Pseudocode for Edelman & Solan (2009)

The present document contains the pseudocode that did not fit into the eight pages allotted to our paper, *Machine Translation Using Automatically Inferred Construction-based Correspondence and Language Models*, in the Proceedings of the 23rd Pacific Asia Conference on Language, Information, and Computation (PACLIC-23), Hong Kong, December 2009. The paper itself can be found in S.E.'s online archive.

Shimon Edelman and Zach Solan
September 2009

---

1. Learn the source and target languages:
   (a) Learn a grammar G^A for the source language (A).
   (b) Estimate a structural statistical language model $SSLM^A$ for (A). Given a grammar (consisting of terminals and nonterminals) and a partial sentence (sequence of terminals $(t_1 \ldots t_i)$), an SSLM assigns probabilities to the possible choices of the next terminal t_{i+1}.
   (c) Learn a grammar G^B for the target language (B).
   (d) Estimate a structural statistical language model $SSLM^B$ for (B).
2. Learn (automatically or manually) a one-to-many *translation candidate* mapping $\mathcal{T}$ from (A) to (B). This is an association function $\mathcal{T} : a_{s_j} \rightarrow b_{s_j}$ that for each sentence s_j in a training corpus maps sets of symbols (terminals and nonterminals) $A_{s_j} \subset G^A$ evoked by s_j to the corresponding sets of symbols $B_{s_j} \subset G^B$.

Figure 1: Algorithm **LearnMT** (outline; the full pseudocode appears below).

---

1. Given a sentence from (A), parse it to obtain a set of symbols L^A that covers it.
2. Use L^A, the association function $\mathcal{T}$, and any other available priors P to obtain the set of translation candidates L^B.
3. Use L^B and $SSLM^B$ to generate a grammatical sentence in (B) that is the most probable translation of the original sentence in (A).

Figure 2: Algorithm **UseMT** (outline; the full pseudocode appears below).

Algorithm 1: LearnMT

Require: Two CFGs: $G^A = \{a_j\}$, $G^B = \{b_k\}$.
$\{$Each grammar (set of terminals and nonterminals, along with the rules and their probabilities) is acquired by the ADIOS algorithm (Solan et al., 2005).$\}$
Require: Two parallel matched corpora A, B; $|A| = |B| = n$.
Ensure: Translation candidate map $\mathcal{T} : \{a_j\} \to \{b_j\}$, for $\{a_j\} \subset G^A$, $\{b_j\} \subset G^B$.
$\{$First, initialize $\mathcal{T}$ using a bilingual machine-readable dictionary; next, modify $\mathcal{T}$ iteratively using two probability ("distance") matrices, $P(a_{j_1}, a_{j_2})$ for $a_{j_{1,2}} \in G^A$ and $P(b_{k_1}, b_{k_2})$, for $b_{k_{1,2}} \in G^B$ (see text for explanations) $\}$

initialize $\mathcal{T}$ from the MRD;
$\{$PASS 1 — update $\mathcal{T}(a, b)$ with parallel-corpus data (optional); update $P(a_{j_1}, a_{j_2})$ and $P(b_{k_1}, b_{k_2})$:$\}$
for $s_i^A \in A$ and $s_i^B \in B$; $i = 1 \dots n$ **do**
  $L_i^A \Leftarrow \mathrm{parse}(s_i^A)$ $\{L_i^A \subset G^A$ such that $\mathrm{covers}(L_i^A, s_i^A)\}$
  $L_i^B \Leftarrow \mathrm{parse}(s_i^B)$ $\{L_i^B \subset G^B$ such that $\mathrm{covers}(L_i^B, s_i^B)\}$
  for all $a_j \in L_i^A$ **do**
    for all $b_k \in L_i^B$ **do**
      update $\mathcal{T}(a_j) \to b_k$;
    end for
  end for
  for all $a_{j_1} \in L_i^A$ **do**
    for all $a_{j_2} \in L_i^B$ **do**
      update $P(a_{j_1}, a_{j_2})$;
    end for
  end for
  for all $b_{k_1} \in L_i^B$ **do**
    for all $b_{k_2} \in L_i^B$ **do**
      update $P(b_{k_1}, b_{k_2})$;
    end for
  end for
end for
$\{$PASS 2 — update $\mathcal{T}(a, b)$ using $P(a_{j_1}, a_{j_2})$ and $P(b_{k_1}, b_{k_2})$:$\}$
for $s_i^A \in A$ and $s_i^B \in B$; $i = 1 \dots n$ **do**
  $L_i^A \Leftarrow \mathrm{parse}(s_i^A)$ $\{$Reuse L_i^A from Pass 1.$\}$
  $L_i^B \Leftarrow \mathrm{parse}(s_i^B)$ $\{$Reuse L_i^B from Pass 1.$\}$
  for all $a_j \in L_i^A$ **do**
    for all $b_k \in L_i^B$ **do**
      update $\mathcal{T}(a_j, b_k)$ using distance spectrum relaxation, with $P(a_{j_1}, a_{j_2})$ and $P(b_{k_1}, b_{k_2})$ as the corresponding "distance" matrices.
    end for
  end for
end for

Algorithm 2: UseMT

Require: Two CFGs: $G^A = \{a_j\}$, $G^B = \{b_k\}$. {Both learned by ADIOS.}
Require: $\mathcal{T}(a, b)$. {Estimated by Algorithm 1.}
Require: A target sentence $s^A \in \mathcal{L}(G^A)$.
Ensure: The most probable sentence $s^B \in \mathcal{L}(G^A)$, given s^A.

{Use the structured language model over G^B, SSLM^B, to generate the most probable translation of s^A, taking into account prior probabilities dictated by $\mathcal{T}$ and possibly extra sources $P(b|D)$, where $b \in G^B$ and D is the discourse context.}

```
L^A ⇐ parse(s^A);
{The information sources used to determine the discourse context D may include textual and extralinguistic settings of s^A.}
determine D from L^A and any other relevant information sources;
{Map the list L^A into its counterpart L^B using the translation candidate mapping T:}
L^B ⇐ T(L^A);
for all b_j ∈ L^B do
   initialize the prior attached to b_j in the SSLM^B language model;
end for
for all b_i ∈ G^B do
   update the prior of b_i using P(b_i|D);
end for
run SSLM^B starting with the priors computed above, to generate a list S of possible translations ranked by likelihood;
{Post-process (re-rank) S using any additional criteria such as thematic fit:}
for all s_m = (t_1, ..., t_i) ∈ S do
   P(s_m) ⇐ ∏_{n=1:i} P(t_n)
   C(s_m, s^A) ⇐ corresp(parse(s_m), parse(s^A), T) {Goodness of thematic correspondence.}
end for
s^B = arg max_{s_m} (βP(s_m) + (1 − β) C(s_m, s^A));
```

References

Solan, Z., Horn, D., Ruppin, E., and Edelman, S. (2005). Unsupervised learning of natural languages. *Proceedings of the National Academy of Science*, 102:11629–11634.