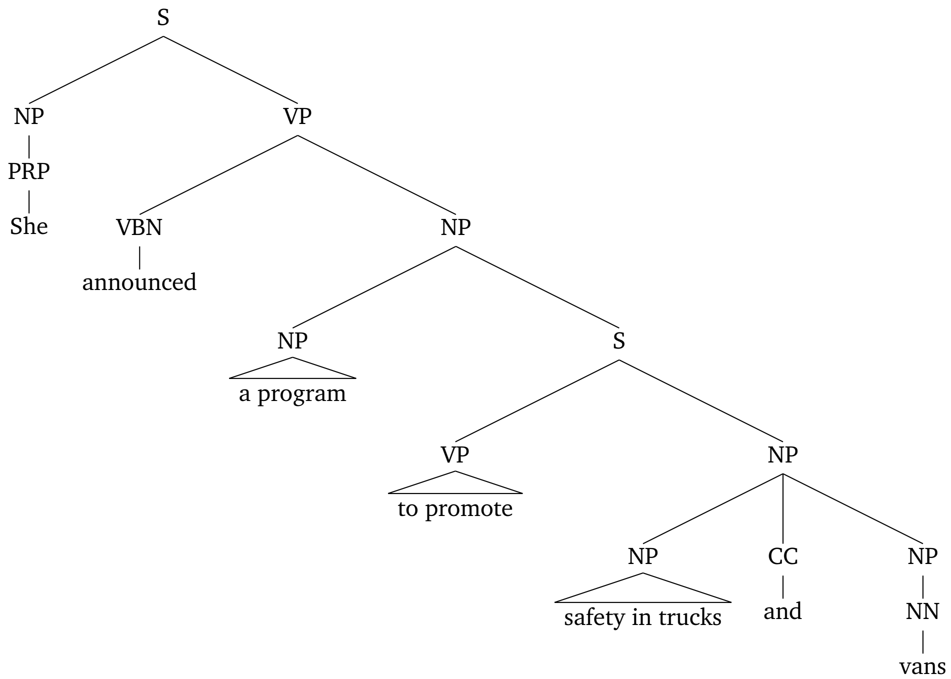
Advanced Human Language Technologies
Exercises on Parsing

Context Free Grammars

Exercise 1.

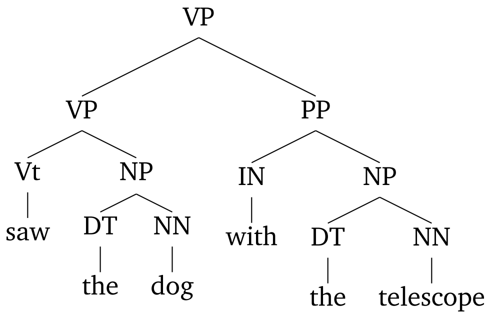
Consider the sentence *She announced a program to promote safety in trucks and vans*
and the following syntactic tree of one of its possible interpretations, in which the program promotes safety in trucks, and also promotes vans:

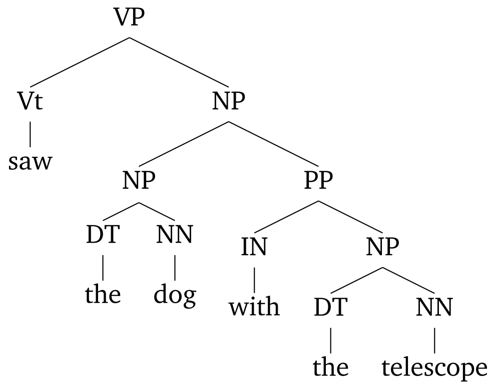


1. Draw the trees for at least three other interpretations for this sentence
2. Draw the trees for at least two interpretations for each of the following sentences
   - *The post office will hold out discounts and service concessions as incentives*
   - *They are hunting lions and tigers*
   - *Monty flies like mosquitoes*

Exercise 2.

Say we have the phrase *saw the dog with the telescope* and we are given the gold parse tree (left) and the predicted parse tree (right):





What are the precision and recall of this predicted parse tree?

Exercise 3.

Consider the following CFG:

S $\rightarrow$ NP VP	DT $\rightarrow$ the	NN $\rightarrow$ park
NP $\rightarrow$ DT NN	NN $\rightarrow$ man	VB $\rightarrow$ saw
NP $\rightarrow$ NP PP	NN $\rightarrow$ dog	IN $\rightarrow$ with
PP $\rightarrow$ IN NP	NN $\rightarrow$ cat	IN $\rightarrow$ under
VP $\rightarrow$ VB NP		

1. How many parse trees are there under this grammar for the sentence *the man saw the dog in the park* ?
2. How many parse trees are there under this grammar for the sentence *the man saw the dog in the park with the cat* ?
3. Consider a sentence that is grammatical under the above context-free grammar, and has exactly k prepositions following the verb, and 0 prepositions before the verb (a preposition is any word with the tag IN). How many parse trees will this sentence have? Reason why.

> *The n^{th} Catalan number is defined as $C_n = \frac{(2n)!}{(n+1)!n!}$ (see Wikipedia for a full description). It can be shown that $C_n$is the number possible different binary trees with $n+1$ leaves.*

Exercise 4.

Consider the following CFG:

S $\rightarrow$ NP VP	DT $\rightarrow$ the	NNS $\rightarrow$ cats
NP $\rightarrow$ DT NN	NN $\rightarrow$ man	NNS $\rightarrow$ parks
NP $\rightarrow$ DT NNS	NN $\rightarrow$ dog	VB $\rightarrow$ see
NP $\rightarrow$ NP PP	NN $\rightarrow$ cat	VB $\rightarrow$ sees
PP $\rightarrow$ IN NP	NN $\rightarrow$ park	IN $\rightarrow$ in
VP $\rightarrow$ VB NP	NNS $\rightarrow$ dogs	IN $\rightarrow$ with
VP $\rightarrow$ VP PP		

This grammar overgenerates incorrect Englihsh sentences, such as:

the dog see the cat

the dog in the park see the cat

the dog in the park see the cat in the park

the dogs sees the cat

the dogs in the park sees the cat

the dogs in the park sees the cat in the park

1. Modify the grammar so that all generated sentences respect third-person subject-verb agreement rules for English

Exercise 5.

Consider the following CFG:

S $\rightarrow$ NP VP	DT $\rightarrow$ the	VB $\rightarrow$ saw
NP $\rightarrow$ DT NN	NN $\rightarrow$ man	IN $\rightarrow$ with
PP $\rightarrow$ IN NP	NN $\rightarrow$ dog	IN $\rightarrow$ under
VP $\rightarrow$ VB NP	NN $\rightarrow$ telescope	
VP $\rightarrow$ VP PP		

An infinite number of sentences can be generated by this grammar, for example:

the man saw the dog

the man saw the dog with the telescope

the man saw the dog with the telescope under the dog

the man saw the dog under the telescope with the dog under the telescope

etc.

The language $\mathcal{L}(G)$ generated by a context-free grammar G is defined as the set of sentences that can be derived with a sequence of grammar rule applications.

A hidden Markov model (HMM), defines a distribution $P(x_1 \dots x_n, y_1 \dots y_n)$ over sentences $x_1 \dots x_n$ paired with PoS tag sequences $y_1 \dots y_n$.
The language generated by a HMM is defined as the set of sentences $x_1 \dots x_n$ such that: $\max_{y_1 \cdots y_n} P(x_1 \dots x_n, y_1 \dots y_n) > 0$, that is, sentences with at least one possible PoS-tag sequence $y_1 \dots y_n$ that gives a non-zero value for the probability $P(x_1 \dots x_n, y_1 \dots y_n)$.

1. Write a bigram HMM that generates the same language than the context-free grammar given above.

Exercise 6.

Consider the following CFG:

S $\rightarrow$ NP VP	WH $\rightarrow$ that
VP $\rightarrow$ Vt NP	DT $\rightarrow$ the
VP $\rightarrow$ Vdt NP NP	NN $\rightarrow$ man
NP $\rightarrow$ DT NN	NN $\rightarrow$ dog
NP $\rightarrow$ NP RELC	NN $\rightarrow$ cat
RELC $\rightarrow$ WH SGAP	NN $\rightarrow$ park
SGAP $\rightarrow$ VP	Vt $\rightarrow$ saw
SGAP $\rightarrow$ NP VGAP	Vdt $\rightarrow$ gave
VGAP $\rightarrow$ Vt	
VGAP $\rightarrow$ Vdt NP	

1. Draw parse trees for the sentences:

the man that saw the dog saw the cat
the man that the cat saw saw the dog

2. Write a sentence that is grammatical under the above grammar, and contains the trigram: *saw saw saw*. Draw the parse tree for the sentence.

3. Assume that we add the following rules to the grammar, so that the sentence *the man said the cat saw the dog* can be parsed correctly:

$$\text{VP} \rightarrow \text{V3 S}$$
$$\text{V3} \rightarrow \text{said}$$

What additional rules should be added to the grammar so that the sentence *the dog that the man said the cat saw saw the park* can be parsed?

Probabilistic Context Free Grammars

Exercise 7.

Using the following PCFG in CNF:

S → NP VP	1.0	P → with	1.0
NP → NP PP	0.4	V → saw	1.0
PP → P NP	1.0	NP → astronomers	0.1
VP → V NP	0.7	NP → ears	0.18
VP → VP PP	0.3	NP → saw	0.04
		NP → stars	0.18
		NP → telescopes	0.1

Work with the sentence: *astronomers saw stars with ears*

- How many correct parses are there for this sentence?
- Write them, along with their probabilities.

Exercise 8.

Given the following PCFG:

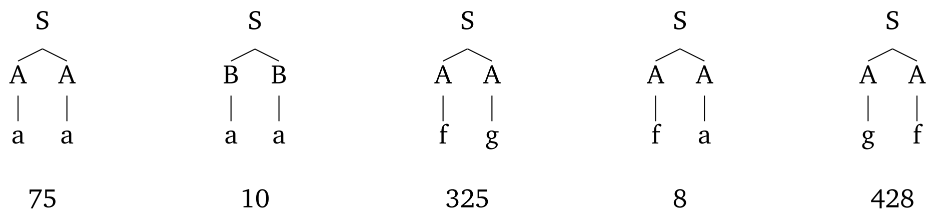
S → NP VP	1.0	N → time	0.4
NP → N N	0.25	N → flies	0.2
NP → D N	0.4	N → arrow	0.4
NP → N	0.35	D → an	1.0
VP → V NP	0.6	ADV → like	1.0
VP → V ADVP	0.4	V → flies	0.5
ADVP → ADV NP	1.0	V → like	0.5

and the sentence *time flies like an arrow*

1. Write two parse trees that this grammar generates for this sentence
2. Compute the probability of each tree.
3. Convert the grammar to CNF and emulate the behaviour of the CKY algorithm on this sentence. Provide the final chart.
4. Emulate the behaviour of the Earley algorithm on this sentence (ignoring rule probabilities). Provide the final chart.

Exercise 9.

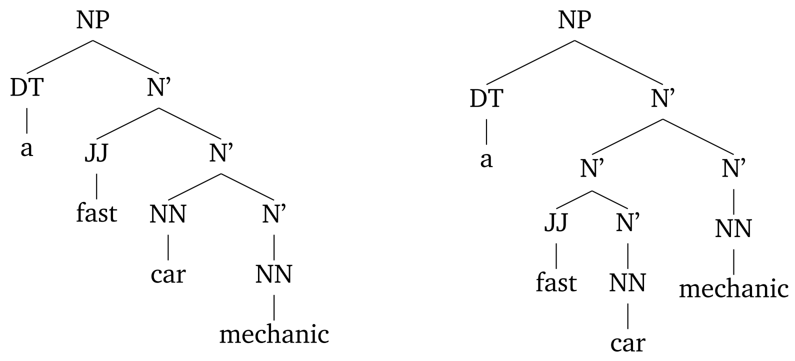
Consider that you have as a training corpus a treebank containing the following trees. Each tree was observed the number of times indicated below it.



1. What PCFG would one get from this treebank (using MLE)?
2. Given the obtained grammar:
   - What is the most likely parse of the string *a a*?
   - Is this a reasonable result? Discuss why.

Exercise 10.

Consider the two following parse trees:



Discuss whether the following statements are true or false and why:

1. The two parse trees receive the same probability under any PCFG
2. The first parse tree receives higher probability if $P(\text{N}' \rightarrow \text{NN N}') > P(\text{N}' \rightarrow \text{N}' \text{ N}')$
3. The first parse tree receives higher probability if $P(\text{N}' \rightarrow \text{NN N}') > P(\text{N}' \rightarrow \text{N}' \text{ N}') + P(\text{N}' \rightarrow \text{NN})$

Exercise 11.

Consider the following PCFG:

$\text{S} \rightarrow \text{V N}$	0.6
$\text{S} \rightarrow \text{D N}$	0.4
$\text{D} \rightarrow \text{a}$	0.2
$\text{D} \rightarrow \text{the}$	0.8
$\text{N} \rightarrow \text{president}$	1.0
$\text{V} \rightarrow \text{support}$	0.6
$\text{V} \rightarrow \text{hate}$	0.4

1. List all sentences generated by this grammar, along with their probability
2. Define a bigram language model that gives the same probability distribution $p(x)$ than the PCFG shown above. The vocabulary of the language model should be $\Sigma = \{\text{a}, \text{the}, \text{president}, \text{support}, \text{hate}\}$. Specify the value for each parameter of the language model.

> *A bigram language model consists of a finite vocabulary Σ, and a parameter $q(u, v)$ for each bigram (u, v) such that $u \in \Sigma \cup \{\text{START}\}$ and $v \in \Sigma \cup \{\text{STOP}\}$*
> *The value for $q(u, v)$ can be interpreted as the probability of seeing word v immediately after word u, i.e. $P(v|u)$*
> *For any sentence $x_1, \ldots, x_n$ where $x_i \in \Sigma$, the probability of the sentence under the bigram language model is $p(x_1, \ldots, x_n) = \prod_{i=1}^{n+1} q(x_{i-1}, x_i)$, where we define $x_0 =$ START and $x_{n+1} =$ STOP.*

Exercise 12.

Consider the following PCFG

$\text{S} \rightarrow \text{NP VP}$	1.0	$\text{VP} \rightarrow \text{sleeps}$	1.0
$\text{NP} \rightarrow \text{DT NBAR}$	1.0	$\text{DT} \rightarrow \text{the}$	1.0
$\text{NBAR} \rightarrow \text{NN}$	0.7	$\text{NN} \rightarrow \text{mechanic}$	0.1
$\text{NBAR} \rightarrow \text{NBAR NBAR}$	0.3	$\text{NN} \rightarrow \text{car}$	0.2
		$\text{NN} \rightarrow \text{metal}$	0.7

1. What is the parse tree with highest probability for the sentence *the metal car mechanic sleeps* ?

2. Modify the grammar above so that the sentece *the human language technology rules* has two interpretations (one about *human language* and another about *human technology*). Draw the trees for both interpretations, and point out which is the most likely.

Exercise 13.

This exercise considers several forms of language models that compute the probability of sentences $P(\mathbf{x})$. In each of the following cases you need to write an expression that indicates how the particular language model computes the probability $P(\mathbf{x})$, making clear what parameters of the model are used to compute the probability for the example sentence.

1. **n-gram language models.** The model considers only the words of $\mathbf{x}$, the rest of the linguistic structure is ignored. Write the expression for $n = 2$ and $n = 3$.

2. **Hidden Markov Models (HMM).** The model represents pos tags in the state sequence and words in the observation sequence. The syntactic tree is ignored. Write the expression of $P(\mathbf{x})$ for a bigram HMM, where states correspond to single PoS tags, and for a trigram HMM, where states correspond to two adjacent pos tags.

3. **Probabilistic Context-Free Grammars (PCFG).** The model considers the full syntactic tree.