School of Computing and Information Systems
The University of Melbourne
COMP90042
WEB SEARCH AND TEXT ANALYSIS (Semester 1, 2017)

Sample solutions for discussion exercises: Week 4

Discussion

1. What is **chart parsing**? Why is it important?
   - **Parsing** in general is the process of identifying the structure(s) of a sentence, according to a grammar of the language.
   - In general, the search space is too large to do this efficiently, so we use a dynamic programming method to keep track of partial solutions. The data structure we use for this is a **chart**, where entries in the chart correspond to partial parses (licensed structures) for various spans (sequences of tokens) within the sentence.
2. Consider the following simple **context–free grammar**:

```
S -> NP VP
VP -> V NP | V NP PP
PP -> P NP
V -> "saw" | "walked"
NP -> "John" | "Bob" | Det N | Det N PP
Det -> "a" | "an" | "the" | "my"
N -> "man" | "cat" | "telescope" | "park"
P -> "on" | "by" | "with"
```

   (a) What changes need to be made to the grammar to make it suitable for **CYK parsing**?
      - For CYK parsing, a grammar needs to be written in Chomsky Normal Form, where each rule consists of either:
        - a (single) non-terminal which re-writes as a single terminal, or
        - a (single) non-terminal which re-writes as exactly two non-terminals
      - Here, we have two rules where a non-terminal re-writes as three non-terminals (`VP -> V NP PP` and `NP -> Det N PP`); we remove these rules and replace them with the following:

```
VP -> V X
X -> NP PP
NP -> Det Y
Y -> N PP
```

   (b) Using the CYK strategy and the above grammar in CNF, parse the following sentences:
      i. "a man saw John"
      ii. "an park by Bob walked an park with Bob"
      iii. "park by the cat with my telescope"
         - This sentence has no parse, because the cell [0,7] doesn't have an `S`.

a	*man*	*saw*	*John*
[0,1] Det	[0,2] NP	[0,3] -	[0,4] S
	[1,2] N	[1,3] -	[1,4] -
		[2,3] V	[2,4] VP
			[3,4] NP

an	*park*	*by*	*Bob*	*walked*	*an*	*park*	*with*	*Bob*
[0,1] Det	[0,2] NP	[0,3] -	[0,4] NP, X	[0,5] -	[0,6] -	[0,7] S	[0,8] -	[0,9] S, S
	[1,2] N	[1,3] -	[1,4] Y	[1,5] -	[1,6] -	[1,7] -	[1,8] -	[1,9] -
		[2,3] P	[2,4] PP	[2,5] -	[2,6] -	[2,7] -	[2,8] -	[2,9] -
			[3,4] NP	[3,5] -	[3,6] -	[3,7] S	[3,8] -	[3,9] S, S
				[4,5] V	[4,6] -	[4,7] VP	[4,8] -	[4,9] VP, VP
					[5,6] Det	[5,7] NP	[5,8] -	[5,9] NP, X
						[6,7] N	[6,8] -	[6,9] Y
							[7,8] P	[7,9] PP
								[8,9] NP

park	*by*	*the*	*cat*	*with*	*my*	*telescope*
[0,1] N	[0,2] -	[0,3] -	[0,4] Y	[0,5] -	[0,6] -	[0,7] Y
	[1,2] P	[1,3] -	[1,4] PP	[1,5] -	[1,6] -	[1,7] PP
		[2,3] Det	[2,4] NP	[2,5] -	[2,6] -	[2,7] NP, X
			[3,4] N	[3,5] -	[3,6] -	[3,7] Y
				[4,5] P	[4,6] -	[4,7] PP
					[5,6] Det	[5,7] NP
						[6,7] N

3. What is a **probabilistic grammar** and what problem does it attempt to solve?

   - A probabilistic grammar is one where each production (rule) is associated with a probability.
   - These probabilities can be estimated from a parsed corpus (**treebank**), and reflect the fact that some constructions, while possible according to the grammar, are not very likely in actual language.
   - More generally, it allows us to disambiguate sentences with multiple parses. While such sentences might be legitimately ambiguous, under normal circumstances only one reading is actually intended.
   - In the case of the CYK chart, we mark up each non-terminal in each cell with its corresponding probability. When we are filling in a new cell, we calculate the probability of the derived non-terminal by multiplying three values: the probability of each of the two sub-spans, and the probability associated with the production.
   - For example, in the case of cell [0,2] in the chart for "an park by Bob walked an park with Bob", we would observe an `NP`, whose probability would be the product of the probabilities of the following three rules: `Det->"an"`, `N->"park"`, and `NP->Det N`.
   - For the case of cell [0,4], the probabilities don't help us to distinguish between the productions leading to `NP` and `X`, although if there was true ambiguity, they might matter in later productions.
   - On the other hand, at the cell [4,9], we have two ambiguous `VP` productions. Only one of these can be part of the correct (intended) reading: using a probabilistic parser, we can reject the less likely (lower probability) reading now, which means that there will be less ambiguity later in the chart (specifically, at [3,9] and [0,9]).
     You might like to work through the details, but effectively we will decide between the two ambiguous structures for this `VP` based on whether the product of the probabilities of the rules `NP->Det N PP` and `VP->Det N` is greater than `NP->Det N` and `VP->Det N PP`. (The probabilities of the rest of the derivation are equivalent, because the same rules appear, and multiplication is associative.)

   Parse the sentences from the Discussion section using the Earley strategy.

   - Note that we immediately find no analyses for the third sentence, because the top-down strategy of Earley infers that sentences must begin with `Det`, according to this grammar.

0	1	2	3
γ->·S [0] S->·NP VP [0] NP->·Det N [0] NP->·Det N PP [0]	Det->"a"· [0] NP->Det·N [0] NP->Det·N PP [0]	N->"man"· [1] NP->Det N· [0] NP->Det N·PP [0] S->NP·VP [0] PP->·P NP [2] VP->·V NP [2] VP->·V NP PP [2]	V->"saw"· [2] VP->V·NP [2] VP->V·NP PP [2] NP->·Det N [3] NP->·Det N PP [3]
4			
NP->"John"· [3] VP->V NP· [2] VP->V NP·PP [2] S->NP VP· [0] PP->·P NP [3] γ->S· [0]			

0	1	2	3
γ->·S [0] S->·NP VP [0] NP->·Det N [0] NP->·Det N PP [0]	Det->"an"· [0] NP->Det·N [0] NP->Det·N PP [0]	N->"park"· [1] NP->Det N· [0] NP->Det N·PP [0] S->NP·VP [0] PP->·P NP [2] VP->·V NP [2] VP->·V NP PP [2]	P->"by"· [2] PP->P·NP [2] NP->·Det N [3] NP->·Det N PP [3]
4	5	6	7
NP->"Bob"· [3] PP->P NP· [2] NP->Det N PP· [0] S->NP·VP [0] VP->·V NP [4] VP->·V NP PP [4]	V->"walked"· [4] VP->V·NP [4] VP->V·NP PP [4] NP->·Det N [5] NP->·Det N PP [5]	Det->"an"· [5] NP->Det·N [5] NP->Det·N PP [5]	N->"park"· [6] NP->Det N· [5] NP->Det N·PP [5] VP->V NP· [4] VP->V NP·PP [4] PP->·P NP [7] S->NP VP· [0]
8	9		
P->"with"· [7] PP->P·NP [7] NP->·Det N [8] NP->·Det N PP [8]	NP->"Bob"· [8] PP->P NP· [7] NP->Det N PP· [5] VP->V NP PP· [4] VP->V NP· [4] VP->V NP·PP [4] S->NP VP· [0] PP->·P NP [9] γ->S· [0]		

0	1	2	3
γ->·S [0] S->·NP VP [0] NP->·Det N [0] NP->·Det N PP [0]	N->"park"· [0]	P->"by"· [1]	Det->"the"· [2]
4	5	6	7
N->"cat"· [3]	P->"with"· [4]	Det->"my"· [5]	N->"telescope"· [6]