# What are Treebank Grammars?

Detlef Prescher, Remko Scha, Khalil Sima'an
*University of Amsterdam*

Andreas Zollmann
*Carnegie Mellon University*

# Motivation

- Probabilistic parsers based on Stochastic Tree Grammars (STGs) achieve **state-of-the-art performance**
- STGs formally generalise probabilistic context-free grammars (PCFGs) because **STGs express contextual evidence by productions that are partial parse-trees**
- It is well-known that maximum-likelihood estimation yields excellent model instances for PCFGs; **By contrast, we still do not know how to estimate STGs with desirable theoretical properties**
- **This talk**: On results of the NWO Project LeStoGram (Oct 2003 - Sept 2006): Bringing together Standard Estimation Theory and Natural Language Processing...

# Overview

- Current Practice in Natural Language Processing
- Standard Estimation Theory
- Treebank Grammars and Estimation Theory
- Related Corpus-Based Methods
- Conclusion

# Natural Language Processing (NLP)

# Applications of NLP

Natural language and NLP play a central role in systems that

- **augment textual or spoken data with information** (e.g. automatic transcription of speech signals, part-of-speech tagging, named-entity recognition, parsing/chunking, word-sense disambiguation)
- **transform textual or spoken data** (e.g. text-to-speech, speech-to-text, spelling correction, text summarization, machine translation)
- **extract information from textual or spoken data** (e.g. information retrieval, question answering, information extraction, data mining)
- **communicate with people** (dialog systems)

# The Aim of NLP

**Scientific**: Build models reflecting the human use of language and speech.

**Technological**: Build models that serve in technological applications.

The main NLP questions are:

1. What are the kind of things that people say and write?
2. What do these things mean?
3. How to incorporate the knowledge about these things into algorithms?

# How to build models of NLP?

**Traditional View: Competence** (Chomsky, $\sim 1960$)
*Grammaticality of sentences in a language* is defined via a set membership test:

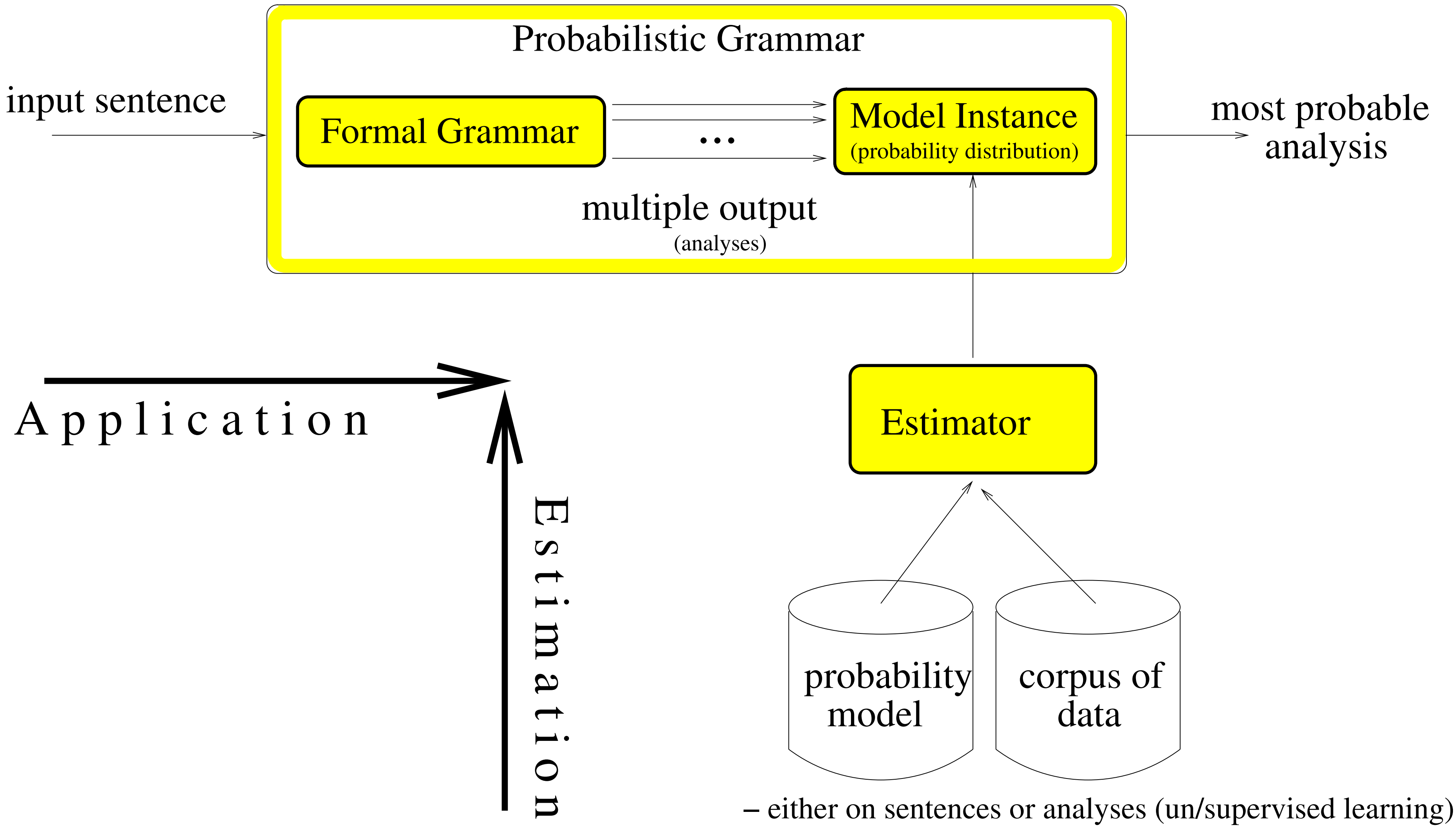
- A *sentence* is a sequence of words,
- A *language* is a set of sentences,
- A *formal grammar* is a device defining the language,

**Modern View: Performance** ($\sim 1990$)
Given a specific NLP task and a specific domain of language use, the human language-behavior is modeled by a

**(black-box) function: input** $\longrightarrow$ **output**,

the output that humans perceive as the most plausible for a given input.

# Building Models of NLP




## *Example: Grammar Estimation*

# Somes Issues in Modeling for NLP

- **How to obtain the symbolic grammar?**
  - Broad-coverage, linguistically motivated, manually constructed grammars: Utilised by early parsing systems; some ongoing activities with Unification Grammars...
  - Treebank Grammars: In current state-of-the art systems, rules are simply read off a corpus of analysed sentences

- **How to estimate the grammar's probabilities?**
  - Context-Free Grammars : Maximum-Likelihood Estimation
  - Tree-Substitution Grammars: The original estimator (DOP1) is biased and inconsistent ; MLE overfits ; ...
  - Unification Grammars: current estimators yield parse 'probabilities' that sum to a value greater one...

# Standard Estimation Theory

# Statistics

RANDOM EXPERIMENT: an experiment whose outcome cannot be predicted with certainty.

RANDOM VARIABLE $X$: a measurement in a random experiment, characterised by a probability distribution $p_X(x) = p(X = x)$ on the set of the outcomes $x$ of $X$.

RANDOM SAMPLE $< X_1, ..., X_n >$: a sequence of *independent* random variables $X_1, ..., X_n$ with the same distribution as the variable $X$ above.

STATISTIC: a random variable derived from the random sample, e.g. the *sample mean* $\overline{X}_n = \frac{1}{n} \sum_{i=1}^{n} X_i$ or the *sample variance* $s_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \overline{X}_n)^2$ .

# Estimators

ESTIMATION THEORY: Guessing the distribution of the random variable $X$ from an observation sequence $<x_1, ..., x_n>$.

MODEL $\mathcal{M}$: The set of admissible distributions. The 'true' distribution of $X$ is assumed to be an instance of $\mathcal{M}$.

PARAMETERS $\Theta$: Typically, the model is characterised by a finite-dimensional set $\Theta \subseteq \mathbb{R}^k$ of *parameter vectors*, i.e., $\mathcal{M} = \{p_\theta | \theta \in \Theta\}$.

ESTIMATOR $\mathrm{est}_n$: a statistic with range $\Theta$.

ESTIMATE: an estimator's parameter guess based on an observation sequence $\langle x_1, \ldots, x_n \rangle$. For example, the *maximum-likelihood estimate* $\arg\max_{\theta \in \Theta} \prod_{i=1}^{n} p_\theta(x_i)$.

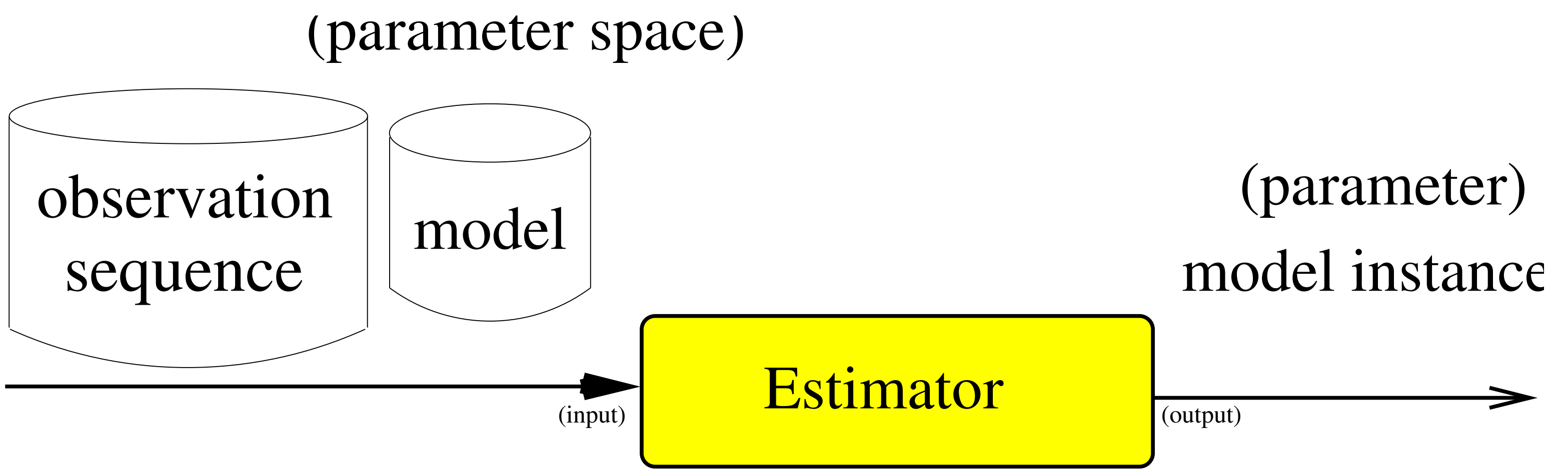
# Properties of Estimators

BIAS: The expected error made by an estimator, i.e., $\mathrm{bias}_\theta(\mathrm{est}_n) = \mathrm{E}(\mathrm{est}_n - \theta)$. If $\mathrm{bias}_\theta(\mathrm{est}_n) = 0$ for all $\theta \in \Theta$, then $\mathrm{est}_n$ is said to be *unbiased*.

CONSISTENCY: Using a loss function $\mathrm{loss}_\theta(\mathrm{est}_n) = ||\mathrm{est}_n - \theta||^2$ for errors, a sequence of estimators $\mathrm{est}_n$ is called *consistent* if for each $\theta \in \Theta$, the expected loss approaches zero as $n$ tends to infinity: $\lim_{n\to\infty} \mathrm{E}(\mathrm{loss}_\theta(\mathrm{est}_n)) = 0$.

MINIMAL SUFFICIENCY: A statistic $U = h(X_1, \ldots, X_n)$ is called *sufficient for* $\theta$ if $U$ contains all of the information about $\theta$ that is available in the entire sample. Thus a sufficient statistic $U$ taking values in an $m-$dimensional space with $m < n$ yields a data reduction with no loss of information. Typically, one looks for sufficient statistics with smallest dimension possible.

# Current Practice in Parameter Estimation




**Standard estimation theory**:

- build a model with a finite-dimensional parameter space
- ensure that the model contains the 'true' distribution
- search for unbiased and/or consistent estimators
- base estimation on minimal sufficient statistics

# Treebank Grammars

# and

# Standard Estimation Theory

# What Are the Parameters of Probabilistic Parsing?

From an Estimation Theory perspective, probability estimation from a corpus of syntactic annotations is used for two tasks:

- **Task 1**: Estimate the production probabilities of an a priori fixed grammar

  $\Longrightarrow$ <u>parameters = production probabilities</u>

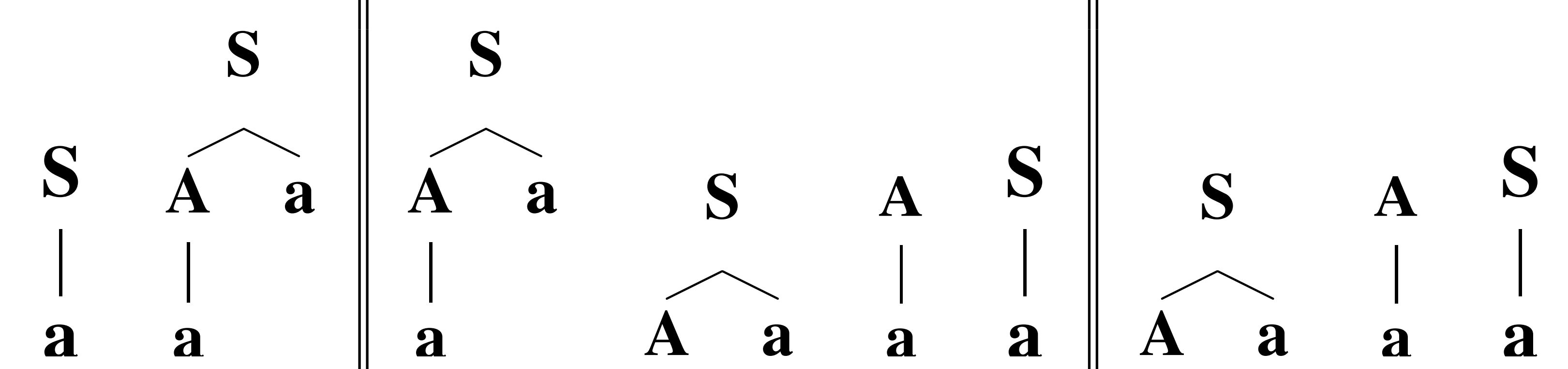
- **Task 2**: Estimate the probability distribution over the parses themselves

  $\Longrightarrow$ <u>parameters = parse probabilities</u>

# Choose the Right Parameters!

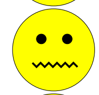
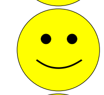
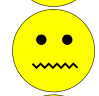
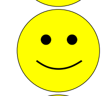
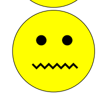
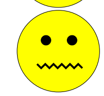
**Example:** Different tree-substitution grammars with the same parse distribution.

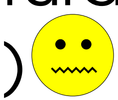
| PARSES | | GRAMMAR1 | | | | GRAMMAR2 | | |
|---|---|---|---|---|---|---|---|---|
| $t_1$ | $t_2$ | 0.25 | 0.25 | 1.0 | 0.5 | 0.5 | 1.0 | 0.5 |



***Two different tasks: Estimating a probabilistic grammar is not equivalent to estimating a parse distribution!***

# Pro/Cons for Estimating Production/Parse Probs

| **ESTIMATION** | via **productions** | via **parses** |
| --- | --- | --- |
| finite-dimensional model? |  |  |
| true distribution in the model? |  |  |
| consistent estimators? |  |  |
| minimal sufficient statistics? |  |  |

**Linguistic Perspective**: (i) Estimating production probabilities implies pinning down a grammar prior to estimation. The chosen grammar has to reflect the exact nature of natural language syntax (which is a very strong assumption) (ii) For ambiguity resolution, the alternative parses have to be ranked by parse probabilities (and should therefore be parameterised)

# The Parameters of Probabilistic Parsing

- The actual goal is to **estimate parse distributions** (of which the treebank is a finite sample)

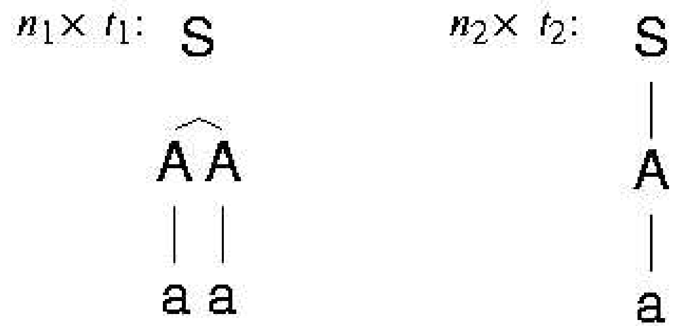
- **Paradigm shift**: Assume that *some grammar* — but not an *a priori* constructed and fixed one — generates the parse distribution

- Search for a minimal sufficient statistics to reduce the infinite-dimensional parse space to a **finite-dimensional model**

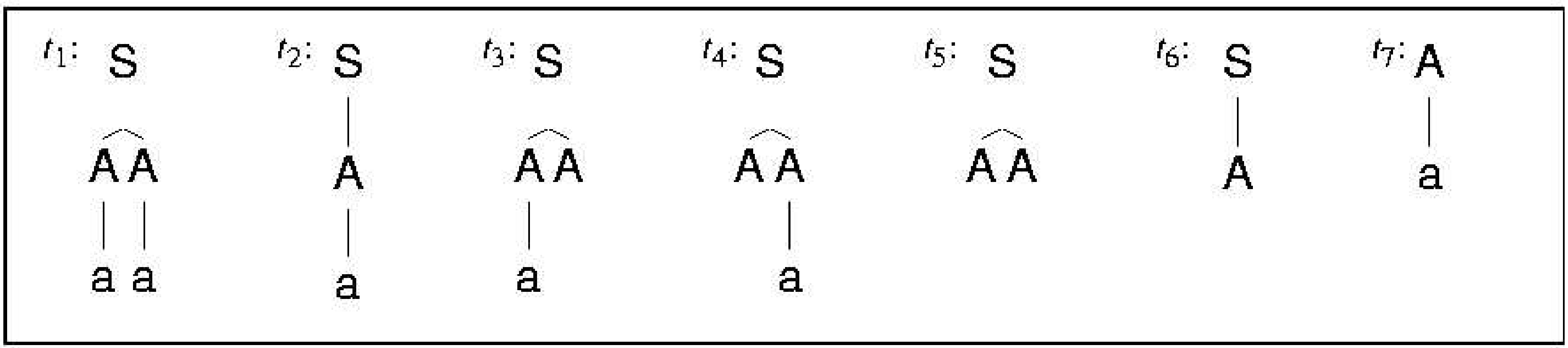
$\Longrightarrow$ Explore **Treebank Grammars** i.e. probabilistic grammars with **productions directly projected from the treebank**

# Treebank-Grammar Approaches

**Treebank** (Example by Johnson, 2002):




**Tree fragments:**

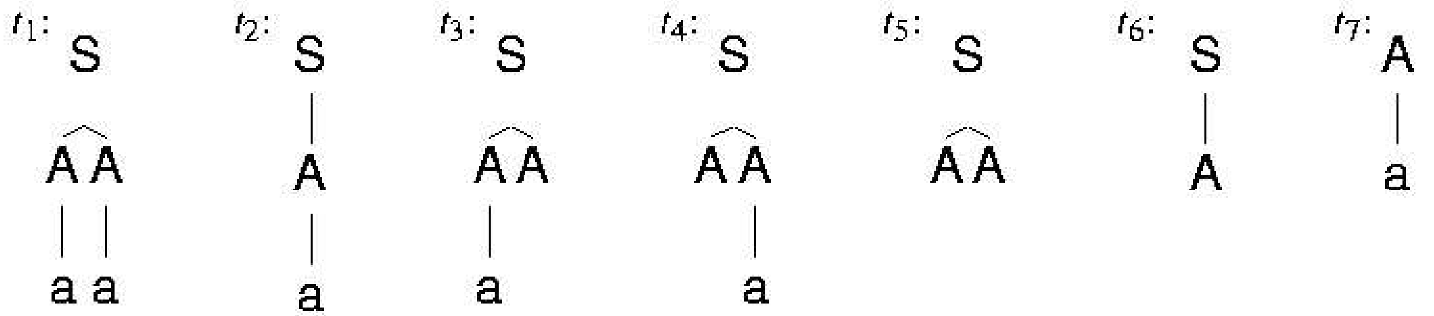



**Tree derivations** (Trees with hidden breakpoints):

$$D(t_1) = \{t_1,\ t_3 \circ t_7,\ t_4 \circ t_7,\ t_5 \circ t_7 \circ t_7\} \quad \text{and} \quad D(t_2) = \{t_2,\ t_6 \circ t_7\}$$

## *Example: Data-Oriented Parsing (DOP)*

# DOP Estimation: Properties...




| Fragments | $t_1$ | $t_2$ | $t_3$ | $t_4$ | $t_5$ | $t_6$ | $t_7$ |
|---|---|---|---|---|---|---|---|
| $f_{\mathrm{DOP1}}$ | $n_1$ | $\boxed{n_2}$ | $n_1$ | $n_1$ | $n_1$ | $\boxed{n_2}$ | $n_2$ |
| $\pi_{\mathrm{DOP1}}$ | $\frac{n_1}{4n_1+2n_2}$ | $\boxed{\frac{n_2}{4n_1+2n_2}}$ | $\frac{n_1}{4n_1+2n_2}$ | $\frac{n_1}{4n_1+2n_2}$ | $\frac{n_1}{4n_1+2n_2}$ | $\boxed{\frac{n_2}{4n_1+2n_2}}$ | 1 |
| $p_{\mathrm{DOP1}}$ | $\frac{4n_1}{4n_1+2n_2}$ | $\frac{2n_2}{4n_1+2n_2}$ | | | | | |
| $p_{\mathrm{PCFG}}$ | $\frac{n_1}{n_1+n_2}$ | $\frac{n_2}{n_1+n_2}$ | | | $\left(\frac{n_1}{n_1+n_2}\right)$ | $\left(\frac{n_2}{n_1+n_2}\right)$ | (1) |

***The original DOP1 estimator is biased and inconsistent***

# DOP Estimation: More Problems...

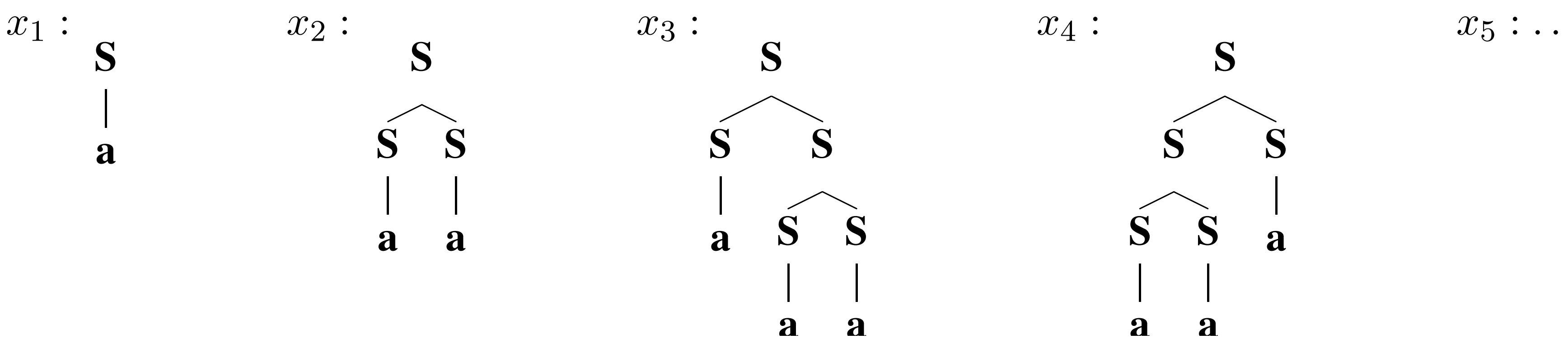
**Maximum-likelihood estimation (MLE)** (Fisher 1912), typically yields an excellent estimate if the given corpus is large:

- under certain conditions which are typically satisfied in practical problems, they are **consistent estimators**,
- unlike the relative-frequency estimator, maximum-likelihood estimators typically **do not over-fit the given corpus** in practice.

**Unfortunately**: MLE results in a completely over-fitting instance of the standard DOP model, *which does not assign a positive probability to any tree outside the given treebank...*

# DOP Estimation: A More Fundamental Problem

In sharp contrast to PCFG estimation, the typical asymptotic behavior of DOP estimation is that *the symbolic backbone of DOP's probability model grows as the treebank grows*




***In the limit of the treebank size, DOP risks learning an arbitrarily large grammar — even if the treebank is generated by a finite grammar.***

# Related Corpus-Based Methods

# The Unknown-Word Problem

**Unknown words**: words that have not occurred in the training data but that will occur in new sentences... Unknown words have been linked to Zipf's law: as a corpus grows there are always new phenomena to be expected to occur in the future...

**Examples:** Open category words like proper nouns and compound nouns, but also verbs are made up on the fly all the time (e.g. 'googling someone').

**Unknown-Word Problem:** One cannot determine a **finite** set of allowed words (the terminal symbols in the formal-grammar terminology) a priori to estimation...

# Generalising the Unknown-Word Problem

The unknown-word problem may be streched to:

- **unknown categories**: words for which some part-of-speech categories are not in the corpus
- **unknown productions**: many productions in the well-known Penn Wall Street Journal treebank occur only once, hinting at the fact that other novel productions are likely to occur in new utterances...

# Current Solution for the Unknown-Word Problem

Most NLP systems based on probability models over word sequences utilise:

**Smoothing Techniques**:

1. Estimate the parameters of a (finite) grammar, including a special symbol `UNKNOWN`, a category of unknown events
2. Use a mapping from a word to itself if it is known, or else to the `UNKNOWN` category
3. Reserve and distribute probability mass to the map into `UNKNOWN`

**Problem**: The second step (the mapping) can only be described by an infinite set of rules that maps a novel word to its `UNKNOWN`...

# Conclusion

- We raise a question as to whether any probabilistic instance of an **a priori fixed, finite grammar** can reflect natural-language syntax

- DOP (and any other higher-order STG) aims at estimating an infinite-dimensional parameter vector, implying that **DOP estimation is incompatible with Estimation Theory**

- Similiarly, other corpus-based methods in NLP (like smoothing) can only be described by an infinite set of rules

- **It seems necessary and reasonable to lift certain finiteness restrictions on the formal grammar that is assumed to generate a natural language**

Thank you!

# The Elements of NLP

**Phonetics/Phonology**: map acoustic signals to phoneme and/or grapheme sequences and vice versa (speech recognition/synthesis)

**Morphology**: analyze the structure of words (morphological analysis)

**Syntax**: identify the category of words (POS tagging), analyze the structure of sentences (parsing/generation)

**Semantics**: calculate the meaning of words/sentences (lexical/compositional semantics)

**Discourse**: analyze the structure of dialog or text (discourse representation)

**Pragmatics**: incorporate world knowledge, cultural convention
a specific use of language.

# What changed NLP?

**Competence Models:** In contrast to people, the linguistic view of language as a set does not care about problems caused by ambiguity. Competence models

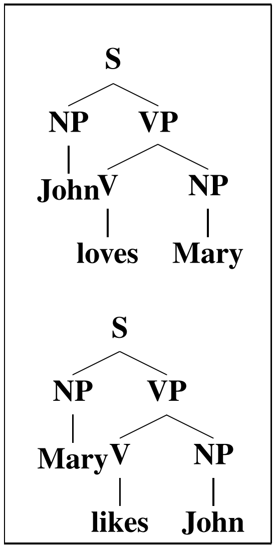
- **cannot resolve multiple output**
- **cannot handle multiple input (noisy utterances)**
- **cannot express multiple levels of grammaticality**

**Performance Models:** Mimic people's language behavior and are specifically designed to resolve ambiguity. They

- handle uncertainty with Probability Theory and Statistics
- utilise competence models as components
- have even the potential to model extra-linguistic factors

# Current Practice: Treebank Grammars

TREEBANK

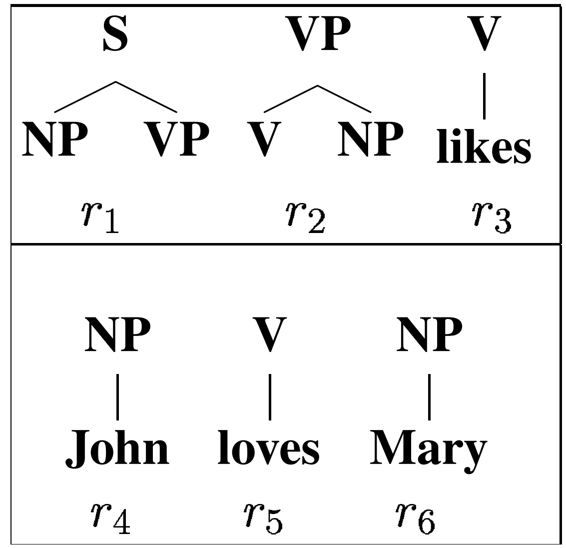



collect rules

$\Longrightarrow$

(e.g. local trees)

PROBABILISTIC GRAMMAR




RULE PROBABILITY: relative-frequency estimate on the corpus of all rules with the same left-hand side, e.g.

$$\pi(r_4) = \frac{\mathtt{count}(r_4)}{\mathtt{count}(r_4) + \mathtt{count}(r_6)}$$

DERIVATION PROBABILITY: product of rule probabilities

TREE PROBABILITY: sum of derivation probabilities

# Current Practice in DOP Estimation

- **DOP Back-Off (Burrato and Sima'an 2003)**: Stick with the 'All-Fragments Approach' of DOP but give up Maximum-Likelihood Estimation for DOP. Use instead back-off distributions based on fragments and their counts...

- **DOP* (Zollmann and Sima'an 2005)**: Stick with Maximum-Likelihood Estimation for DOP but give up the 'All-Fragments Approach' of DOP...

***Have we to give up the spirit of DOP saying that DOP is some kind of a Memory-Based-Learning approach??!***