Stochastic Lexical-Functional Grammars

Mark Johnson

Brown University

LFG 2000 Conference

July 2000

Overview

- What is a stochastic LFG?
- Estimating property weights from a corpus
- Experiments with a stochastic LFG
- Relationship between SLFG and OT-LFG.

Motivation: why combine grammar and statistics?

- Statistics has nothing to do with grammar: *WRONG*
- Statistics $\equiv$ inference from uncertain or incomplete data

  $\Rightarrow$ Language acquisition is a statistical inference problem

  $\Rightarrow$ Sentence interpretation is a statistical inference problem
- How can we do statistical inference over linguistically realistic representations?

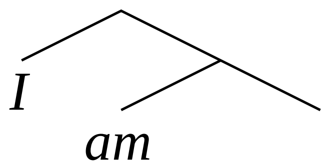
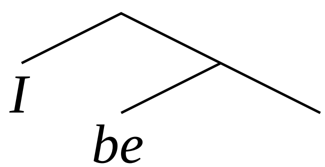
What is a Stochastic LFG?

(*stochastic* $\equiv$ *incorporating a random component*)

A Stochastic LFG consists of:

- A non-stochastic component: an LFG G, which defines Ω, the universe of input-candidate pairs
- A stochastic component: An *exponential model* over Ω
  - A finite set of *properties* or features $f_1, \ldots, f_n$.
    Each property f_i maps $x \in \Omega$ to a real number $f_i(x)$
  - Each property f_i has a *property weight* w_i.
    w_i determines how f_i affects the distribution of candidate representations

A simple SLFG

Input-candidate pairs			**Properties**		
Input	*c-structure*	*f-structure*	$f_{\star 1}$	$f_{\star \mathrm{SG}}$	f_{FAITH}
$\begin{bmatrix} \mathrm{BE}, 1, \mathrm{SG} \\ \cdots \end{bmatrix}$		$\begin{bmatrix} \mathrm{BE}, \mathbf{1}, \mathbf{SG} \\ \cdots \end{bmatrix}$	1	1	0
$\begin{bmatrix} \mathrm{BE}, 1, \mathrm{SG} \\ \cdots \end{bmatrix}$		$\begin{bmatrix} \mathrm{BE} \\ \cdots \end{bmatrix}$	0	0	1

- If $w_{\mathrm{FAITH}} < w_{\star 1} + w_{\star \mathrm{SG}}$ then *I am* is preferred
- If $w_{\star 1} + w_{\star \mathrm{SG}} < w_{\mathrm{FAITH}}$ then *I be* is preferred

(Apologies to Bresnan 1999)

Exponential probability distributions

$$\Pr(x) = \frac{1}{Z} e^{w_1 \cdot f_1(x) + w_2 \cdot f_2(x) + \ldots + w_n \cdot f_n(x)}$$

where Z is a normalization constant.

The weights w_i can be negative, zero, or positive.

- Exponential distributions have lots of nice properties
  - *Maximum Entropy* distributions are exponential
- Many familiar distributions (e.g., PCFGs, HMMs, Harmony theory) are exponential or log linear

Conditional distributions

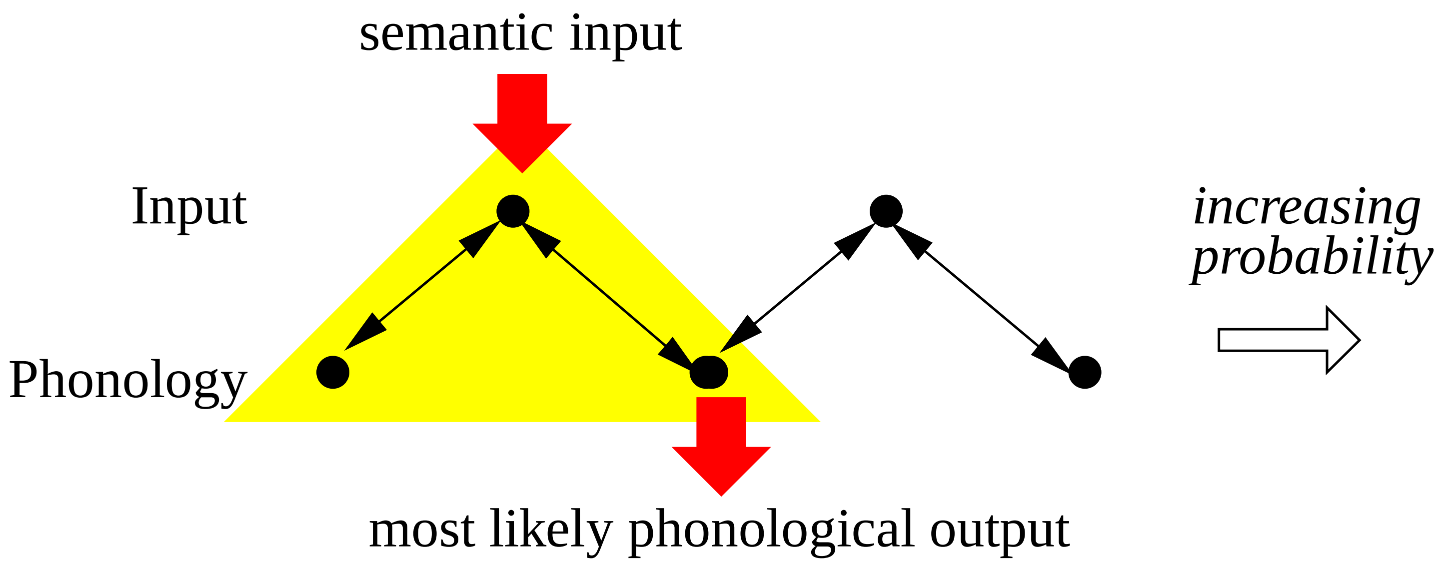
Conditional distributions tell us how likely a structure is given certain conditions.

- For *parsing*, we need to know how likely an input-candidate pair x is, *given a particular phonological string* p, i.e., $\Pr(x|Phonology = p)$
- For *generation*, we need to know how likely an input-candidate pair x is, *given a particular semantic input* s, i.e., $\Pr(x|Input = s)$

Conditional distributions

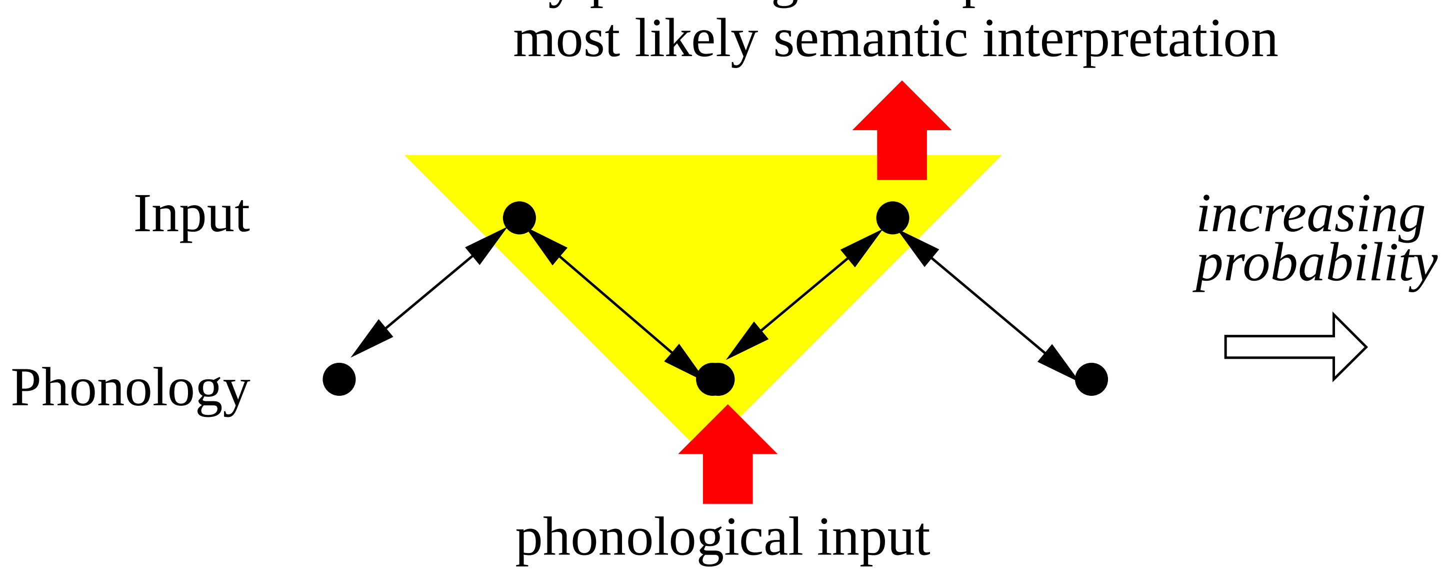
Generation

$\Pr(x|Input)$



Parsing

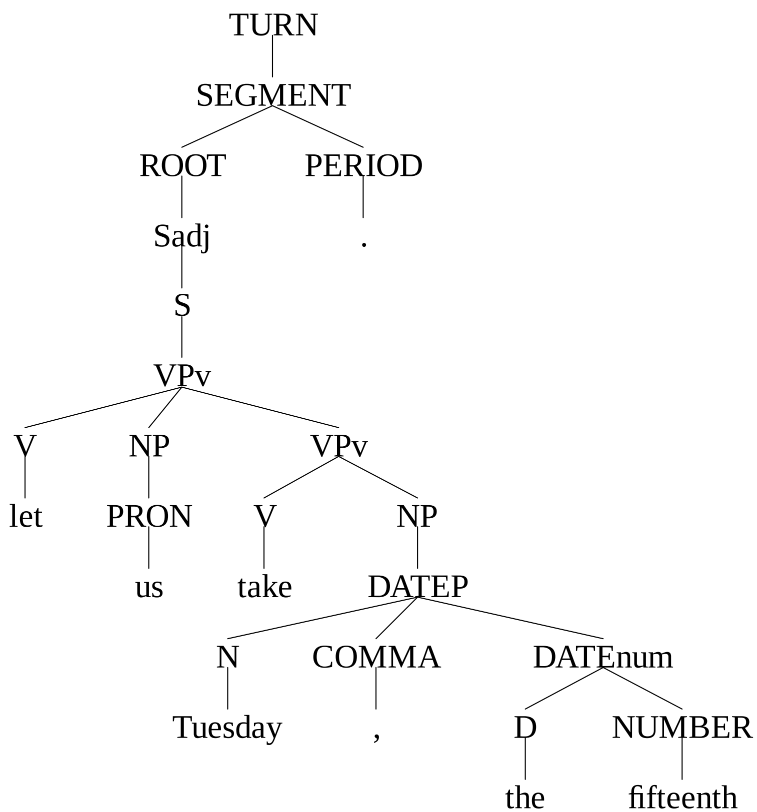
$\Pr(x|Phonology)$

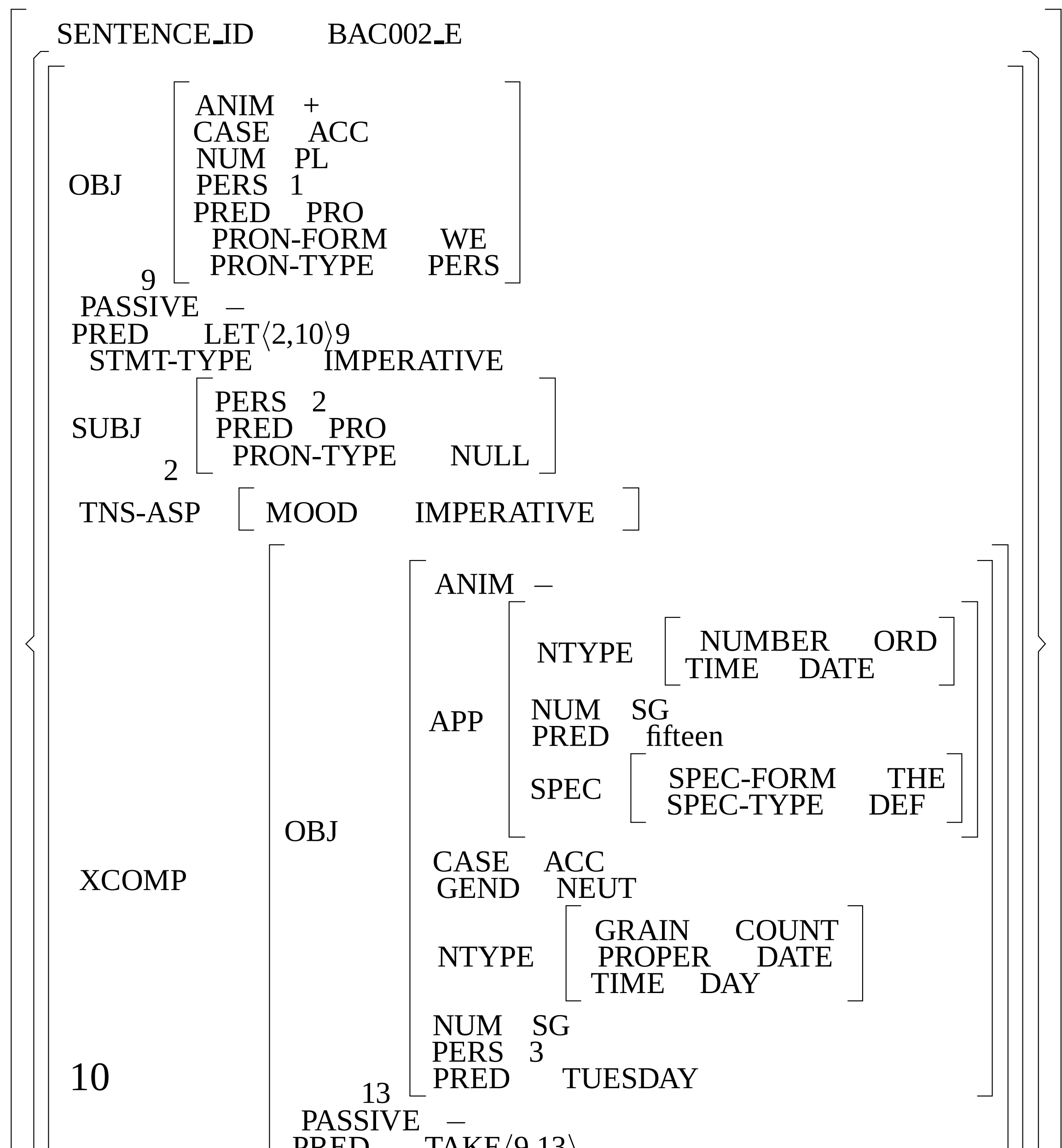


SLFG for parsing

- We used the parses of a conventional LFG (supplied by Xerox PARC)
  - On average each ambiguous sentence has 8 parses
  - Our SLFG should identify the correct one
- We wrote our own property functions
- We estimated the property weights from a hand-corrected parsed training corpus
  - The weights are chosen to maximize the *conditional probability* (pseudo-likelihood) of the correct parses given the phonological strings (Johnson et. al. 1999)

Sample parses





Property functions

- The property functions can be any (efficiently computable) function of the candidate representations
- If the grammar is a CFG then estimating property weights is simple if the property functions count rule use
- If the grammar is not a CFG, then the simple estimator that works for PCFGs is *inconsistent* (Abney 1998)
- OT constraints can be used as property functions
- c/f-str fragments can be used as property functions, yielding consistent LFG-DOP estimators (B. Cormons)

The property functions we used

Rule properties: For every non-terminal N, $f_N(x)$ is the number of times N occurs in c-structure of x

Attribute value properties: For every attribute a and every atomic value v, $f_{a=v}(x)$ is the number of times the pair $a = v$ appears in x

Argument and adjunct properties: For every grammatical function g, $f_g(x)$ is the number of times g appears in x

Additional property functions

Non-rightmost phrases: $f_{NR}(x)$ is the number of c-structure phrasal nodes that have a right sibling. (Right association)

Coordination parallelism: $f_{C_i}(x), i = 1, \ldots, 4$ is the number of coordinate structures in x that are parallel to depth i

Consistency of dates, times, locations: $f_D(x)$ is the number of non-date subphrases in date phrases. Similarly for times and locations.

Additional property functions

Lexical dependency properties: For all predicates p_1, p_2 and grammatical functions g, $f_{\langle p_1, g, p_2 \rangle}(x)$ is the number of times the head of p_1's g function is p_2.

For example, in *Al ate George's pizza*, $f_{\langle \text{eat,OBJ,pizza} \rangle} = 1$.

- Our LFG training corpus was too small to estimate the lexical dependency property weights
- We developed a method for incorporating property weights that are estimated in other ways (Johnson et. al. 2000)
- Lexical properties were not very useful with English data, but they were useful with German data

Stochastic LFG experiment

- Two parsed LFG corpora provided by Xerox PARC
- Grammars unavailable, but corpus contains all parses and hand-identified correct parse
- Properties chosen by inspecting Verbmobil corpus only

	Verbmobil corpus	**Homecentre corpus**
# of sentences	540	980
# of ambiguous sentences	324	424
Av. amb. sentence length	13.8	13.1
# of amb. parses	3245	2865
# of nonlexical properties	191	227
# of rule properties	59	57

SLFG parsing performance evaluation

	Verbmobil corpus 324 sentences		Homecentre corpus 424 sentences	
	C	$-\log PL$	C	$-\log PL$
Random	88.8	533.2	136.9	590.7
SLFG	180.0	401.3	283.25	580.6

- Corpus only contains ambiguous sentences; 10-fold cross-validation scores
- C is the number of maximum likelihood parses of held-out test corpus that were the correct parses
- PL is the conditional probability of the correct parses
- Combined system performance: 75% of MAP parses are correct

Further Extensions

- **Expectation maximization:**
  A technique for estimating property weights from corpora which *do not indicate which parse is correct* (Riezler et. al. 2000)

- **Automatic property selection:**
  New property functions are constructed "on the fly" based on the most useful current properties, and incorporated into the SLFG only if they are useful.

Research question: can these two techniques be combined?

Trading hard for soft constraints

- Many linguistic dependencies can be expressed either as *a hard grammatical constraint* or as *a soft stochastic property*
- Advantages of using stochastic properties
  - *greater robustness:* more sentences can be interpreted
  - *property weights can be automatically learnt* but not the underlying LFG

Generality of the approach

- Approach extends to *virtually any theory of grammar*
  - The universe of candidate representations is defined by a grammar (LFG, HPSG, P&P, Minimalist, etc.)
  - Property functions map candidate representations to numbers (OT constraints, parameters, etc.)
  - A learning algorithm estimates property weights from a corpus (parameter values)

SLFG and OT-LFG are closely related

OT constraints interact via strict domination, while SLFG properties do not.

- Let $F = \{f_1, \ldots, f_m\}$ be a set of OT constraints. F is *strictly bounded* iff $f_j(x) < c$, for all $f_j \in F$ and $x \in \Omega$
- **Observation:** If the OT constraints F are strictly bounded then for any constraint ordering $f_1 \gg \ldots \gg f_m$ there are property weights so that the exponential distribution on properties $f_1, \ldots, f_m$ satisfies:

$$x \text{ is more optimal than } x' \Leftrightarrow \Pr(x) > \Pr(x')$$

English auxiliaries (Bresnan 1999)

Input: [1 SG]

			*PL, *2	FAITH	*SG, *1, *3
☞	‘am’:	[1 SG]			**
	‘art’:	[2 SG]	*!	*	*
	‘is’:	[3 SG]		*!	**
	???:	[1 PL]	*!	*	*
	???:	[2 PL]	*!*	*	
	???:	[3 PL]	*!	*	*
	‘are’:	[ ]		*!	

Emergence of the unmarked

Input: [2 SG]

		*PL, *2	FAITH	*SG, *1, *3
	‘am’: [1 SG]		*	*!*
	‘art’: [2 SG]	*!		*
	‘is’: [3 SG]		*	*!*
	???: [1 PL]	*!	*	*
	???: [2 PL]	*!*	*	
	???: [3 PL]	*!	*	*
☞	‘are’: [ ]		*	

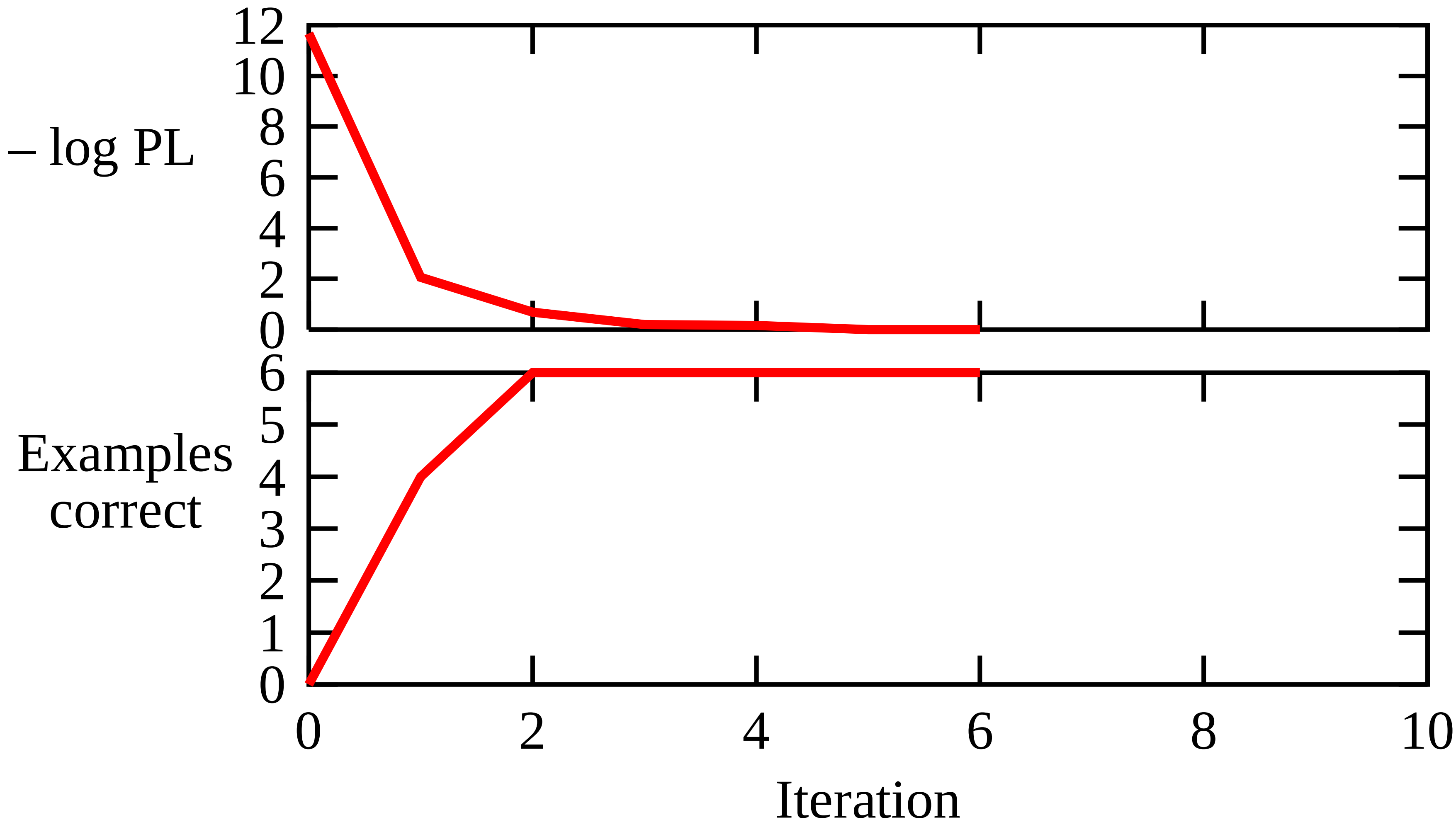
Input to OT and SLFG learners

Constraints: $[f_{\star 1}, f_{\star 2}, f_{\star 3}, f_{\star \mathrm{SG}}, f_{\star \mathrm{PL}}, f_{Faith}]$

Optimal x_i	Suboptimal competitors $\Omega_i - \{x_i\}$
[1 SG] – 'am' : [1 0 0 1 0 0]	[1 SG] – 'art' : [0 1 0 1 0 1], [1 SG] – 'are' : [0 0 0 0 0 1], .
[2 SG] – 'are' : [0 0 0 0 0 1]	[2 SG] – 'art' : [0 1 0 1 0 0], [2 SG] – 'is' : [0 0 1 1 0 1], . . .
[3 SG] – 'is' : [0 0 1 1 0 0]	[3 SG] – 'am' : [1 0 0 1 0 1], [3 SG] – 'are' : [0 0 0 0 0 1], .
. . .	. . .

- **OT learner:** find a constraint ordering so each x_i is more optimal than its competitors Ω_i
- **SLFG learner:** find weights that maximize the conditional probability of x_i given its competitors Ω_i

PL estimation of "Standard English"

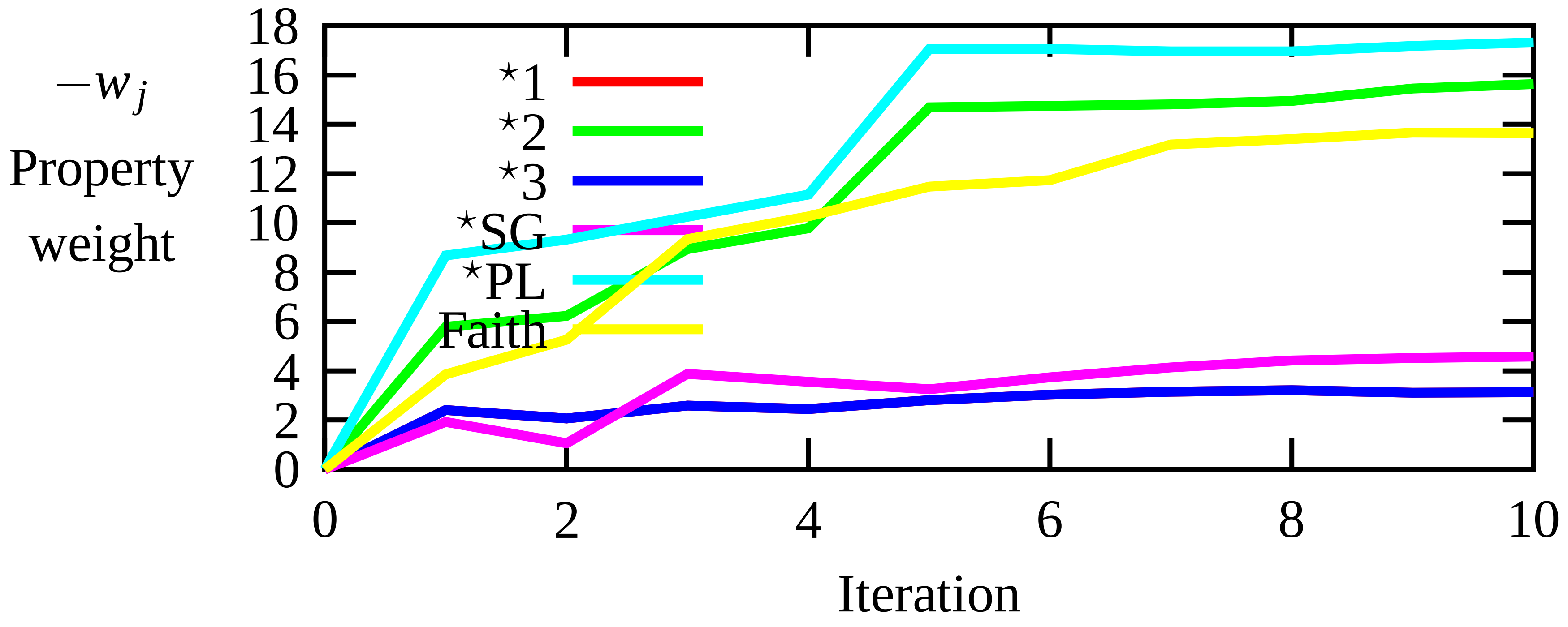


“Standard English” property weights

I am	we are
you are	you are
she is	they are

Bresnan: $\star$PL, $\star$2 $\gg$ FAITH $\gg$ $\star$SG, $\star$1, $\star$3

SLFG: $\star$PL $>$ $\star$2 $>$ FAITH $>$ $\star$SG $>$ $\star$1 $=$ $\star$3

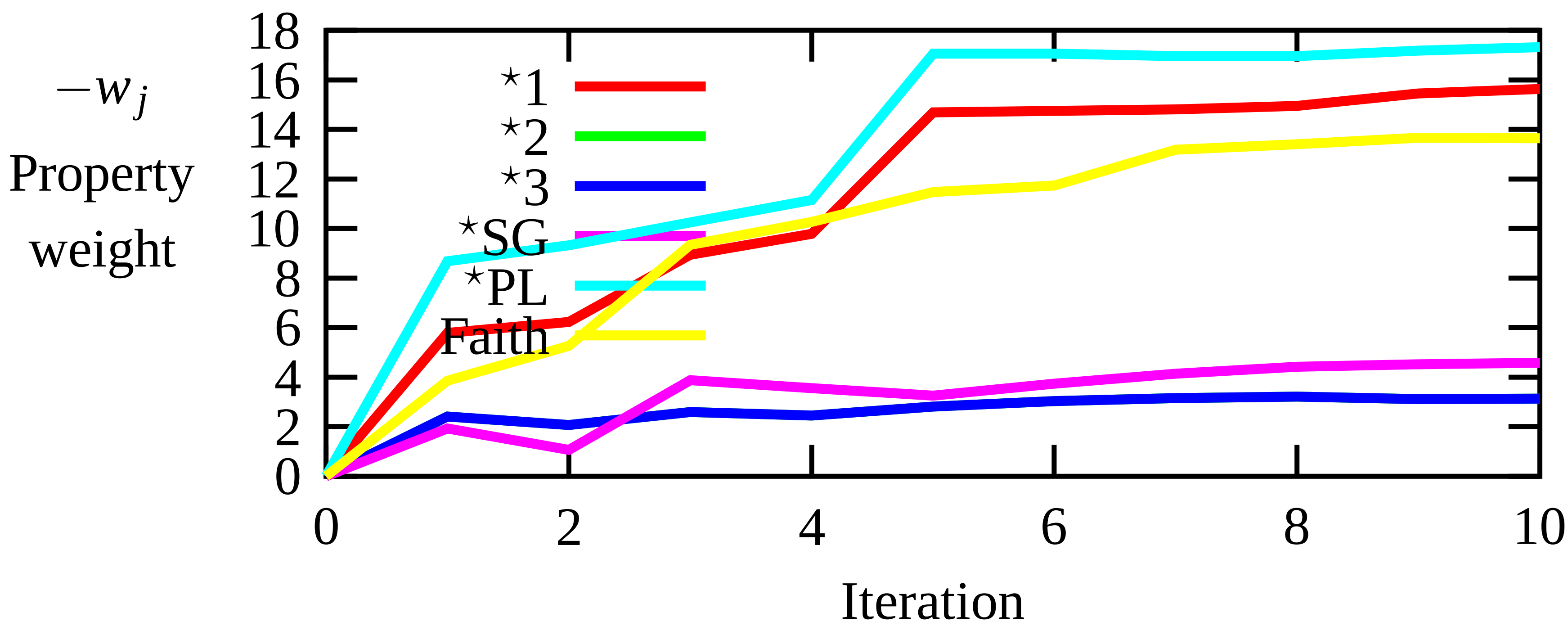


Somerset English property weights

be	be
art	be
is	be

Bresnan: *PL, *1 $\gg$ FAITH $\gg$ *SG, *2, *3

PL: *PL $>$ *1 $>$ FAITH $>$ *SG $>$ *2 $=$ *3

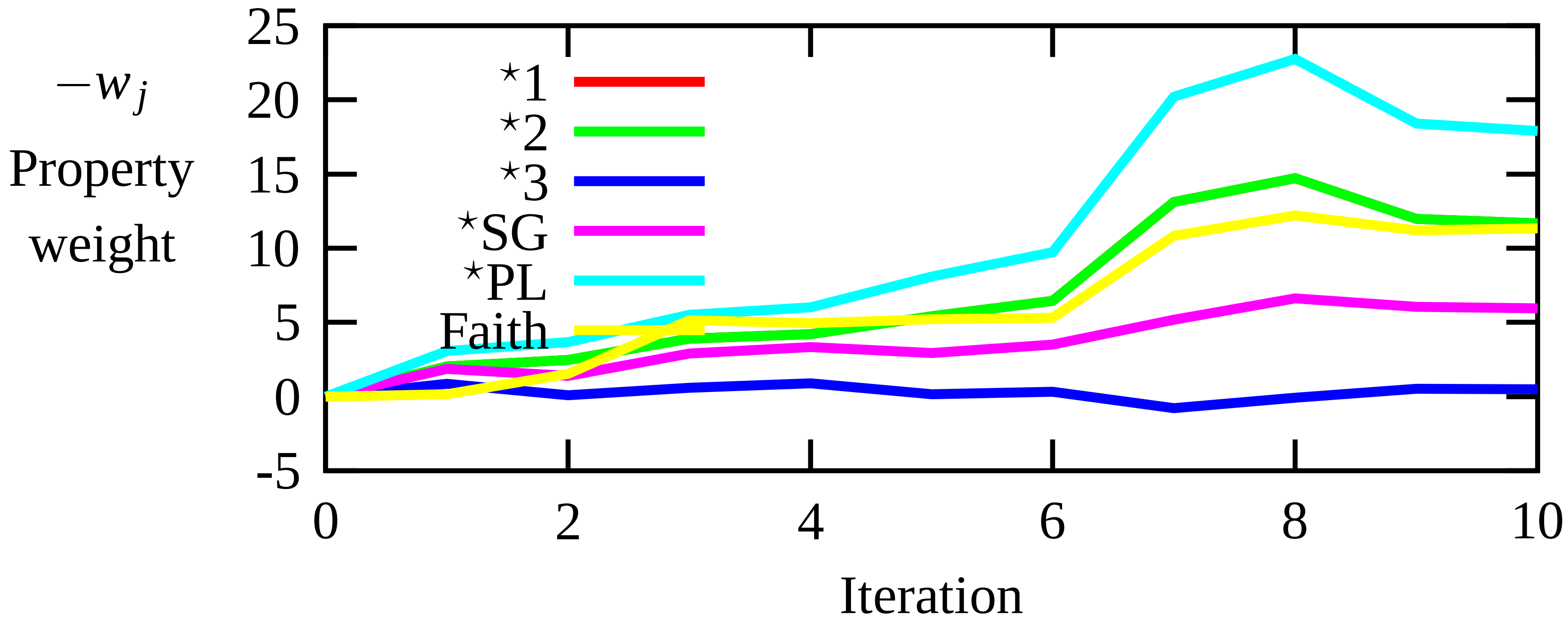


Southern and East Midlands

are	are
are	are
is	are

Bresnan: *PL, *1, *2 ≫ FAITH ≫ *SG, *3

PL: *PL > *1 = *2 ≈ FAITH > *SG > *3



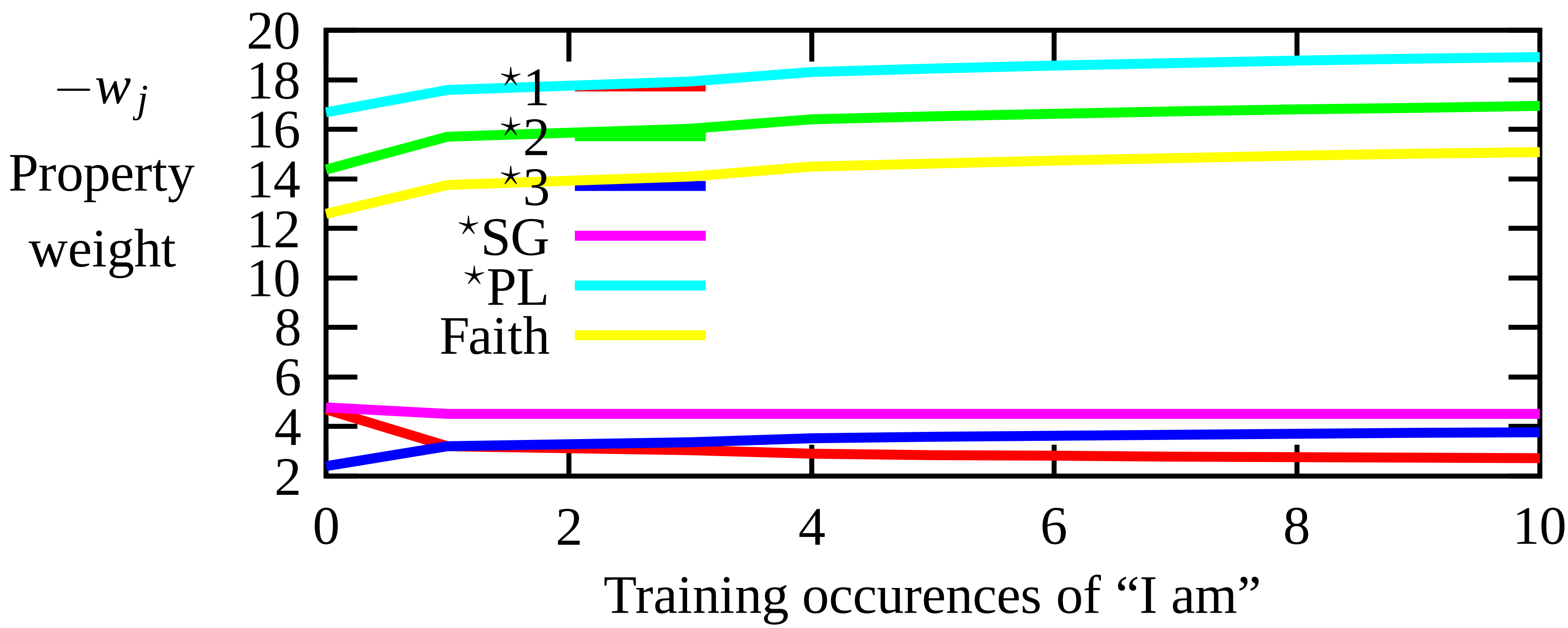
Effect of frequency on weights

I am	we are
you are	you are
she is	they are

Bresnan: *PL, *2 $\gg$ FAITH $\gg$ *SG, *1, *3

0 “I am”: *PL $>$ *2 $>$ FAITH $>$ *SG $>$ *1 $>$ *3

10 “I am”: *PL $>$ *2 $>$ FAITH $>$ *SG $>$ *3 $>$ *1



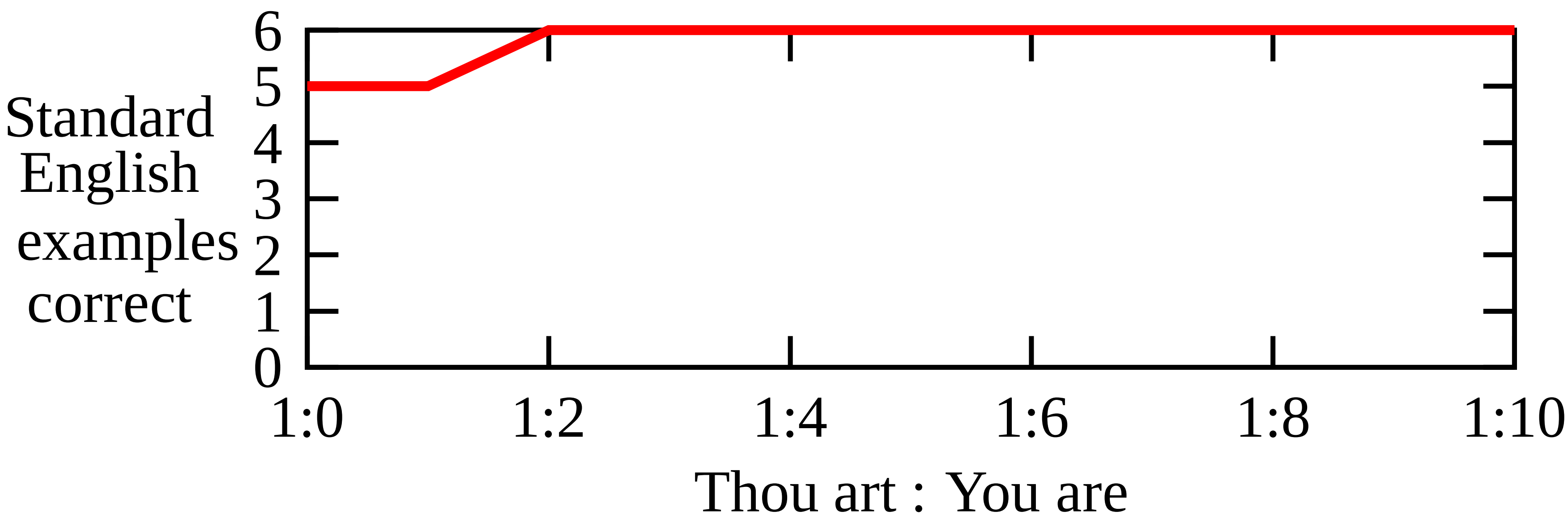
Learning from inconsistent data

are	are
art	are
is	are

are	are
are	are
is	are

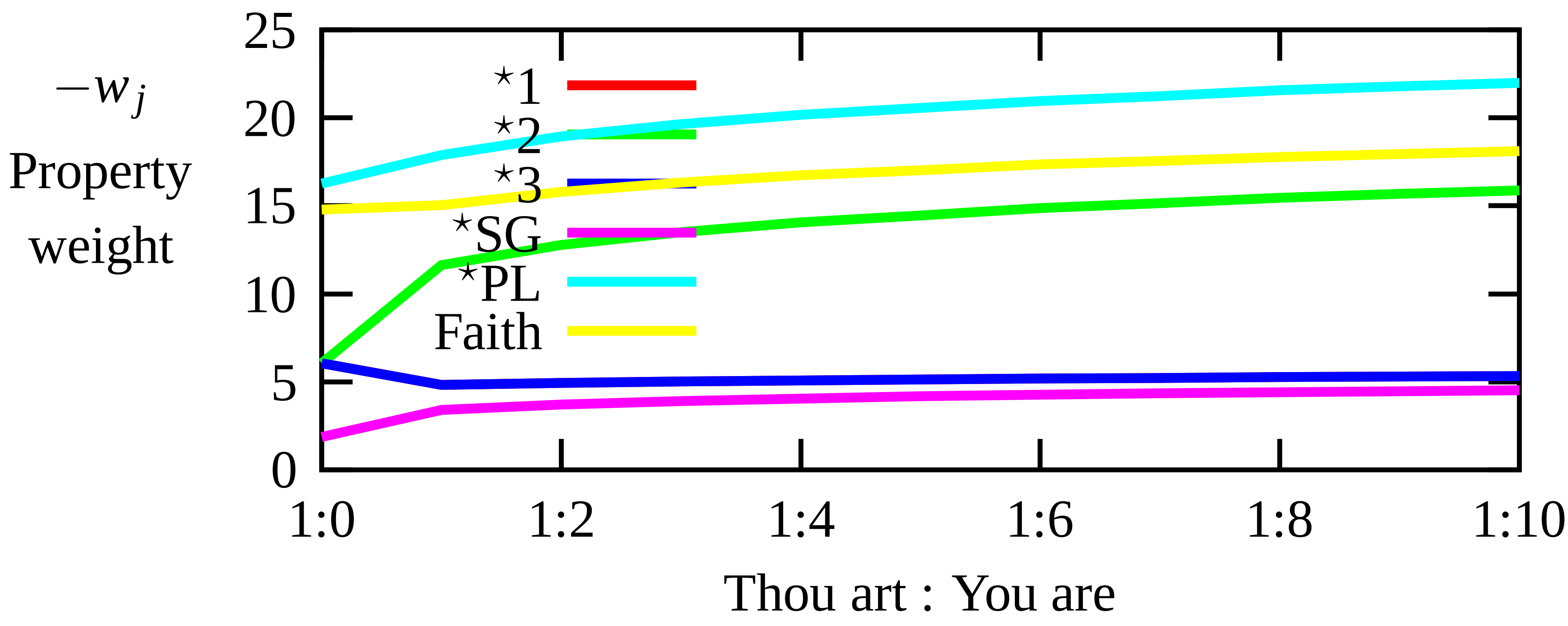
*PL ≫ FAITH ≫ *SG, *1, *2, *3

*PL, *2 ≫ FAITH ≫ *SG, *1, *3



Learning from inconsistent data

am	are	am	are	*PL $\gg$ FAITH $\gg$ *SG, *1, *2, *3
art	are	are	are	*PL, *2 $\gg$ FAITH $\gg$ *SG, *1, *3
is	are	is	are	*PL $>$ FAITH $>$ *2 $>$ *1 $=$ *3 $>$ *SG



Conclusions

- Statistical methods can be applied to realistic linguistic representations!
- Statistical methods can improve parser accuracy
- Statistical methods can be used to study language acquisition
- OT and exponential models are closely related
- Statistical estimation may be more robust to noisy data than current OT learners

http://www.cog.brown.edu/~mj

Acknowledgements: This work is supported by 3 NSF awards, including an NSF *Integrated Graduate Education Research and Training Award.*

Selected References:

S. Abney (1997) “Stochastic Attribute-Value Grammars”. *Computational Linguistics* 23.4, 597–617.

M. Johnson, S. Geman, S. Canon, Z. Chi and S. Riezler (1999) “Estimators for Stochastic ‘Unification-Based’ Grammars”. *Proc. 37th ACL*, 535–541.

M. Johnson and S. Riezler (2000) “Exploiting Auxiliary distributions in Stochastic Unification-Based Grammars”. *Proc. 1st NAACL*, 154–161.

S. Riezler, D. Prescher, J. Kuhn and M. Johnson “Lexicalized Stochastic Modelling of Constraint-Based Grammars using Log-Linear Measures and EM Training”, to appear *Proc ACL 2000*.