Reordering

Philipp Koehn

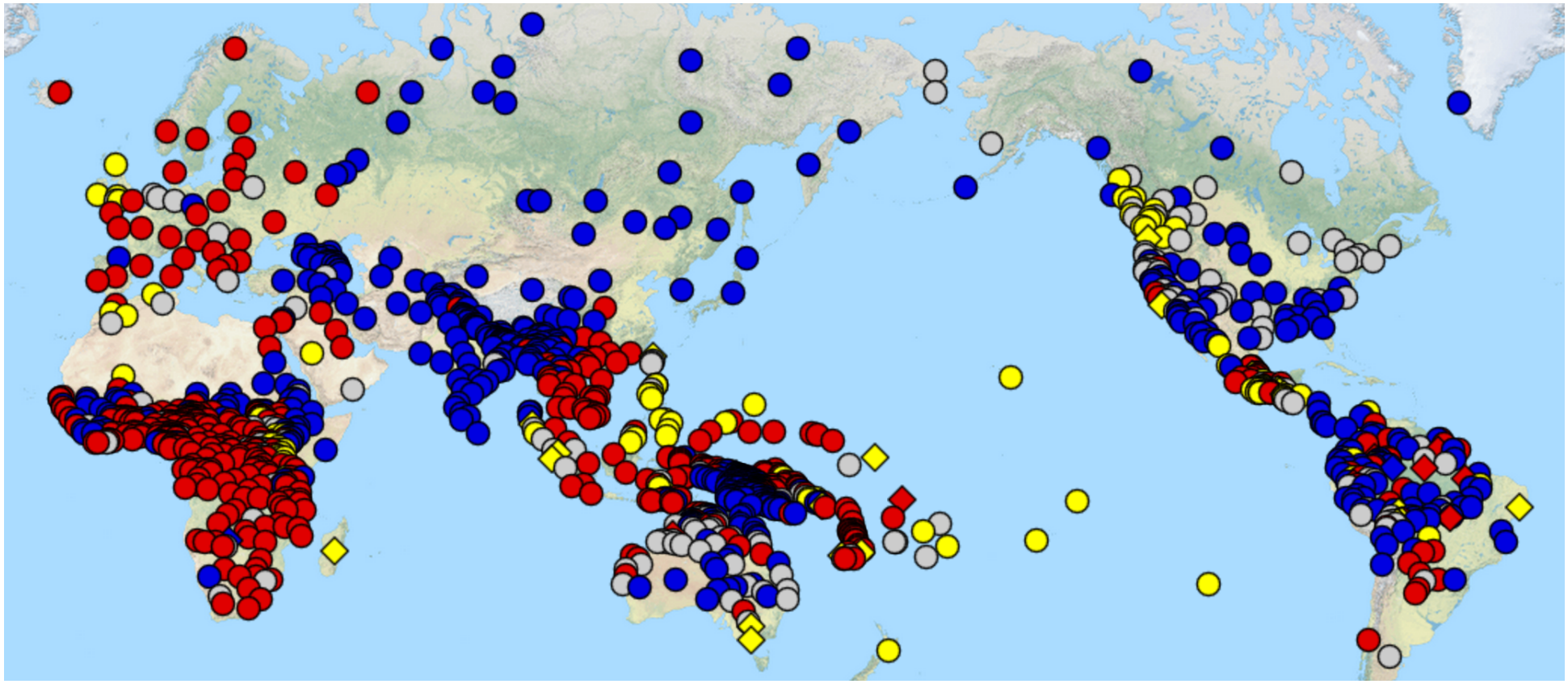
31 October 2017

Why Word Order?

- Language has words to name
  - things (nouns)
  - actions (verbs)
  - properties (adjectives, adverbs)
- Function words help to glue sentences together
- Word order also helps to define relationships between words

differences in word order

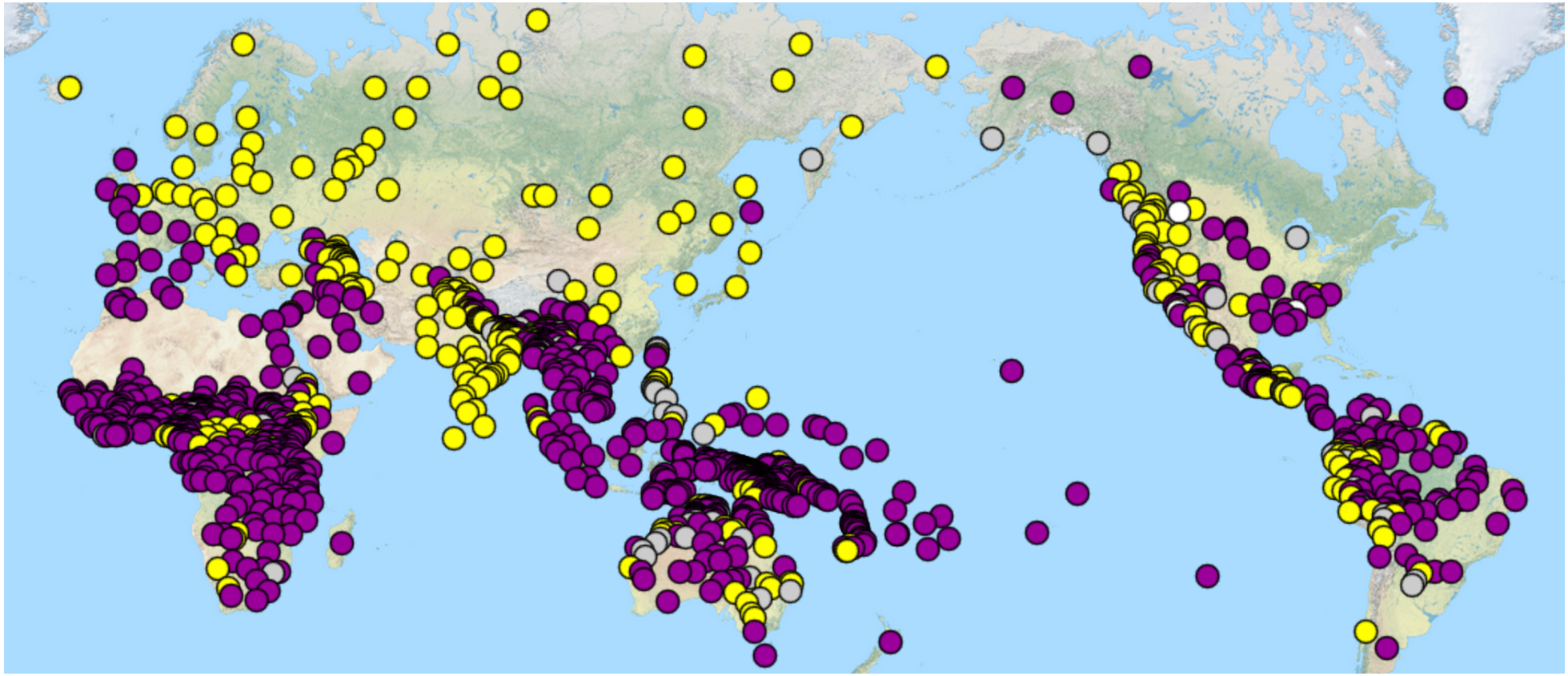
Subject, Verb, Object



● SOV (565 languages) ● SVO (488) ● VSO (95) ♦ VOS (25) ♦ OVS (11) ♦ OSV (4)

Source: World Atlas of Language Structures http://wals.info/

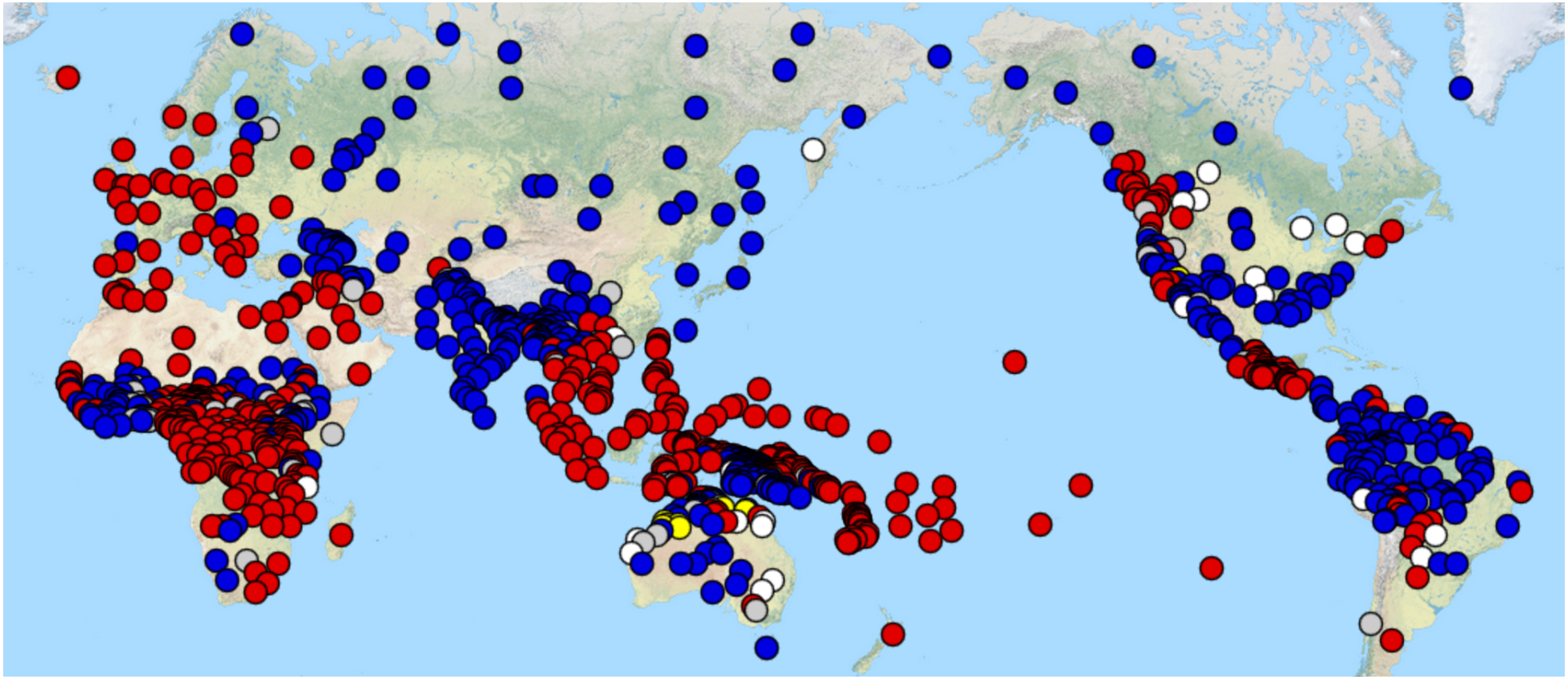
Adjective, Noun



● Adj-N (373 languages) ● N-Adj (878) ● no dominant order (110)

Source: World Atlas of Language Structures http://wals.info/

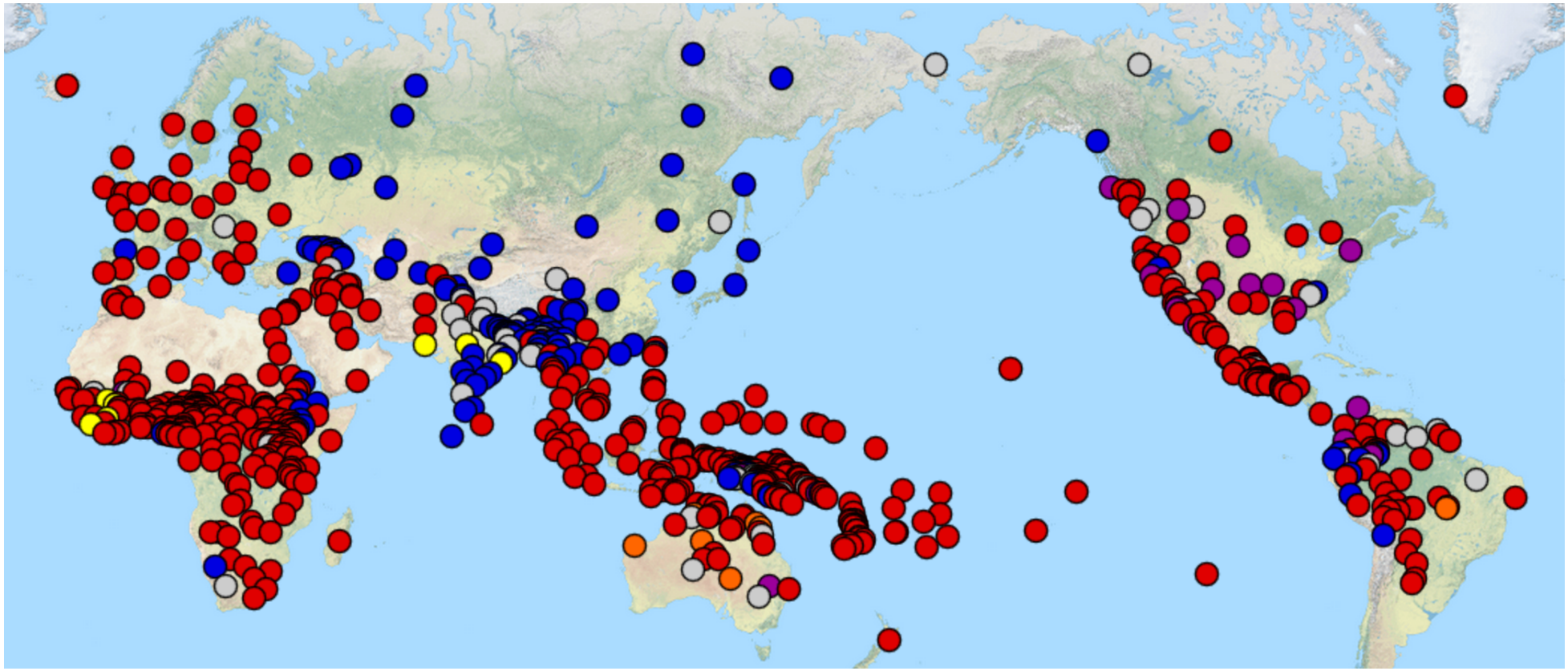
Adposition, Noun Phrase



● postposition (576 languages) ● preposition (511)
● inposition (8) ● no dominant order (58)

Source: World Atlas of Language Structures http://wals.info/

Noun, Relative Clause



● N-Rel (579 languages) ● Rel-N (141) ● internally headed (24)

Source: World Atlas of Language Structures http://wals.info/

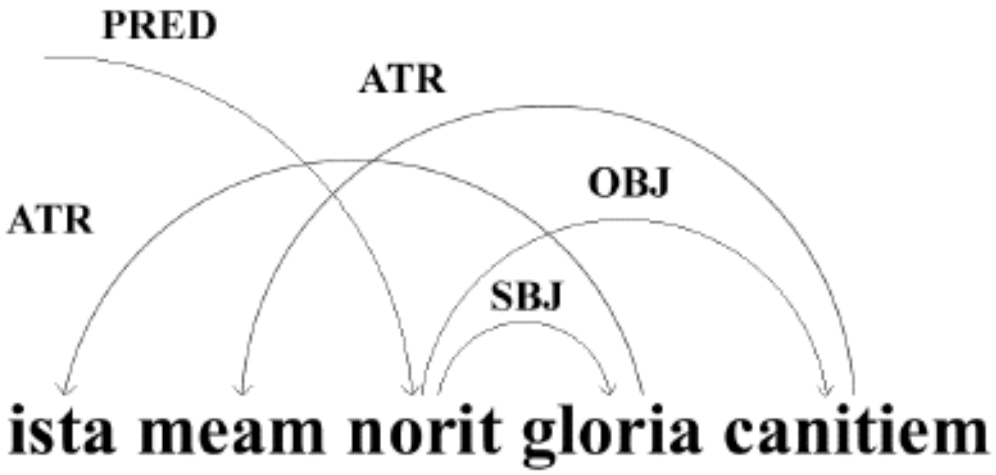
Free Word Order

- Sometimes the word order is not fixed
- The following German sentences mean the same:

  Der Mann gibt der Frau das Buch.
  Das Buch gibt der Mann der Frau.
  Der Frau gibt der Mann das Buch.
  Der Mann gibt das Buch der Frau.
  Das Buch gibt der Frau der Mann.
  Der Frau gibt das Buch der Mann.

- Placing of content words allows for nuanced emphasis
- Role of noun phrases (subject, object, indirect object) handled by morphology

Non-Projectivity



this my will-know glory old-age

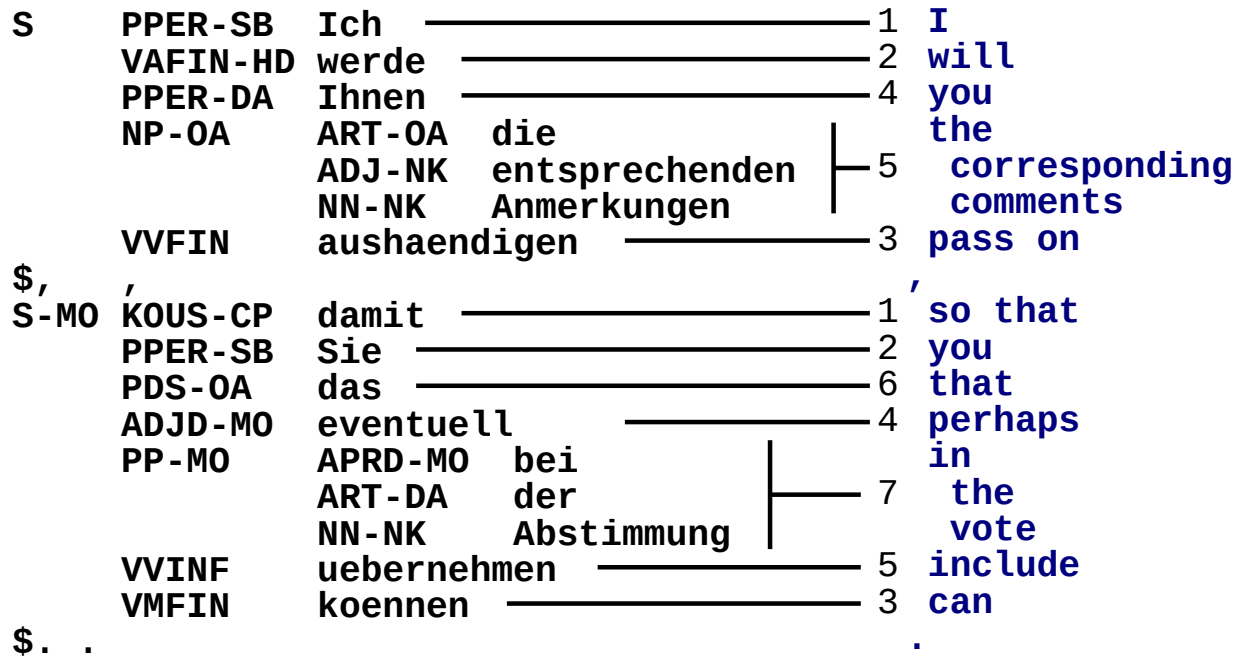
- Non-projectivity = crossing dependencies in a dependency parse
- Sentence does not decompose into contiguous phrases
- Latin example
  - NP meam ... canitiem = my old-age
  - NP ista ... gloria = that glory

pre-reordering rules

Hand-Written Reordering Rules

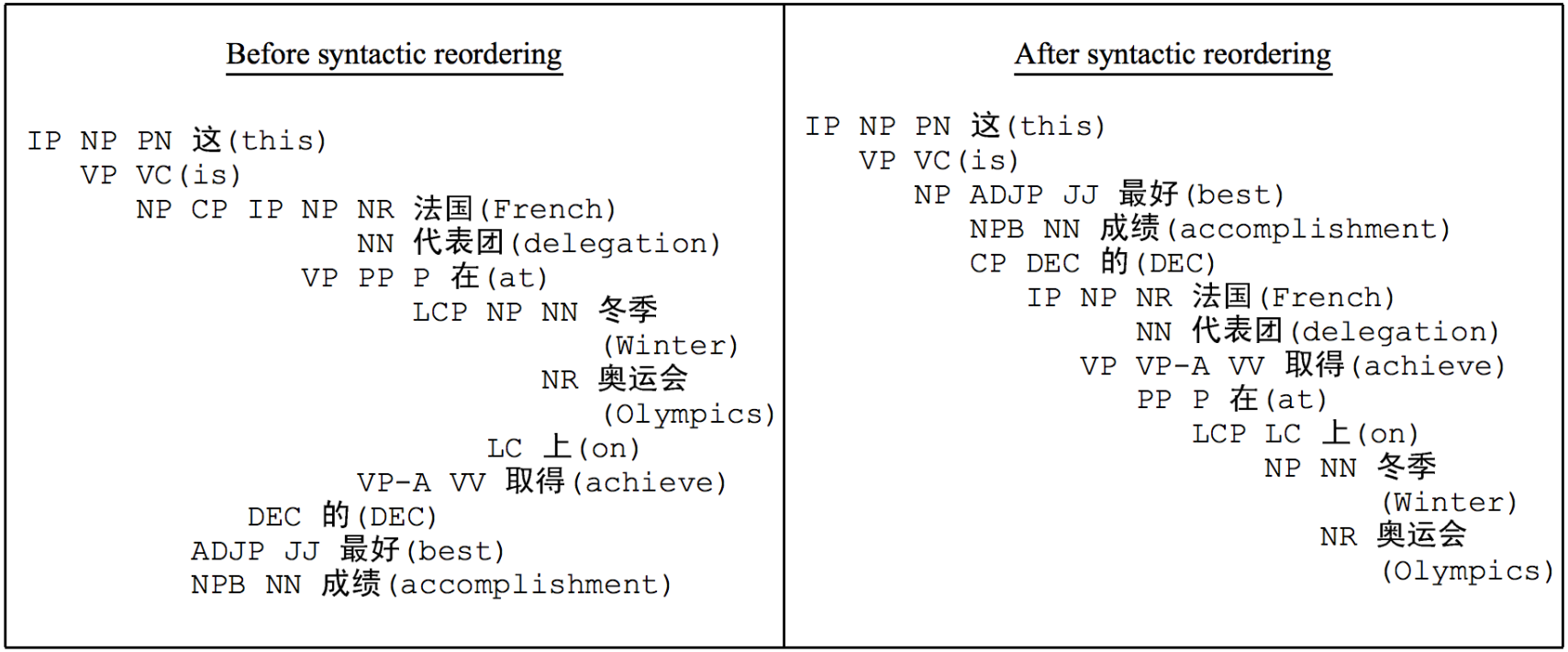
- Differences between word orders are syntactic in nature
- Simple hand-written rules may be enough
- Preprocessing: reorder source sentence into target sentence order
  - parse the source sentence
  - apply rules
- Preprocess both training and test data

German–English



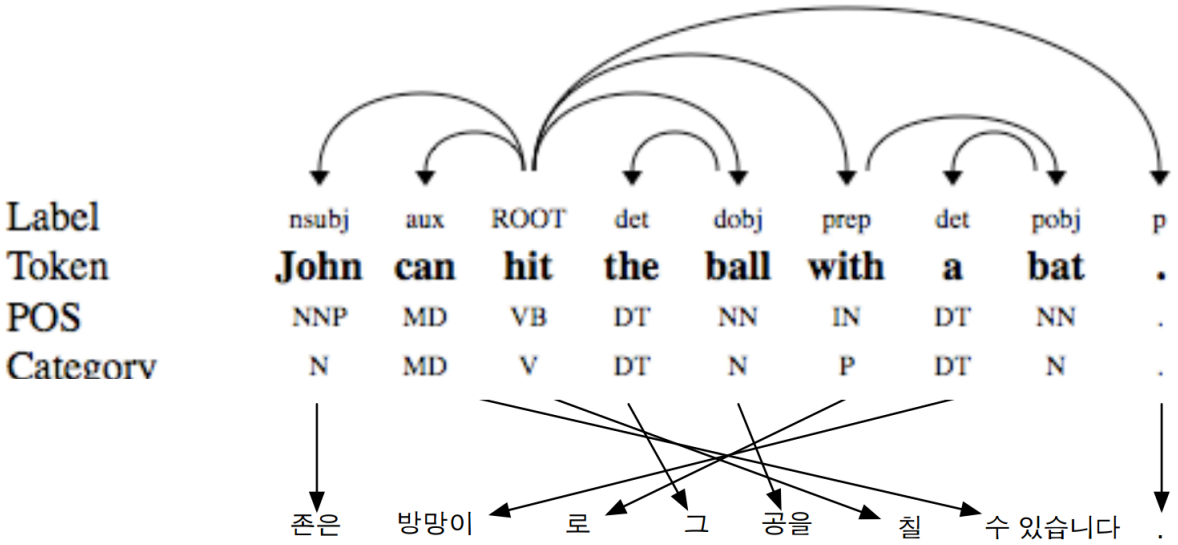
- Apply a sequence of reordering rules
  1. in any verb phrase move head verbs into initial position
  2. in sub-ordinate clauses, move the (main verb) directly after complementizer
  3. in any clause, move subject directly before head
  4. move particles in front of verb
  5. move infinitives after finite verbs
  6. move clause-level negatives after finite verb

Chinese–English



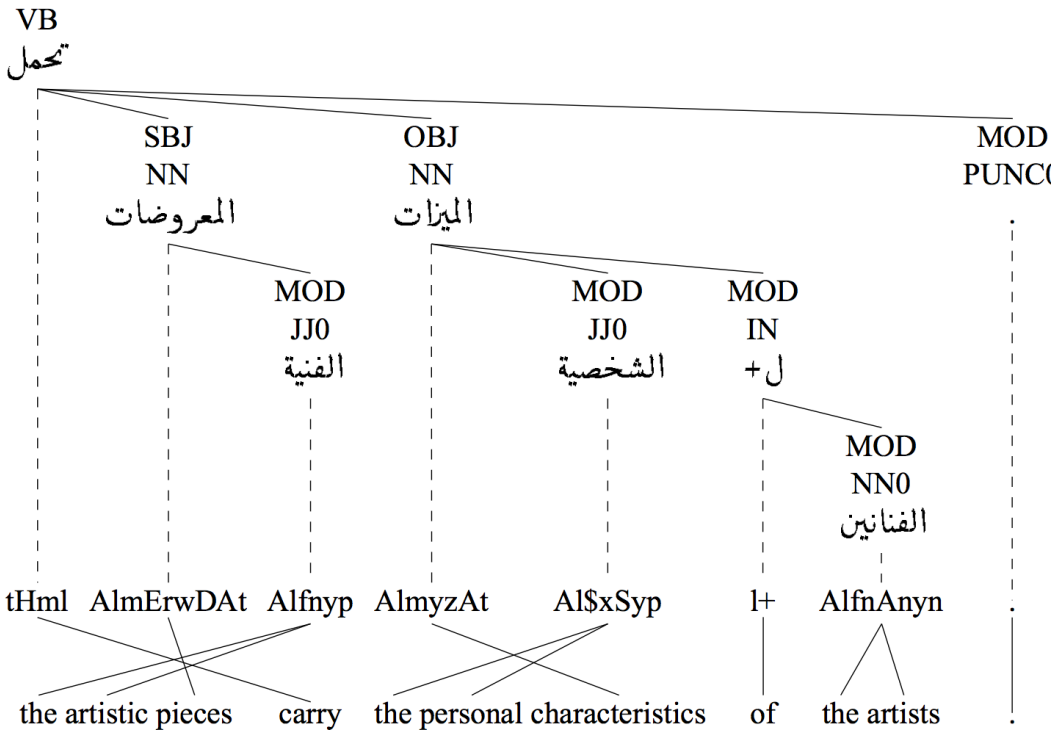
- Reordering based on constituent parse
  - PP modifying a VP are moved after it
  - temporal NP modifying a VP are moved after it
  - PP and relative clauses (CP) modifying NPs are moved after it
  - postpositions are moved in front of monied NP

English–Korean



- Based on dependency parse, group together dependents of verbs (VB*)
  - phrasal verb particle (prt)
  - auxiliary verb (aux)
  - passive auxiliary verb (auxpass)
  - negation (neg)
  - verb itself (self) together
- Reverse their positions and move them to the end of the sentence
- Same reordering also works for Japanese, Hindi, Urdu, and Turkish

Arabic–English



- Three main types of reordering
  - verb subjects may be: (a.) pro-dropped, (b.) pre-verbal, or (c.) post-verbal.
  - adjectival modifiers typically follow their nouns
  - clitics need to split and reordered book+his → his book

Word of Caution

- Example German sentence

Den	Vorschlag	verwarf	die	Kommission	.
the	proposal	rejected	the	commission	.

- Classic case of OVS $\rightarrow$ SVO transformation

The commission rejected the proposal.

- But a translator may prefer to restructure the sentence into passive (this keeps the German emphasis on the proposal)

The proposal was rejected by the commission.

- In actual data, evidence of even more drastic syntactic transformations to keep sentence order.

learning pre-reordering

Pre-Reordering Rules

- Reordering rules are language specific

  $\Rightarrow$ for each language pair, a linguist has to find the best ruleset

- Complex interactions between rules

  $\Rightarrow$ a specific sequence of reordering steps has to be applied

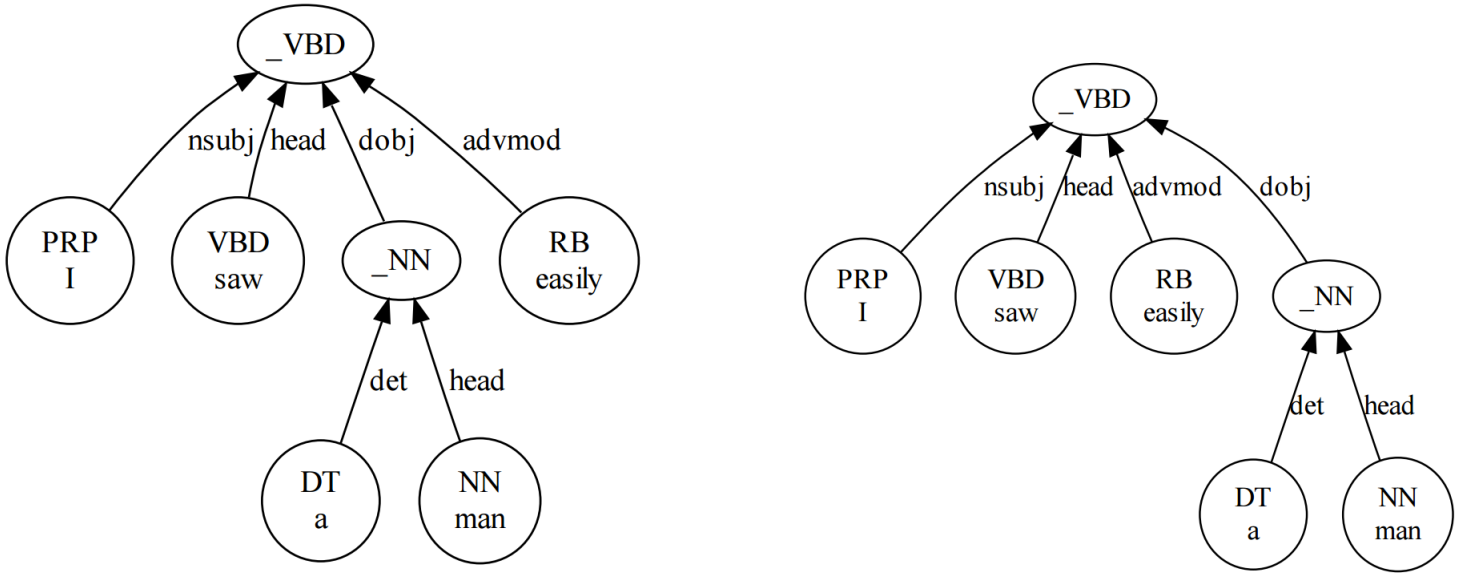
- Evaluating a reordering ruleset not straightforward

  - training an entire machine translation system too costly
  - automatically generated word alignments may be flawed
  - not many large manual word alignments available

Learning Pre-Reordering Rules

- One successful method: Genzel [COLING 2010]
- Learn a sequence of reordering rules based on dependency parse
- Rule application
  - applies to tree top-down
  - only reorder children of same node
  - rule format: conditioning context $\rightarrow$ action
- Successful across a number of language pairs
  (English to Czech, German, Hindi, Japanese, Korean, Welsh)

Types of Rules



Rule: nT=VBD, 1T=PRP, 1L=nsubj, 3L=dobj → (1,2,4,3)

- Conditioning context: conjunction of up to 5 conditions, each
  - matching POS tag (T) / syntactic label (L)
  - of current node (n), parent node (p), 1st child, 2nd child, etc.
- Action: permutation such as (1,2,4,3), i.e., reordering 3rd and 4th of 4 children

Learning Algorithm

- Greedy learning of rules

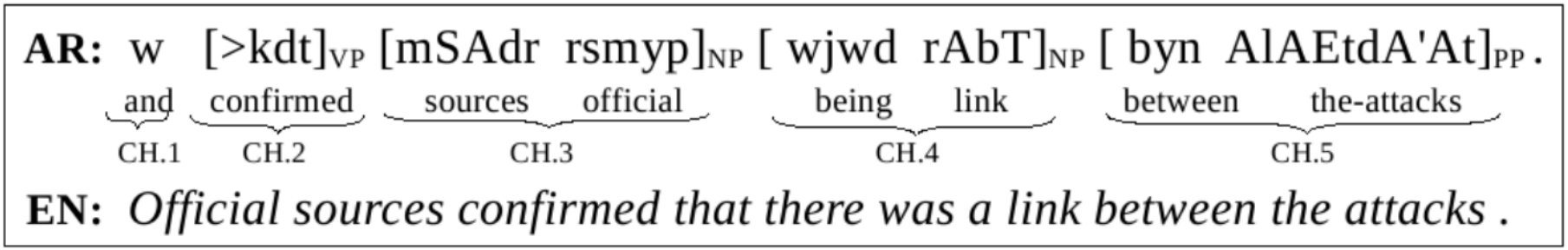
  1. start with empty sequence, un-reordered parallel corpus
  2. consider all possible rules
  3. pick the one the reduces reordering error the most
  4. append to the sequence, apply to all sentences
  5. go to step 2, until convergence

- Evaluate against IBM Model 1 word alignment

  – higher IBM Models have monotone bias
  – metric: number of crossing alignment links

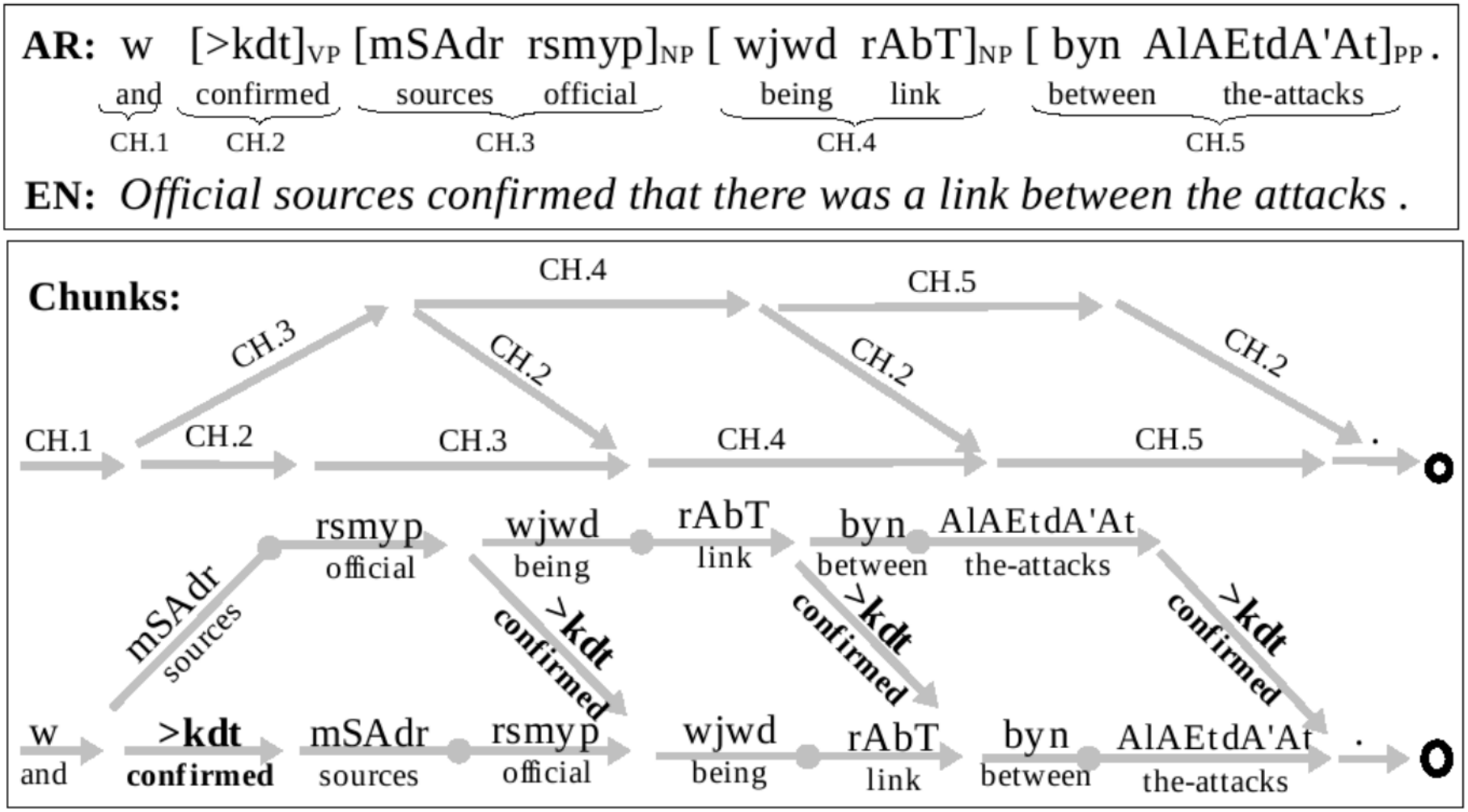
reordering lattice

Ambiguity in Arabic Verb Reordering



- Arabic is VSO, so the verb has to be moved behind the subject
- Where does the subject end?
  - subject may have modifiers (prepositional phrases)
  - pro-drop: there may not even be a subject

Encode Multiple Reorderings in Lattice



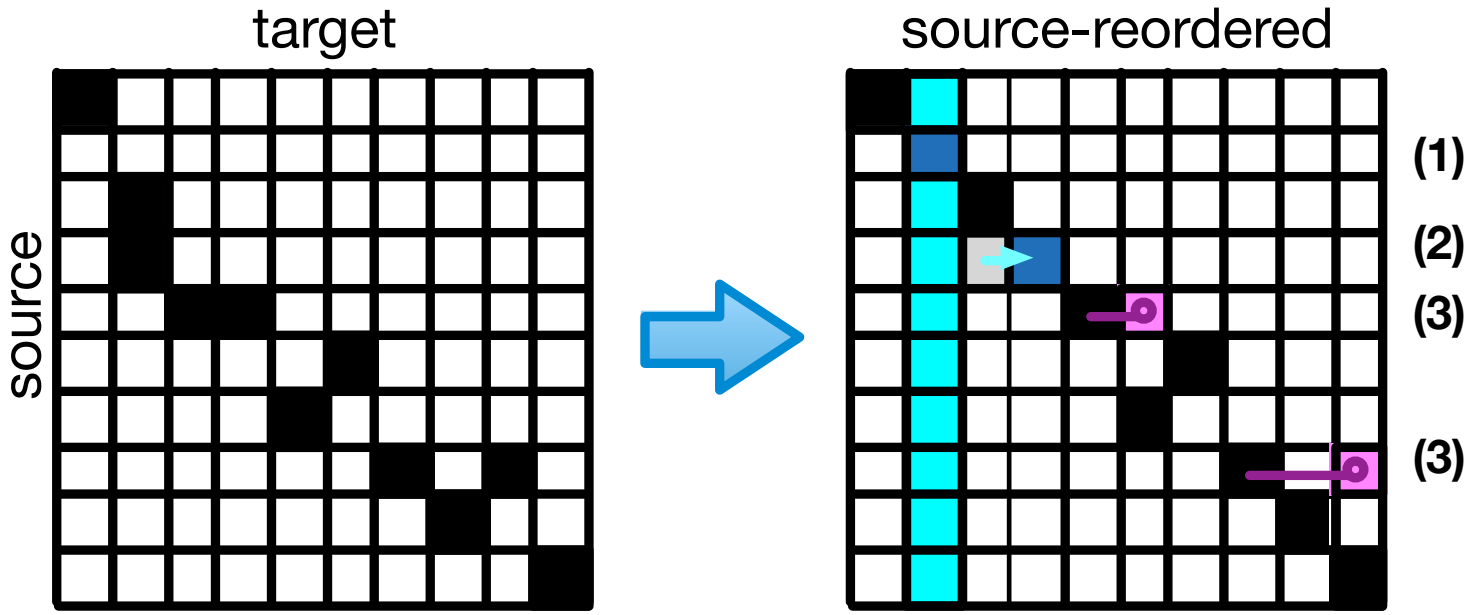
- Allow decoder explore multiple input paths

evaluation

LR Score

- BLEU not very good at measuring reordering quality
- Alignment metric that compares reordering between
  - machine translation vs. source
  - reference vs. source
- Ignores lexical accuracy

Permutations



- Convert source-target alignment to source permutation

1. unaligned source words
   → position immediately after target word position of previous source word
2. multiple source words aligned to same target word → make monotone
3. source words aligned to multiple target words → aligned to first target word

Compare MT and Reference Permutation

- Two permutations π and σ
- Hamming distance (exact match distance)

$$d_H(\pi,\sigma) = 1 - \frac{\sum_{i=1}^{n} x_i}{n} \text{ where } x_i = \begin{cases} 0 & \text{if } \pi(i) = \sigma(i) \\ 1 & \text{otherwise} \end{cases}$$

- Kendall tau distance (swap distance)

$$d_\tau(\pi,\sigma) = 1 - \frac{2}{n^2 - n} \sum_{i=1}^{n} \sum_{j=1}^{n} z_{ij}$$

$$z_{ij} = \begin{cases} 1 & \text{if } \pi(i) < \pi(j) \text{ and } \sigma(i) > \sigma(j) \\ 0 & \text{otherwise} \end{cases}$$

Combination with Lexical Score

- Reordering distance ignores lexical accuracy
- Can be combined with traditional metrics (e.g., BLEU) to form full metric
  - interpolation with BLEU

$$\text{LRscore} = \alpha R + (1 - \alpha)\text{BLEU}$$

  - reordering score includes brevity penalty

$$R = d \times \text{BP}$$

$$\text{BP} = \begin{cases} 1 & \text{if } t > r \\ e^{1-\frac{r}{t}} & \text{if } t \leq r \end{cases}$$

- Shown to correlate better with human judgment

summary

Summary

- Languages differ a lot in word order
  - anything that one language places to the left, another one places to the right
  - things that are closely related may not even be closely located
- Pre-reordering rules
  - hand-written
  - successful for many language pairs
- Learning pre-reordering rules
- Preserving ambiguity: lattices, distortion matrices
- LR Score

Other Approaches

- Lexicalized reordering models – various refinements
  - hierarchical lexicalized reordering
  - learn a maximum entropy model, not just probabilistic model
  - encode as sparse features
- Syntax-based models
  - integrate syntactic parse tree into the translation model
  - translation rules include syntactic reordering patterns