Question 1b (1 pt)

For which sentences, if any, does your grammar produce more

than one parse? Explain whether this is as it should be, or not.

If there are sentences with multiple parses, indicate in your explanation

which of these parses corresponds to the most likely interpretation of

the sentence, according to you, and why

Answer:

Thirteen sentences had one parse, and two had two parses. The two sentences that had two parses were:

- "research fans hope for spinal injuries"

  - Ambiguity: One interpretation is that research fans are hoping to get or have spinal injuries. The other is that, even if not named, there is some hope for spinal injuries, and research fans are interested in that hope.

  - It comes down to parsing hope as either a noun or a verb.
    So, either:

    - S → NP [research fans] PP [hope for spinal injuries]

    - S → NP [research fans] NP [hope] PP [for spinal injuries]

  - More likely: The second one, because it makes more sense that there is some hope for spinal injuries than that research fans are hoping to get spinal injuries.

- "sports fans hope for likes"

  - Ambiguity: One interpretation is that sports fans are hoping to get likes, and the other is that sports fans have some hope for likes.
    So, either:

    - S → NP [sports fans] PP [hope for likes]

    - S → NP [sports fans] VP [hope] PP [for likes]

  - More likely: We believe the first one is more natural, As it's more common to say that sports fans are hoping for likes than to talk about a "hope for likes" that they possess.

Question 2a (1 pt) Convert the grammar to Chomsky Normal Form (CNF).

Answer:

Grammar rules

S → NP VP

VP → V NP

VP → V NP PP --not in CNF

NP → Nominal --not in CNF

NP → Det Nominal

NP → Nominal PP

NP → Det Nominal PP --not in CNF

Nominal → Noun --not in CNF

PP → P NP

Lexicon

Noun → students, problems, AI

P → with, in

V → solve, like, hate

Det → a, the

Chomsky normal form CNF version of the grammar

S -> NP VP

VP -> V NP

VP -> V VP1

VP1 -> NP PP

NP -> students | problems | AI

NP -> Det Nominal

NP -> Nominal PP

NP -> Det NP1

NP1 -> Nominal PP

PP -> P NP

Nominal -> students | problems | AI

V -> solve | like | hate

P -> with | in

Det -> a | the

_____

Question 2b (2 pt) Using the CNF version of the grammar, draw a CKY parse table

for the sentence students solve problems with AI. For the last column,

show step-by-step how it is filled, in the same way as is done in Figure
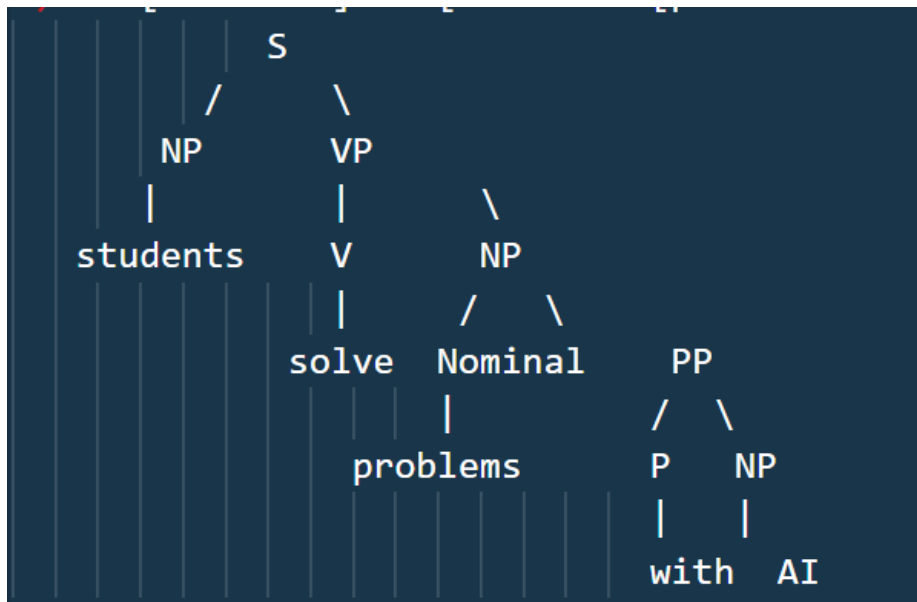
18.14 of J&M Ch. 18.

Answer:

STUDENTS	SOLVE	PROBLEMS	WITH	AI
N?, Nominal				S, VP1, NP, NP1
[0,1)	[0,2]	[0,3]	[0,4]	[0,5] NP1
	V	VP		VP
	[1,2)	[1,3]	[1,4]	[1,5]
		NP, Nominal		NP1, VP1 / NP
		[2,3]	[2,4]	[2,5]
			P	PP
			[3,4]	[3,5]
				N?, Nominal
				[4,5]

[3,5]

PP → P NP   [3,4], [4,5]

[2,5]

VP1 → N P PP   [2,3] [3,5]

NP1 → Nominal PP   [2,3], [3,5]

N? → Nominal PP   [2,3], [3,5]

[1,5]

VP → V VP1   [1,2] [2,5]

VP → V NP   [1,2], [2,5]
            [1,2], [4,5]

[0,5]

S → NP VP   [0,1] [1,5]

VP1 → NP PP   [0,1] [3,5]

NP → Nominal PP   [0,1] [3,5]
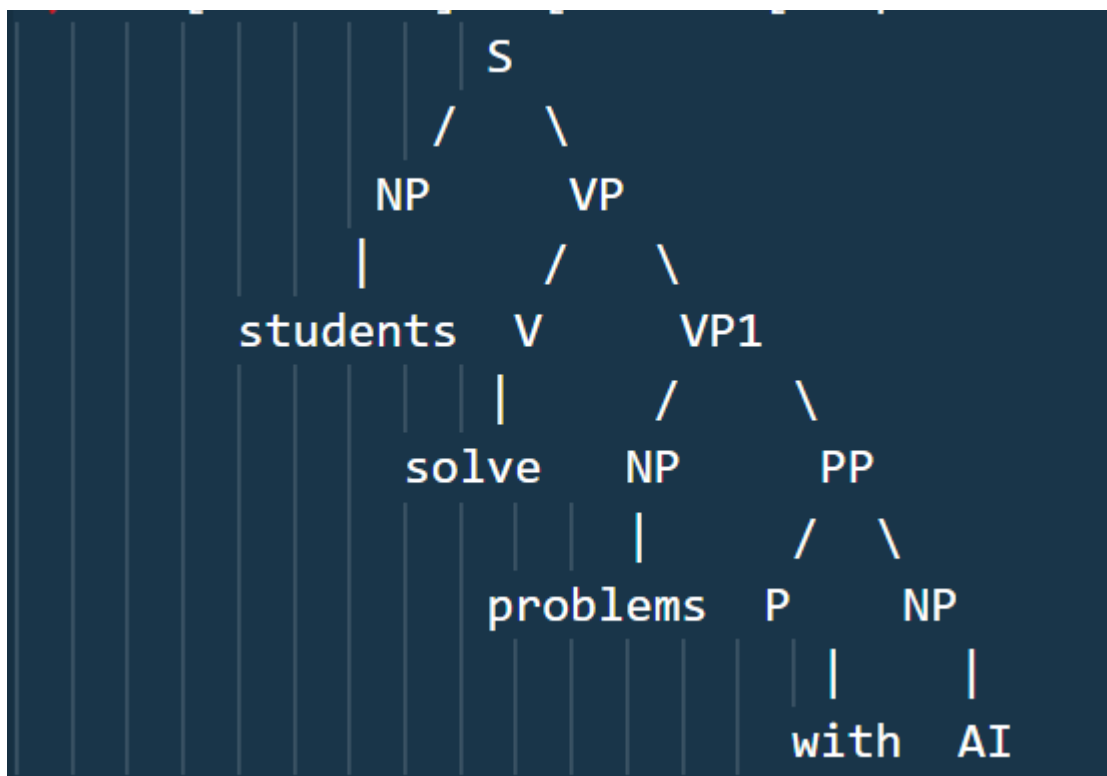
NP1 → Nominal PP   [0,1] [3,5]

Question 2c (0.5 pt) Draw the parse tree or parse trees (with root S) that can be

derived from the filled CKY parse table.

Answer:

a) NP [students] VP [solve NP [problems PP [with AI]]]

```
            S
          /   \
      NP         VP
      |          |      \
  students       V        NP
                 |       /   \
              solve   Nominal    PP
                         |      /  \
                      problems  P   NP
                                |   |
                              with  AI
```

b) NP [students] VP[solve [NP problems with AI]]

```
                  S
                /   \
            NP         VP
            |         /   \
        students   V       VP1
                   |      /     \
                solve   NP        PP
                        |       /   \
                     problems  P     NP
                               |     |
                             with   AI
```

Several dependency parsers are available as part of NLP toolkits, such as NLTK1 or the Stanford CoreNLP package2

In this exercise we try out the dependency parser of SpaCy, another open-source NLP software library. We

use the web demo of its dependency visualizer: https://explosion.ai/demos/display

Question 3a (0.5 pt) Use the web demo to parse the ambiguous sentence "I shot a man in clown shoes." .You have to uncheck the "Merge phrases" checkbox. What do you think about the way this sentence was parsed? Is it a good parse or not? Why (or why not)?

Answer:

We do think it is a good dependency parse. It assigns phrase "in clown shoes" to the noun man, meaning subject "I" shot a person who was wearing clown shoes. This is one of the two possible meanings of the sentence

The other meaning would be that "I" was wearing clown shoes when "I shot a man". Syntactically, both parses are good. Neither meanings would be more probable in this context.

_____

Question 3b (1 pt) Now, use the web demo to parse the following variations of the

previous sentence. Keep the "merge phrases" checkbox unchecked.

  1. I shot an elephant in clown shoes

  2. I shot a man in my clown shoes

  3. I love an elephant in my clown shoes

  4. women hate men in clown shoes

  5. men hate women in clown shoes

Discuss the differences between the dependency parses SpaCy assigns

to these sentences. Do you think they are good parses? Why (not)?

Can you think of reasons for the different parses?

Answer:

  1. I shot an elephant in clown shoes

  - > parsed as: I shot an elephant that was wearing clown shoes

  2. I shot a man in my clown shoes

  - > parsed as: I shot a man while I was wearing my clown shoes.

  3. I love an elephant in my clown shoes

  - > parsed as: I love an elephant while wearing my clown shoes.

  4. women hate men in clown shoes

- > parsed as: women hate men that are wearing clown shoes.

5. men hate women in clown shoes

- > parsed as: men hate women while wearing clown shoes.

They are good parses because they all make sense and are one of the possible meanings of the sentences. Therefore, they are all syntactically correct. Issue is that probable meaning of the sentences is not the one that was parsed. We would therefore argue that semantically they are not good parses. First sentence makes no sense, as interpreted by the online tool. Another issue is that 4th and 5th sentence are parsed differently, even though they are syntactically identical. The only difference is the subject and object are swapped, or we could say genders of subject and object were swapped. This should not change the parse.

The reason for the different parses could be that the model was trained on a corpus where men are more often being described wearing something (eg in a black suit). This is a bias that presents a problem, since no gender swap should yield a different parse.

To conclude, the tool is good at parsing syntactically, but not semantically. Issues arise due to training data/lexicon bias, NP/VP attachment problems and possessiveness placed outside their usual context.

Question 3c (2 pt) Take the dependency parse produced by SpaCy for I shot a

man in clown shoes (from 3a). Provide the trace of this dependency

parse as it would be produced by a transition-based, arc-standard

dependency parser. Use the notation that is also used in Figure 19.6

of J&M Chapter 19.

Answer:

0 [root][I shot a man in clown shoes] SHIFT

1 [root I][shot a man in clown shoes] SHIFT

2 [root I shot][a man in clown shoes] LEFT-ARC (shot → I)

3 [root shot][a man in clown shoes]   SHIFT

4 [root shot a][man in clown shoes]   SHIFT

5 [root shot a man][in clown shoes]   LEFT-ARC (man → a)

6 [root shot man][in clown shoes]   SHIFT

7 [root shot man in][clown shoes]   SHIFT

8 [root shot man in clown][shoes]   SHIFT

9 [root shot man in clown shoes]   LEFT-ARC (shoes → clown)

10 [root shot man in shoes]   RIGHT-ARC (in → shoes)

11 [root shot man in]   RIGHT-ARC (man → in)

12 [root shot man][]   RIGHT-ARC (shot → man)

13 [root shot][]   RIGHT-ARC (root → shot)

14 [root][]   DONE