

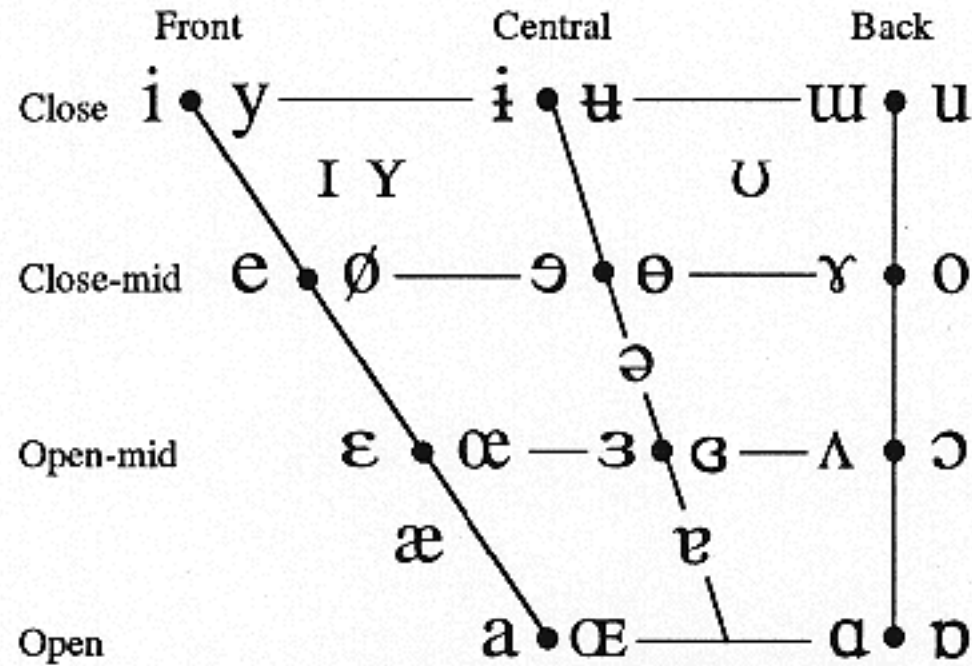
**Data-Oriented Language Learning –
moving beyond negative learnability results**

**Jelle Zuidema
ILLC
University of Amsterdam**

Plan of the talk

1. Context-Free Grammars as models of Language
2. Learning Context-Free Grammars
3. Stochastic Tree Grammars as models of Language
4. Learning Stochastic Tree Grammars

VOWELS



Where symbols appear in pairs, the one to the right represents a rounded vowel.

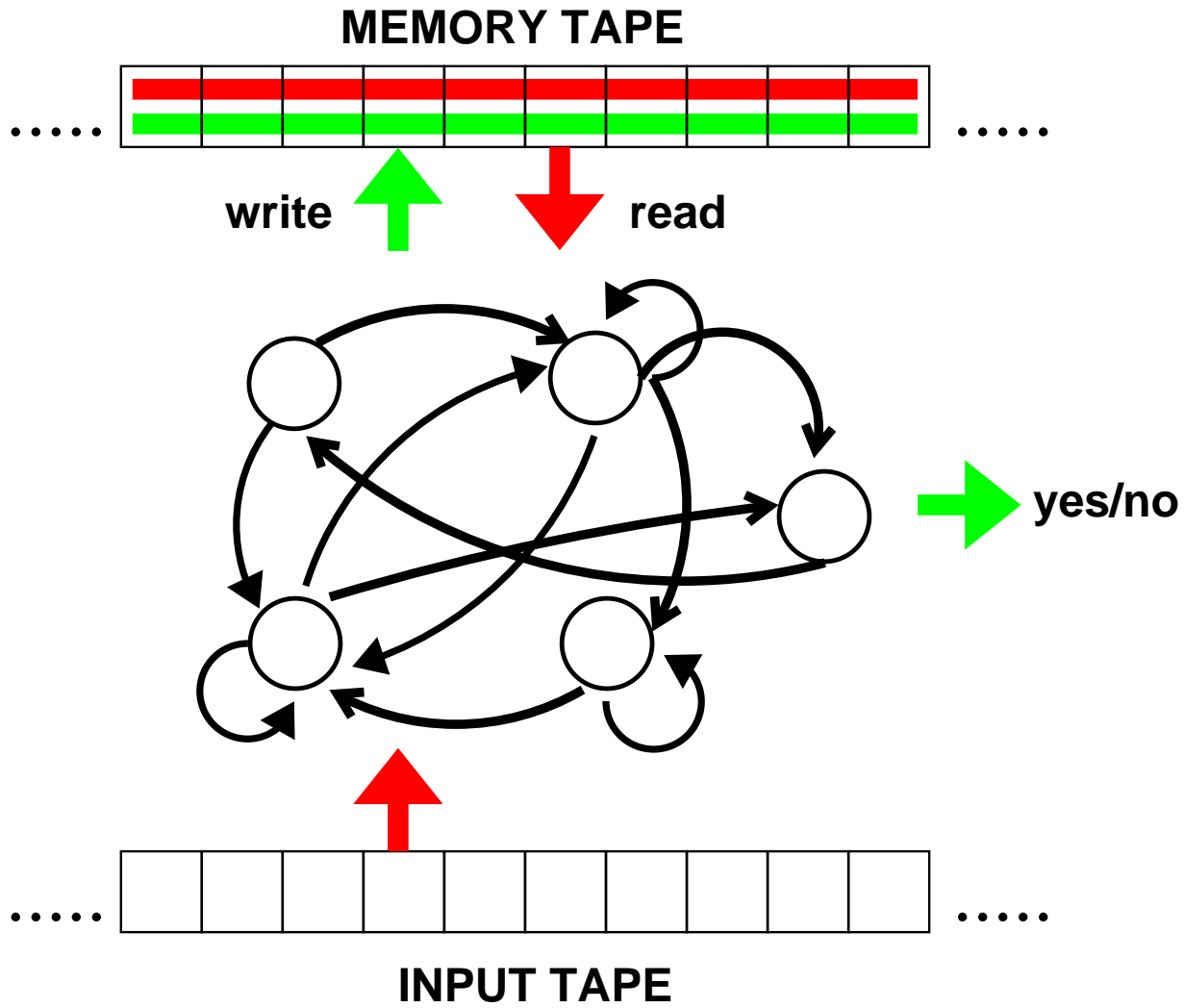
$$\Sigma = \{x \mid x \text{ is a phoneme of English}\}$$

(Chomsky, 1957)

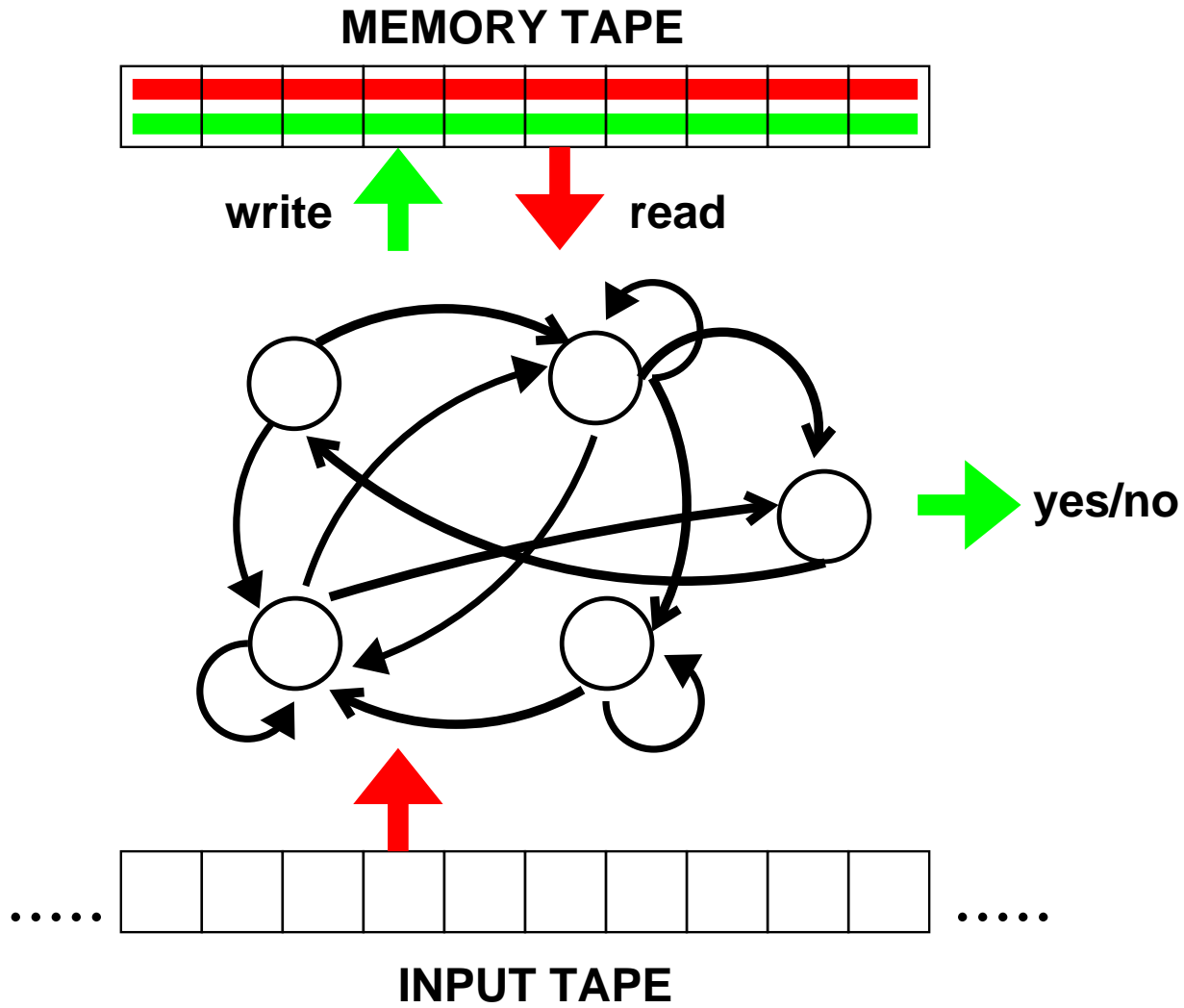
- (1) Colorless green ideas sleep furiously.
- (2) *Furiously sleep ideas green colorless.
- (3) have you a book on modern music?
- (4) the book seems interesting.
- (5) *read you a book on modern music?
- (6) *the child seems sleeping.

grammaticality : $\Sigma^* \mapsto \{yes, no\}$

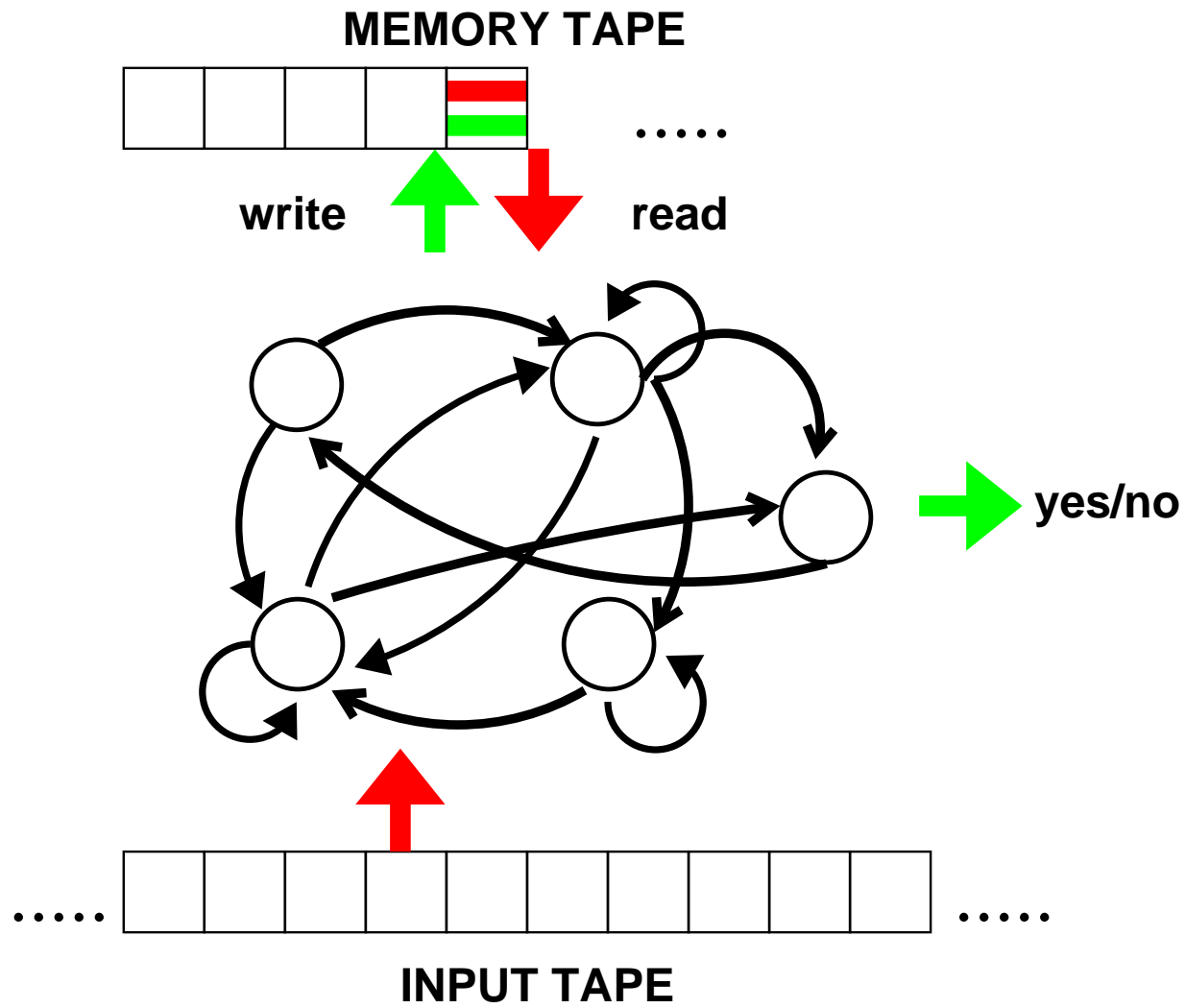
Turing Machine (Type 0)



Linear Bounded Automaton (Type 1)

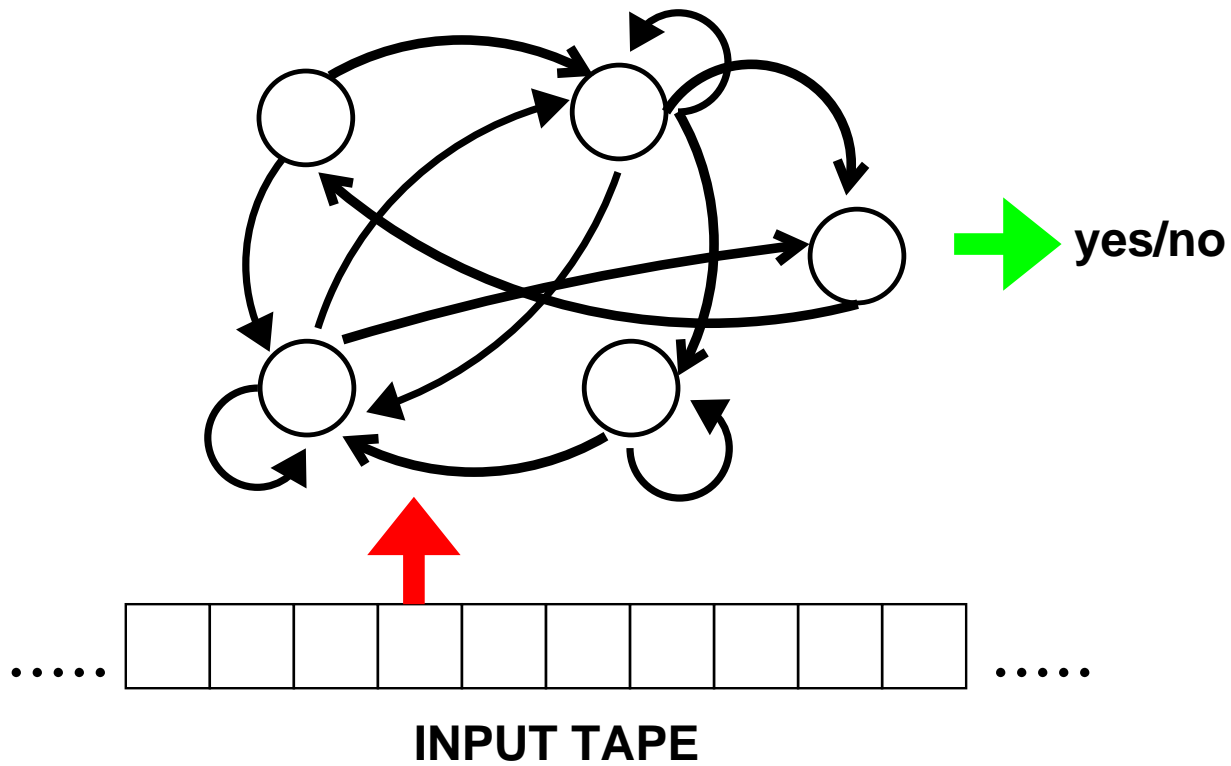


Push-down Automaton (Type 2)



Finite-state Automaton (Type 3)

(NO MEMORY TAPE)



Finite-state Automata are inadequate

(Chomsky, 1957)

Let S_1, S_2, S_3, S_4 be simple declarative sentences in English.

(7) If S_1 , then S_2 .

(8) Either S_3 or S_4 .

(9) The man who said that S_5 , is arriving today

(Context-Free) Phrase Structure Grammars

- Sentence \rightarrow NP + VP
- NP \rightarrow T + N
- VP \rightarrow Verb + NP
- T \rightarrow *the*
- N \rightarrow *man, ball, etc.*
- Verb \rightarrow *hit, took, etc.*

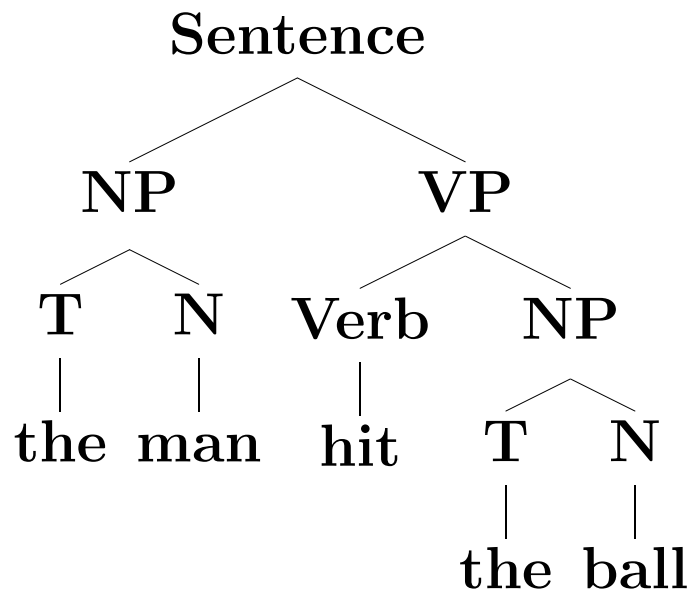
(Context-Free) Phrase Structure Grammars

- Sentence \rightarrow NP + VP
- NP \rightarrow T + N
- VP \rightarrow Verb + NP
- T \rightarrow *the*
- N \rightarrow *man, ball, etc.*
- Verb \rightarrow *hit, took, etc.*

1. *Sentence*
2. NP + VP
3. T + N + VP
4. T + N + Verb + NP
5. *the* + N + Verb + NP
6. *the* + *man* + Verb + NP
7. *the* + *man* + *hit* + NP
8. *the* + *man* + *hit* + NP
9. *the* + *man* + *hit* + T + N
10. *the* + *man* + *hit* + *the* + N
11. *the* + *man* + *hit* + *the* + *ball*

(Context-Free) Phrase Structure Grammars

- Sentence \rightarrow NP + VP
- NP \rightarrow T + N
- VP \rightarrow Verb + NP
- T \rightarrow *the*
- N \rightarrow *man, ball, etc.*
- Verb \rightarrow *hit, took, etc.*



1. *Sentence*
2. NP + VP
3. T + N + VP
4. T + N + Verb + NP
5. the + N + Verb + NP
6. the + man + Verb + NP
7. the + man + hit + NP
8. the + man + hit + NP
9. the + man + hit + T + N
10. the + man + hit + the + N
11. the + man + hit + the + ball

Context-Free Grammars = Type 2

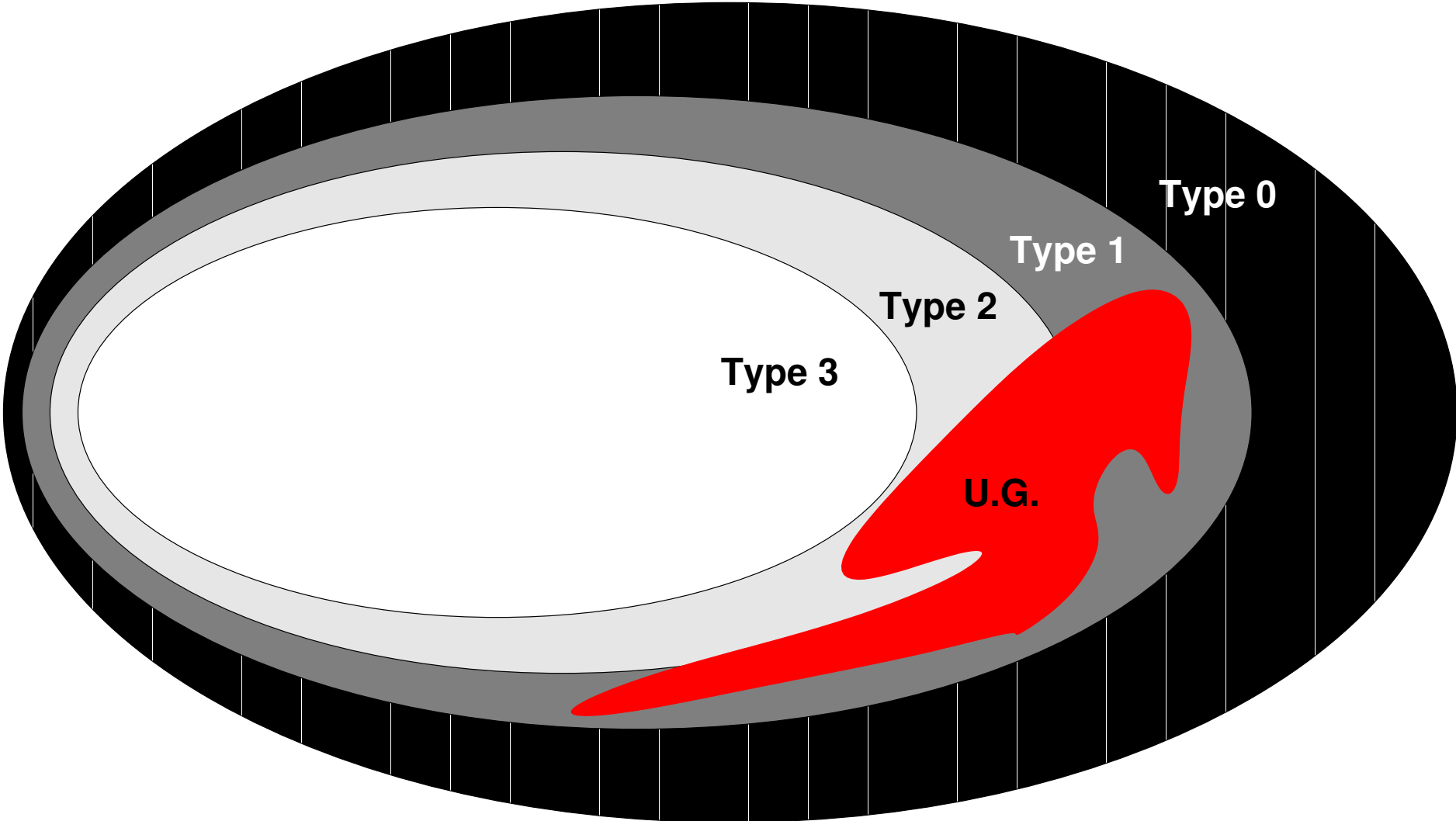
- Sentence \rightarrow NP + VP
- NP \rightarrow T + N
- VP \rightarrow Verb + NP
- T \rightarrow *the*
- N \rightarrow *man, ball, etc.*
- Verb \rightarrow *hit, took, etc.*

A rewriting grammar $G = \langle P, S, V_{nt}, \Sigma \rangle$, where:

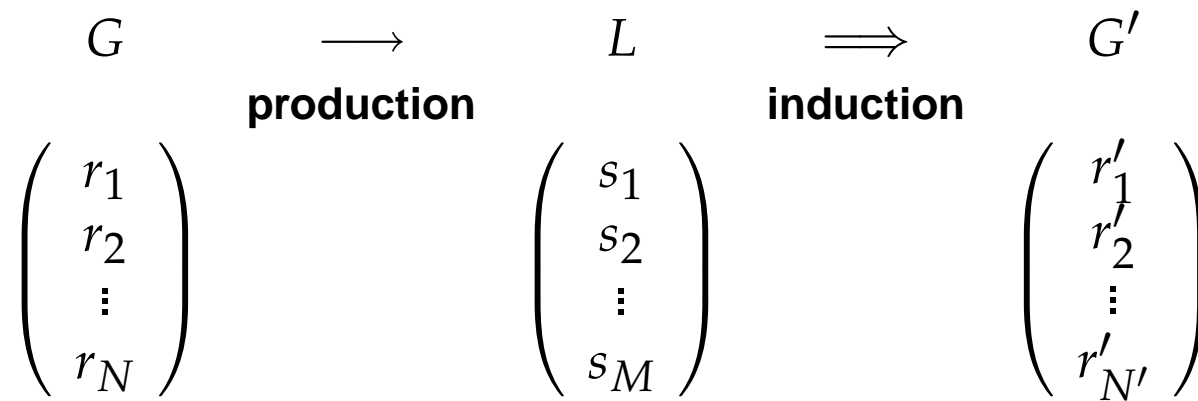
- S is the set of start symbols;
- V_{nt} is the set of *nonterminal* symbols (variables);
- Σ is the set of *terminal* symbols;
- P is the set of rewrite rules

A grammar is of TYPE 2 (the “context-free grammars”) if every rule is of the form $A \mapsto w$, where $A \in V_{nt}$, and $w \in (V_{nt} \cup \Sigma)^*$.

The Chomsky Hierarchy



Learnability



Identification in the limit

(Gold, 1967)

“A class of language will be called identifiable in the limit [...] if there is an effective learner [...] with the following property: Given any language of the class and given any allowable training sequence for this language, the language will be identified in the limit.”

- Positive evidence: *Text*
- Negative evidence: *Informant sequence*

Context-free grammars are not learnable from text

Infinite languages can not be identified, because there exists an infinite sequence of finite languages that are indistinguishable for any amount of training samples.. I.e. no matter how many examples you have seen, you'll never know whether you've seen the whole language or whether you should generalize to (infinitely) more.

<i>infinite language</i>	<i>finite languages</i>
$S \mapsto Sa$	$S \mapsto a$
$S \mapsto a$	$S \mapsto aa$
	$S \mapsto aaa$
	$S \mapsto aaaa$
	$S \mapsto aaaaa$
	...

Principle & Parameter grammars are learnable from text

(Wexler & Culicover, 1980)

E.g., by *identification through enumeration*. The algorithm considers a finite number of hypotheses; it sticks to an hypothesis until it receives a counter example; it will always receive a counter example within a finite amount of time. If it considers hypotheses in the right order, it will therefore always arrive and stay at the correct hypothesis within a finite amount of time.

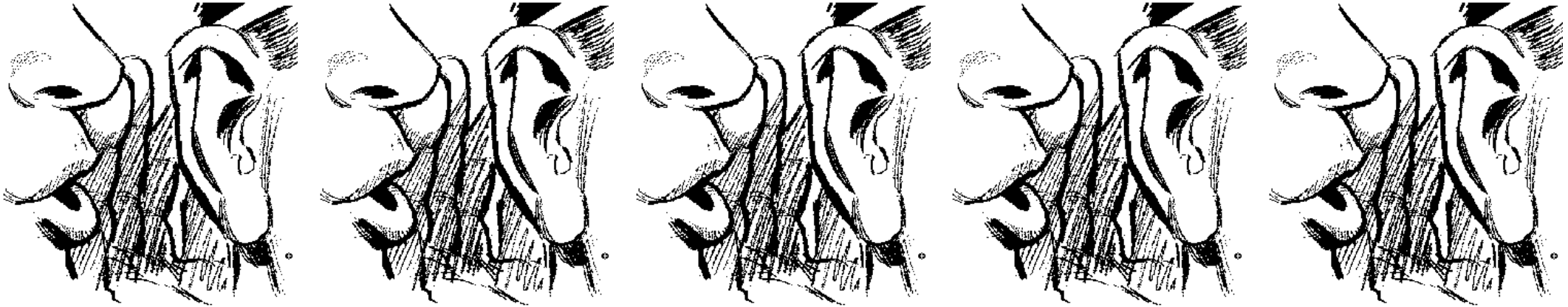
$S \mapsto aSb$	$S \mapsto aS aA$
$S \mapsto ab$	$A \mapsto Ab b$
<hr/>	<hr/>
<i>aabb</i>	<i>aabb</i>
<i>aaaabbbb</i>	<i>aaaabbbb</i>
<i>aaaaabbbbb</i>	<i>aaaaabbbbb</i>
<i>aaaaabbbbb</i>	<i>aabbbbb</i>

“The basic results of the field [of learnability theory] include the formal, mathematical demonstration that without serious constraints on the nature of human grammar, no possible learning algorithm can in fact learn the class of human grammars.”

(Wexler, 1999, “Innateness of Language”, MIT Encyclopedia of Cognitive Science)

Iterated Learning

(Zuidema, 2003; Kirby 1999)



Representation

Context-free grammars

- Rules of the forms: $A \mapsto t$, $A \mapsto BC$, $A \mapsto Bt$
- Start symbol S , terminal symbols (lexicon), non-terminal symbols

Lexical

$S \mapsto$ mary shouts
 $S \mapsto$ angry mary shouts
 $S \mapsto$...

Combinatorial

$S \mapsto$ N V
 $N \mapsto$ mary
 $N \mapsto$ angry mary
 $N \mapsto$...
 $V \mapsto$ walks

Recursive

$S \mapsto$ N V
 $N \mapsto$ A N
 $N \mapsto$ mary
 $A \mapsto$ beautiful
 $A \mapsto$ angry
 $V \mapsto$ walks

Learning algorithm

Incorporation: extend the language, such that it includes the encountered string

Compression: substitute frequent and long substrings with a nonterminal (the grammar becomes smaller and the language remains unchanged)

Generalization: equate two nonterminals if they occur frequently in the same context

Example

Training sentences: abcd, abcababcd, abcabcabcd

(a) Incorporation

S \mapsto abcd
S \mapsto **abcabcd**
S \mapsto **abcabcabcd**

(b) Compression

S \mapsto **abcd**
S \mapsto Xd
S \mapsto Xabcd
X \mapsto **abcabc**

(c) Compression

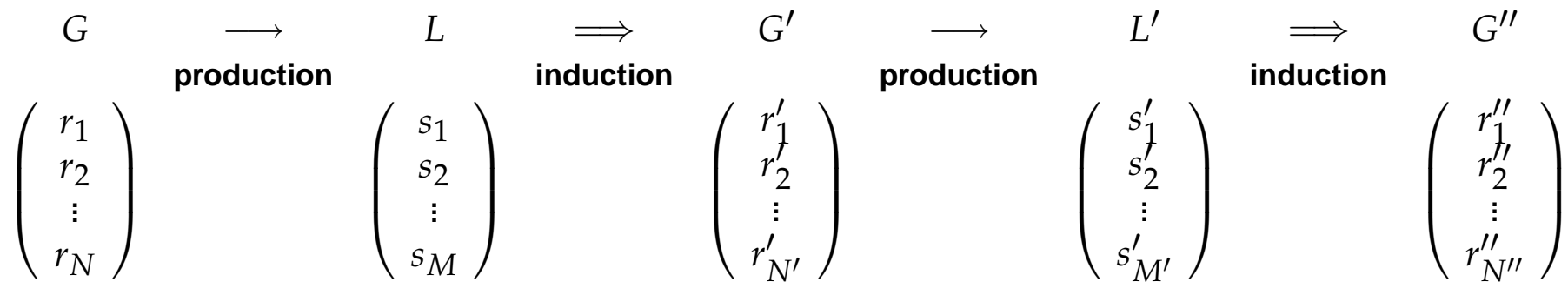
S \mapsto **Yd**
S \mapsto **Xd**
S \mapsto Xabcd
X \mapsto YY
Y \mapsto abc

(d) Generalization

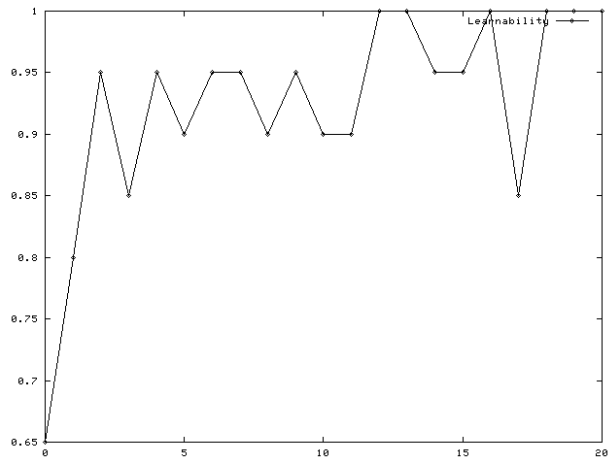
S \mapsto **Xd**
S \mapsto Xabcd
X \mapsto XX
X \mapsto abc

Iterated Learning

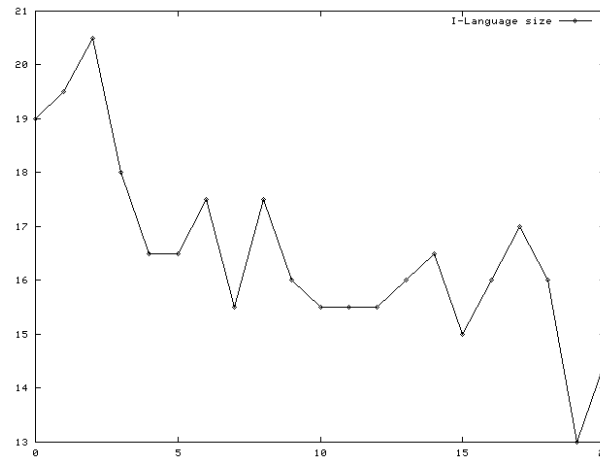
Individuals in a *chain* learn from the previous individual and teach to the next



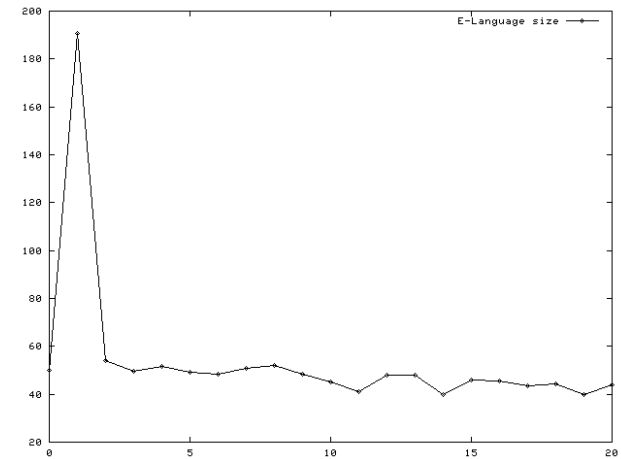
Iterated Learning - results



learnability



number of rules



number of sentences

Parameters: $V_t = \{a, b, c, d\}$, $V_{nt} = \{S, X, Y, Z, A, B, C\}$, $T=30$, $E=20$, $l_0=3$.
Shown are the average values of 2 simulations.

Example

1. “ada”, “ddac”, “adba”, “bcbd”, “cdca”
2. “dcac”, “bcac”, “caac”, “daac”
3. $S \mapsto dcX$, $S \mapsto bcX$, $S \mapsto caX$, $S \mapsto daX$, and $X \mapsto ac$
4. $Y \mapsto b$
5. “dcac”, “bcac”, “caac”, “bcb”, “cab”, “dab”,

Language Adaptation

1. Languages are transmitted culturally and are subject to change
2. Languages will change more if they are difficult to learn
3. Over time, languages that are easy to learn are more likely to occur

The Uniformity Fallacy

Assuming every *possible* language is equally likely is NOT the proper assumption for natural languages.

- The average learnability of a class can be very good, if easy languages occur much more often than difficult languages
- Biased learning algorithms can be more successful than the “best possible” learner

Probabilistic Linguistics

(Abney 1996)

(10) a. #the a are of I word salad?

b. John saw Mary unambiguous?

(11) a. a hectare is a hundred ares

b. As described in section I paragraph a ...

c. The a paragraph of I is hardly readable.

d. Typhoid Mary

e. the Russia house butler

Probabilistic Context-Free Grammars

r1.	S	→ NP VP	.7
r2.	S	→ NP	.3
r3.	NP	→ N	.8
r4.	NP	→ N N	.2
r5.	N	→ John	.6
r6.	N	→ walks	.4
r7.	V	→ walks	1.0

John walks S or NP?

$$P(r1 \circ r3 \circ r5 \circ r7) = .7 \times .8 \times .6 \times 1.0 = .336$$

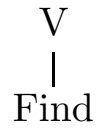
$$P(r2 \circ r4 \circ r5 \circ r6) = .3 \times .2 \times .6 \times .4 = .0144$$

Constructions

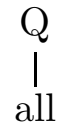
- (12) a. Grandpa kicked the bucket.
b. One everyone will kick the bucket.
c. Logician after logician wants to do linguistics.
- (13) a. What time is it?
b. #How late is it?
- (14) a. When is the next train *from* Amsterdam *to* Paris?
b. Show me the *nearest* airport *to* Denver.
c. BA carried *more* people *than* cargo in 1987.

(Fillmore, Kay, Goldberg, Jackendoff; Bod, 1998)

Tree Substitution Grammars



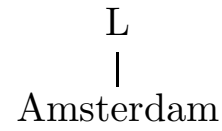
(a) t_1



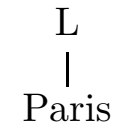
(b) t_2



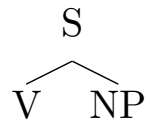
(c) t_3



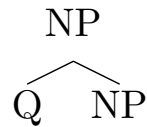
(d) t_4



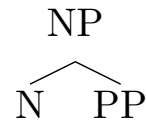
(e) t_5



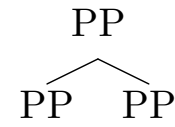
(f) t_6



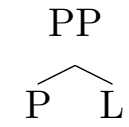
(g) t_7



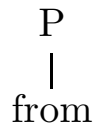
(h) t_8



(i) t_9



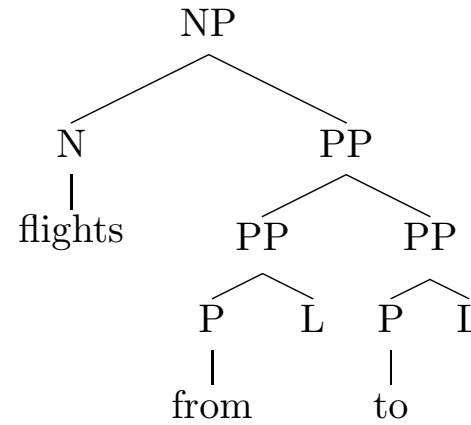
(j) t_{10}



(k) t_{11}



(l) t_{12}



(m) t_{13}

Data-Oriented Parsing (DOP1)

(Bod, 1998)

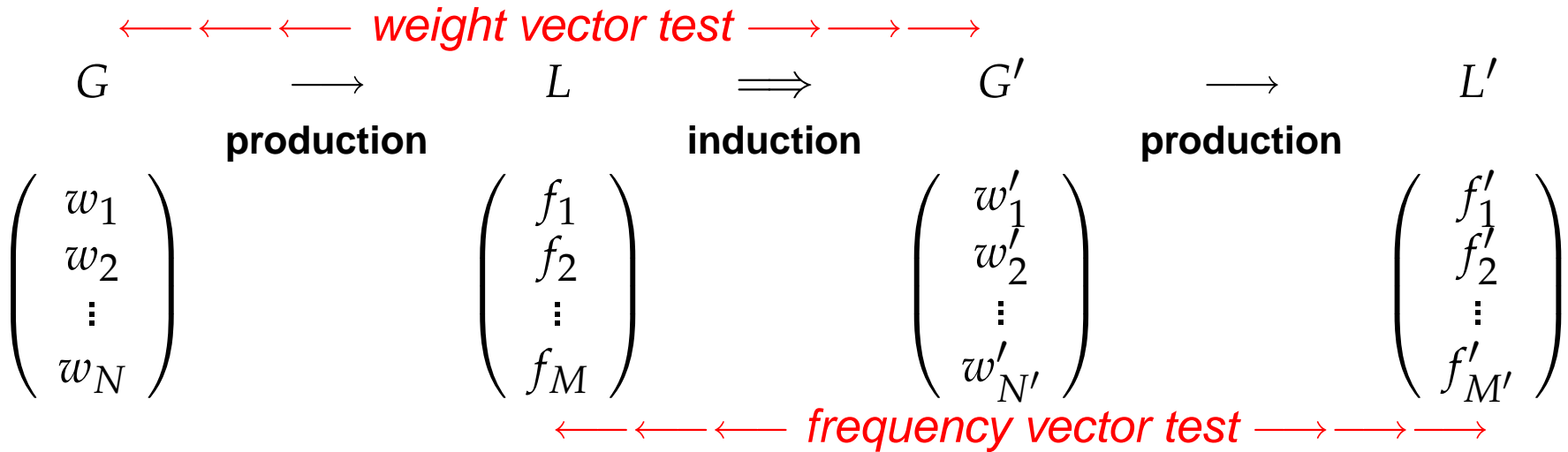
weights $P(t_i) = w_i = \frac{f_i}{\sum_{j: \text{root}(t_j) = \text{root}(t_i)} (f_j)}$

probability of a derivation $P(d = t_1 \circ \dots \circ t_n) = \prod_{i=1}^n (P(t_i))$

probability of a parse $P(p = \hat{d}_1 = \dots = \hat{d}_n) = \sum_{i=0}^n (P(d_i))$

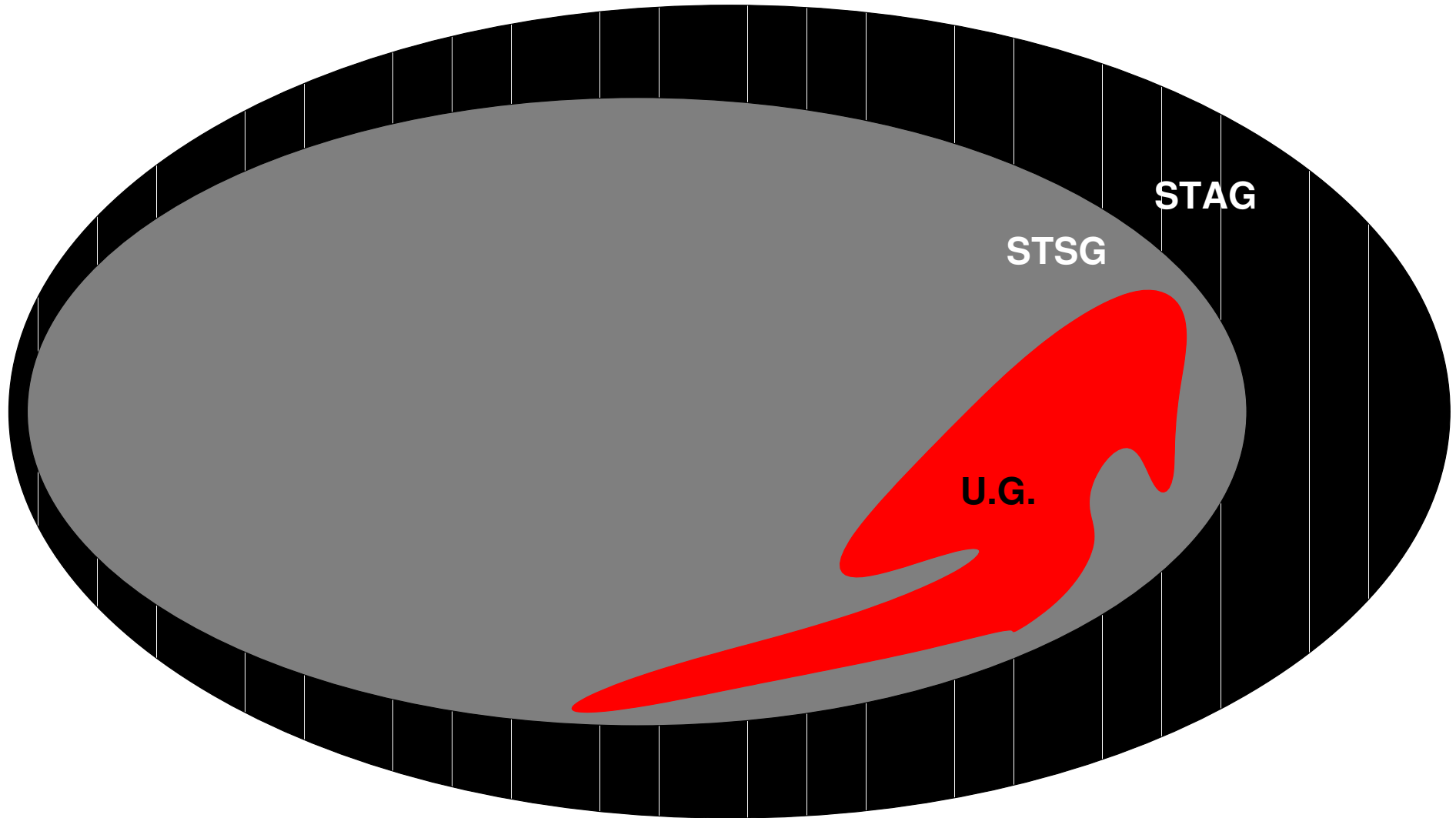
Excellent empirical results on ATIS and Penn Tree Bank

Johnson (2002) presents an example STSG for which the $E[w'_i] \neq w_i$, independent of how much data is seen: DOP1 is *biased* and *inconsistent*.



$$\begin{array}{ccccccc}
G & \xrightarrow{\text{production}} & L & \xRightarrow{\text{induction}} & G' & \xrightarrow{\text{production}} & L' & \xRightarrow{\text{induction}} & \dots & \xrightarrow{\text{production}} & G'' & \xrightarrow{\text{production}} & L'' \\
\begin{pmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{pmatrix} & & \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_M \end{pmatrix} & & \begin{pmatrix} w'_1 \\ w'_2 \\ \vdots \\ w'_{N'} \end{pmatrix} & & \begin{pmatrix} f'_1 \\ f'_2 \\ \vdots \\ f'_{M'} \end{pmatrix} & & \dots & & \begin{pmatrix} r''_1 \\ r''_2 \\ \vdots \\ r''_{N''} \end{pmatrix} & & \begin{pmatrix} s''_1 \\ s''_2 \\ \vdots \\ s''_{M''} \end{pmatrix}
\end{array}$$

Consistency on a superclass is not necessary



The real problem...

