Combining Labeled and Unlabeled Data in Statistical Natural Language Parsing

Simon Fraser University – April 18, 2002

Anoop Sarkar
Department of Computer and Information Science
University of Pennsylvania

anoop@linc.cis.upenn.edu
http://www.cis.upenn.edu/~anoop

- Task: find the most likely parse for natural language sentences

- Approach: rank alternative parses with statistical methods trained on data annotated by experts (labeled data)

- Focus of this talk:

  1. Motivate a particular probabilistic grammar formalism for statistical parsing: tree-adjoining grammar

  2. Combine labeled data with unlabeled data to improve performance in parsing using co-training

Overview

- Introduction to Statistical Parsing
- Tree Adjoining Grammars and Statistical Parsing
- Combining Labeled and Unlabeled Data in Statistical Parsing
- Summary and Future Directions

Applications of Language Processing Algorithms

- Information Extraction: converting unstructured data (text) into a structured form
- Improving the word error rate in speech recognition
- Human-Computer Interaction: dialog systems, machine translation, summarization, etc.
- Cognitive Science: computational models of human linguistic behaviour
- Biological structure prediction: formal grammars for RNA secondary structures

A Key Problem in Processing Language: Ambiguity

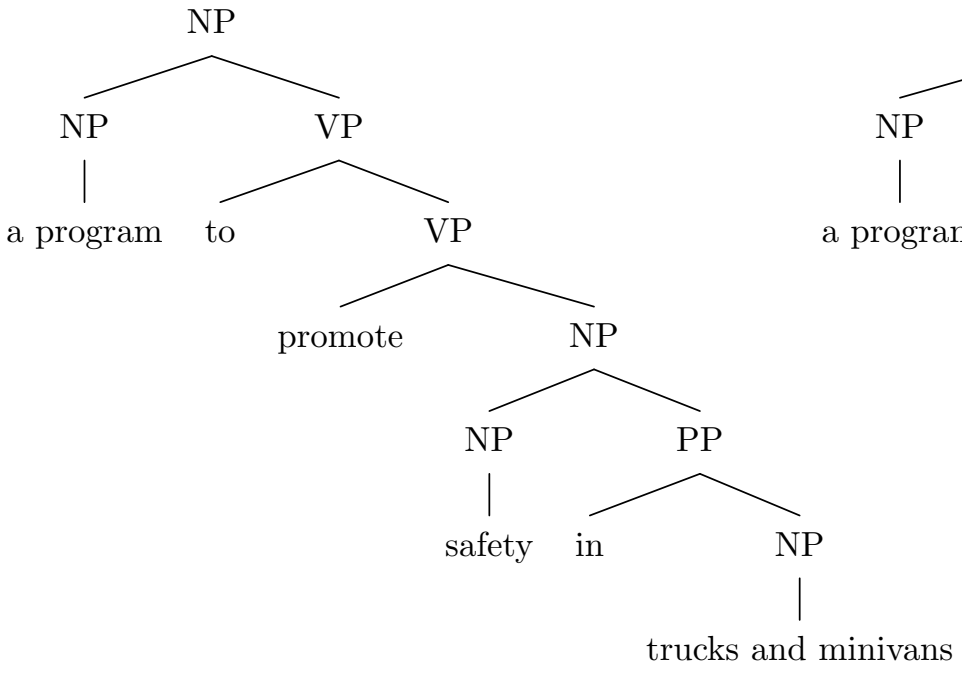
(Church and Patil 1982; Collins 1999)

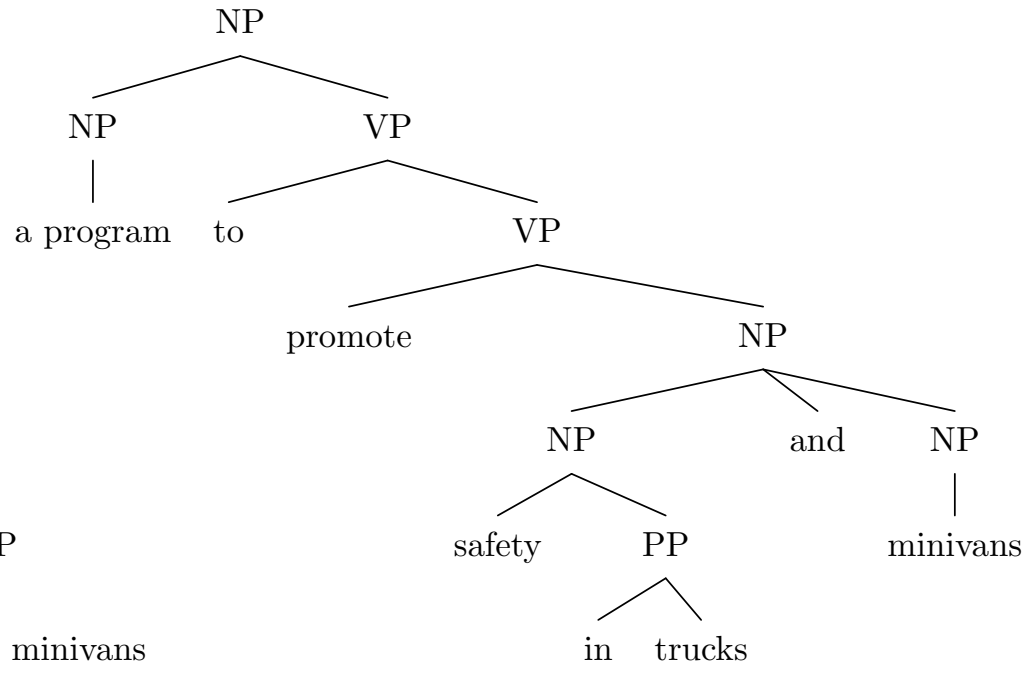
- Part of Speech ambiguity
  `saw` → noun
  `saw` → verb

- Structural ambiguity: Prepositional Phrases
  `I saw (the man) with the telescope`
  `I saw (the man with the telescope)`

- Structural ambiguity: Coordination
  `a program to promote safety in ((trucks) and (minivans))`
  `a program to promote ((safety in trucks) and (minivans))`
  `((a program to promote safety in trucks) and (minivans))`

Ambiguity ← attachment choice in alternative parses





Parsing as a machine learning problem

- S = a sentence
  T = a parse tree
  A statistical parsing model defines $P(T \mid S)$

- Find best parse: $\arg\max_T P(T \mid S)$

- $P(T \mid S) = \frac{P(T,S)}{P(S)} = P(T, S)$

- Best parse: $\arg\max_T P(T, S)$

- e.g. for PCFGs: $P(T, S) = \prod_{i=1\ldots n} P(\mathrm{RHS}_i \mid \mathrm{LHS}_i)$

Parsing as a machine learning problem

- Training data: the Penn WSJ Treebank (Marcus et al. 1993)
- Learn probabilistic grammar from training data
- Evaluate accuracy on test data
- A standard evaluation:
  Train on 40,000 sentences
  Test on 2,300 sentences
- The simplest technique: PCFGs perform badly
  Reason: not sensitive to the words

Machine Learning for ambiguity resolution: prepositional phrases

V	N1	P	N2	Attachment
making	paper	for	filters	N
join	board	as	director	V
is	chairman	of	N.V.	N
using	crocidolite	in	filters	V
bring	attention	to	problem	V
is	asbestos	in	products	N
including	three	with	cancer	N

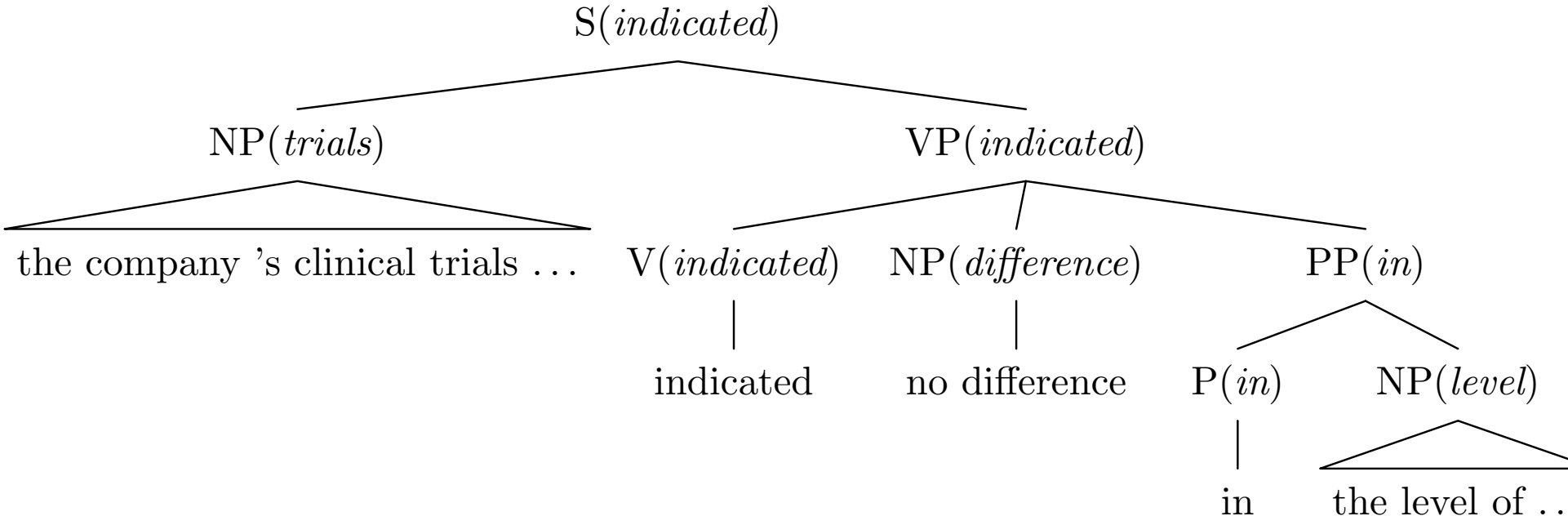
↑
Supervised learning

Machine Learning for ambiguity resolution: prepositional phrases

Method	Accuracy
Always noun attachment	59.0
Most likely for each preposition	72.2
Average Human (4 head words only)	88.2
Average Human (whole sentence)	93.2
Lexicalized Model (Collins and Brooks 1995)	84.0
Lexicalized Model + Wordnet (Stetina and Nagao 1998)	88.0

Statistical Parsing:

the company 's clinical trials of both its animal and human-based insulins indicated no difference in the level of hypoglycemia between users of either product



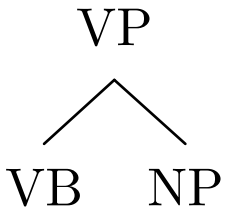
Bilexical CFG: dependencies between pairs of words

- Full context-free rule:
  VP(*indicated*) → V-hd(*indicated*) NP(*difference*) PP(*in*)

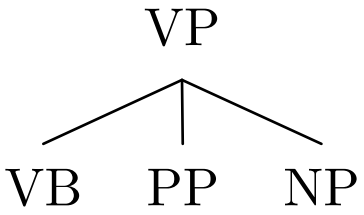
- Each rule is generated in three steps (Collins 1999):
  1. Generate head daughter of LHS: VP(*indicated*) → V-hd(*indicated*)
  2. Generate non-terminals to *left* of head daughter: STOP . . . V-hd(*indicated*)
  3. Generate non-terminals to *right* of head daughter:
     - V-hd(*indicated*) . . . NP(*difference*)
     - V-hd(*indicated*) . . . PP(*in*)
     - V-hd(*indicated*) . . . STOP

Independence Assumptions

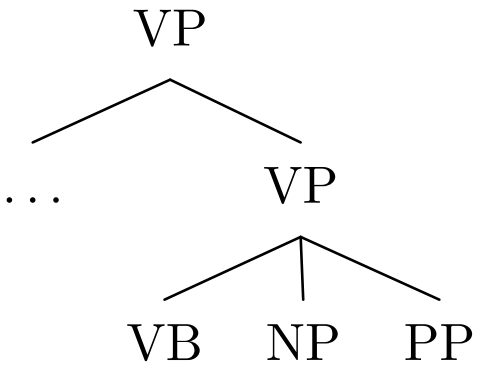
60.8%



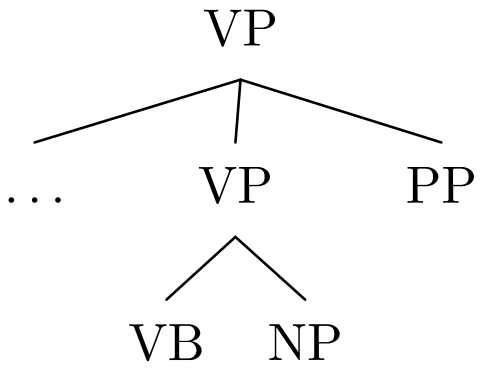
0.7%



2.23%



0.06%



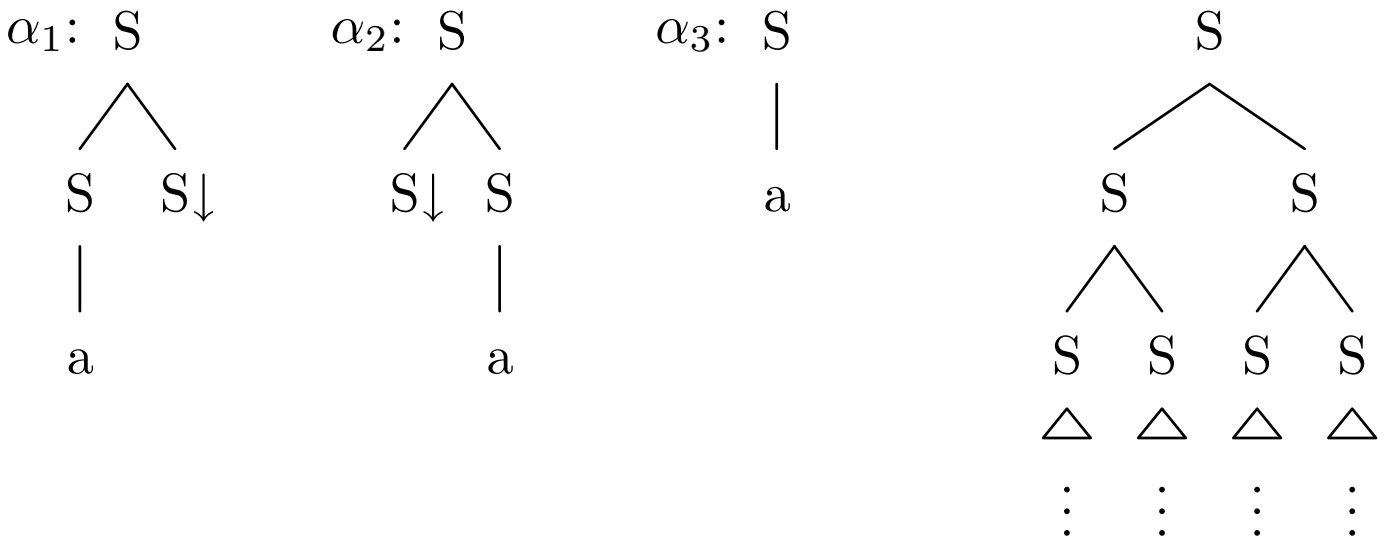
Overview

- Introduction to Statistical Parsing
- Tree Adjoining Grammars and Statistical Parsing
- Combining Labeled and Unlabeled Data in Statistical Parsing
- Summary and Future Directions

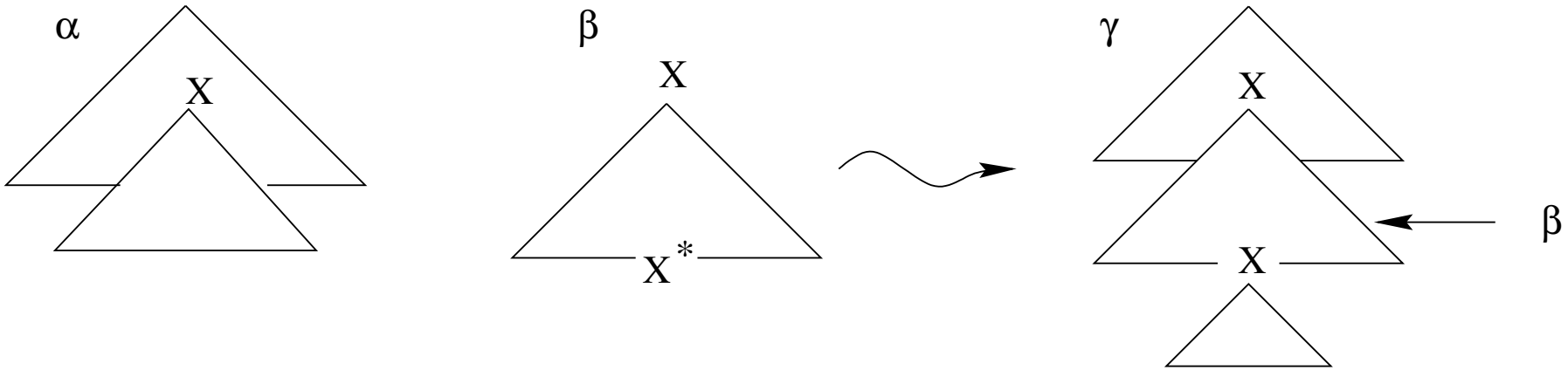
Lexicalization of Context-Free Grammars

- CFG G: $(r_1)\ S \rightarrow S\ S \quad (r_2)\ S \rightarrow a$

- Tree-substitution Grammar G':



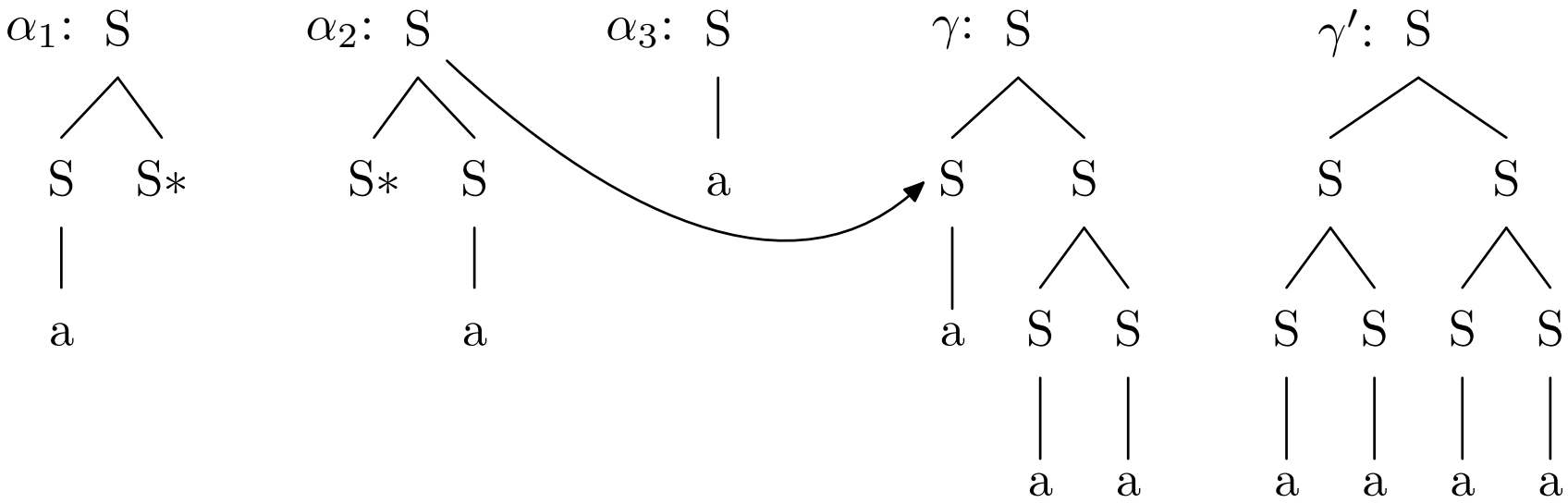
Lexicalization of Context-Free Grammars



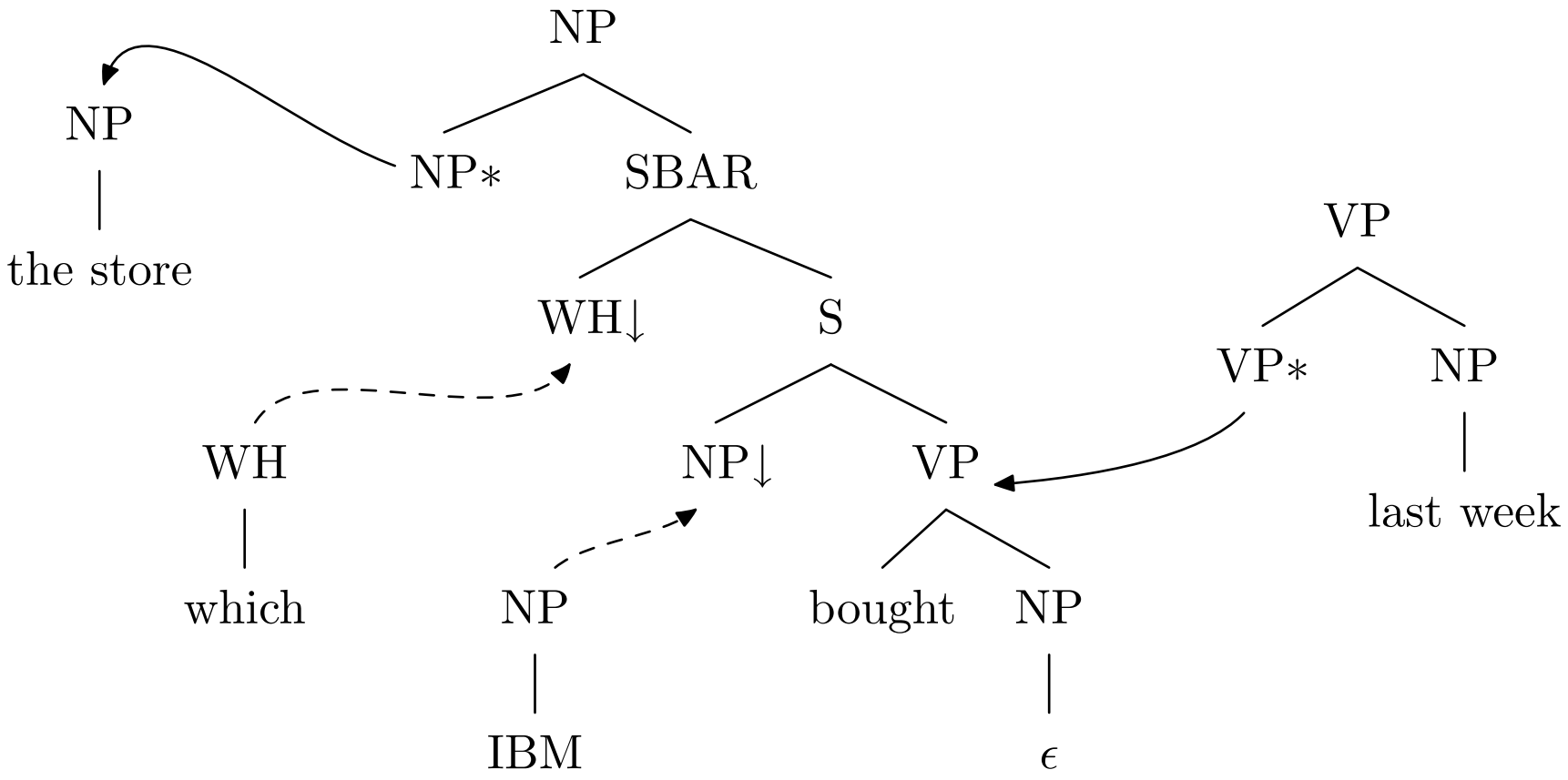
Lexicalization of Context-Free Grammars

- CFG G: $(r_1)\ S \rightarrow S\ S \quad (r_2)\ S \rightarrow a$

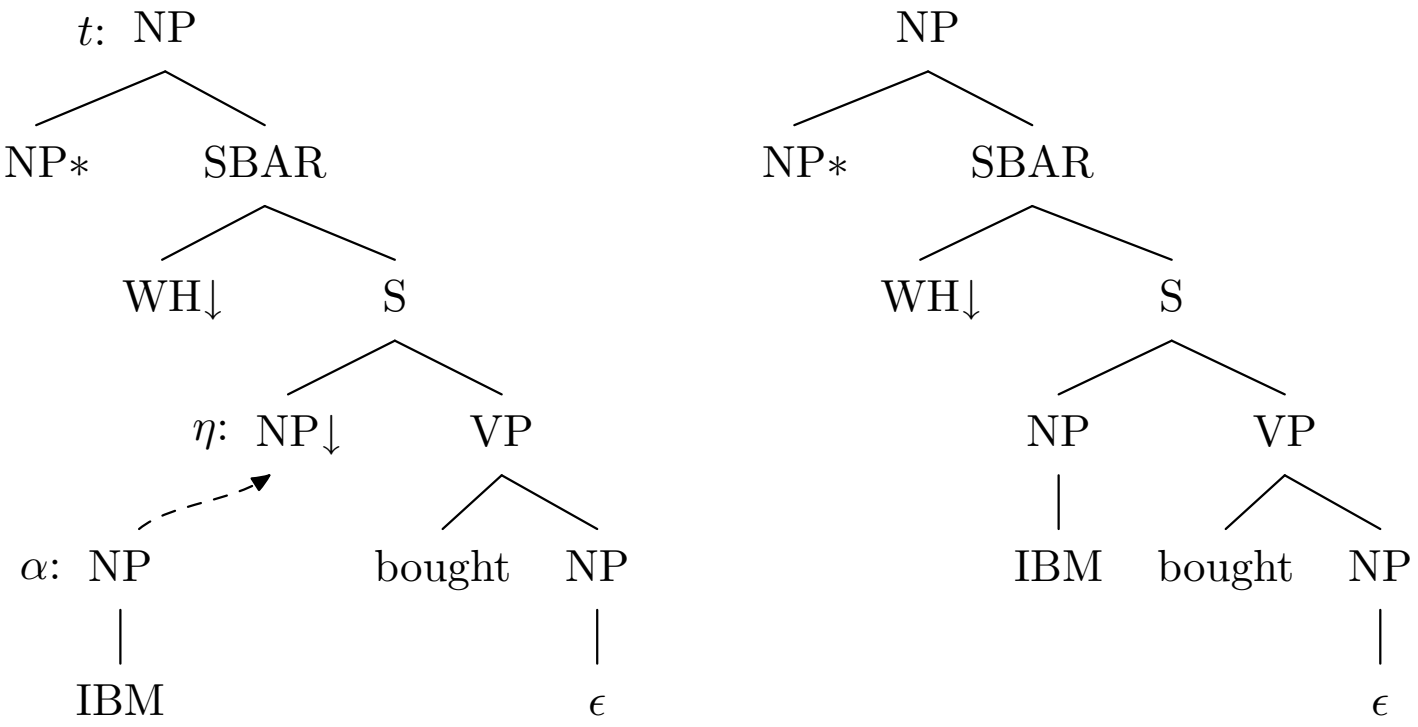
- Tree-adjoining Grammar G'':



Tree Adjoining Grammars: Different Modeling of Bilexical Dependencies

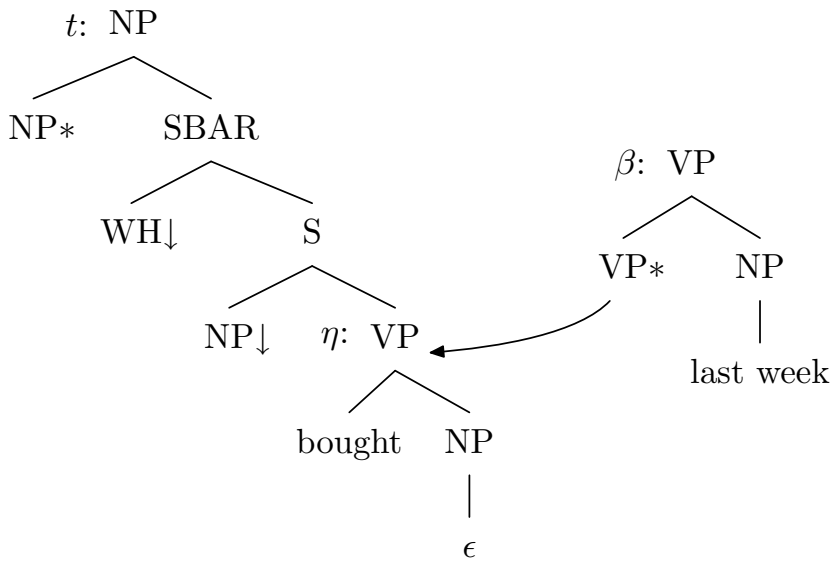


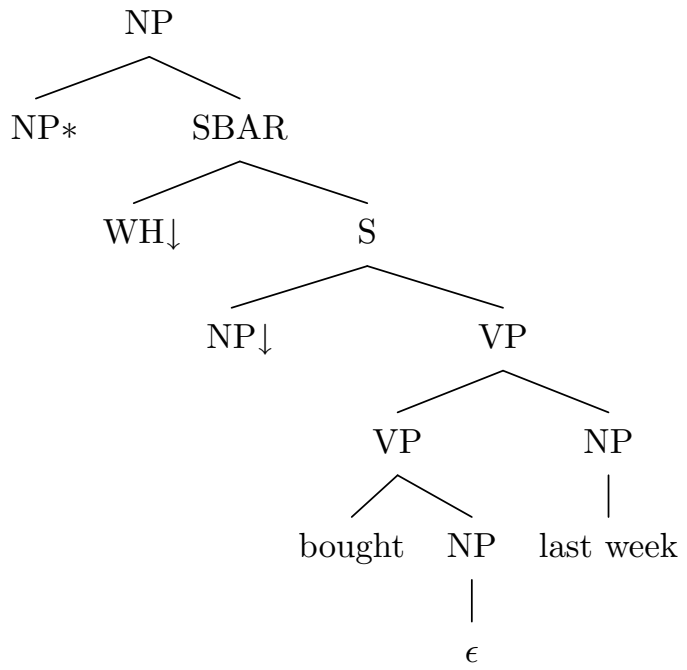
Probabilistic TAGs: Substitution



$$\sum_{\alpha} P_s(t, \eta \rightarrow \alpha) = 1$$

Probabilistic TAGs: Adjunction





$$P_a(t, \eta \rightarrow \text{NA}) + \sum_{\beta} P_a(t, \eta \rightarrow \beta) = 1$$

Tree Adjoining Grammars

- Start of a derivation: $\sum_{\alpha} P_i(\alpha) = 1$

- Probability of a derivation:

$$Pr(\mathcal{D}, w_0 \ldots w_n) =$$
$$P_i(\alpha, w_i) \times \prod_{p} P_s(\tau, \eta, w \rightarrow \alpha, w') \times$$
$$\prod_{q} P_a(\tau, \eta, w \rightarrow \beta, w') \times \prod_{r} P_a(\tau, \eta, w \rightarrow \text{NA})$$

- Events for these probability models can be extracted from an expert-annotated set of derivations (e.g. Penn Treebank)

Performance of supervised statistical parsers

System	$\leq 40 wds$ LP	$\leq 40 wds$ LR	$\leq 100 wds$ LP	$\leq 100 wds$ LR
(Magerman 95)	84.9	84.6	84.3	84.0
(Collins 99)	88.5	88.7	88.1	88.3
(Charniak 97)	87.5	87.4	86.7	86.6
(Ratnaparkhi 97)			86.3	87.5
Current	86.0	85.2		
(Chiang 2000)	87.7	87.7	86.9	87.0

- Labeled Precision $= \dfrac{\text{number of correct constituents in proposed parse}}{\text{number of constituents in proposed parse}}$

- Labeled Recall $= \dfrac{\text{number of correct constituents in proposed parse}}{\text{number of constituents in treebank parse}}$

Theory of Probabilistic TAGs

PCFGs: (Booth and Thompson 1973); (Jelinek and Lafferty 1991)

- A probabilistic grammar is well-defined or consistent if:

$$\sum_{n=1}^{\infty} \sum_{a_1 a_2 \ldots a_n \in \mathcal{V}} P(s \rightarrow a_1 a_2 \ldots a_n) = 1$$

- What is the single most likely parse (or derivation) for input string $a_1, \ldots, a_n$?

- What is the probability of $a_1, \ldots, a_i$, where $a_1, \ldots, a_i$ is a prefix of some string generated by the grammar? $\sum_{w \in \Sigma^*} P(a_1, \ldots, a_i w)$

Tree Adjoining Grammars

- Locality and independence assumptions are captured elegantly with a simple and well-defined probability model.

- Parsing can be treated in two steps:

  1. Classification: structured labels (elementary trees) are assigned to each word in the sentence.

  2. Attachment: the elementary trees are connected to each other to form the parse.

- Produces more than just the phrase structure of each sentence. It directly gives the predicate-argument structure.

Overview

- Introduction to Statistical Parsing
- Tree Adjoining Grammars and Statistical Parsing
- Combining Labeled and Unlabeled Data in Statistical Parsing
- Summary and Future Directions

Training a Statistical Parser

- How should the rule probabilities be chosen?

- Alternatives:
  - EM algorithm: completely unsupervised (Schabes 1992)
  - Supervised training from a Treebank (Chiang 2000)
  - Weakly supervised learning:
    exploit new representation to combine labeled and unlabeled data

Co-Training

- Pick two “views” of a classification problem.
- Build separate models for each of these “views” and train each model on a small set of labeled data.
- Sample an unlabeled data set and to find examples that each model independently labels with high confidence.
- Pick confidently labeled examples and add to labeled data. Iterate.
- Each model labels examples for the other in each iteration.

Co-training for simple classifiers (Blum and Mitchell 1998)

- Task: Build a classifier that categorizes web pages into two classes, +: *is a course web page*, −: *is not a course web page*

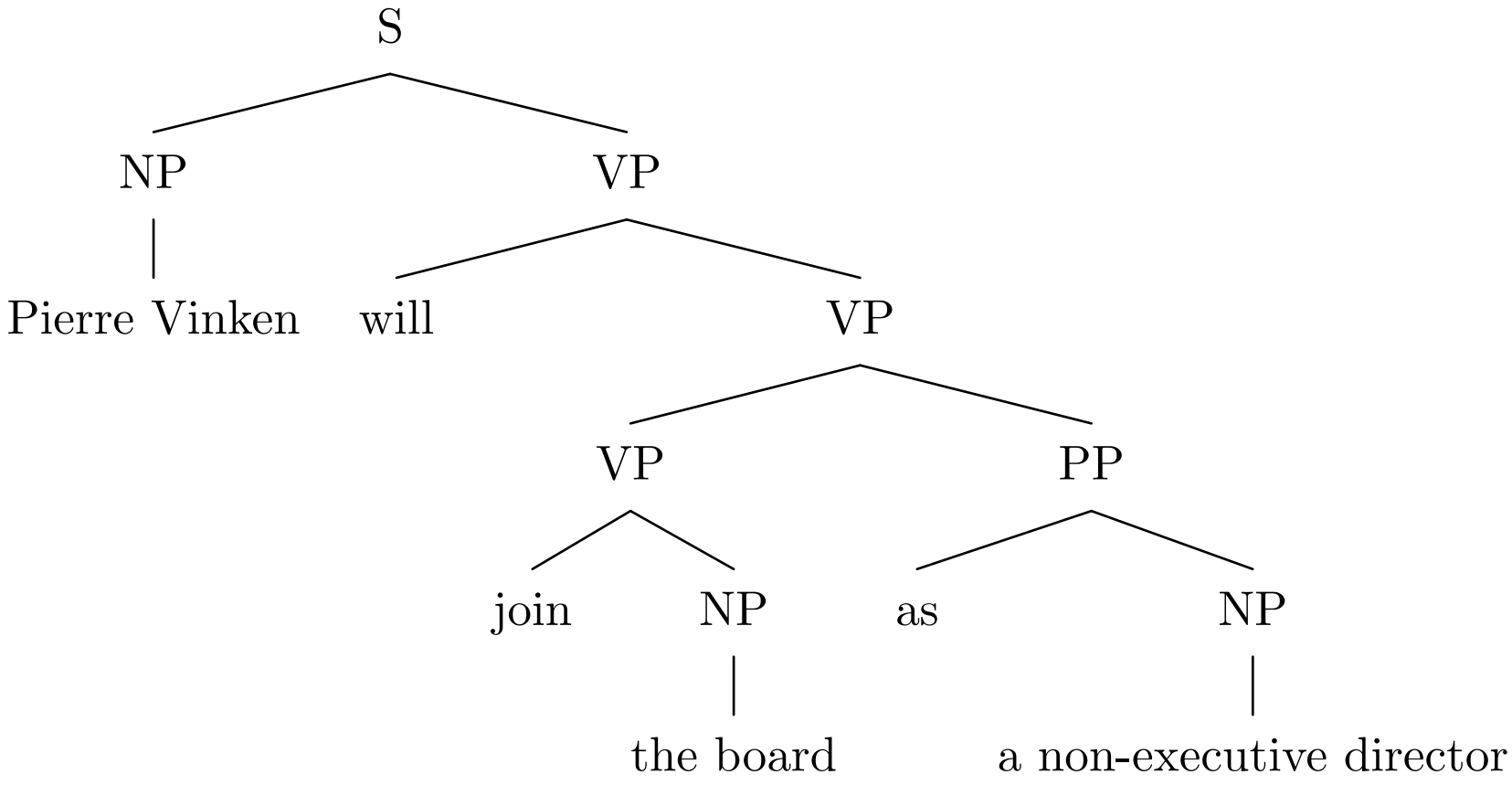
- Each labeled example has two views:

  1. Text in hyperlink: `<a href="...">` CSE 120, Fall semester `</a>`

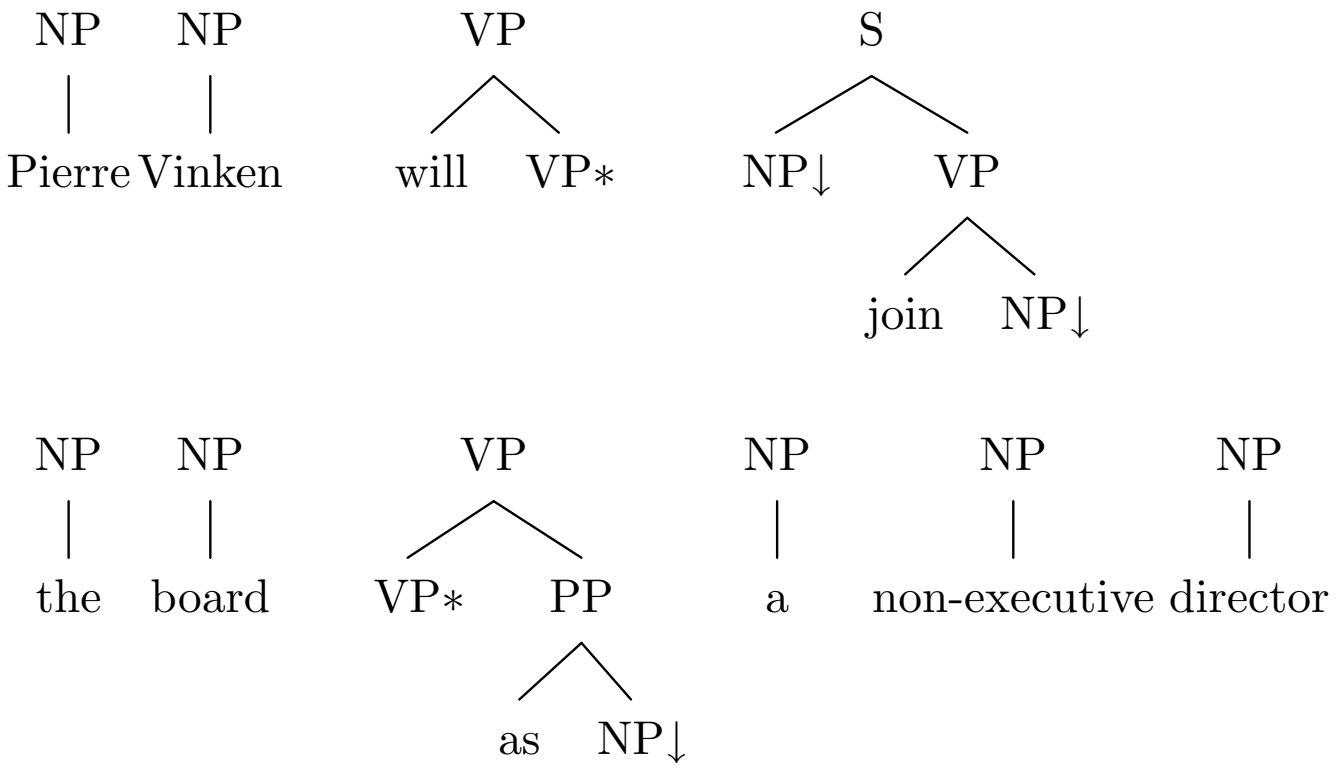
  2. Text in web page: `<html>...` Assignment #1 `...</html>`

- Combining labeled and unlabeled data outperforms only using labeled data

Pierre Vinken will join the board as a non-executive director

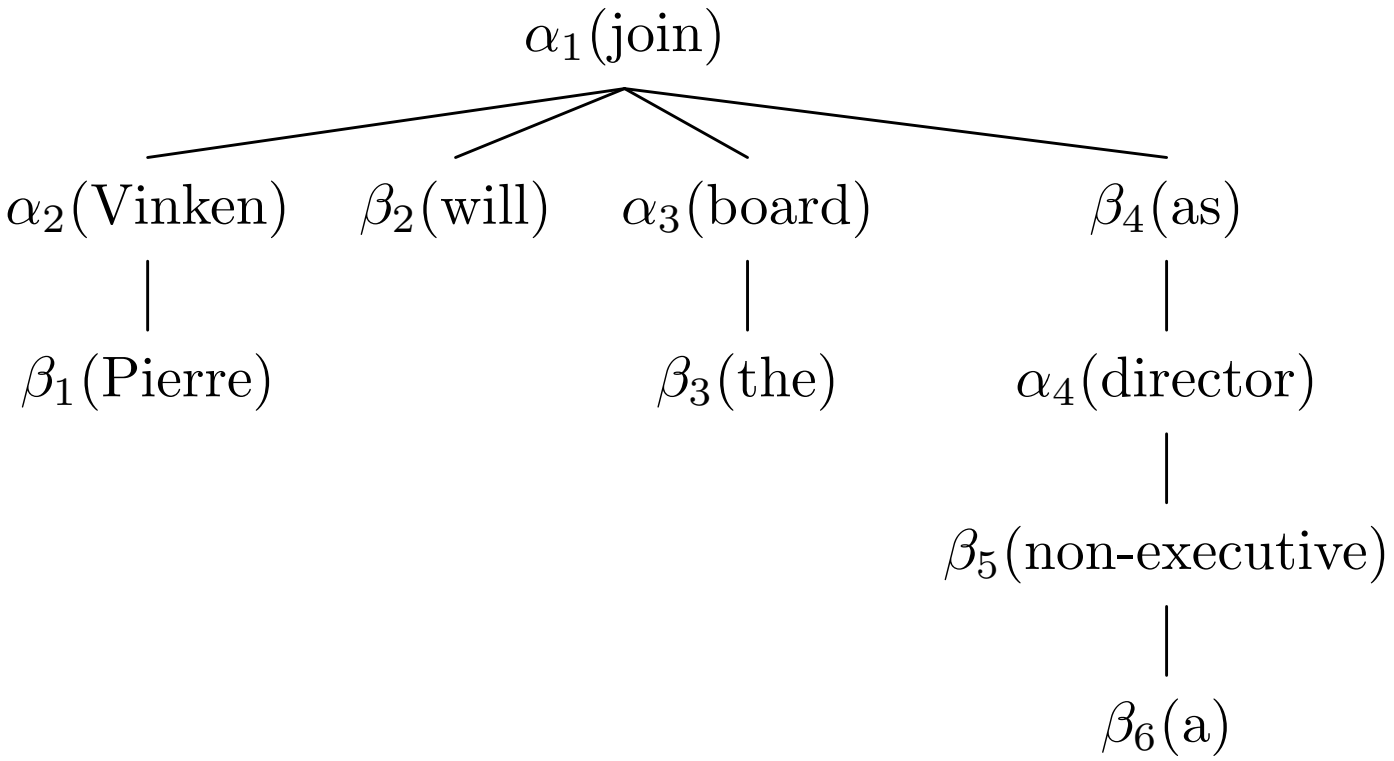


Parsing = n-best Tree Classification and Stapling: (Srinivas 1997)



Model H1: $P(T_i \mid T_{i-2}T_{i-1}) \times P(w_i \mid T_i)$

Parsing = Finding Best Bilexical Dependencies



Model H2: $P(w, T \mid \text{TOP}) \times \Pi_i P(w_i, T_i \mid \eta, w, T)$

The Co-Training Algorithm for Parsing

1. Input: *labeled* and *unlabeled*
2. Update cache
   - Randomly select sentences from *unlabeled* and refill *cache*
   - If *cache* is empty; exit
3. Train models H1 and H2 using *labeled*
4. Apply H1 and H2 to cache.
5. Pick most confidently labeled n from H1 and add to *labeled*.
6. Pick most confidently labeled n from H2 and add to *labeled*
7. $n = n + k$; Go to Step 2

Experiment

- *labeled* was set to Sections 02-06 of the Penn Treebank WSJ
  (9625 sentences)

- *unlabeled* was 30137 sentences
  (Section 07-21 of the Treebank stripped of all annotations).

- A tree dictionary of all lexicalized trees from *labeled* and *unlabeled*.
  Similar to the approach of (Brill 1997)
  New trees were treated as unknown tree tokens

- The *cache* size was 3000 sentences.

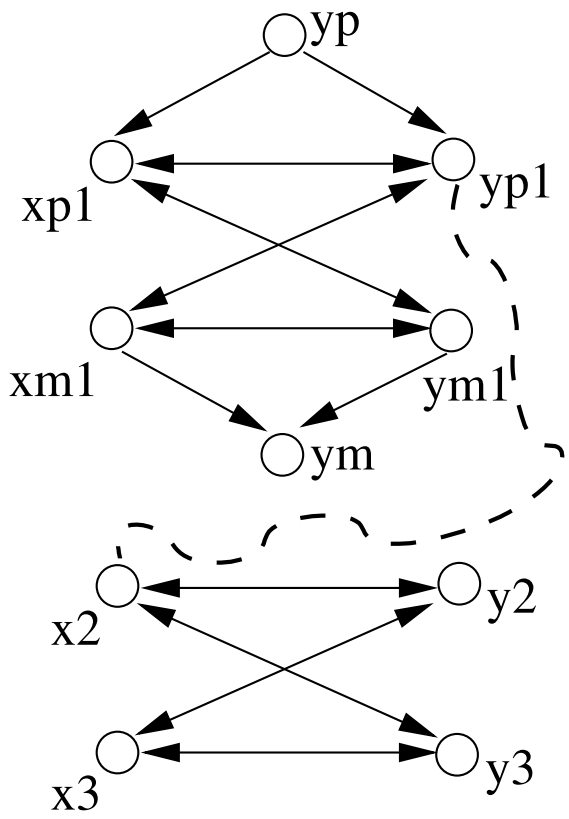
Results

- Test set: Section 23
- Baseline Model was trained only on the *labeled* set:
  Labeled Bracketing Precision = 72.23% Recall = 69.12%
- After 12 iterations of Co-Training:
  Labeled Bracketing Precision = 80.02% Recall = 79.64%
- Evaluation of an unsupervised approach is directly comparable to other supervised parsers (unlike previous work).

Experiment with large set of labeled data

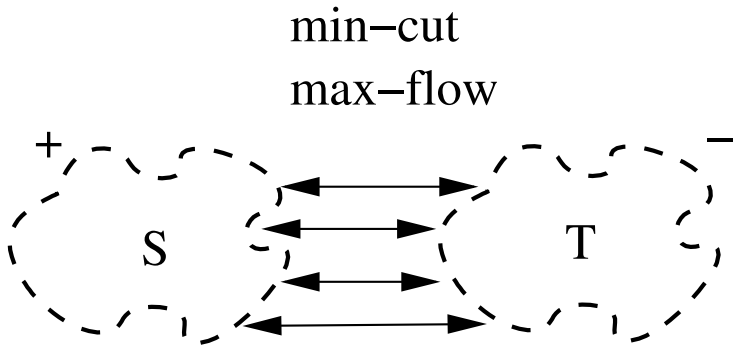
- Still needs human supervision to create the tree dictionary. For small datasets, this is unavoidable.
- Another application: use a large labeled dataset. But improve performance using a much larger unlabeled dataset.
- Expt: 1M words *labeled* and 23M words *unlabeled*. Tree dictionary is completely defined by the labeled set.
- Even after 12 iterations of co-training, performance did not improve significantly over the baseline of LR 85.2% and LP 86%.

Co-Training and Graph Mincuts (Blum and Mitchell 1998; Blum and Chawla 2001)



labeled

unlabeled



Seq

Teq

Co-Training and EM

	max likelihood over full unlabeled set	iterative selection from unlabeled set
$Q_0 \parallel Q_\infty$	EM†	self-training
conditionally independent features	co-EM*	Co-Training

* (Nigam and Ghani, 2000)

† Discriminative Objective f; $Q_0 \parallel Q_{dis}$ (Mitchell, to appear)

Overview

- Introduction to Statistical Parsing
- Tree Adjoining Grammars and Statistical Parsing
- Combining Labeled and Unlabeled Data in Statistical Parsing
- Summary and Future Directions

Future Directions

- Co-training multiple parsers: (JHU summer workshop 2002)
- Applications of statistical parsing, e.g. information extraction, data mining from text
- Predicting RNA secondary structures, protein folding
- Effective use of unlabeled data in machine learning (in other applications)
- Multilingual statistical parsing: English, Korean, Czech, Hindi, Chinese, Arabic

Other Contributions of the Dissertation (not presented in this talk)

- Theoretical Work
  - Consistency of Probabilistic TAGs (COLING-ACL 1998)
  - Prefix Probabilities from Probabilistic TAGs (COLING-ACL 1998)
  - Head-corner parsing algorithm for TAGs (NAACL 2001, TAG+ 2000)
    (implementation used in the XTAG system)
- Corpus-Based Work (combining labeled and unlabeled data)
  - Learning unknown verb subcategorization frames in Czech (COLING 2000)
  - Applying SF learning to verb alternation classes
  - Multilingual parsing: Korean, Hindi

Summary

- Provided new approach to parsing: parsing is treated as two steps: classification and attachment, each with associated probability model

- First application of co-training to the complex problem of statistical parsing (previous work only on binary classifiers)

Summary

- Showed experimental evidence that one can bootstrap from small amounts of labeled data in statistical parsing
- Results are competitive with methods that use larger amounts of labeled data
- First unsupervised approach to statistical parsing that produces the same output as supervised approaches
- Allows direct comparison of unsupervised and supervised approaches