February 21 2006

Dependencies vs. Constituents for Tree-Based Alignment

Daniel Gildea

Presented by Michael Tepper

Tree-Based Alignments

- Some background.
- Motivation for dependency alignments.
- How does it work?
- How does it do?

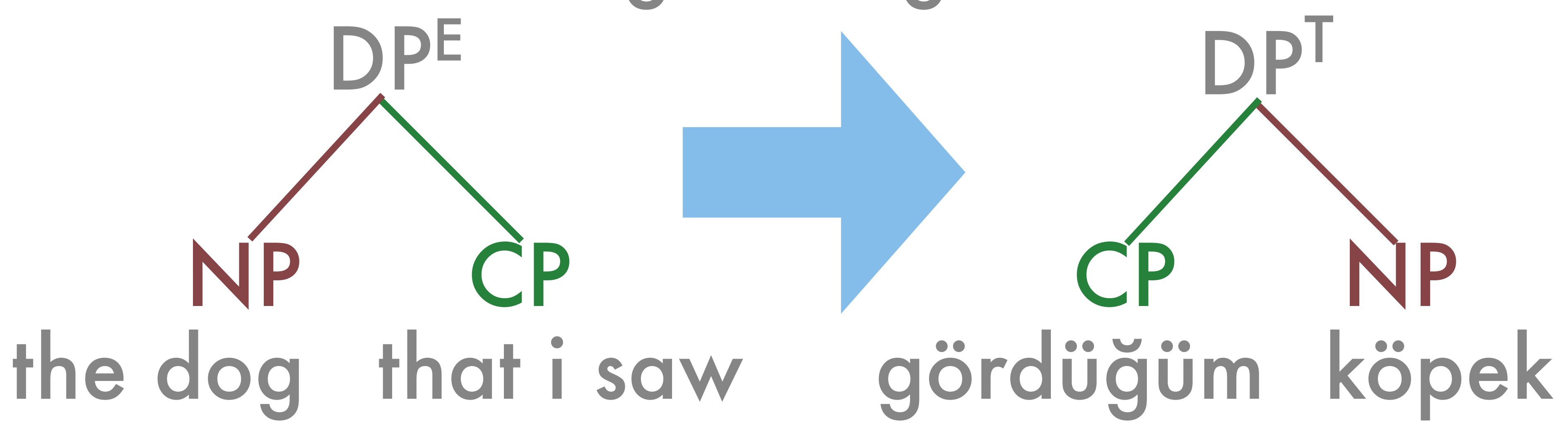
The Forest

- Syntactically motivated approach to MT
  - Basic Idea
  - Assign a tree structure to one or both sides of a sentence pair
  - Model alignment process by reordering operations defined on tree structure itself

Trees have advantages

- Captures the fact that syntactic constituents tend to move as a unit
- Systematic differences between two grammars *should* be represented
    - EXAMPLE: English ⟶ Turkish

Systematic Ordering Change of Two Constituents



Prior work

- Wu's Inversion Transduction Grammar, 1997 - Presented by David
- Alshawi's Finite State Head Transducers over trees, 2000 - presented by Yow-Ren
- Yamada and Knight, 2001 - presented by Gabriel
- Gildea's Tree-to-tree formalism, 2003 - just presented by Scott

Wu 1997

- Tree-based formalism the goal of which is to restrict word-level alignments
    - Uses simple grammar with a single nonterminal
    - Restricts alignments to those allowable by reordering operations on binary branching trees

Constraint allows EM convergence in polynomial time!

r	ITG	all matchings	ratio
0	1	1	1.000
1	1	1	1.000
2	2	2	1.000
3	6	6	1.000
4	22	24	0.917
5	90	120	0.750
6	394	720	0.547
7	1,806	5,040	0.358
8	8,558	40,320	0.212
9	41,586	362,880	0.115
10	206,098	3,628,800	0.057
11	1,037,718	39,916,800	0.026
12	5,293,446	479,001,600	0.011
13	27,297,738	6,227,020,800	0.004
14	142,078,746	87,178,291,200	0.002
15	745,387,038	1,307,674,368,000	0.001
16	3,937,603,038	20,922,789,888,000	0.000

Alshawi 2000

- Tree transduction is an interesting and expressive formalism
- Technique to induce hierarchical alignments, then probabilistically weight transducers to generate those alignments
- But formalism is very difficult to understand... and to implement?

Yamada and Knight 2001

- Tree-string model, taking an English parse as a “tree mobile”, rotating its arms in order to come up with the French reordering.
- Limited by the extent to which it over restricts ordering options. e.g. It cannot generate VSO from SVO input, assuming binary branching trees.
- Model Estimation procedure converges at $O(n^4)$, quartic with the size of the input sentence.

Gildea 2003

from tree to tree to alignment

- Fundamental Idea: Transform source language tree into target language tree through applying a series of operations.
  - Operations in our toolbox:
    - Reorder a node's children.
    - Insert a child node or delete one.
    - Translate words at leaves.
    - And cloning.

Gildea 2003 – II

- A generative model.
  - Calculating $P(T_b \mid T_a)$ takes place from the root of the tree down, given
    1. Current node ε_a might be grouped with one of its children with

       $P_{elem}(t_a \mid \varepsilon_a \Rightarrow children(\varepsilon_a))$

    2. Alignment of the children of this node is given by

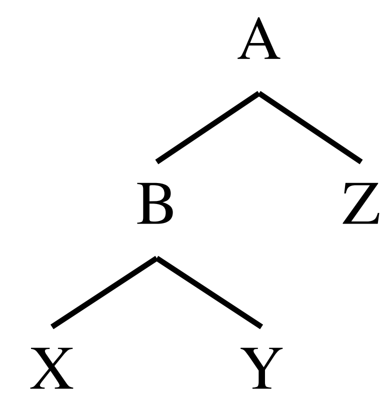
       $P_{align}(\alpha \mid \varepsilon_a \Rightarrow children(t_a))$

       - Alignment α can include insertions and deletions of individual children
       - If two nodes have been grouped (in 1.) their nodes are reordered in one step

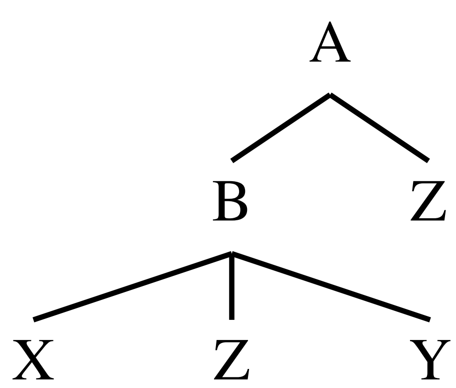
    3. Lexical items at the leaves of the tree are translated into the target language according to the distribution $P_t(f \mid e)$

Gildea 2003 – III

- In order to allow pairings with divergent tree structures, cloning is introduced
- Takes a copy of a (translated) English subtree and inserts it at any point in the translated French sentence, at some cost



before cloning



node Z can be cloned, allowing the ordering XZY

clone fertility of a node: $P_{ins}(\text{clone}|\varepsilon_j)$

choose node to clone:

$$P_{clone}(\varepsilon_i|\text{clone}=1) = \frac{P_{makeclone}(\varepsilon_i)}{\sum_k P_{makeclone}(\varepsilon_k)}$$

Dependency Trees

- Shown to be more consistent than constituent structure between French and English in Fox 2002.
- The basic finding is that *head* spans intersecting with spans of their modifiers are reduced when you use dependencies (in English--French)

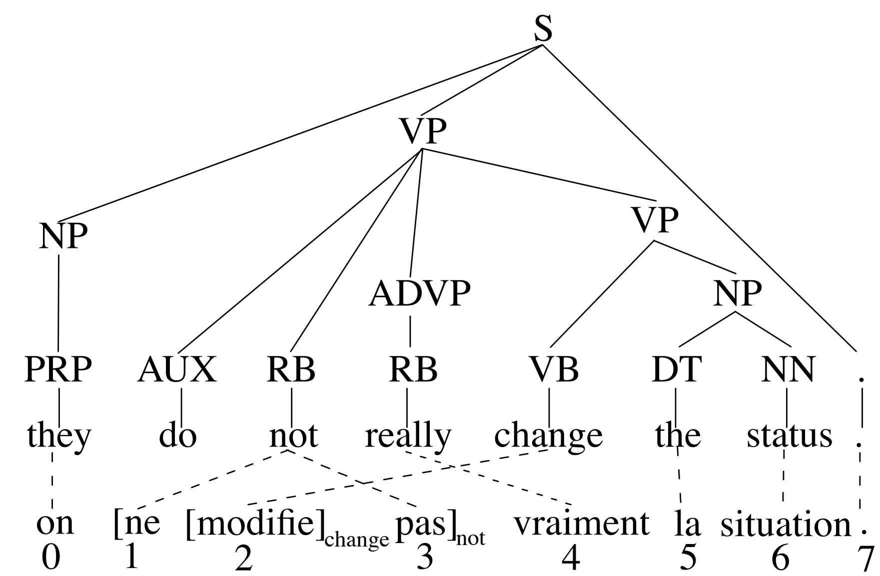


Figure 1: Alignment Example with Crossing

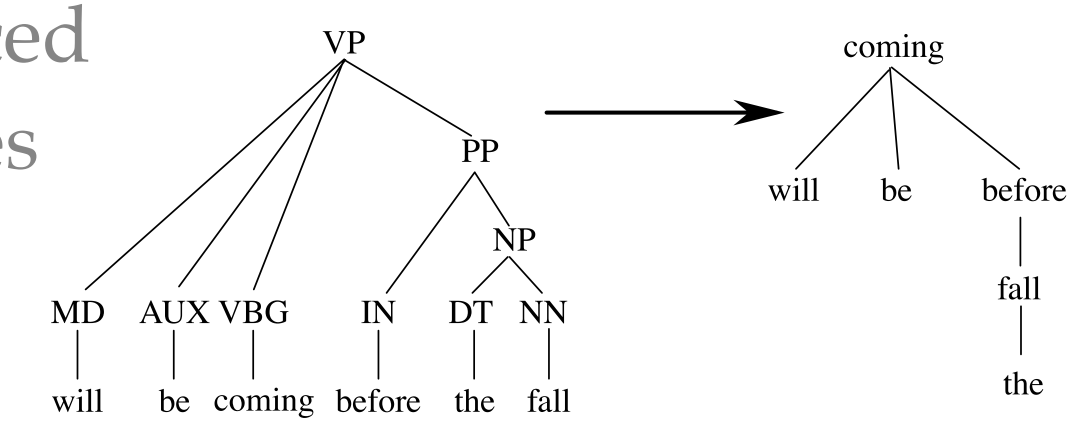
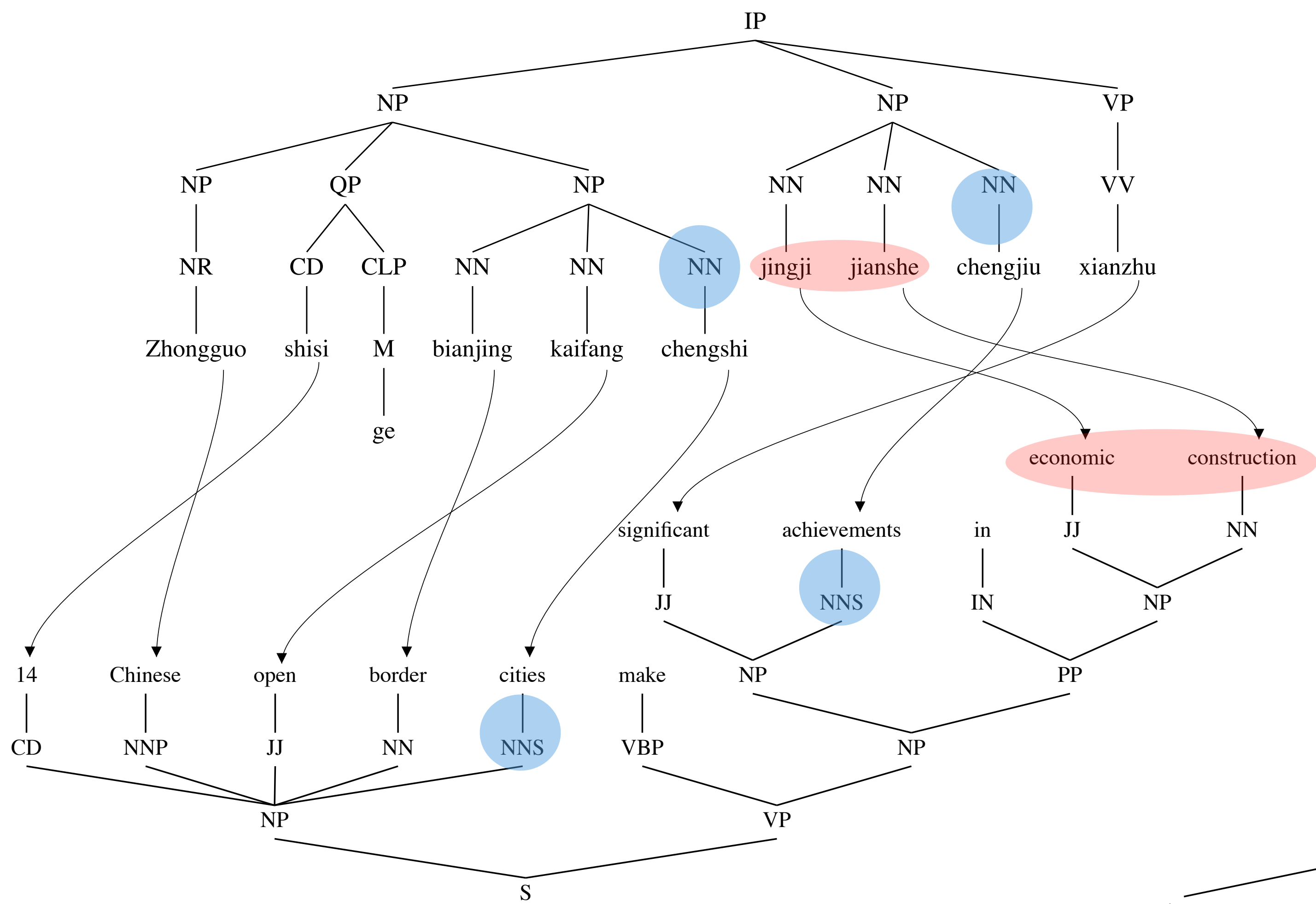


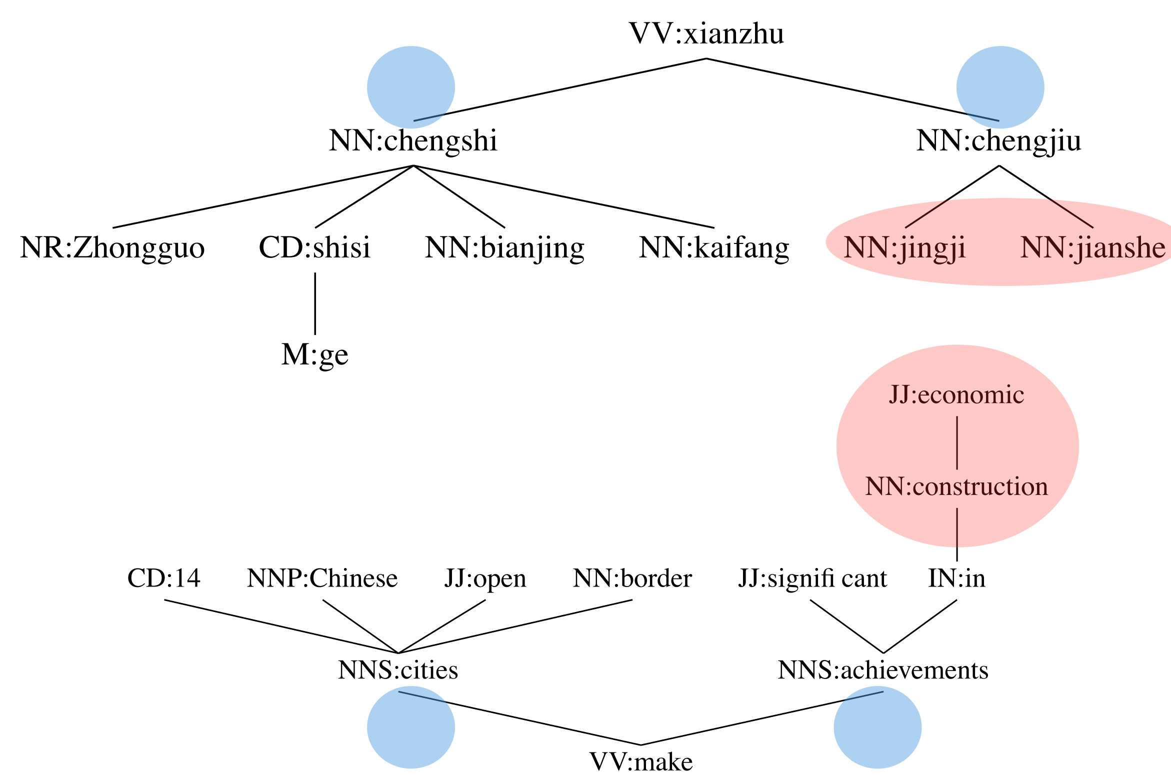
Figure 9: Extracting Dependencies

From Fox 2002

Constituent and Dependency Trees + Alignments for Chinese-English Bitext



Flattening effect, moving the English noun phrase headed by "achievements" to the same level of the tree as its Chinese counterpart.

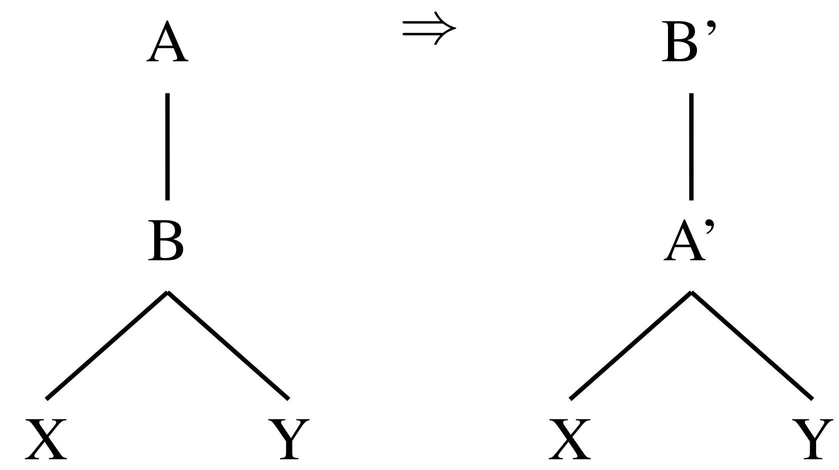


However, a more divergent structure is created in the dependency bi-tree. Even though the constituent trees for "economic construction" locally similar, different heads are selected for each.

Dependency Alignment Model

- Similar generative model for constituent trees, with some changes to keep in mind:
  - Lexical translation takes place at every node of the tree, rather than just at the leaves.
  - Lexical translation costs are then factored in as alignment costs for elementary trees.
  - There is a new parameter P_{swap} that gives the probability that a child node will swap positions with the node above it.

a derivation



First calculate probability of elementary tree

$$P_{elem}(\mathrm{AB}|\mathrm{A}\Rightarrow\mathrm{B})$$

Swap probability

$$P_{swap}$$

Translation Probabilities

$$P_t(\mathrm{A}'|\mathrm{A})$$
$$P_t(\mathrm{B}'|\mathrm{B})$$

Alignment of the Children

$$P_{align}(\{(1,1)(2,2)\}|\mathrm{A}\Rightarrow \mathrm{XY}).$$

Dependency Alignment Model – II

- Lexicalized extension
  - Alignment probabilities are extended so that reorderings are conditioned on categories of node's children, as well as lexical item.
  - e.g. $P_{align}(\{(1,1,),(2,2)\} \mid A \Rightarrow XY)$ becomes $P_{align}(\{(1,1,),(2,2)\} \mid A(\text{lex}) \Rightarrow XY)$
  - Smoothed by linear interpolation with unlexicalized probabilities.

Alignment Experiments

- Corpora
- Error Metric
- Results & Discussion

EXPERIMENTS – Corpora

- Trained on Xinhua newswire Chinese-English corpus.
- Restricted to sentences of at most 25 words in either language
  - Resulted in 18,773 sentence pairs.
    - 276,113 (23,783 unique) Chinese Words
    - 315,415 (27,075 unique) English Words
    - Chinese parsed with Bikel's 2002 Parser
    - English parsed with Collins' 2000 Parser
  - Two sets were held out and hand aligned
    - 48 pair set used for development
    - 49 pair test used for final test, to control for overfitting

EXPERIMENTS – Corpora

- Sentences with very high fertility nodes were actually not used in training, for computational efficiency
  - at most 5 children for constituent model, which limited training pairs to 11,422
  - at most 6 children for dependency model, limiting training pairs to 10,662

EXPERIMENTS – Error Measure

- For scoring alignments, the alignment error rate (AER) developed by Och and Ney (2000) was used.

$$P = \frac{|A \cap G|}{|A|}$$

- $$AER = 1 - \frac{2|A \cap G|}{|A| + |G|}$$

$$R = \frac{|A \cap G|}{|G|}$$

- **A** is the set of aligned word pairs generated by the automatic system
- **G** is the set of aligned word pairs indicated by the manually aligned data.

EXPERIMENTS – Results

	Precision	Recall	Alignment Error Rate
IBM Model 1	.56	.42	.52
IBM Model 4	.67	.43	.47
Constituent Tree-to-Tree	.51	.48	.50
Dependency Tree-to-Tree	.44	.38	.60
Dependency, lexicalized reordering	.41	.37	.61

Discussion

Note that the constituent-based model significantly outperforms the dependency based model, and achieves higher recall rate than both IBM models

In order to investigate the differences between *constituent* and *dependency* structure, the authors took aligned constituents in their hand annotated data

In only 40% of cases, an English tree production is consistently aligned to a Chinese constituent. For an example see slide 13.

Where alignments were consistent, heads corresponded 67% of the time. Head swapping should take care of this as long as nodes are adjacent. Are we talking about the dependency structure "swap" operation here?