Language and Speech Processing
Treebank Grammars

Reut Tsarfaty
rtsarfat@science.uva.nl

November 13, 2007

1 What we (you) did so far

1.1 Analyze Structures

A Formal Generative Device (A Grammar)

- symbols
- productions
- derivation

Structures

- N-gram Models over words
- Markov Models over POS sequences
- Tree Structures

1.2 Disambiguate Structure

A Probabilistic Component (A Probabilistic Grammar)

- production probabilities
- derivation probability
- structure probability

Objective Functions

- Most Probable Word sequence
- Most Probable POS sequence over words
- Most Probable Parse

1.3 Make Predictions

- Next word prediction
- Assignment of syntactic categories
- Recovery of Grammatical Relations

Depending on the Task at Hand

- Speech Recognition
- Information Retrieval
- Natural Language "Understanding" ($\rightsquigarrow$ our goal)

2 Probabilistic Context Free Grammars

2.1 The Grammar

2.1.1 The Formal Component: CFG

A Context-Free Grammar CFG consists of

- a finite set of terminal symbols V_T
- a finite set of nonterminal symbols N_T
- a finite set of production (rewrite) rules

$$\{A \to B | A \in N_T, B \in (N_T \cup V_T)^*\}$$

- a designated start symbol $S \in N_T$

2.1.2 The Probabilistic Component: PCFG

A Context-Free Grammar CFG consists of

- a finite set of terminal symbols V_T
- a finite set of nonterminal symbols N_T
- a finite set of production (rewrite) rules $R = \{A \to B | A \in N_T, B \in (N_T \cup V_T)^*\}$
- a designated start symbol $S \in N_T$
- a **probability mass function** $P : R \to (0, 1]$ **s.t.**

$$\sum_{\alpha} P(A \to \alpha | A) = 1$$

2.2 The Parser (Last lecture)

2.2.1 The Input

- A PCFG
- A Sentence

2.2.2 The Objective Function

- The Goal: The Most Probable Parse
- The Objective: The Most Probable Derivation

$$argmax_T P(T|S) = argmax_T \frac{P(T,S)}{P(S)} = argmax_T P(T,S) = argmax_T \prod_{r_i \in T} r_i$$

2.2.3 The Algorithm

- Chart Parsing
- Dynamic Programming

2.3 How do we Obtain a PCFG? (This lecture)

	The Formal Component	The Probabilistic Component
(a)	linguists	linguists
(b)	linguists	annotated corpora
(c)	annotated corpora	annotated corpora

3 Treebank Grammars

3.1 Informally:

A Treebank is a body of text annotated with syntactic analyses (parse trees)

A Treebank Grammar is a grammar which is acquired from the treebank

A Treebank PCFG is a PCFG in which both the *productions* and the *production probabilities* are acquired from the treebank using *relative frequency*

3.2 Formally:

3.2.1 Terminology

- **A Set of Parse Trees:** T
- **Frequency Function:** $F_T : T \rightarrow N$
- **A Treebank:** $tb = \langle T, F_T \rangle$

[Note: The set of trees determine the set of sentences in the treebank]

- **A MultiSet of Productions:** R
- **Frequency Function:** $F_R : R \rightarrow N$
- **A Set of Productions:** $\Pi_{tb} = \langle R, F_R \rangle$

3.2.2 Acquiring a Treebank PCFG - step 1: Productions

1. View the parses in tb as derivations of some PCFG
2. Decompose rules into productions obtaining Π_{tb}

3.2.3 Acquiring a Treebank PCFG - step 2: Probabilities

1. Constraint:
   The production probabilities $P : R \rightarrow (0, 1]$ must fulfill

$$\sum_{A \rightarrow \beta \in R} P(A \rightarrow \beta | A) = 1$$

2. Estimate:
   We compute the relative frequency of production $A \rightarrow \alpha$ as follows

$$rf(A \rightarrow \alpha | \Pi_{tb}) = \frac{F_R(A \rightarrow \alpha)}{\sum_\beta F_R(A \rightarrow \beta)}$$

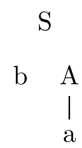
$$\hat{P}(A \rightarrow \alpha | A) = rf(A \rightarrow \alpha | \Pi_{tb})$$

4 Examples

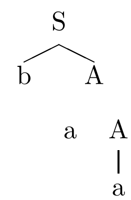
4.1 A Treebank Grammar

Given a set of trees T:

(a)



(b)



And a frequency function F_T:

$F_T(a) = 4$ $F_T(b) = 3$

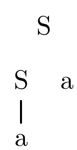
Calculate:

1. R ?
2. F_R ?
3. $\hat{P}(A \rightarrow a|A)$?
4. $\hat{P}((\mathrm{b}))$?

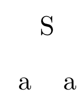
4.2 A Biased Treebank Grammar

Given a set of trees T:

(a)



(b)



And a frequency function F_T:

$F_T(a) = 4$ $F_T(b) = 2$

Calculate:

1. $rf((\mathrm{a}))$? $rf((\mathrm{b}))$?
2. $\hat{P}((\mathrm{a}))$? $\hat{P}((\mathrm{b}))$?

Contemplate:

1. What went wrong ?
2. Why?

Performance of Treebank Grammars: 75% (Charniak 1996) Why?

5 Shortcomings of Treebank PCFGs

5.1 Inadequate Assumptions

Formal Model Assumption: Generated by a Context Free-Grammar

- Domain of Locality
- Cross-Serial Dependencies (Shieber 1987)

Probabilistic Model Assumption: Rule-Independence

- Too Strong (context independence)
- Too Weak (rules cannot decompose)

Statistical Estimation Assumption: Convergence in the limit

- Data set is always finite (underestimation)
- Data set has bias/noise (overtting)

5.2 Inadequate Modeling of Linguistic Phenomena

5.2.1 Selectional Preferences/Subcategorization Frames

- "She ate/picked a banana"
- "She ate/*picked"

5.2.2 Agreement

- " He plays/*play guitar"
- "They play/*plays together"
- "The three little cats play/*plays together"

5.2.3 Long Distance Dependencies

- "Where do you go?"
- "Where do you think you are?"
- "Where the hell do you think you are?"

6 So What do We do?

- **Richer Theories** HPSG, LFG, CCG
- **Richer Models** Data Oriented Parsing
- **Richer PCFGs** Transforms over existing Treebanks

	Pros	**Cons**
Richer Theories	- Linguistically Motivated	- Expensive Annotation - Unclear Usefulness
Richer Models	- Modeling Performance	- Research Stage - Inefficient
Richer PCFGs	- Efficient - Linguistically interesting	- Not readily available - Unclear reuse

[Note: This table should *not* be taken literally!]

7 Enriched PCFG Models

7.1 Treebank PCFG Limitations

- **No Lexicalization** Grammar rules/probabilities are not sensitive to words
- **No Generalization (“weak coverage”)** Shallow structures
- **No Contextualization (“Independence”)** rules treated independently

7.2 Motivation: Why Use Transforms Over Treebanks

Johnson 1998: PCFG Models of Linguistic Tree Representation

7.3 Digression: Heads and Dependencies

7.3.1 The Linguistic Notion of a *Head*

We (/linguists) can identify “Head-Dependent” pairs of words

Examples:

- “last month”
- “a company”
- “sold shares”
- “last month a company sold shares”

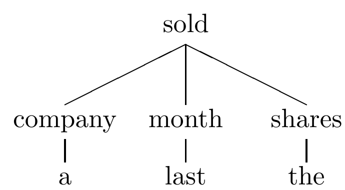
Informal Characterization of Heads:

- The “most important part” (Semantics)

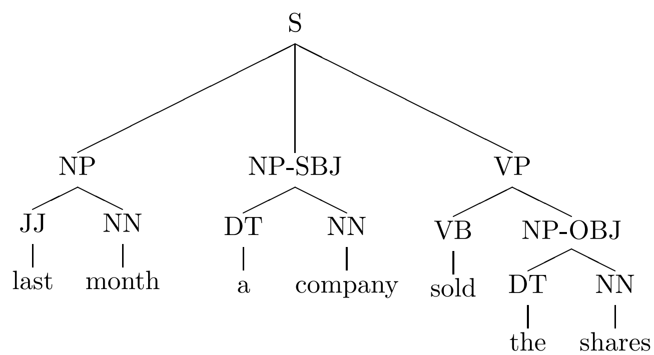
- Impose Selectional preferences (Syntax)
- Define Agreement Features (Morphology)

[Note: These properties do not always coincide!]

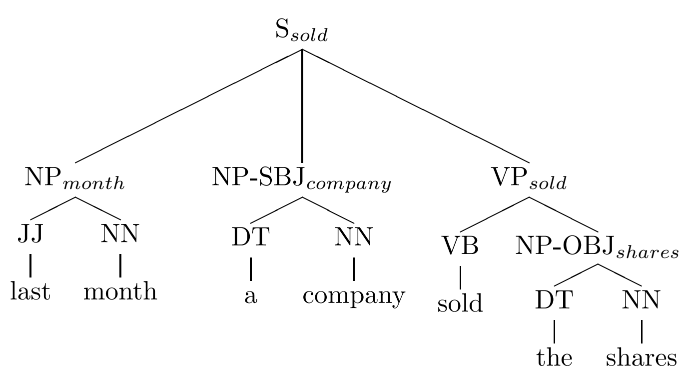
7.3.2 Dependency Structures (DS) as trees



7.3.3 Phrase-Structures (PS) trees



7.3.4 PS-DS structures as trees



Resulting rules exploit bilexical dependencies:

$\mathrm{VP}_{sold} \rightarrow \mathrm{VB}_{sold}\ \mathrm{NP}_{shares}$

$\mathrm{S}_{sold} \rightarrow \mathrm{NP}_{month}\ \mathrm{NP}_{company}\ \mathrm{VP}_{sold}$

Problem: Sparseness. Solution: Linearizion (Later)

7.4 Transforms over Treebank

- Contextualization (Johnson 1998)
- Lexicalization (Collins 1996)
- Linearization (Klein and Manning 2003)
- Refinement (Klein and Manning 2003)

7.5 Parsing the WSJ Penn Treebank

Charniak 1996	75% Treebank Grammar
Johnson 1998	>75% Parent Annotation
Collins 1999	86.6% Head, Markovization, Lexicalization
Klein and Manning 2003	86.3% Parents, Heads, Linearization, Refinement
Petrov et al. 2006	>90% Binarization, Refinement

8 Discussion Points

- Other Structures?
- Other Languages?
- Other Linguistic Modules? (semantics, morphology)
- Other Machine Learning Techniques? (not generative)

8.1 How do we (you) go from here?

Other Models: Data Oriented Parsing (Khalil, Next Lectures)

Other Theories: Recommended course: *Formal Approaches to Grammar* (Reut)
Spring 2008, Block B. Registration: via studieweb
Important: register *before* December 15th 2007.

Other Machine Learning Techniques: Recommended course: *Machine Learning: Pattern Recognition* Winter 2008, Block A/B.

A Parseval Measures

Notation:

- $C(T)$ is the set of constituents of the form $\langle i, X, j \rangle$ in a tree T

Given a Test Set:

- a set of correct parse trees $\{T_c^1, T_c^2, ..., T_c^n\}$
- a set of output parse trees $\{T_o^1, T_o^2, ..., T_o^n\}$

Evaluation Metrics:

- Labeled Recall

$$\frac{\sum_{i=1}^{n} \frac{|C(T_c^i) \cap C(T_o^i)|}{|C(T_c^i)|}}{n}$$

- Labeled Precision

$$\frac{\sum_{i=1}^{n} \frac{|C(T_c^i) \cap C(T_o^i)|}{|C(T_o^i)|}}{n}$$

Implementation Notes

- Discard Top Node
- Discard Part-of-Speech Level
- (Optional: discard punctuation)