Inversion Transduction Grammars

Wilker Aziz
5/4/16

Discussion

- IBM models do not constrain mismatch in word order
- Distortion step must consider

  all the *m*! permutations

  of *m* French words

All permutations: sensible or not?

If we do not impose structural constraints
(but they do exist)

- the model will have to learn (rather *implicitly*) how to not violate them
- which ought to require more data

Practical consequences

Practical consequences

Estimation

- modelling outcomes that even though possible are not plausible (unlikely to be observed)

Practical consequences

Estimation

- modelling outcomes that even though possible are not plausible (unlikely to be observed)

Generation

- NP-completeness!

All permutations

Let $\Sigma_n = \{a_1, ..., a_n\}$

- $S \rightarrow A_{\Sigma_n}$
- $A_X \rightarrow a\ A_{X-\{a\}}$ for $\#X \geq 2$
- $A_{\{a\}} \rightarrow a$

Regular grammar (there is an equivalent FSA)

Asveld (2006, 2008)

Complexity

Note that nonterminals are indexed by subsets of Σ_n

i.e. power set of Σ

- 2^n nonterminals (states)
- $n \times 2^n$ productions (transitions)
- $n!$ strings (paths)

Example: 3 elements

$S \rightarrow A_{123}$

$A_{123} \rightarrow a_1\ A_{23} \mid a_2\ A_{13} \mid a_3\ A_{23}$

$A_{12} \rightarrow a_1\ A_2 \mid a_2\ A_1$

$A_{13} \rightarrow a_1\ A_3 \mid a_3\ A_1$

$A_{23} \rightarrow a_2\ A_3 \mid a_3\ A_2$

$A_1 \rightarrow a_1$

$A_2 \rightarrow a_2$

$A_3 \rightarrow a_3$

"IBM constraint"

Distortion limit in **generation** but not in **estimation**

- any reasons why that may be unsatisfactory?

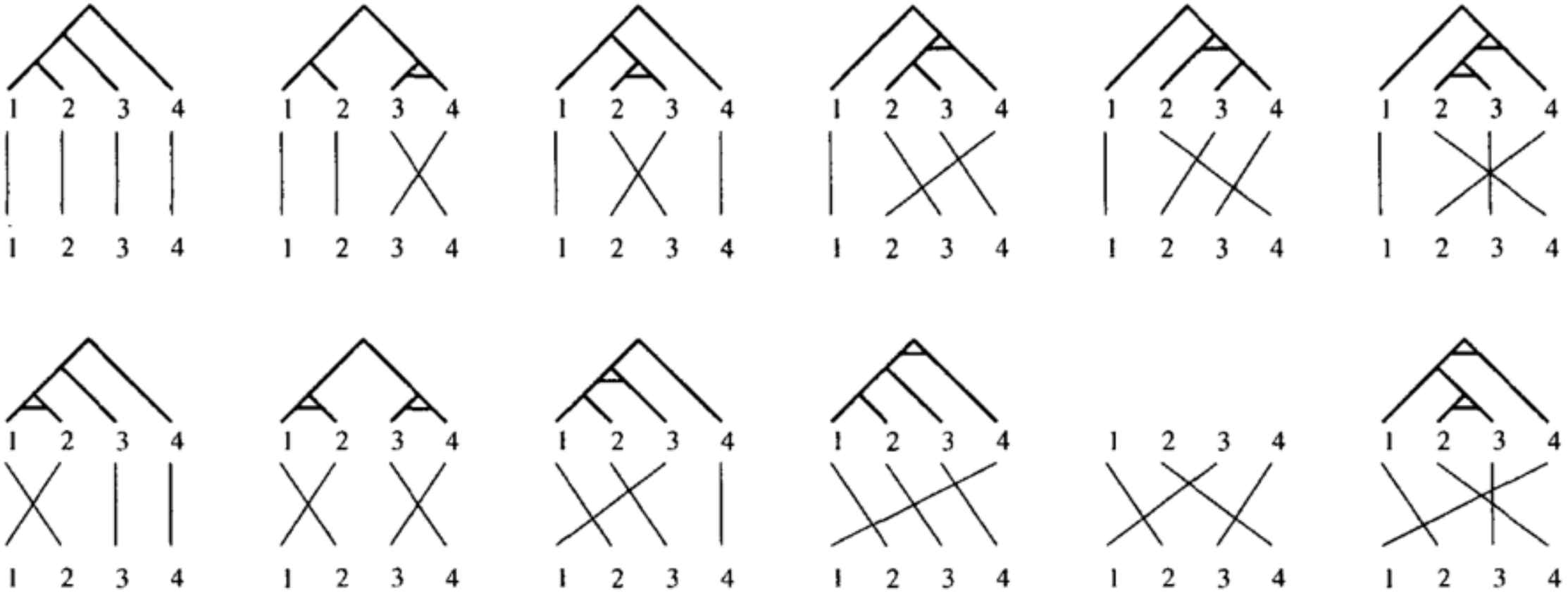
	Inference	Generation
1	Exact	Local search (and distortion limit)
2	Exact	
≥3	Approximate	

Constraining permutations without a distortion limit

Inversion Transduction Grammars (ITGs)

- Binarizable permutations
  - two streams are simultaneously generated
  - context-free structure

Wu (1995, 1997)



Wu (1997)

ITG

ITG

English French

ITG

	English	French	
S →	X	X	copy

ITG

	English	French	
S →	X	X	copy
X →	X_1 X_2	X_1 X_2	copy

ITG

	English	French	
S →	X	X	copy
X →	X_1 X_2	X_1 X_2	copy
	X_2 X_1		invert

ITG

	English	French	
S →	X	X	copy
X →	X_1 X_2	X_1 X_2	copy
		X_2 X_1	invert
X →	e	f	transduce

ITG

	English	French	
S →	X	X	copy
X →	X_1 X_2	X_1 X_2	copy
		X_2 X_1	invert
X →	e	f	transduce
X →	e	ε	delete

ITG

	English	French	
S →	X	X	copy
X →	X_1 X_2	X_1 X_2	copy
		X_2 X_1	invert
X →	e	f	transduce
X →	e	ε	delete
X →	ε	f	insert

Model

Model

Joint probability model P(E, F, A)

Model

Joint probability model P(E, F, A)

- Multinomial: one parameter per rule

Model

Joint probability model P(E, F, A)

- Multinomial: one parameter per rule
  - $\theta_{[]}$ one parameter for **monotone** copy

Model

Joint probability model P(E, F, A)

- Multinomial: one parameter per rule
  - $\theta_{[]}$ one parameter for **monotone** copy
  - $\theta_{<>}$ one parameter for copy in **inverted** order

Model

Joint probability model P(E, F, A)

- Multinomial: one parameter per rule
  - $\theta_{[]}$ one parameter for **monotone** copy
  - $\theta_{<>}$ one parameter for copy in **inverted** order
  - $\theta_{e/f}$ one parameter per **word pair**

Model

Joint probability model P(E, F, A)

- Multinomial: one parameter per rule
  - $\theta_{[]}$ one parameter for **monotone** copy
  - $\theta_{<>}$ one parameter for copy in **inverted** order
  - $\theta_{e/f}$ one parameter per **word pair**
  - $\theta_{e/\varepsilon}$ one parameter per **English** word

Model

Joint probability model P(E, F, A)

- Multinomial: one parameter per rule
  - $\theta_{[]}$ one parameter for **monotone** copy
  - $\theta_{<>}$ one parameter for copy in **inverted** order
  - $\theta_{e/f}$ one parameter per **word pair**
  - $\theta_{e/\varepsilon}$ one parameter per **English** word
  - $\theta_{\varepsilon/f}$ one parameter per **French** word

MLE

We do not typically construct treebanks of ITG trees

- potential counts instead of counts

$$\theta_{X\to\alpha} = \frac{\langle n(X \to \alpha)\rangle_{P(A|F,E)}}{\sum_{\alpha'}\langle n(X \to \alpha)\rangle_{P(A|F,E)}}$$

Expectations from parse forests

- Inside-Outside [Baker, 1979; Lari and Young, 1990; Goodman, 1999]

Typically initialised with IBM1

Difficulties

- Complexity: $O(l^3m^3)$
- Too few reordering parameters
  - try and compare to IBM2

Bibliography

- Asveld, Peter R. J. 2006. Generating All Permutations by Context-free Grammars in Chomsky Normal Form. In *Theoretical Computer Science*. Elsevier Science Publishers Ltd.

- Asveld, Peter R. J. 2008. Generating All Permutations by Context-free Grammars in Greibach Normal Form. In *Theoretical Computer Science*. Elsevier Science Publishers Ltd.

- Wu, D. 1995. An Algorithm for Simultaneously Bracketing Parallel Texts by Aligning Words. In *Proceedings of the 33rd Annual Meeting of the Association for Computational Linguistics*. ACL.

- Wu, D. 1997. Stochastic Inversion Transduction Grammars and Bilingual Parsing of Parallel Corpora. In *Computational Linguistics*. MIT Press.

- James K. Baker. 1979. Trainable grammars for speech recognition. In *Proceedings of the Spring Conference of the Acoustical Society of America*.

- Karim Lari and Steve J. Young. 1990. The estimation of stochastic context-free grammars using the inside--outside algorithm. In *Computer Speech and Language*.

- Goodman, Joshua. 1999. Semiring parsing. In Computational Linguistics.