

# Semi-Supervised Approaches for Learning to Parse Natural Languages

Rebecca Hwa

University of Maryland

*hwa@umiacs.umd.edu*

# Natural Language Processing

- Applications using language technologies are ubiquitous
  - Search engines
  - Machine translation
  - Question answering
  - Summarization
  - Speech recognition
  - Dialogue systems
  -

# The Role of Parsing in Language Applications...

- As a stand-alone application
  - Grammar checker
- As a pre-processing step
  - Q&A, information extraction, dialogue systems
- As an integral part of a model
  - Speech Recognition
    - language models [*Chelba and Jelinek, 1998*]
  - Machine Translation
    - word alignment [*Yamada and Knight, 2001*]

# Programming Languages

# Natural Languages

Unambiguous, precise

Ambiguous, imprecise

`(a == b ? 1 : 0)`

*“That depends on what the definition of ‘is’ is.”*

Grammar:

Grammar:

*type-qualifier*: one of  
`const volatile`

*struct-or-union*: one of  
`struct union`

...

?

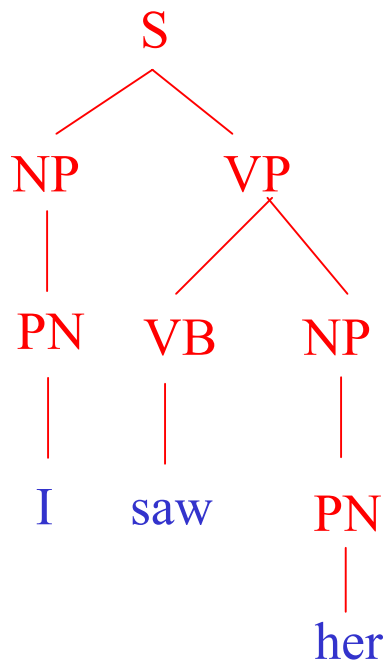
Learn syntactic structures by combining linguistic knowledge and statistical techniques

# Roadmap

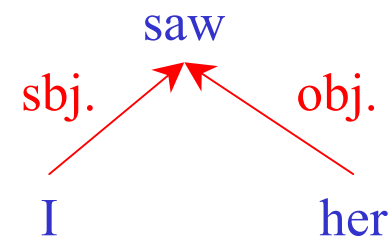
- Parsing as a learning problem
  - Why train a parser?
  - Challenges in training parsers
- 3 Semi-supervised approaches
- Conclusion and further directions

# Parsing

Input: *I saw her*



Constituency tree

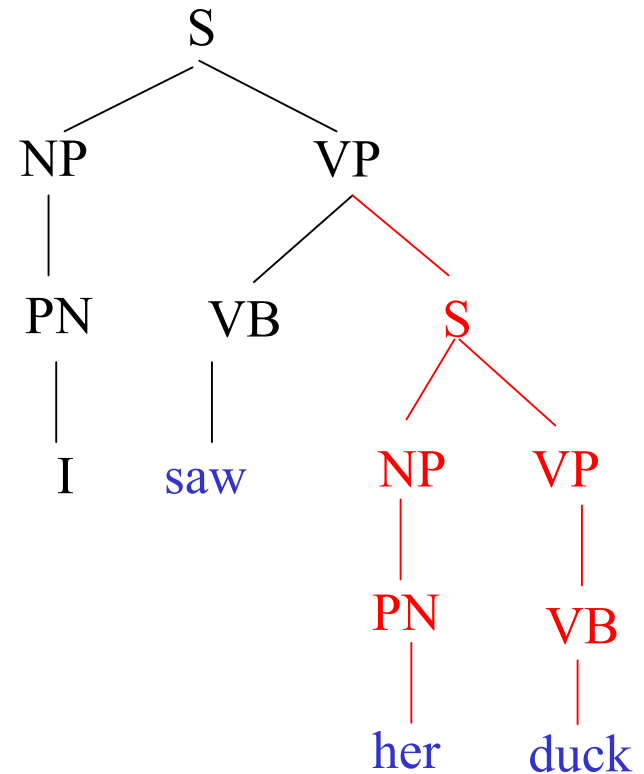
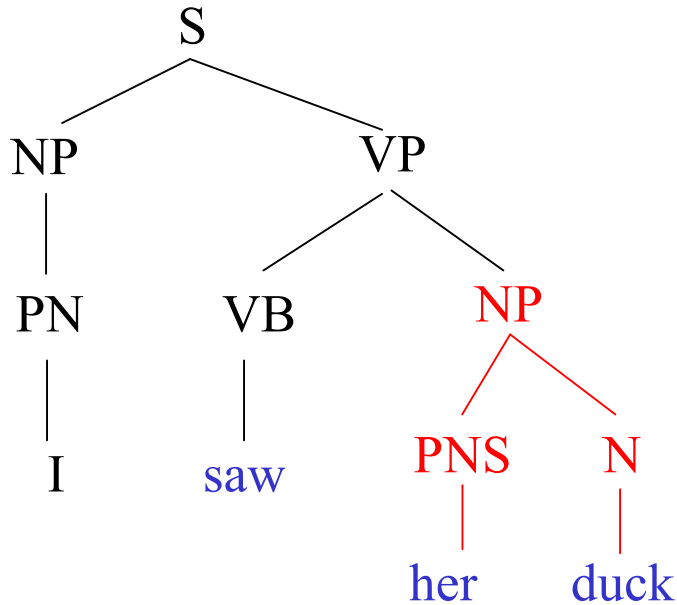


Dependency tree

- Parsers provide syntactic analyses of sentences

# Parsing Ambiguities

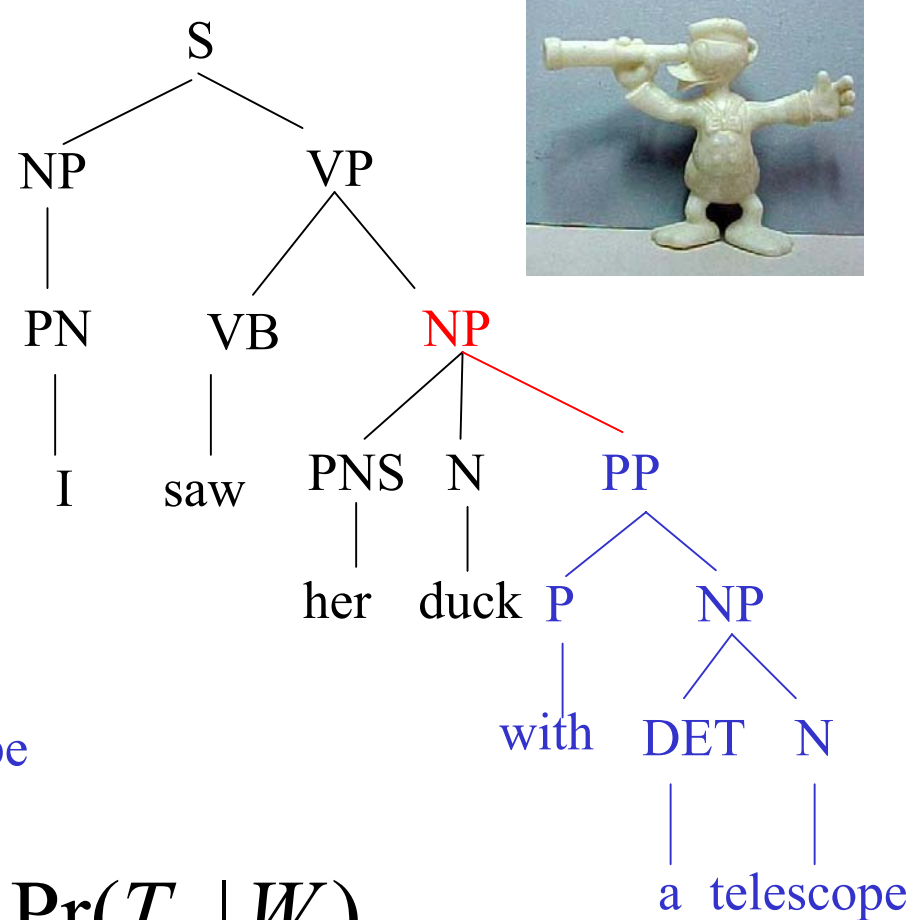
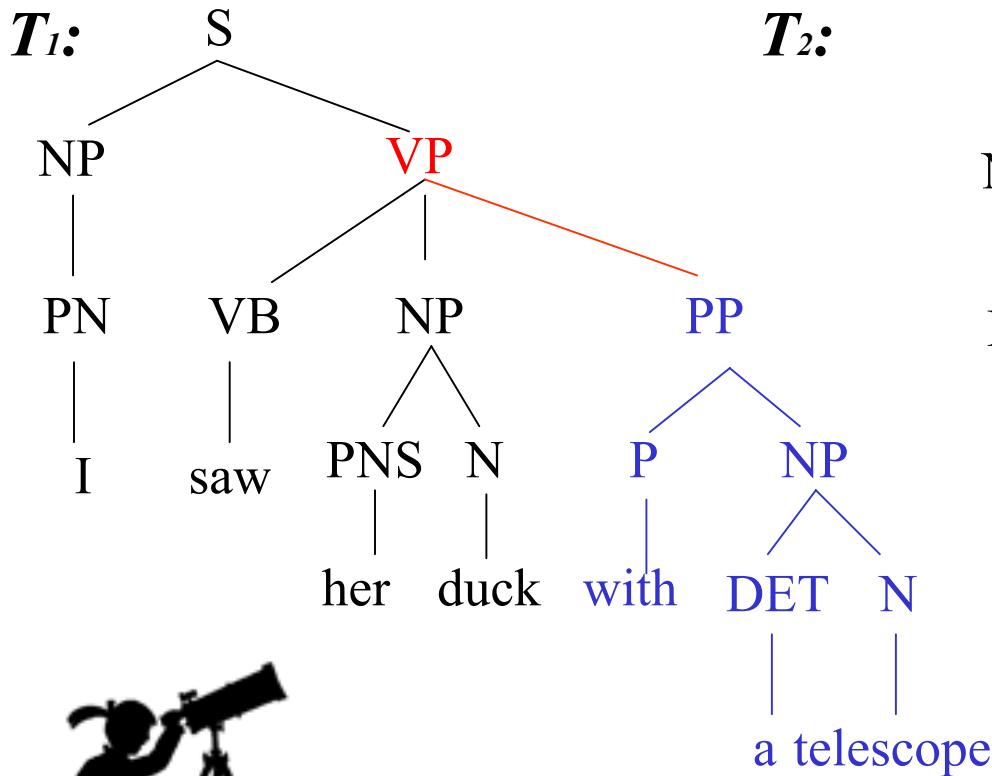
Input: *I saw her duck*



- Lexical ambiguity
  - e.g., multiple word senses, multiple parts-of-speech
- Structural ambiguity

# Disambiguation with Statistical Parsing

$W = \text{"I saw her duck with a telescope"}$



$$\Pr(T_1 | W) > \Pr(T_2 | W)$$

# A Statistical Parsing Model

- Probabilistic Context-Free Grammar (PCFG)
- Associate probabilities with production rules
- Likelihood of the parse is computed from the rules used
- Learn rule probabilities from training data

*Example of PCFG rules:*

0.7 NP → DET N

0.3 NP → PN

0.5 DET → a

0.1 DET → an

0.4 DET → the

...

$$\arg \max_{T_i \in \text{Trees}(W)} \Pr(T_i | W) = \arg \max_{T_i \in \text{Trees}(W)} \frac{\Pr(T_i, W)}{\Pr(W)}$$

$$\Pr(T_i, W) = \prod_r \Pr(RHS_r | LHS_r)$$

# Learning to Classify

Train a model to decide: should a **prepositional phrase** modify the **verb** before it or the **noun**?

*Training examples:*

(v, saw, duck, with, telescope)

(n, saw, duck, with, feathers)

(v, saw, stars, with, telescope)

(n, saw, stars, with, Oscars)

...

# Learning to Parse

Train a model to decide: what is the most likely **parse** for a **sentence W**?

```
[S [NP-SBJ [NNP Ford] [NNP Motor] [NNP Co.]]  
  [VP [VBD acquired]  
      [NP [NP [CD 5] [NN %]]  
          [PP [IN of]  
              [NP [NP [DT the] [NNS shares]]  
                  [PP [IN in]  
                      [NP [NNP Jaguar]  
                          [NNP PLC]]]]]]] . ]
```

```
[S [NP-SBJ [NNP Pierre] [NNP Vinken]]  
  [VP [MD will]  
      [VP [VB join]  
          [NP [DT the] [NN board]]  
          [PP [IN as] [NP [DT a] [NN director]]] . ]
```

...

# Supervised Learning

- Training examples are pairs of **problems** and **answers**
- Training examples for parsing: a collection of **sentence**, **parse tree** pairs (**Treebank**)
  - From the treebank, get maximum likelihood estimates for the parsing model [*Charniak, 1997; Collins 1999; etc.*]
- Treebanks are difficult to obtain
  - Needs human experts
  - Takes years to complete

# Building Treebanks

Language	Amount of Training Data	Time to Develop	Parser Performance
English (WSJ)	1M words 40k sent.	~5 years	~90%
Chinese (Xinhua News)	100K words 4k sent.	~2 years	~75%
Others (e.g., Hindi, Cebuano)	?	?	?

# Alternative Approaches

- Resource rich methods
  - Use additional context (e.g., morphology, semantics, etc.) to reduce training examples [*Hermjakob, 1997*]
- Resource poor (unsupervised) methods
  - Do not require labeled data for training
  - Typically have poor parsing performance
  - Can use some labels to improved performance [*Pereira and Schabes, 1992; Hwa, 1998; Hwa, 1999; Chiang, 2002*]

# Our Approach

- Sample selection
  - Reduce the amount of training data by picking more useful examples
- Co-training
  - Improve parsing performance from unlabeled data
- Cross-language projection
  - Bootstrap a parser for a new language from available resources within months

# Roadmap

- Parsing as a learning problem
- Semi-supervised approaches
  - Sample selection [*Hwa, 2000; Hwa, 2001*]
    - Overview
    - Scoring functions
    - Evaluation
  - Co-training
  - Cross-language projection
- Conclusion and further directions

# Sample Selection

- Assumption
  - Have lots of unlabeled data (cheap resource)
  - Have a human annotator (expensive resource)
- Iterative training session
  - Learner selects sentences to learn from
  - Annotator labels these sentences
- Goal: Predict the *benefit* of annotation
  - Learner selects sentences with the highest *Training Utility Values (TUVs)*
  - Key issue: scoring function to estimate TUV

# Scoring Function

- Approximate the TUV of each sentence
  - True TUVs are not known
- Need relative ranking
- Ranking criteria
  - Knowledge about the domain
    - e.g., sentence clusters, sentence length, ...
  - Output of the hypothesis
    - e.g., error-rate of the parse, uncertainty of the parse, ...
  - ...

# Proposed Scoring Functions

- Using domain knowledge
  - $f_{len}$  long sentences tend to be complex
- Uncertainty about the output of the parser
  - $f_{te}$  tree entropy
- Minimize mistakes made by the parser
  - $f_{error}$  use an **oracle** scoring function find sentences with the most parsing inaccuracies

# Entropy

- Measure of uncertainty in a distribution
  - Uniform distribution  $\Rightarrow$  very uncertain
  - Spike distribution  $\Rightarrow$  very certain
- Expected number of bits for encoding a probability distribution,  $X$

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \log p(x)$$

# Tree Entropy Scoring Function

- Distribution over parse trees for sentence  $W$ :

$$\sum_{T_i \in \text{Trees}(W)} \Pr(T_i | W) = 1$$

- Tree entropy: uncertainty of the parse distribution

$$TE(W) = - \sum_{T_i \in \text{Trees}(W)} \Pr(T_i | W) \log \Pr(T_i | W)$$

- Scoring function: ratio of actual parse tree entropy to that of a uniform distribution

$$f_{te} = \frac{TE(W)}{\log(|\text{Trees}(W)|)}$$

# Oracle Scoring Function

- $f_{error}$  1 - the accuracy rate of the most-likely parse
- Parse accuracy metric: f-score

f-score = harmonic mean of precision and recall

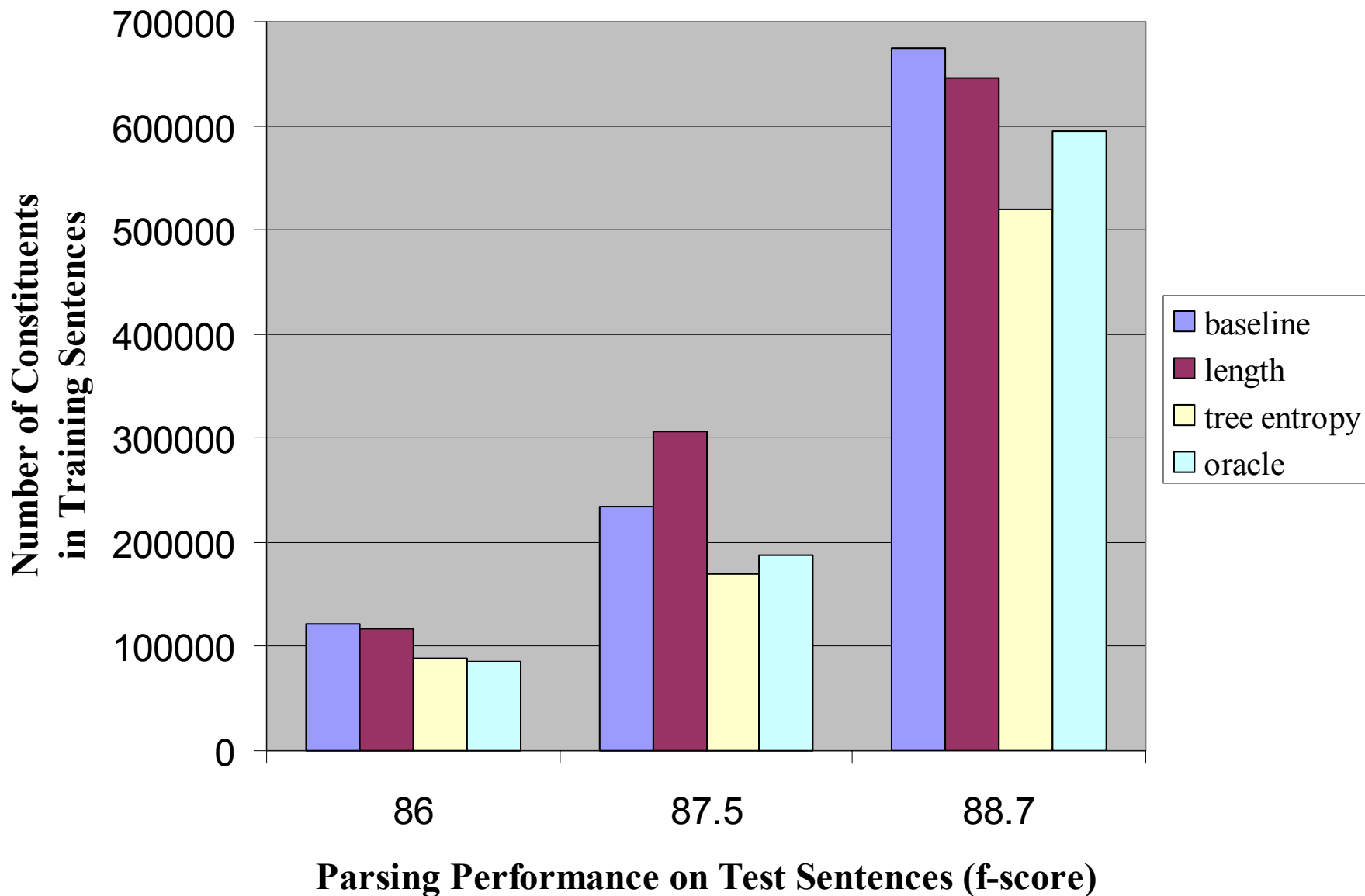
$$\text{Precision} = \frac{\# \text{ of correctly labeled constituents}}{\# \text{ of constituents generated}}$$

$$\text{Recall} = \frac{\# \text{ of correctly labeled constituents}}{\# \text{ of constituents in correct answer}}$$

# Experimental Setup

- Parsing model:
  - Lexicalized PCFG [*Collins, 1999*]
- Candidate pool
  - WSJ sec 02-21, with the annotation stripped
  - Initial labeled examples: 500 sentences
  - Per iteration: add 100 sentences
- Testing metric: f-score (precision/recall)
- Test data:
  - ~2000 unseen sentences (from WSJ sec 00)
- Baseline
  - Annotate data in sequential order

# Parsing Performance Vs. Constituents Labeled

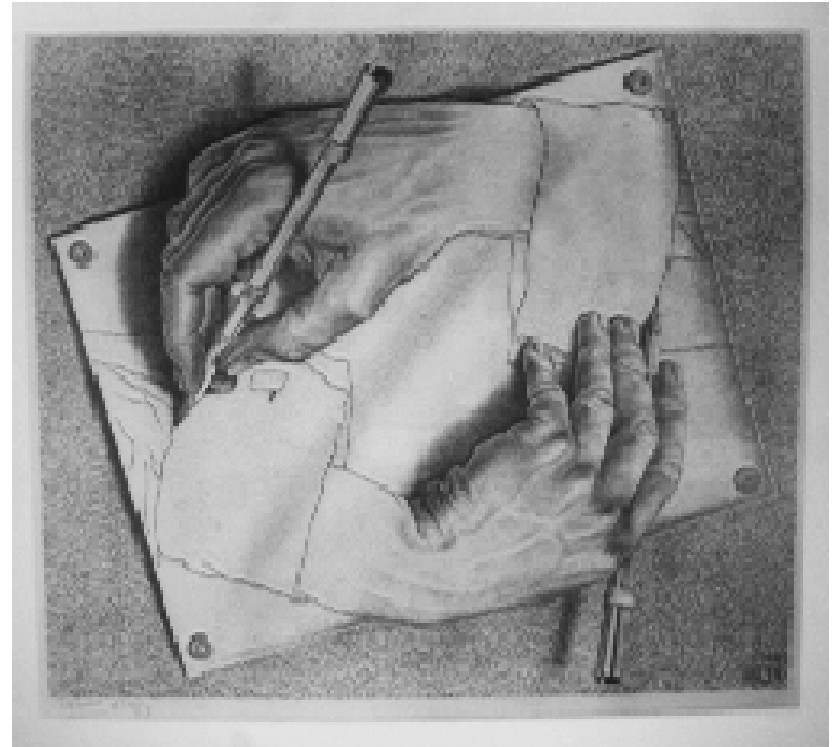


# Roadmap

- Parsing as a learning problem
- **Semi-supervised approaches**
  - Sample selection
  - Co-training [*with Steedman et al., 2003a, 2003b*]
  - Cross-language projection
- Conclusion and further directions

# Co-Training [*Blum and Mitchell, 1998*]

- Assumptions
  - Have a small treebank
  - No further human assistance
  - Have multiple learners with different views of the problem



# Co-Training

- Iterative learning process
  - Each learner labels data for the other learner
  - Which machine labeled data should be added to the training set? [*Dasgupta et al., 2002; Abney, 2002; Steedman and Hwa et al., 2003*]
- Goals
  - Minimize adding errors
  - Maximize training utilities

# Our Co-Training Findings

- Need effective scoring function to evaluate the parsers' output
- Need selection methods that consider training utility as well as output accuracy
- Co-training improves parsers when...
  - the size of the initial training set is small
  - unlabeled data is from a different domain
- Corrected Co-training
  - Relationship to active learning [*Muslea et al., 2000; Pierce and Cardie, 2001*]

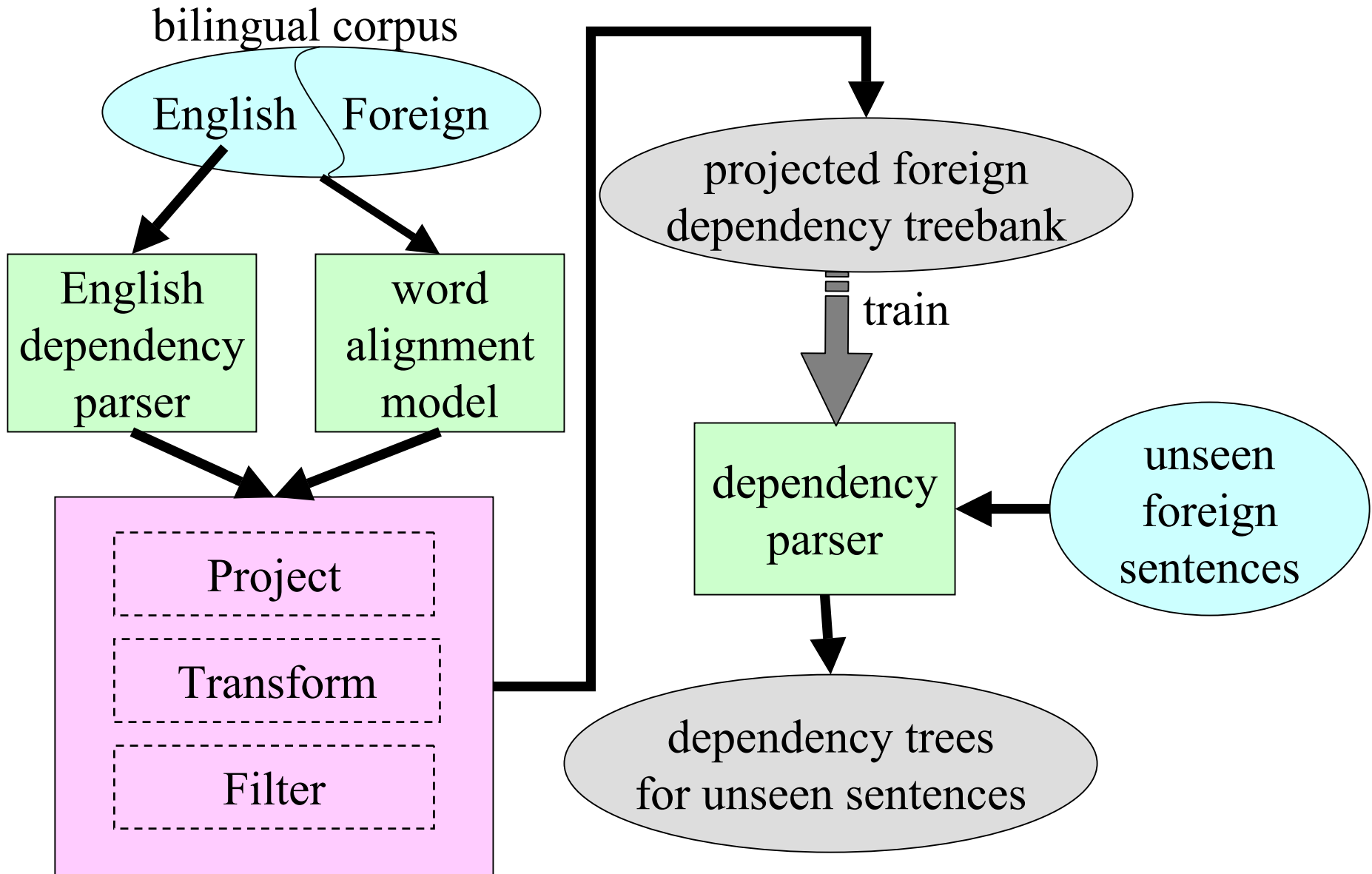
# Roadmap

- Parsing as a learning problem
- **Semi-supervised approaches**
  - Sample selection
  - Co-training
  - **Cross-language projection** [*Hwa et al., 2002*]
    - Overview
    - Direct Projection Algorithm
    - Evaluation
- Conclusion and further directions

# Cross-Language Projection

- Assumptions
  - Have no annotated resource in new language
  - Have human for a short time (~1 month)
  - Have English resources and bilingual data
- Approach
  - Syntactic dependency holds across languages
  - Bootstrap from English to induce a treebank for the new language
- Goal
  - Use the induced treebank to train a parser for the new language

# System Architecture



# Necessary Resources:

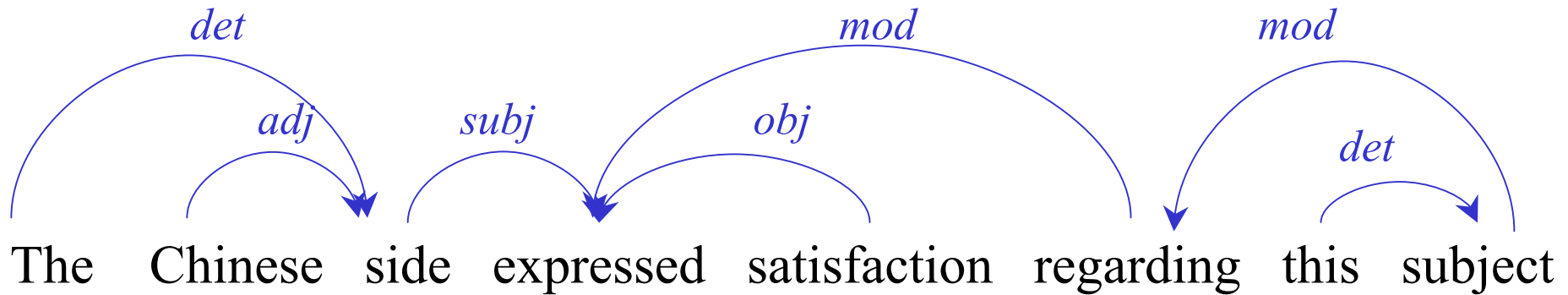
## 1. Bilingual Sentences

The Chinese side expressed satisfaction regarding this subject

中国方面对此表示满意

# Necessary Resources

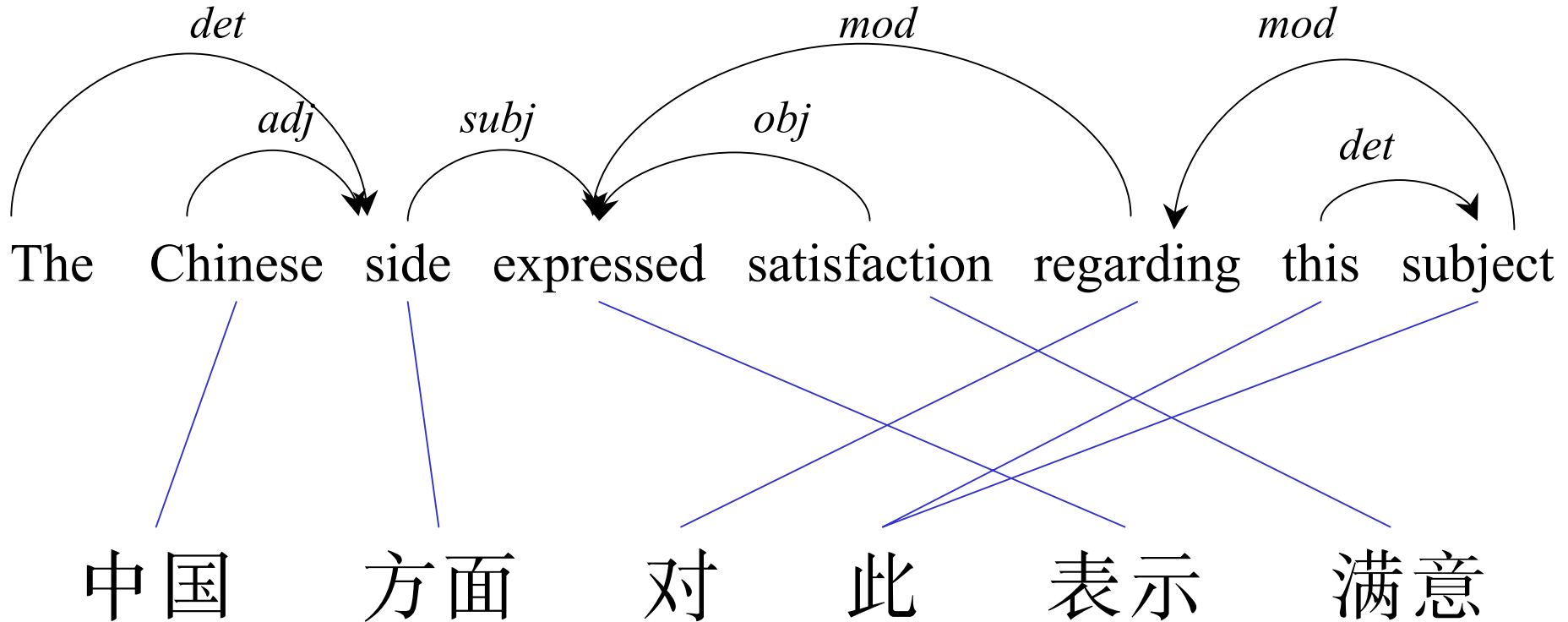
## 2. English (Dependency) Parser



中国 方面 对 此 表示 满意

# Necessary Resources

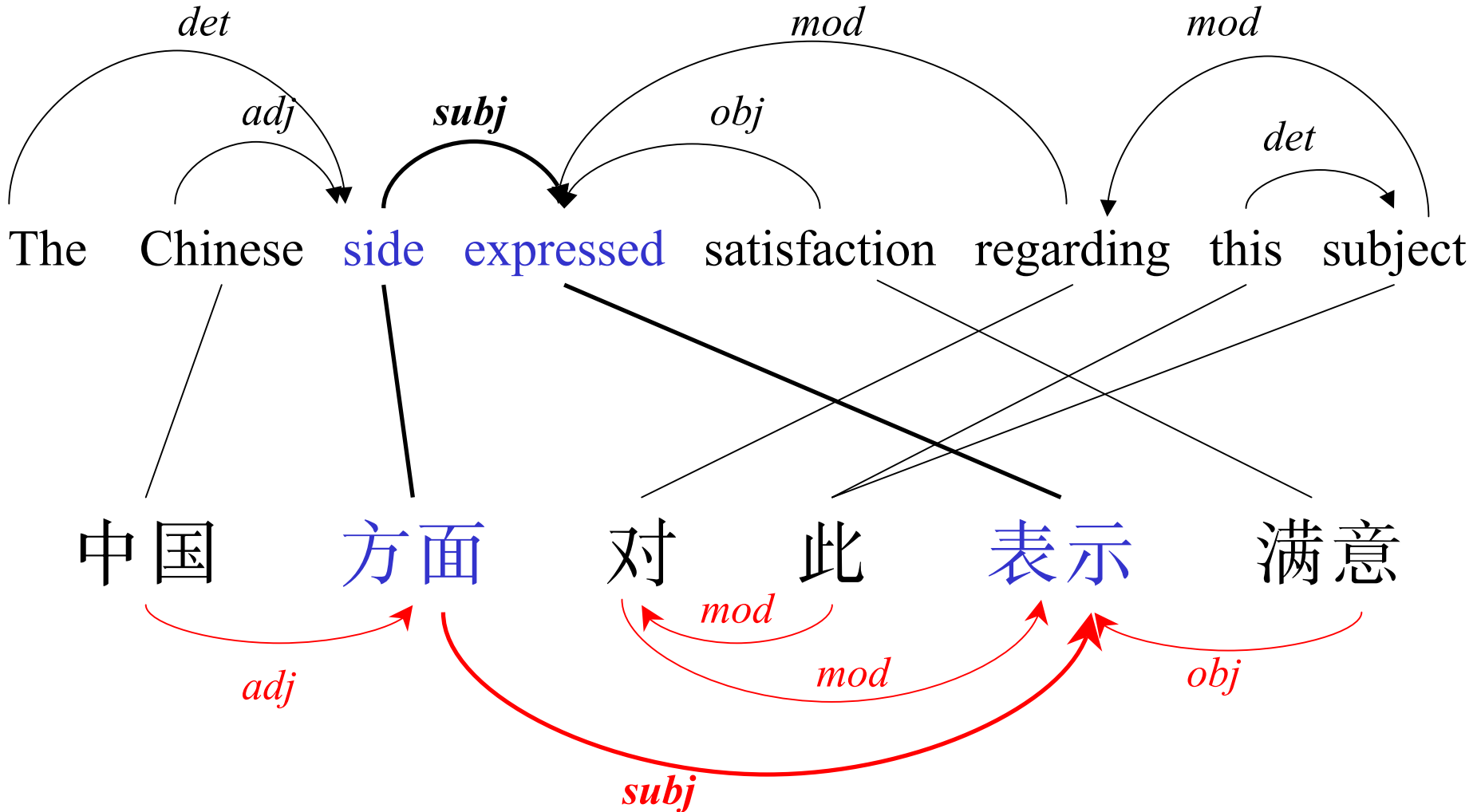
## 3. Word Alignment



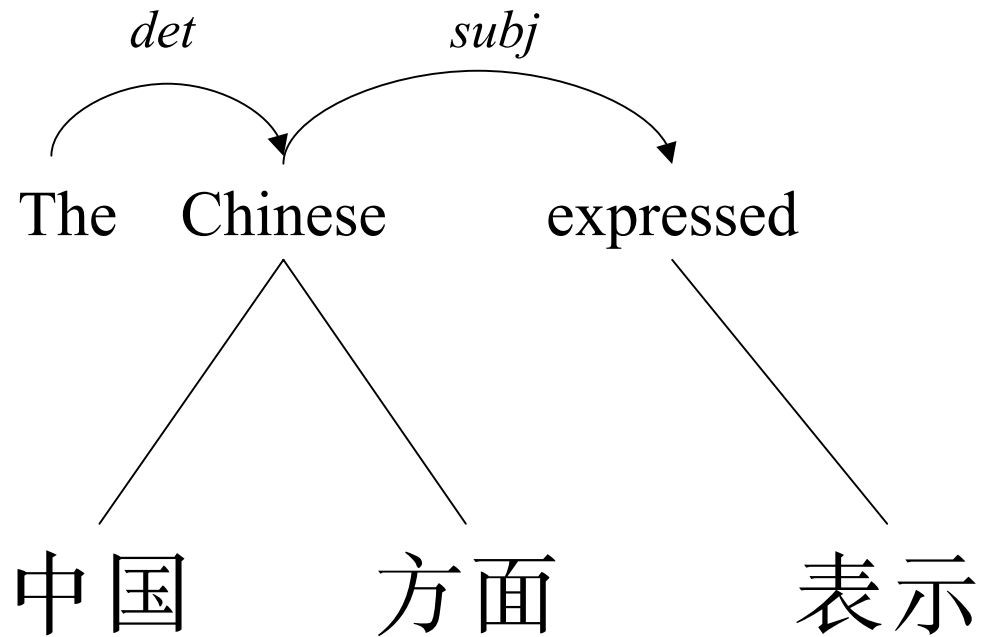
# Direct Projection Algorithm (DPA)

- If there is a syntactic relationship between two English words, then the same syntactic relationship also exists between their corresponding foreign words

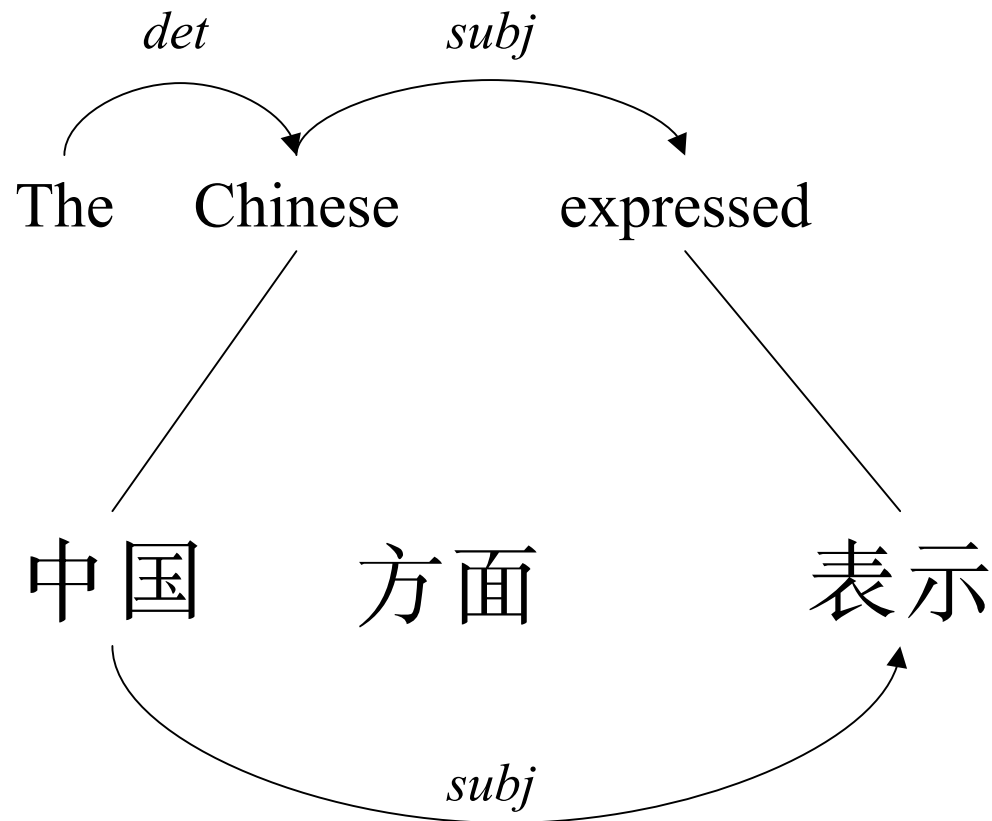
# Projected Chinese Dependency Tree



# Problematic Case: 1-to-Many



# Problematic Case: Unaligned Foreign Word



# Post-Projection Transformation

- Use a human speaker's linguistic knowledge
- Handle one-to-many mapping
  - Select **head** based on (English) part-of-speech categories
- Handle some unaligned foreign word cases
  - Only addressing **closed-class** words
    - Functional words (e.g., aspectual, measure words)
    - Easily enumerable lexical categories (e.g., pronouns, prepositions)

# Evaluating the Projected Trees

- Compare against gold standard
  - Effects of post-projection transformation and alignment
- Two pilot studies (English-Chinese and English-Spanish)
  - Chinese: 88 sentences from Xinhua News
  - Spanish: 100 sentences from UN / FBIS / Bible

	Word Alignment	f-Score (Chinese/Spanish)
Projection	human	38% / 42%
Projection + transformation	human	67% / 70%
Projection	automatic	12% / 37%
Projection + transformation	automatic	39% / 65%

# Filtering the Induced Treebank

- Projected treebank is noisy
  - Projection mismatch
  - Cascading component errors
- Automatically filter out bad training examples from projected treebank
  - Too many words were unaligned
  - Too many words are aligned to the same word
  - Projected tree has too many crossing dependencies.

# Training a Spanish Parser from Projected Treebank

Method	Training Corpus	Corpus Size	Parser Accuracy (100 test sent.)
Modify Prev (baseline)	-	-	34%
Stat. Parser	UN/FBIS/Bible (no filter)	98K	67%
Stat. Parser	UN/FBIS/Bible (w/ filter)	20K	72%
Commercial Parser	-	-	69%

# Conclusion

- Sample selection
  - Tree-entropy reduces the number of training examples by 35% and the number of labeled constituents by 23%.
- Co-training
  - Utility-based selection methods improve parsing performance by using unlabeled data.
- Cross-language projection
  - Projected treebank can train a Spanish parser with performances comparable to a commercial hand-crafted parser.

# Further Directions (1)

- Machine learning methods for parsing
  - Better understanding of relationships between different learning techniques
  - Scoring functions for sample selection and co-training
  - Selection methods for co-training
  - Interaction with human in supervised training
- Applications of parsing: multilingual language processing
  - Word alignment [*with Lopez et al., 2002*]
  - Structural correspondences [*with Dorr et al., 2002*]
  - Machine translation

# Further Directions (2)

- Learning to represent complex but regular structures
  - Text and data mining applications
- Data sparsity
  - Efficient use of available resources
- Human-computer cooperation in learning tasks
  - Combining the strengths of each

# Bibliography

- M. Steedman, R. Hwa, S. Clark, M. Osborne, A. Sarkar, J. Hockenmaier, P. Ruhlen, S. Baker, and J. Crim. [Example Selection for Bootstrapping Statistical Parsers](#). To appear in the *Proceedings of the Annual Meeting of the North American Chapter of the ACL*, Edmonton, Canada, 2003.
- M. Steedman, M. Osborne, A. Sarkar, S. Clark, R. Hwa, J. Hockenmaier, P. Ruhlen, S. Baker, and J. Crim. [Bootstrapping Statistical Parsers from Small Datasets](#). To appear in the *Proceedings of the Annual Meeting of the European Chapter of the ACL*, Budapest, Hungary, 2003.
- R. Hwa, P. Resnik, A. Weinberg, and O. Kolak. [Evaluating Translational Correspondence using Annotation Projection](#). In the *Proceedings of the 40th Annual Meeting of the ACL*, Philadelphia, PA, 2002.
- R. Hwa. [Sample selection for statistical grammar induction](#). In the *Proceedings of the 2000 Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pp. 45-52, October 2000.
- R. Hwa. [Supervised Grammar Induction using Training Data with Limited Constituent Information](#). In the *Proceedings of the 37th Annual Meeting of the ACL*, pp. 73-79, June 1999.
- R. Hwa. [An Empirical Evaluation of Probabilistic Lexicalized Tree Insertion Grammars](#). In the *Proceedings of ACL-COLING 1998*, pp. 557-563.