University of Oslo : Department of Informatics

INF4820: Algorithms for Artificial Intelligence and Natural Language Processing

Live Coding, Parser Evaluation, Quiz

Stephan Oepen & Erik Velldal

Language Technology Group (LTG)

November 11, 2015

Overview

Last Time

- Generalized Chart Parsing
- Inside the Parse Forest
- Viterbi Tree Decoding

Today

- Exhaustive Unpacking
- Viterbi Tree Decoding
- Parser Evaluation
- Wrap-Up Quiz

Chart Parsing: Key Ideas

- The parse *chart* is a two-dimensional matrix of *edges* (aka chart items);
- an edge is a (possibly partial) rule instantiation over a substring of input;
- the chart indexes edges by start and end string position (aka vertices);
- dot in rule RHS indicates degree of completion: $\alpha \rightarrow \beta_1 \ldots \beta_{i-1} \bullet \beta_i \ldots \beta_n$;
- *active* edges (aka *incomplete* items) — partial RHS: $[1, 2, \text{VP} \rightarrow \text{V} \bullet \text{NP}]$;
- *passive* edges (aka *complete* items) — full RHS: $[1, 3, \text{VP} \rightarrow \text{V NP} \bullet]$;

The Fundamental Rule

$$[i,\ j,\ \alpha \rightarrow \beta_1 \ldots \beta_{i-1} \bullet \beta_i \ldots \beta_n] + [j,\ k,\ \beta_i \rightarrow \gamma^+ \bullet]$$
$$\mapsto [i,\ k,\ \alpha \rightarrow \beta_1 \ldots \beta_i \bullet \beta_{i+1} \ldots \beta_n]$$

Ambiguity Packing in the Chart

General Idea

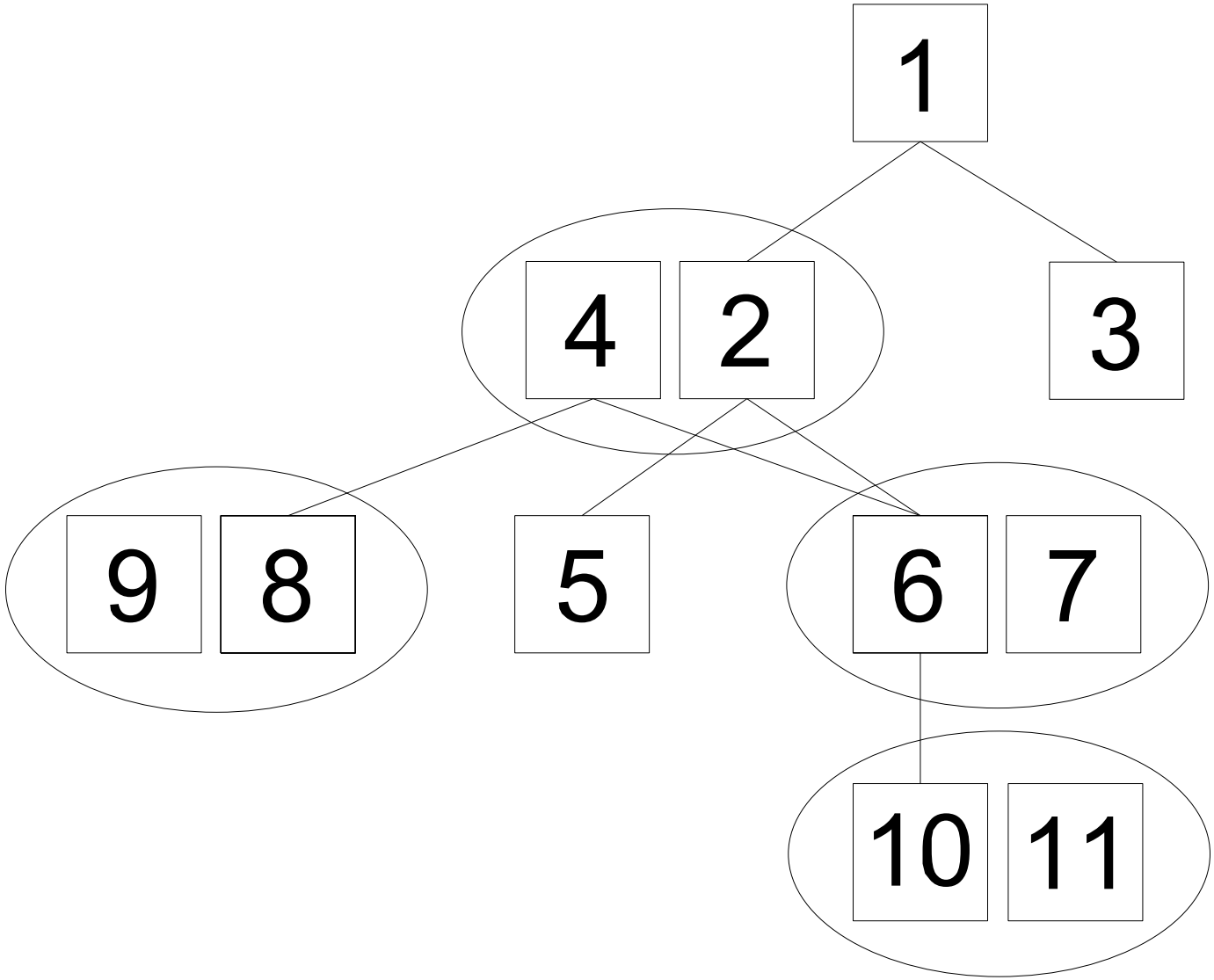
- Maintain only one edge for each α from i to j (the 'representative');
- record alternate sequences of daughters for α in the representative.

Implementation

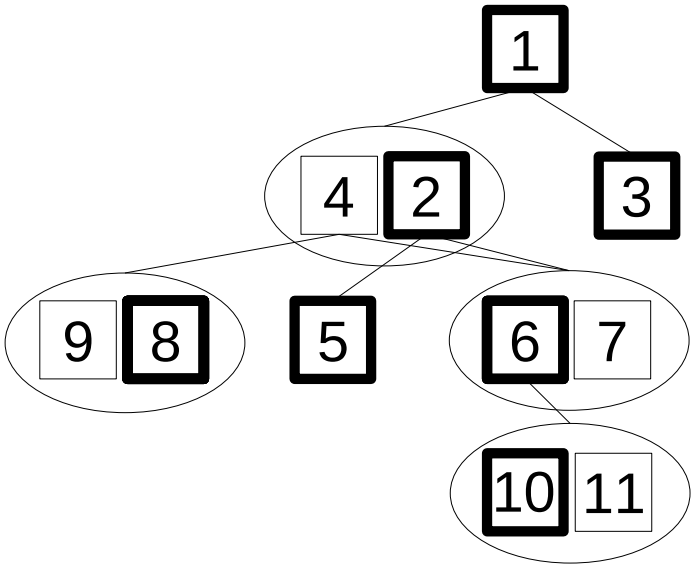
- Group passive edges into *equivalence classes* by identity of α, i, and j;
- search chart for existing equivalent edge (h, say) for each new edge e;
- when h (the 'host' edge) exists, *pack* e into h to record equivalence;
- e *not* added to the chart, no derivations with or further processing of e;

$\rightarrow$ *unpacking* multiply out all alternative daughters for all result edges.

An Example (Hypothetical) Parse Forest



Unpacking: Cross-Multiplying Local Ambiguity



$$
\begin{aligned}
1 &\rightarrow \langle 2\ 3 \rangle \mid \langle 4\ 3 \rangle \\
2 &\rightarrow \langle 5\ 6 \rangle \mid \langle 5\ 7 \rangle \\
4 &\rightarrow \langle 8\ 6 \rangle \mid \langle 8\ 7 \rangle \mid \langle 9\ 6 \rangle \mid \langle 9\ 7 \rangle \\
6 &\rightarrow \langle 10 \rangle \mid \langle 11 \rangle
\end{aligned}
$$

How many complete trees in total?

Live Coding: Exhaustive Unpacking

Probability Theory and Natural Language?

The most important questions of life are, for the most part, really only questions of probability. (Pierre-Simon Laplace, 1812)

Special wards in lunatic asylums could well be populated with mathematicians who have attempted to predict random events from finite data samples. (Richard A. Epstein, 1977)

But it must be recognized that the notion 'probability' of a sentence is an entirely useless one, under any known interpretation of this term. (Noam Chomsky, 1969)

Every time I fire a linguist, system performance improves. (Fredrick Jelinek, 1980s)

Viterbi Decoding over the Parse Forest

- Recall the Viterbi algorithm for HMMs

$$v_i(x) = \max_{k=1}^{L} [v_{i-1}(k) \cdot P(x|k) \cdot P(o_i|x)]$$

- Over the (complete, result edges from the) parse forest, compute Viterbi scores for sub-trees of increasing size:

$$v(e) = \max \left[P(\beta_1, \ldots \beta_n|\alpha) \times \prod_i v(\beta_i) \right]$$

- Similar to HMM decoding, we also need to keep track of the set of daughters that led to the maximum probability.

Parser Evaluation

There are a number of aspects to consider in judging parser performance:

- **Coverage** the percentage of inputs for which we we found an analysis.
- **Overgeneration** the percentage of *ungrammatical* inputs (incorrectly) assigned an analysis.
- **Efficiency** time and memory used by the parser.
- **Accuracy** Sentence accuracy measures the percentage of input sentences which received the correct tree.

  Since full trees can be quite complex, this is a very strict metric, and so most statistical parsers report accuracy according to the granular ParsEval metric.

ParsEval

- The ParsEval metric (Black, et al., 1991) measures constituent overlap.
- The original formulation only considered the shape of the (unlabeled) bracketing.
- The modern 'standard' uses a tool called `evalb`, which reports precision, recall and F_1 score for labeled brackets, as well as the number of crossing brackets.

ParsEval

Gold Standard

(NP (DT *a*)
(ADVP (RB *pretty*)
(JJ *big*))
(NOM (NN *dog*)
(POS *'s*)
(NN *house*)))

0,6	NP	1,2	RB	3,4	NN
0,1	DT	2,3	JJ	4,5	POS
1,3	ADVP	3,6	NOM	5,6	NN

System Output

(NP (DT *a*)
(JJ *pretty*)
(NOM (JJ *big*)
(NOM (NN *dog*)
(POS *'s*)
(NN *house*))))

0,6	NP	2,6	NOM	3,4	NN
0,1	DT	2,3	JJ	4,5	POS
1,2	JJ	3,6	NOM	5,6	NN

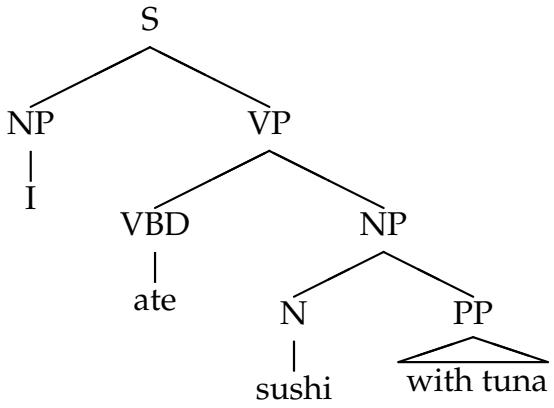
Recall: $\frac{Correct}{Gold} = \frac{7}{9}$ Precision: $\frac{Correct}{System} = \frac{7}{9}$ F_1 score: $\frac{7}{9}$

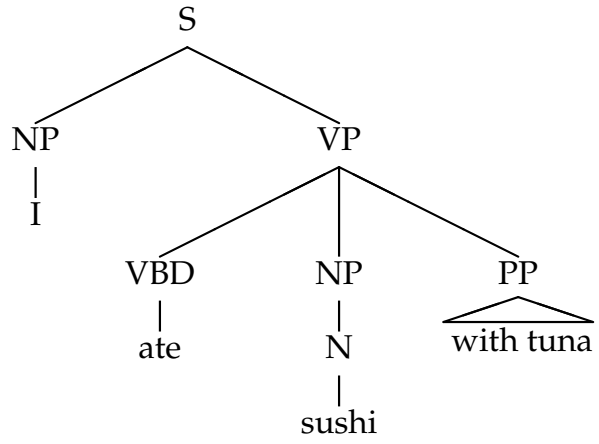
Crossing Brackets: 1

In Conclusion

In the second half of the class, we set out to determine:

- which string is most likely: ✓
  - *How to recognise speech* vs. *How to wreck a nice beach*
- which tag sequence is most likely for *flies like flowers*: ✓
  - **NNS VB NNS** vs. **VBZ P NNS**
- which syntactic analysis is most likely: ✓





Finally: Need Some Bonus Points?

Rules of the Game

- Up to four bonus points towards completion of Obligatory Exercise (3).
- Get one post-it; at the top, write down your first and last name.
- Further, write down your UiO account name (e.g. `oe`, in my case).
- Write each answer on a line of its own, prefix by question number.
- Do not consult with your neighbors; they might just mess things up.

After the Quiz

- Post your answers at the front of your table, we will collect all notes.
- Discuss your answers with your neighbor(s); find out who is right.

Question (1): Natural Language Ambiguity

Assume the following 'toy' grammar of English:

$$S \rightarrow NP$$
$$NP \rightarrow Det\ N$$
$$N \rightarrow N\ N$$

$$Det \rightarrow \textit{the}$$
$$N \rightarrow \textit{kitchen} \mid \textit{gold} \mid \textit{towel} \mid \textit{rack}$$

(1) How many different syntactic analyses, if any, does the grammar assign to the following strings?

(a) *the kitchen towel rack*
(b) *the kitchen gold towel rack*

Question (2): CKY Parsing

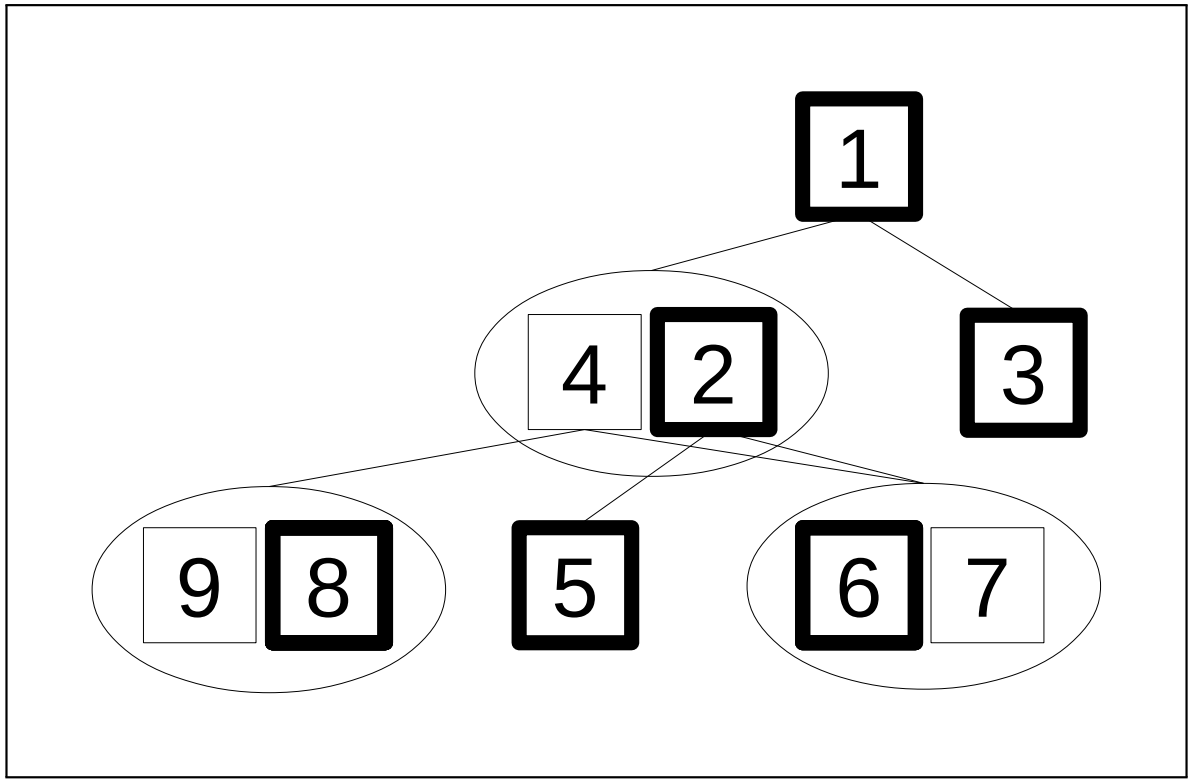
Assume the following grammar and CKY parse table:

$S \rightarrow NP\ VP$
$VP \rightarrow V\ NP$
$VP \rightarrow VP\ PP$
$NP \rightarrow NP\ VP$
$PP \rightarrow P\ NP$

	1	2	3	4	5
0	NP		S		S
1		V	VP		VP
2			NP		NP
3				P	PP
4					NP

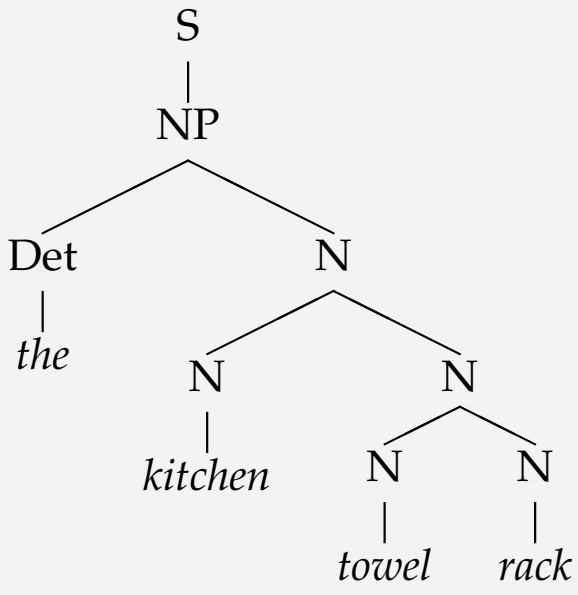
(2) Which pair(s) of 'input' cells and which production(s) give rise to the derivation of category S in 'target' cell $\langle 0, 5 \rangle$?

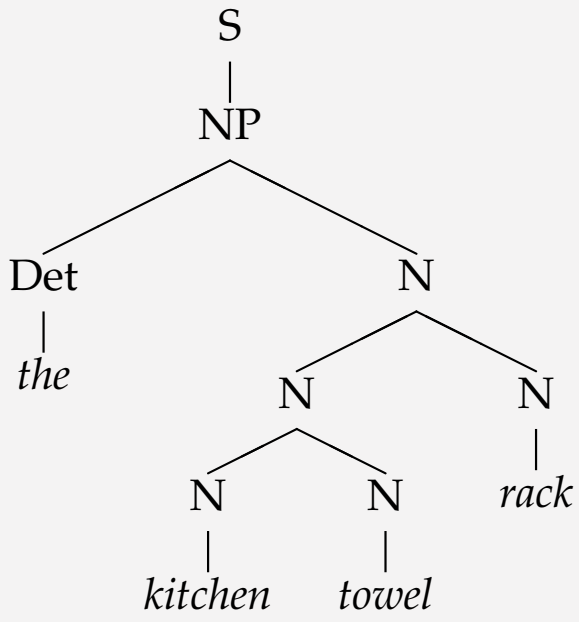
Question (3): Packed Parse Forests



(3) How many complete trees are represented in this forest?

Question (4): Parser Evaluation





(4) What are the ParsEval precision and recall scores for this pair of trees (gold on the left; system on the right)?