NRC-CNRC

Institute for Information Technology

Statistical Machine Translation

George Foster

National Research Council Canada
Conseil national de recherches Canada

Canada

A Brief History of MT

Origins (1949): WW II codebreaking success suggests statistical approach to MT

A Brief History of MT

Origins (1949): WW II codebreaking success suggests statistical approach to MT

Classical period (1950–1966): rule-based MT and pursuit of FAHQT

A Brief History of MT

Origins (1949): WW II codebreaking success suggests statistical approach to MT

Classical period (1950–1966): rule-based MT and pursuit of FAHQT

Dark ages, post ALPAC (1966–1990): find applications for flawed technology

A Brief History of MT

Origins (1949): WW II codebreaking success suggests statistical approach to MT

Classical period (1950–1966): rule-based MT and pursuit of FAHQT

Dark ages, post ALPAC (1966–1990): find applications for flawed technology

Renaissance (1990's): IBM group revives statistical MT

A Brief History of MT

Origins (1949): WW II codebreaking success suggests statistical approach to MT

Classical period (1950–1966): rule-based MT and pursuit of FAHQT

Dark ages, post ALPAC (1966–1990): find applications for flawed technology

Renaissance (1990's): IBM group revives statistical MT

Modern era (2000–present): intense research activity, steady improvement in quality, new commercial applications

Why is MT Hard?

- structured prediction problem: difficult for ML
- word-replacement is NP-complete (Knight 99), via grouping *and* ordering
- performance grows as log(data-size): state-of-the-art models are huge and computationally expensive
- some language pairs are very distant
- evaluation is ill-defined

Statistical MT

$$\hat{t} = \operatorname*{argmax}_{t} p(t|s)$$

Statistical MT

$$\hat{t} = \operatorname*{argmax}_{t} p(t|s)$$

Statistical MT

$$\hat{t} = \operatorname*{argmax}_{t} p(t|s)$$

Statistical MT

$$\hat{t} = \operatorname*{argmax}_{t} p(t|s)$$

Statistical MT

$$\hat{t} = \operatorname*{argmax}_{t} p(t|s)$$

Two components:

- model
- search procedure

SMT Model

Noisy-channel decomposition, “fundamental equation of SMT”:

$$\begin{aligned} p(t|s) &= p(s|t)\, p(t) \,/\, p(s) \\ &\propto p(s|t)\, p(t) \end{aligned}$$

SMT Model

Noisy-channel decomposition, "fundamental equation of SMT":

$$\begin{aligned} p(t|s) &= p(s|t)\, p(t) \;/\; p(s) \\ &\propto p(s|t)\, p(t) \end{aligned}$$

Modular and complementary:

- translation model $p(s|t)$ ensures t translates s
- language model $p(t)$ ensures t is grammatical (typically n-gram model, trained on target-language corpus)

Log-linear Model

Tweaking the noisy channel model is useful:

$$p(t|s) \quad \propto \quad p(s|t)^{\alpha}\, p(t)$$

Log-linear Model

Tweaking the noisy channel model is useful:

$$\begin{aligned} p(t|s) &\propto p(s|t)^{\alpha}\, p(t) \\ &\propto p(s|t)^{\alpha}\, p'(t|s)^{\beta}\, p(t) \quad ?? \end{aligned}$$

Log-linear Model

Tweaking the noisy channel model is useful:

$$\begin{aligned} p(t|s) &\propto p(s|t)^{\alpha}\, p(t) \\ &\propto p(s|t)^{\alpha}\, p'(t|s)^{\beta}\, p(t) \quad ?? \end{aligned}$$

Generalize to log-linear model:

$$\log p(t|s) = \sum_i \lambda_i f_i(s,t) - \log Z(s)$$

- features $f_i(s,t)$ are interpretable as log probs; always include at least LM and TM
- weights λ_i are set to maximize system performance

⇒ All mainstream SMT approaches work like this.

Translation Model

Core of an SMT system: $p(s|t)$ - dictates search strategy

Capture relation between s and t using hidden *alignments*:

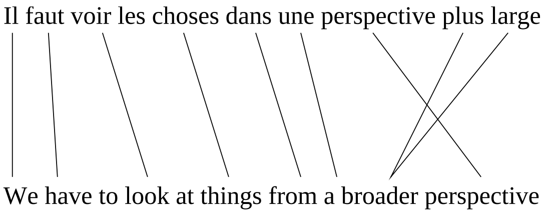
$$\begin{aligned} p(s|t) &= \sum_a p(s,a|t) \\ &\approx p(s,\hat{a}|t) \quad \text{(Viterbi assumption)} \end{aligned}$$

Different approaches model $p(s,a|t)$ in different ways:

- word-based
- phrase-based
- tree-based

Word-Based TMs (IBM Models)

- Alignments consist of word-to-word links.
- Asymmetrical: source words have 0 or 1 connections; target words have have zero or more:

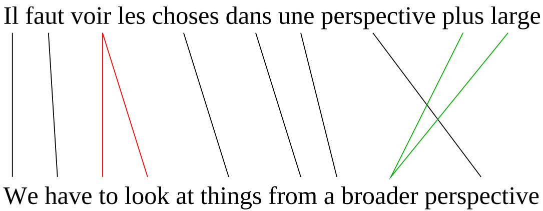


Il faut voir les choses dans une perspective plus large

We have to look at things from a broader perspective

Word-Based TMs (IBM Models)

- Alignments consist of word-to-word links.
- Asymmetrical: source words have 0 or 1 connections; target words have have zero or more:


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

IBM 1

Simplest of 5 IBM models:

- alignments are equally probable: $p(s, a|t) \propto p(s|a, t)$
- given an alignment, $p(s|a, t)$ is product of conditional probs of linked words, eg:

$$p(il_1, faut_2, voir_4, \ldots |we, have, to, look, \ldots) = \\ p(il|we)p(faut|have)p(voir|look) \times \cdots$$

- parameters: $p(w_{src}|w_{tgt})$ for all w_{src}, w_{tgt} (the *ttable*)
- interpretation of IBM1: 0-th order HMM, with target words as states and source words as observed symbols

Other IBM Models

IBM models 2–5 retain ttable, but add other sets of parameters for increasingly refined modeling of word connection patterns:

Other IBM Models

IBM models 2–5 retain ttable, but add other sets of parameters for increasingly refined modeling of word connection patterns:

- IBM2 adds *position* parameters $p(j|i, I, J)$: probability of link from source pos j to target pos i (alternative is *HMM* model: link probs depend on previous link).

Other IBM Models

IBM models 2–5 retain ttable, but add other sets of parameters for increasingly refined modeling of word connection patterns:

- IBM2 adds *position* parameters $p(j|i, I, J)$: probability of link from source pos j to target pos i (alternative is *HMM* model: link probs depend on previous link).
- IBM3 adds *fertility* parameters $p(\phi|w_{tgt})$: probability that target word w_{tgt} will connect to ϕ source words.

Other IBM Models

IBM models 2–5 retain ttable, but add other sets of parameters for increasingly refined modeling of word connection patterns:

- IBM2 adds *position* parameters $p(j|i, I, J)$: probability of link from source pos j to target pos i (alternative is *HMM* model: link probs depend on previous link).
- IBM3 adds *fertility* parameters $p(\phi|w_{tgt})$: probability that target word w_{tgt} will connect to ϕ source words.
- IBM4 replaces position parameters with *distortion* parameters that capture location of translations of current target word given same info for previous target word.

Other IBM Models

IBM models 2–5 retain ttable, but add other sets of parameters for increasingly refined modeling of word connection patterns:

- IBM2 adds *position* parameters $p(j|i, I, J)$: probability of link from source pos j to target pos i (alternative is *HMM* model: link probs depend on previous link).
- IBM3 adds *fertility* parameters $p(\phi|w_{tgt})$: probability that target word w_{tgt} will connect to ϕ source words.
- IBM4 replaces position parameters with *distortion* parameters that capture location of translations of current target word given same info for previous target word.
- IBM5 fixes normalization problem with IBM3/4.

Training IBM Models

Given parallel corpus, use *coarse-to-fine* strategy: each model in the sequence serves to initialize parameters of next model.

1. Train IBM1 (ttable) using exact EM (convex, so starting values not important).
2. Train IBM2 (ttable, positions) using exact EM.
3. Train IBM3 (ttable, positions, fertilities) using approx EM.
4. Train IBM4 (ttable, distortion, fertilities) using approx EM.
5. Optionally, train IBM5.

Ttable Samples

$w_{en}\ \ w_{fr}\ \ p(w_{fr}|w_{en})$:

city ville 0.77
city city 0.04
city villes 0.04
city municipalité 0.02
city municipal 0.02
city québec 0.01
city région 0.01
city la 0.00
city , 0.00
city où 0.00
... 637 more ...

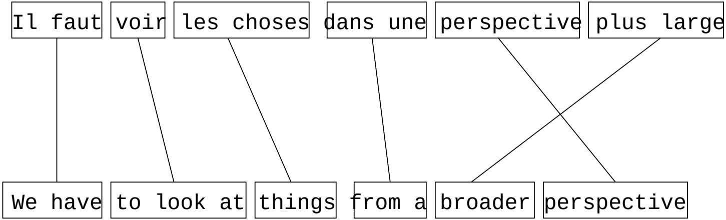
foreign-held détenus 0.21
foreign-held large 0.21
foreign-held mesure 0.19
foreign-held étrangers 0.14
foreign-held par 0.12
foreign-held agissait 0.09
foreign-held dans 0.02
foreign-held s' 0.01
foreign-held une 0.00
foreign-held investissements 0
... 6 more ...

running candidat 0.03
running temps 0.02
running présenter 0.02
running se 0.02
running diriger 0.02
running fonctionne 0.02
running manquer 0.02
running file 0.02
running campagne 0.01
running gestion 0.01
... 1176 more ...

Phrase-Based Translation

Alignment structure:

- Source/target sentences segmented into contiguous "phrases".
- Alignments consist of one-to-one links between phrases.
- Exhaustive: all words are part of some phrase.



Phrase-Based Model

$$p(s, a|t) = p(g|t)p(a|g, t)p(s|a, t)$$

- $p(g|t)$ is a segmentation model, usually uniform
- $p(a|g, t)$ is a distortion model for source phrase positions
- $p(s|a, t)$ models the content of phrase pairs, given alignment:

$$p(il\_faut_1, voir_2, les\_choses_3, \ldots | we\_have, to\_look\_at, things, \ldots) = \\ p(il\_faut|we\_have)p(voir|to\_look\_at)p(les\_choses|things) \times \cdots$$

  - parameters: $p(h_{src}|h_{tgt})$ for all phrase pairs h_{src}, h_{tgt} in a *phrase table* (analogous to ttable, but *much* larger)

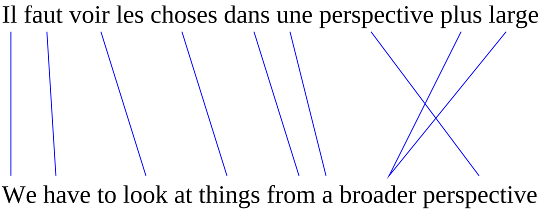
Phrase-Based Model Training

Heuristic algorithm:

1. Train IBM models (IBM4 or HMM) in two directions: $p(s|t)$ and $p(t|s)$.
2. For each sentence pair in parallel corpus:
   - Word-align sentences using *both* IBM4 models.
   - Symmetrize the 2 asymmetrical IBM alignments.
   - Extract phrase pairs that are consistent with symmetrized alignment.
3. Estimate $p(h_{src}|h_{tgt})$ (and reverse) by:
   - relative frequency: $c(h_{src}, h_{tgt})/c(h_{tgt})$
   - lexical estimate: from IBM models, or via link counts

Symmetrizing Word Alignments

1. align $s \rightarrow t$

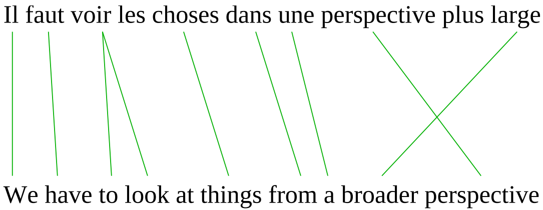


Il faut voir les choses dans une perspective plus large

We have to look at things from a broader perspective

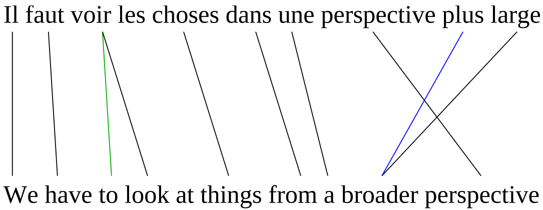
Symmetrizing Word Alignments

1. align $s \longrightarrow t$
2. align $t \longrightarrow s$


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

Symmetrizing Word Alignments

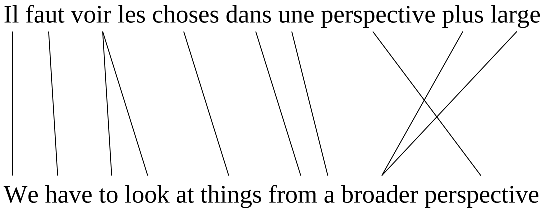
1. align $s \rightarrow t$
2. align $t \rightarrow s$
3. intersect links



Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

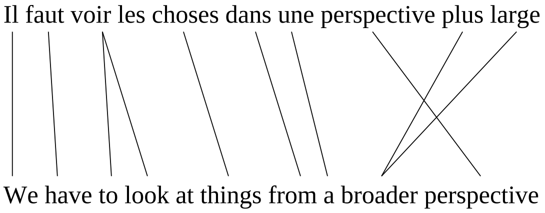
Symmetrizing Word Alignments

1. align $s \rightarrow t$
2. align $t \rightarrow s$
3. intersect links
4. add adjacent links (iteratively)


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

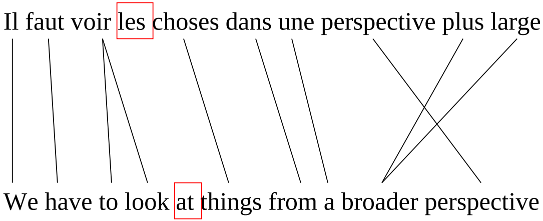
Phrase Extraction

Extract all possible phrase pairs that contain at least one alignment link, and that have no links that "point outside" the phrase pair. (Extracted pairs can overlap.)


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

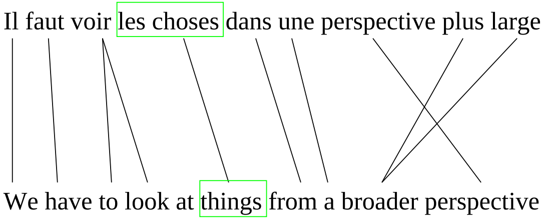
Phrase Extraction

Extract all possible phrase pairs that contain at least one alignment link, and that have no links that “point outside” the phrase pair. (Extracted pairs can overlap.)


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

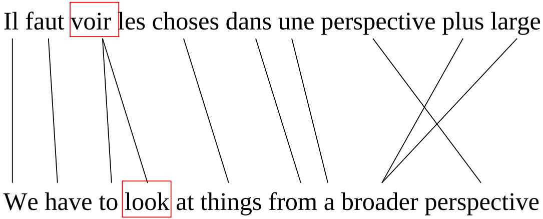
Phrase Extraction

Extract all possible phrase pairs that contain at least one alignment link, and that have no links that "point outside" the phrase pair. (Extracted pairs can overlap.)


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

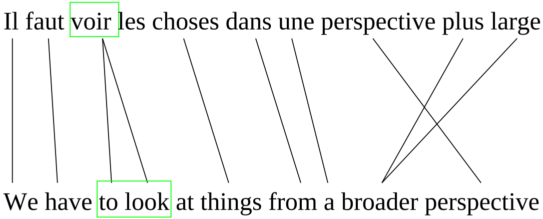
Phrase Extraction

Extract all possible phrase pairs that contain at least one alignment link, and that have no links that “point outside” the phrase pair. (Extracted pairs can overlap.)


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

Phrase Extraction

Extract all possible phrase pairs that contain at least one alignment link, and that have no links that "point outside" the phrase pair. (Extracted pairs can overlap.)


Il faut voir les choses dans une perspective plus large
We have to look at things from a broader perspective

Phrase Table Sample

```
bargaining agents ||| agents négociateurs
bargaining agents ||| agents de négociation
bargaining agents ||| représentants
bargaining agents ||| agents de négociations
bargaining agents ||| les agents négociateurs
bargaining agents ||| agents négociateurs ,
bargaining agents ||| des agents de négociation
bargaining agents ||| d’ agents négociateurs
bargaining agents ||| représentants syndicaux
bargaining agents ||| agents négociateurs qui
bargaining agents ||| agents négociateurs ont
bargaining agents ||| agent de négociation
bargaining agents ||| agents négociateurs pour
bargaining agents ||| agents négociateurs .
bargaining agents ||| les agents négociateurs ,
... 15 more ...
```

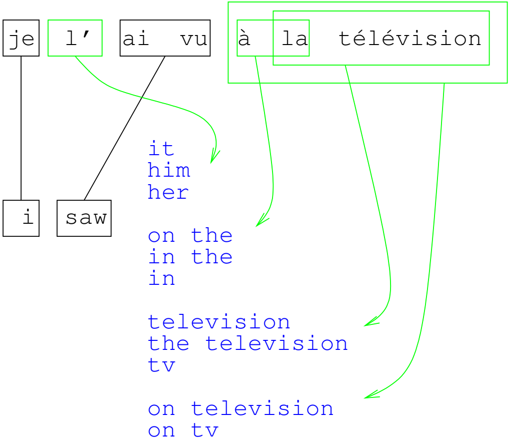
Search with Phrase-Based Model

Strategy: enumerate all possible translation hypotheses left-to-right, tracking phrase alignments and scores for each. Recombine and prune partial hyps to control exponential explosion.

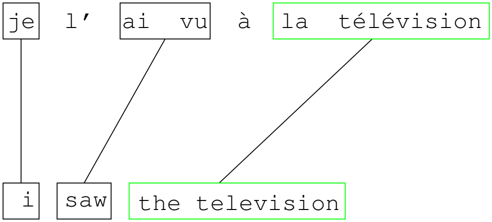
Basic Algorithm:

1. Find all phrase matches with source sentence.
2. Initialize hypothesis list with empty hypothesis.
3. While hypothesis list contains partial translations:
    - Remove next partial translation *h*.
    - Replace h with all its possible 1-phrase extensions.
    - Recombine and prune list.
4. Output highest-scoring hypothesis.

Phrase-Based Search: Hypothesis Extension



Phrase-Based Search: Hypothesis Extension



Update hypothesis scores:

$$\begin{aligned} S_{TM} &\mathrel{+}= \log p(la\_television|the\_television) \\ S_{LM} &\mathrel{+}= \log p(the|i, saw) + \log p(television|i, saw, the) \\ S_{DM} &\mathrel{+}= -1 \end{aligned}$$

$$S = \lambda_{TM} S_{TM} + \lambda_{LM} S_{LM} + \lambda_{DM} S_{DM} + \ldots$$

Phrase-Based Search: Complexity Control

Recombination (dynamic programming): eliminate hypotheses that can never win. Eg, assuming 3-gram LM and simple DM, if two or more hyps have same:

- set of covered source words
- last two target words
- end point for most recent source phrase

⇒ Then need to keep only highest-scoring one.

Pruning (beam search): heuristically eliminate low scoring hyps.

- compare hyps that cover same # of src words
- use scores that include future cost estimate (A* search)
- two strategies: histogram (fix number of hyps), and relative score threshold

Tree-Based Models

Express alignments as mappings between tree structures for source and/or target sentences:

- permits better modeling of linguistic correspondences, especially long-distance movement
- in principle, can impose reordering constraints to speed search

Two kinds:

- asynchronous models: separate parse trees on each side, eg tree-to-tree, tree-to-string, string-to-tree
- synchronous models: one-to-one correspondence between non-terminals, often purely formal syntax without typed non-terminals

Hiero Translation Model

Weighted synchronous CFG, with lexicalized rules of the form, eg:

$$\begin{array}{lcl} X & \Rightarrow & <\text{traverse } X_1 \text{ à } X_2 \text{ , } X_2 \text{ across } X_1 > \\ X & \Rightarrow & < X_1 \text{ profonde}, \text{deep } X_1 > \\ X & \Rightarrow & < \text{la}, \text{the} >, < \text{rivière}, \text{river} >, < \text{nage}, \text{swim} > \end{array}$$

Derivation works top-down, multiplying rule probs to get $p(s, a|t)$:

$$\begin{array}{ll} \Rightarrow & < X_1, X_1 > \\ \Rightarrow & < \text{traverse } X_2 \text{ à } X_3 \text{ , } X_3 \text{ across } X_2 > \\ \Rightarrow & < \text{traverse } X_2 \text{ à nage}, \text{swim across } X_2 > \\ \Rightarrow & < \text{traverse } X_2 \; X_4 \text{ à nage}, \text{swim across } X_2 \; X_4 > \\ \Rightarrow & < \text{traverse la } X_4 \text{ à nage}, \text{swim across the } X_4 > \\ \Rightarrow & < \text{traverse la } X_5 \text{ profonde à nage}, \text{swim across the deep } X_5 > \\ \Rightarrow & < \text{traverse la rivière profonde à nage}, \text{swim across the deep river} > \end{array}$$

Hiero Estimation and Decoding

Rules are induced from phrase table, via aligned sub-phrases, eg:

rivière profonde ||| *deep river*

yields:

$$\begin{aligned} X &\Rightarrow < X_1 \text{ profonde}, \text{deep } X_1 > \\ X &\Rightarrow < \text{rivière } X_1, X_1 \text{ river} > \end{aligned}$$

resulting rule set is very large—requires pruning!

To decode, find highest-scoring parse of source sentence (integrating LM and other features); binarization yields cubic complexity.

Translation Model Recap

Three approaches:

- Word-based: performs poorly, but still used for phrase extraction.
- Phrase-based: state-of-the-art approach; efficient and easy to implement.
- Tree-based (Hiero): better than PB for some language pairs; can be complex and difficult to optimize.

Evaluation of MT Output

Manual evaluation:

- general purpose: adequacy and fluency; system ranking –difficult task for people, especially on long sentences!
- task specific: HTER (postediting); ILR (comprehension testing)
- too slow for system development

Automatic evaluation:

- compare MT output to fixed reference translation
- standard metric is BLEU: document-level n-gram precision (n $= 1 \ldots 4$), with "brevity penalty" to counter precision gaming –flawed, but adequate for comparing similar systems
- many other metrics proposed, eg METEOR, NIST, WER, TER, IQMT, ..., but stable improvement over BLEU in correlation with human judgment has proven elusive

Minimum Error-Rate Training (MERT)

Log-linear model:

$$\log p(t|s) = \sum_i \lambda_i f_i(s,t)$$

Goal: find values of $\lambda's$ to maximize BLEU score. Typically 10's of weights, tuned on dev set of around 1000 sentences.

Problems:

- BLEU(λ) is not convex, and not differentiable (piecewise constant).
- Evaluation at each λ requires decoding, hence is very expensive.

MERT Algorithm

Main idea: use n-best lists (current most probable hypotheses) to approximate complete hypothesis space.

1. Choose initial λ, and set initial n-best lists to empty.
2. Decode using λ to obtain n-best lists. Merge with existing n-bests.
3. Find $\hat{\lambda}$ that maximizes BLEU over n-bests (using Powell's algorithm with custom linemax step for n-best re-ranking).
4. Stop if converged, otherwise set $\lambda \leftarrow \hat{\lambda}$ and repeat from 2.

MERT Algorithm

Main idea: use n-best lists (current most probable hypotheses) to approximate complete hypothesis space.

1. Choose initial λ, and set initial n-best lists to empty.
2. Decode using λ to obtain n-best lists. Merge with existing n-bests.
3. Find $\hat{\lambda}$ that maximizes BLEU over n-bests (using Powell's algorithm with custom linemax step for n-best re-ranking).
4. Stop if converged, otherwise set $\lambda \leftarrow \hat{\lambda}$ and repeat from 2.

MERT yields large BLEU gains, but is highly unstable (more so with larger feature sets). Often difficult to know if gains/losses are due to added feature or MERT variation!

Current SMT Research Directions

- Adaptation: use background corpora to improve in-domain performance, eg by weighting relevant sentences or phrases (Matsoukas et al, EMNLP 2009, Foster et al, EMNLP 2010).
- Applications: CE (Specia et al, MTS 2009); MT/TM combination (He et al, ACL 2010, Simard and Isabelle, MTS 2009); online updates (Hardt et al, AMTA 2010, Levenberg et al, NAACL 2010).
- Discriminative training: stabilize MERT and extend to large feature sets (Chiang et al, EMNLP 2009); principled methods for phrase/rule extraction (DeNero and Klein, ACL 2010; Wuebker et al, ACL2010; Blunsom and Cohn, NAACL 2010).
- Improved syntax and linguistics: restructure trees (Wang et al, ACL 2010), "soft" tree-to-tree structure (Chiang, ACL 2010), model target morphology (Jeong et al, NAACL 2010).