 1. **1 Advp Adv Adv**

This rule breaks down an adverb phrase into two adverbs, so an adverb phrase like “south frequently” would be parsed into “south” (Adv) and “frequently” (Adv)

* 1. **Parse Tree after adding rules**

(START

   (S1

      (@NP-VP

         (NP

            (Det the)

            (ProperOS Britons))

         (VP

            (VerbTS migrate)

            (Advp

               (Adv south)

               (Adv frequently)))) .))

1. **Arthur and Guinevere ride frequently near the castle .**
   1. **The vocabulary rules added**
      1. **1 Conj and**

“Conj” denotes coordinating conjunctions words like “and”

* + 1. 1 VerbTS ride
  1. **The grammar rules added**
     1. **1 NP Proper Conj Proper**

This rule separates a noun phrase into a proper noun for a person’s name, a conjugation, and another proper noun for a person’s name, so a noun phrase like “Arthur and Guinevere” would be parsed into “Arthur” (Proper), “and” (Conj), and “Guinevere” (Proper)

* + 1. **1 VP VerbTS Advp PP**

This rule separates a verb phrase into a verb that is present, plural and third person, an adverb phrase, and a prepositional phrase, so a verb phrase like “ride frequently near the castle” would be parsed into “ride” (VerbTS), “frequently” (Advp), and “near the castle” (PP)

* + 1. **1 Advp Adv**

This rule breaks down an adverb phrase into one adverb

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (@Proper-Conj

               (Proper Arthur)

               (Conj and))

            (Proper Guinevere))

         (VP

            (@VerbTS-Advp

               (VerbTS ride)

               (Advp

                  (Adv frequently)))

            (PP

               (Prep near)

               (NP

                  (Det the)

                  (Nbar

                     (Noun castle)))))) .))

1. **he suggests to grow fruit at home .**
   1. **The vocabulary rules added**
      1. **1 ProP he**

“ProP” to denote personal pronouns like “he”.

* + 1. **1 VerbT suggests**
    2. **1 VerbN grow**
  1. **The grammar rules added**
     1. **1 NP ProP**

This rule introduces a personal noun as the sole part of a noun phrase, so a noun phrase could be “he”, “she”, or “it”.

* + 1. **1 VP VerbT VP**

This rule separates a verb phrase into third person singular verb, and a verb phrase, so a verb phrase like “suggests to grow fruit” would be parsed into “suggests” (VerbT), and “to grow fruit” (VP)

* + 1. **1 VP To VP**

This rule separates a verb phrase into a “To” and a verb phrase, so a verb phrase like “to grow fruit” would be parsed into “to” (To), and “grow fruit” (VP)

* + 1. **1 VP VerbN NP PP**

This rule separates a verb phrase into a normal form verb, a noun phrase and a prepositional phrase, so a verb phrase like “to grow fruit” would be parsed into “grow” (VerbN), “fruit” (NP), and “at home” (PP).

* + 1. **15 NP Nbar**

This rule makes a noun the sole element of a noun phrase. So “home” alone could be a NP (no pun intended).

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (ProP he))

         (VP

            (VerbT suggests)

            (VP

               (To to)

               (VP

                  (@VerbN-NP

                     (VerbN grow)

                     (NP

                        (Nbar

                           (Noun fruit))))

                  (PP

                     (Prep at)

                     (NP

                        (Nbar

                           (Noun home)))))))) .))

1. **riding to Camelot is not hard .**
   1. **The vocabulary rules added**
      1. **1 ProperO Camelot**
      2. **1 Not not**

“Not” to indicate the word “not”.

* + 1. **1 AdjN hard**
  1. **The grammar rules added**
     1. **1 S1 VP VP .**

This rule separates a whole sentence into two verb phrases, so a sentence like “Riding to Camelot is not hard” would be parsed into “Riding to Camelot” (VP) and “is not hard” (VP).

* + 1. **1 VP VerbPrP PP**

This rule separates a verb phase into a verb in present article form and a prepositional phrase, so a phrase like “riding to Camelot” would be parsed into “Riding” (VerbPrP) and “to Camelot” (PP).

* + 1. **1 VP VerbT Not Adj**

This rule separates a verb phase into a third person singular verb, a “not”, and an adjective phrase, so a phrase like “is not hard” would be parsed into “is” (Verb), “not” (Not), “hard” (Adj).

* 1. **Parse Tree after adding rules**

(START

   (S1

      (@VP-VP

         (VP

            (VerbPrP riding)

            (PP

               (To to)

               (NP

                  (Nbar

                     (ProperO Camelot)))))

         (VP

            (@VerbT-Not

               (VerbT is)

               (Not not))

            (Adj

               (AdjN hard)))) .))

1. **do coconuts speak ?**
   1. **The vocabulary rules added**
      1. **1 Do do**

“Do” to denote the “do words” like “do” and “does” in the sense of “do you know him?”, not in the sense of being a verb like “do the dishes”

* + 1. **1 NounS coconuts**
    2. **1 VerbN speak**
  1. **The grammar rules added**
     1. **1 S1 SQ**

This rule signifies that the whole sentence is an SQ, which indicates an inverted yes/no question, or main clause of a wh-question, like this one (Do coconuts speak).

* + 1. **1 SQ Do NP VP ?**

This rule separates an inverted yes/no question into a Do, a noun phrase, a verb phrase and a question mark, so a phrase like “do coconuts speak ?” would be parsed into “do” (Do), “coconuts” (NP), “speak” (VP) and “?” (misc).

* + 1. **1 VP VerbN**

This rule signifies that a verb noun is made by one normal form verb only, in this case, VP is made of “speak” only.

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (SQ

         (@@Do-NP-VP

            (@Do-NP

               (Do do)

               (NP

                  (Nbar

                     (NounS coconuts))))

            (VP

               (VerbN speak))) ?)))

1. **why does England have a king ?**
   1. **The vocabulary rules added**
      1. **1 Whadv why**

“Whadv” category to indicate wh-adverbs like “why”.

* + 1. **1 Do does**
    2. **1 ProperO England**
  1. **The grammar rules added**
     1. **1 S1 DQ**

This rule signifies that the whole sentence is an DQ, which indicates a direct question like this one (“why does England have a king ?).

* + 1. **1 DQ Whadv SQ**

This rule separates a direct question into a wh-adverb and an inverted yes-no question, so a phrase like “why does England have a king ?” would be parsed into “why” (Whadv), and “does England have a king ?” (SQ).

* + 1. **1 VP VerbN NP**

This rule separates a verb phrase into a normal form verb and a noun phrase, so a phrase like “have a king ?” would be parsed into “have” (VerbN), and “a king” (NP).

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (DQ

         (Whadv why)

         (SQ

            (@@Do-NP-VP

               (@Do-NP

                  (Do does)

                  (NP

                     (Nbar

                        (ProperO England))))

               (VP

                  (VerbN have)

                  (NP

                     (Det a)

                     (Nbar

                        (Noun king))))) ?))))

**Challenge Sentences**

1. **what horse does Arthur ride ?**
   1. **The vocabulary rules added**
      1. **1 WhDt what**

“WhDT” vocab category to indicate “wh-determiners” like “what”.

* + 1. **1 VerbN ride**

In this case, ride is used as a normal form verb, instead of the one we had earlier used as a third person plural form verb (“they ride”)

* 1. **The grammar rules added**
     1. **1 DQ WhN SQ ?**

This rule separates a direct question into a what-noun phrase (WhN), an inverted yes-no question (SQ), and a question mark (?), so a phrase like “what horse does Arthur ride ?” would be parsed into “what horse” (WhN), “does Arthur ride” (SQ), and “?” (misc)

* + 1. **1 WhN WhDt Nbar**

This rule separates a what-noun phrase into a wh-determiner and a noun, so a phrase like “what horse” would be parsed into “what” (WhDt) and “horse” (Nbar)

* + 1. **1 SQ Do NP VP**

This rule separates an inverted yes/no question into a Do, a noun phrase and a verb phrase, so a phrase like “does Author ride ?” would be parsed into “does” (Do), “Author” (NP) and “ride” (VP).

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (DQ

         (@WhN-SQ

            (WhN

               (WhDt what)

               (Nbar

                  (Noun horse)))

            (SQ

               (@Do-NP

                  (Do does)

                  (NP

                     (Proper Arthur)))

               (VP

                  (VerbN ride)))) ?)))

1. **who does Arthur suggest she carry ?**
   1. **The vocabulary rules added**
      1. **1 WhPro who**

“WhPro” vocab category to indicate “wh-pronouns” like “who”.

* + 1. **1 VerbN suggest**
    2. **1 ProP she**
    3. **1 VerbN carry**
  1. **The grammar rules added**
     1. **1 WhN WhPro**

This rule signifies that the whole what-noun phrase is made of one single wh-pronoun, like “who”.

* + 1. **1 VP VerbN NP VP**

This rule separates a verb phrase into normal form of verb, a noun phrase, and a verb phrase, so a phrase like “suggest she carry” would be parsed into “suggest” (VerbN), “she” (NP), and “carry” (VerbN).

* 1. **Parse Tree after adding rules**

[PCFGParser] best parse tree:

(START

   (S1

      (DQ

         (@WhN-SQ

            (WhN

               (WhPro who))

            (SQ

               (@Do-NP

                  (Do does)

                  (NP

                     (Proper Arthur)))

               (VP

                  (@VerbN-NP

                     (VerbN suggest)

                     (NP

                        (ProP she)))

                  (VP

                     (VerbN carry))))) ?)))

**Part 2**

1. **make up a sentence that uses some of the same words in the sentences that you already parsed and is an actual English sentence, but cannot be parsed by those rules.**
   1. **Sentence I made up:**

what does she have ?

* 1. **Its S2 parse tree:**

(START

   (S2

      (\_WhDt

         (WhDt what)

         (\_Do

            (Do does)

            (\_ProP

               (ProP she)

               (\_VerbN

                  (VerbN have)

                  (\_Misc

                     (Misc ?))))))))

* 1. **Why it didn’t work out?**

I think the reason “what horse does Arthur ride ?” could be parsed by my rules but not “what does she have ?” is because the “what” in “what horse does Arthur ride ?” is a wh-determiner that determines some kind of horse, but the “what” in “what does she have ?” is a “wh-pronoun” which refers to some actual “thing(s)”. We don’t have a vocab rule for what acting as a wh-pronoun, that’s why it would not parse.

1. **make up a string of words that should not be an actual English sentence, but your grammar will parse it.**
   1. **“Sentence” I made up:**

Arthur suggest she to grow .

* 1. **Its S1 parse tree:**

[PCFGParser] best parse tree:

(START

   (S1

      (@NP-VP

         (NP

            (Proper Arthur))

         (VP

            (@VerbN-NP

               (VerbN suggest)

               (NP

                  (ProP she)))

            (VP

               (To to)

               (VP

                  (VerbN grow))))) .))

[PCFGParser] cross-entropy=5.144 perplexity=3.537e+01

* 1. **Why was it parsed?**
     1. Because of the rule I wrote earlier “VP VerbN NP VP ” parsed “suggest she to grow” successfully into “suggest” (VerbN), “she” (NP), and “to grow” (VP), which did not check whether noun phrase “she” is compatible with normal form verb “suggest” in this instance.
     2. This whole flawed VP is overgeneralized into the rule “S1 NP VP .” This overgeneralization did not check whether VerbN “suggest” in VP is compatible with the proper noun “Author” that came before it, so this grammatically flawed sentence is still parsed.