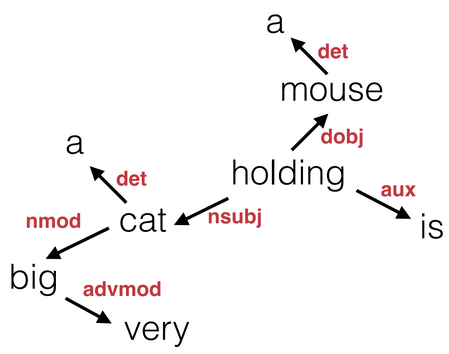
Incremental generative model of sentence linearization and word order variation

Kristina Gulordava
University of Geneva

Frank Keller
University of Edinburgh

Language production and sentence linearization

underlying representation → utterance



"a very big cat is holding a mouse"

Task: unordered dependency tree → order of words

Research goal: a cognitive model

How sentence linearization proceeds word-by-word?

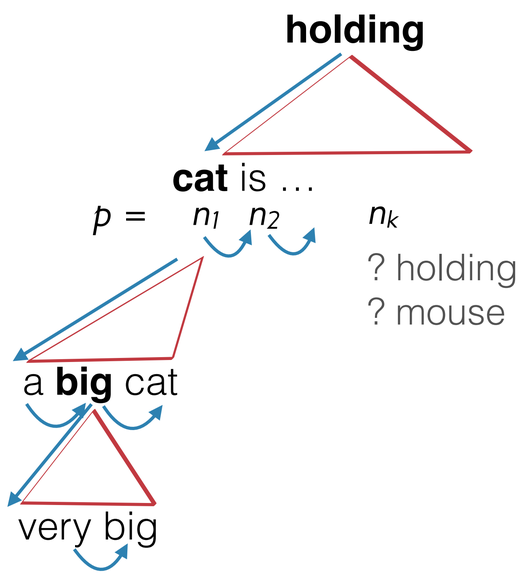
Incrementality

- Speakers don't plan the whole utterance in advance
- What is a plausible degree of incrementality? word-by-word, chunk-by-chunk

Probabilistic nature

- Speakers have access to a probabilistic grammar (e.g. for processing)
- How these probabilities are used in generation?

Linearization process



Recursive procedure

- tree is traversed **top-down**
- each set of *head + immediate children* is ordered **independently**

Greedy choice of the next node in each set results in ~~a~~ word-by-word ~~incremental~~ linearization

- **score function** defines which node is ~~chodes~~ given previously output nodes *p* and the remaning nodes in the set

Probabilistic score function

$$score(p) = \prod_{n_i \in p} score(n_i, p) \cdot \prod_{n_j \notin p} score'(n_j, p)$$

generation score *for nodes in p*

$$P(n_i | n_1 \ldots n_{i-1}) \cdot P(left | n_i, head)$$

ngram *probability* · ***direction*** *probability*

future score *for remaning nodes*

if *head* is in the output nodes *p:*

$$score'(n_j, p) = P(right | n_j, head)$$

else: $$max \begin{matrix} P(left | n_j, head) \\ P(right | n_j, head) \end{matrix}$$

We estimate the **unlexicalized** probabilities from a treebank:

- conditioned on *dependency label*, *part-of-speech tag* (no token information)
- ngram probabilities are estimated as *trigrams*; no smoothing

Re-ranking with size features

Modelling word order variation cases

A cat is staring [at a poor little mouse] [with a hungry look]
A cat is staring [with a hungry look] [at a poor little mouse]

- two alternative grammatical orders with the same semantics, i.e. unordered dependency trees
- some relevant features: **sizes of the phrases**
- **c**hoice between two options *(..., staring, mouse) vs (.., staring, look)* can be modelled as a discriminative re-ranking at each linearization step

Data and set-up

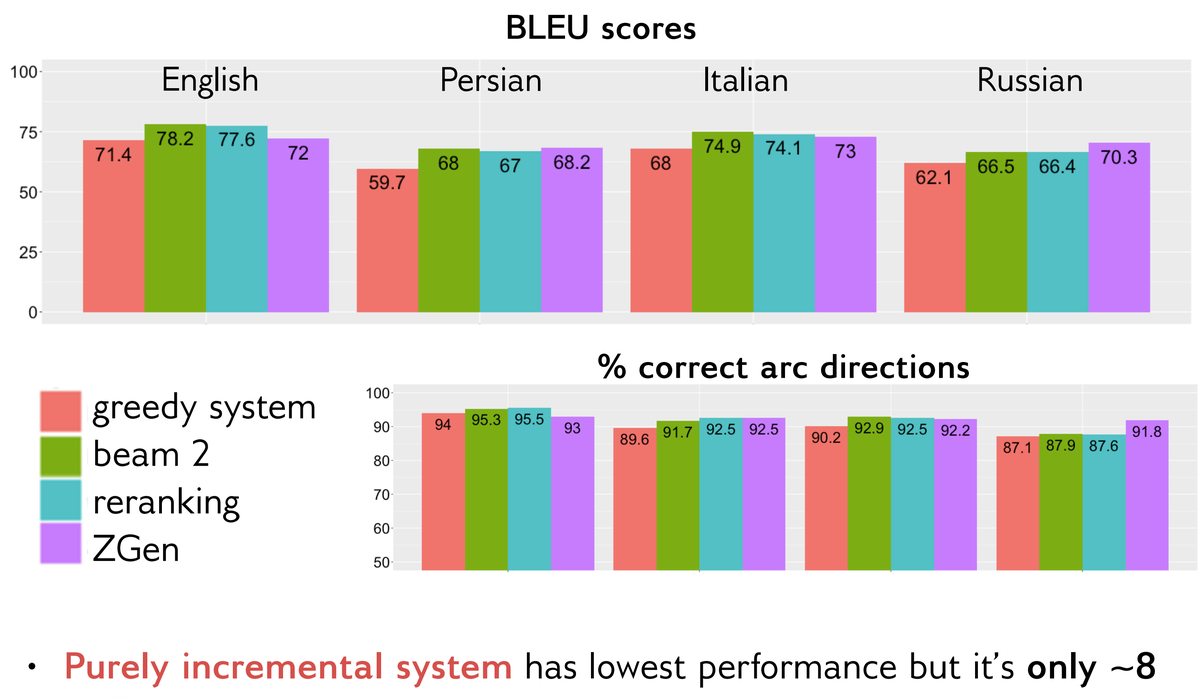
Four UD treebanks: English, Italian, Persian, Russian (development sets)

Pre-processing: only sentences without punctuation,

Point of comparison: **ZGen** (Zhang et al, 2012, 2015) - incremental, transition-based linearization system; lexicalized, **uses large beam (64)**

Measures: **BLEU** and **% of arcs having correct direction**

Results



- Purely incremental system has lowest performance but it's **only ~8 BLEU points** ~~away~~ despite its simple greedy architecture
- System which keeps **two hypotheses** at each linearization step (**beam 2**) improves the results significantly

- Reranking improves significantly over greedy system, reaching almost the performance of the system with beam 2
  - discriminative information in terms of two best nodes is crucial
  - it confims that size features play role in choosing better word orders

Conclusions

We can reach competitive performance using a cognitively plausible architecture with greedy search, probabilistic score function and unlexicalized features