Constrained Recombination in an Example-based Machine Translation System

Monica Gavrila

University of Hamburg
Faculty of Mathematics, Informatics and Natural Sciences

EAMT
(Research Paper)
29.05.2011

Contents

1. The Framework
2. MT Systems
3. Experiments
4. Conclusions

Framework

Example-based Machine Translation

- Translation by analogy (Nagao, 1984).
- A (small) parallel aligned corpus is enough: database of examples.
- Three steps: matching, alignment and recombination.
- Several Approaches: linear, template-based, hybrid etc.

Template: *(...) gave (...) up* ↔ *(...) a abandonat (...)*

- Languages: Romanian, German, English
- Romanian as under-resourced language

Contents

1. The Framework
2. MT Systems
3. Experiments
4. Conclusions

The Implemented MT Systems

1. $Lin - EBMT$
   - The EBMT baseline system
   - A linear EBMT system
2. $Lin - EBMT^{REC+}$
   - Extends $Lin - EBMT$
   - Hybrid system (linear + template-based)
   - Word-order constraints are used in the recombination step. The constraints are extracted from templates.

Lin – EBMT Matching

- Recursive approach
- Based on surface-forms
- Based on the longest common subsequence (LCS) algorithm (Bergroth et al, 2000)
- A token-index is used to reduce the matching space.

LCS Similarity (LCSS)

Given two strings - $s1$ and $s2$ - the LCSS measure is calculated as

$$LCSS(s1, s2) = LCSS_T(s1, s2) - P * noTG, \qquad (1)$$

where

$$LCSS_T(s1, s2) = \frac{Length(LCS(s1, s2))}{Length(s1)}, \qquad (2)$$

Example

Input s1 = "Saving **names and** phone **numbers** (Add name)"
Sentence in the corpus s2 = "Erasing **names and numbers**"
$LCS(s1, s2) =$ "*names and numbers*"
$LCSS(s1, s2) = \frac{3}{9} - 0.01 * 1 = 0.323.$

Lin – EBMT: Alignment

- Uses GIZA++ results and the longest TL aligned subsequence are used

LCS: ”*technical regulations standards*”
Alignments

- ”*technical - tehnice*” (position 8 in TL),
- ”*regulations - reglementările*” (position 7 in TL) and
- ”*standards - standarde*” (position 23 in TL)

We use further the sequences: ”*reglementările tehnice*” and ”*standarde*”.

Lin – EBMT: Recombination

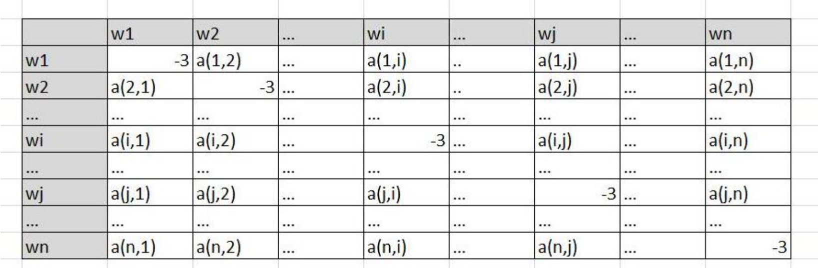
- Input the "*the bag of word sequences*" $\{w_1, w_2, ..., w_n\}$ provided by the alignment step
- The result is the needed translation.
- Uses a "**recombination matrix**"

The Recombination Matrix

Let $A = a(i, j)$ be the "*recombination matrix*". If the outcome of the alignment is n word-sequences $\{w_1, w_2, ..., w_n\}$ which form the output and are not necessarily different, with $w_i = w_{i_1} w_{i_2} ... w_{i_{last}}$, then A is a square matrix of order n that is defined as follows:

$$A = \begin{cases} -3, & \text{if } i = j; \\ -2, & \text{if } i <> j, \\ & w_{i_{last}} w_{j_1} \text{ is} \\ & \text{not in the} \\ & \text{corpus;} \\ \frac{2 * count(w_{i_{last}} w_{j_1})}{count(w_{i_{last}}) + count(w_{j_1})}, & \text{else.} \end{cases} \qquad (3)$$

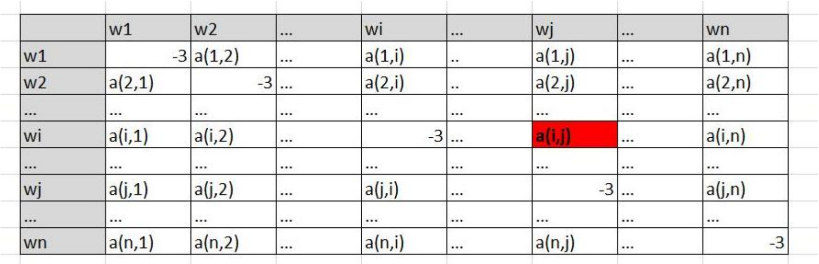
The Recombination Matrix - 2



	w1	w2	...	wi	...	wj	...	wn
w1	-3	a(1,2)	...	a(1,i)	..	a(1,j)	...	a(1,n)
w2	a(2,1)	-3	...	a(2,i)	..	a(2,j)	...	a(2,n)
...	...	...	...	...	...	...	...	...
wi	a(i,1)	a(i,2)	...	-3	...	a(i,j)	...	a(i,n)
...	...	...	...	...	...	...	...	...
wj	a(j,1)	a(j,2)	...	a(j,i)	...	-3	...	a(j,n)
...	...	...	...	...	...	...	...	...
wn	a(n,1)	a(n,2)	...	a(n,i)	...	a(n,j)	...	-3

w_i, $1 \leq i \leq n$, is a sequence.

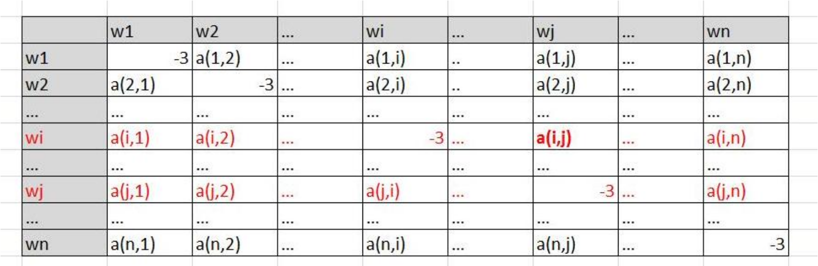
The Recombination Matrix - 2



	w1	w2	...	wi	...	wj	...	wn
w1	-3	a(1,2)	...	a(1,i)	..	a(1,j)	...	a(1,n)
w2	a(2,1)	-3	...	a(2,i)	..	a(2,j)	...	a(2,n)
...	...	...	...	...	...	...	...	...
wi	a(i,1)	a(i,2)	...	-3	...	**a(i,j)**	...	a(i,n)
...	...	...	...	...	...	...	...	...
wj	a(j,1)	a(j,2)	...	a(j,i)	...	-3	...	a(j,n)
...	...	...	...	...	...	...	...	...
wn	a(n,1)	a(n,2)	...	a(n,i)	...	a(n,j)	...	-3

w_i, $1 \leq i \leq n$, is a sequence.

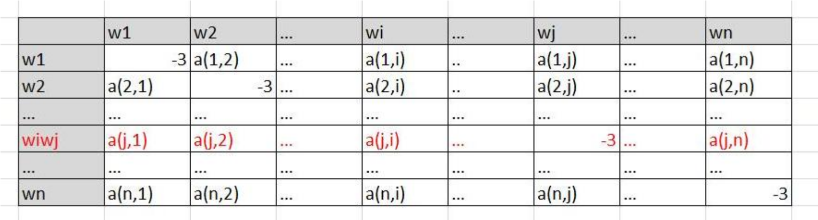
The Recombination Matrix - 2



	w1	w2	...	wi	...	wj	...	wn
w1	-3	a(1,2)	...	a(1,i)	..	a(1,j)	...	a(1,n)
w2	a(2,1)	-3	...	a(2,i)	..	a(2,j)	...	a(2,n)
...	...	...	...	...	...	...	...	...
wi	a(i,1)	a(i,2)	...	-3	...	**a(i,j)**	...	a(i,n)
...	...	...	...	...	...	...	...	...
wj	a(j,1)	a(j,2)	...	a(j,i)	...	-3	...	a(j,n)
...	...	...	...	...	...	...	...	...
wn	a(n,1)	a(n,2)	...	a(n,i)	...	a(n,j)	...	-3

w_i, $1 \leq i \leq n$, is a sequence.

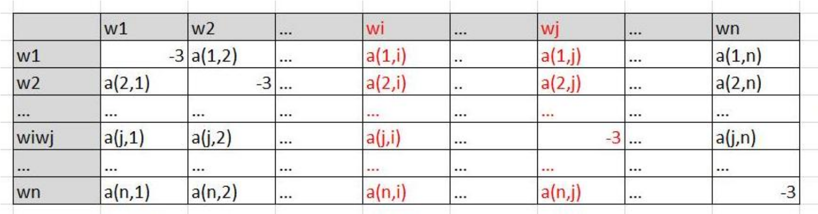
The Recombination Matrix - 2



	w1	w2	...	wi	...	wj	...	wn
w1	-3	a(1,2)	...	a(1,i)	..	a(1,j)	...	a(1,n)
w2	a(2,1)	-3	...	a(2,i)	..	a(2,j)	...	a(2,n)
...	...	...	...	...	...	...	...	...
wiwj	a(j,1)	a(j,2)	...	a(j,i)	...	-3	...	a(j,n)
...	...	...	...	...	...	...	...	...
wn	a(n,1)	a(n,2)	...	a(n,i)	...	a(n,j)	...	-3

w_i, $1 \leq i \leq n$, is a sequence.

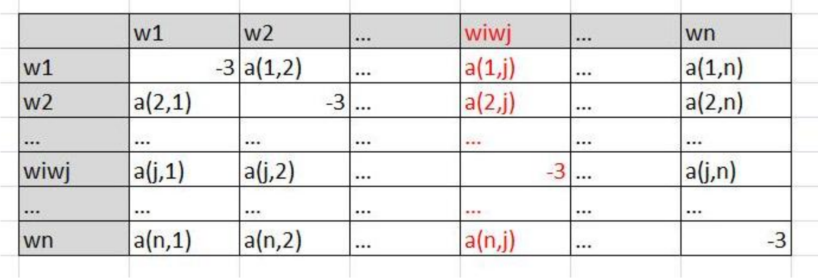
The Recombination Matrix - 2



	w1	w2	...	wi	...	wj	...	wn
w1	-3	a(1,2)	...	a(1,i)	..	a(1,j)	...	a(1,n)
w2	a(2,1)	-3	...	a(2,i)	..	a(2,j)	...	a(2,n)
...	...	...	...	...	...	...	...	...
wiwj	a(j,1)	a(j,2)	...	a(j,i)	...	-3	...	a(j,n)
...	...	...	...	...	...	...	...	...
wn	a(n,1)	a(n,2)	...	a(n,i)	...	a(n,j)	...	-3

w_i, $1 \leq i \leq n$, is a sequence.

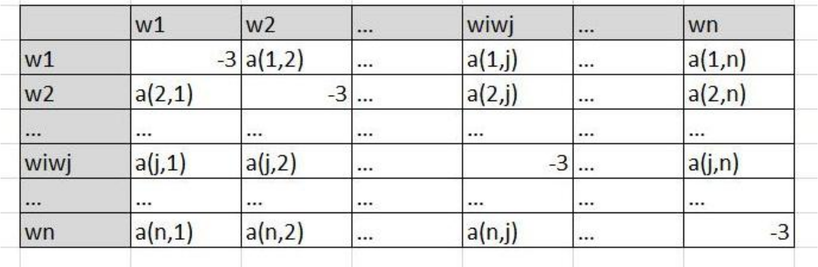
The Recombination Matrix - 2



	w1	w2	...	wiwj	...	wn
w1	-3	a(1,2)	...	a(1,j)	...	a(1,n)
w2	a(2,1)	-3	...	a(2,j)	...	a(2,n)
...	...	...	...	...	...	...
wiwj	a(j,1)	a(j,2)	...	-3	...	a(j,n)
...	...	...	...	...	...	...
wn	a(n,1)	a(n,2)	...	a(n,j)	...	-3

w_i, $1 \leq i \leq n$, is a sequence.

The Recombination Matrix - 2



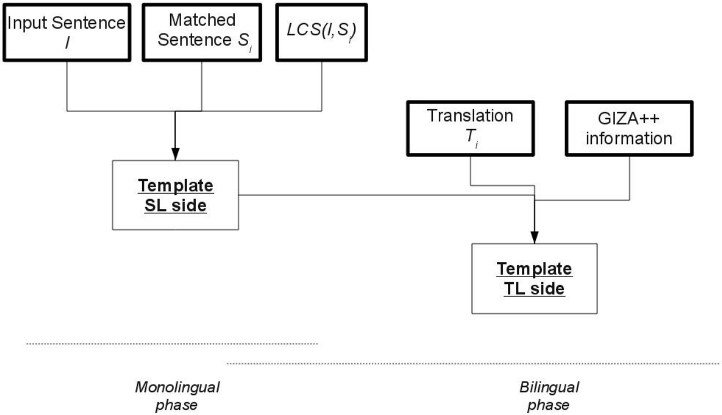
	w1	w2	...	wiwj	...	wn
w1	-3	a(1,2)	...	a(1,j)	...	a(1,n)
w2	a(2,1)	-3	...	a(2,j)	...	a(2,n)
...	...	...	...	...	...	...
wiwj	a(j,1)	a(j,2)	...	-3	...	a(j,n)
...	...	...	...	...	...	...
wn	a(n,1)	a(n,2)	...	a(n,j)	...	-3

w_i, $1 \leq i \leq n$, is a sequence.

$Lin - EBMT^{REC+}$

- Motivation: use the information which is lost in the recombination step of $Lin - EBMT$;
- Mixture of linear and template-based approach;
- Matching and alignment remain as in $Lin - EBMT$;
- Constraints are set on the values from the recombination matrix, by using information extracted from templates.

Template Extraction



$$((TF_{SL})^*(VAR_{SL})^*)^*TF_{SL}((TF_{SL})^*(VAR_{SL})^*)^* \leftrightarrow ((TF_{TL})^*(VAR_{TL})^*)^*$$

Template-Example

The input

press and hold clear to delete the characters more quickly .

Matched sentence and alignment

pentru a sterge simultan toate **caracterele** cand scrieti un mesaj , apasati optiuni si selectati stergeti textul .

to delete all **the characters** at once when writing a message press options and select clear text .

Template-Example

The input

press and hold clear to delete the characters more quickly .

Template

to&&1&& **delete&&2&&** VAR3 **the&&4&&** **characters&&5&&** VAR6 NOALIGN7 VAR8_18 .&&19&& ↔ **pentru&&1&&** **a&&1&&** **sterge&&2&&** VAR6 VAR3 **caracterele&&5&&** VAR8_18 .&&19&&

Constraints

1. **The First-Word-Constraint (C.1)**: A constraint C.1 refers to the first word of the output.
2. **TLSide-Template-Constraint (C.2)**: the C.2 constraints are deduced only from the TL side of each of the templates extracted.
3. **Whole-Template-Constraint (C.3)**: the C.3 constraints are extracted considering each of the templates, together with the input sentence, and the alignment information.

The result: a set $C = \{(word_i, word_j)\}$ of constraints: The sequence $word_i word_j$ is not allowed.

C.1 Constraints

The input

to delete the characters more quickly press and hold clear.

Template

to&&1&& **delete&&2&&** VAR3 **the&&4&&**
characters&&5&& VAR6 NOALIGN7 VAR8_18 **.&&19&&**
↔ **pentru&&1&&** **a&&1&&** **sterge&&2&&** VAR6 VAR3
caracterele&&5&& VAR8_18 **.&&19&&**

C.2 Constraints

Template

to&&1&& **delete&&2&&** VAR3 **the&&4&&** **characters&&5&&** VAR6 NOALIGN7 VAR8_18 **.&&19&&** ↔ **pentru&&1&&** **a&&1&&** **sterge&&2&&** VAR6 VAR3 **caracterele&&5&&** VAR8_18 **.&&19&&**

New Recombination Matrix

$$A = \begin{cases} -3, & \text{if } i = j; \\ -2, & \text{if } i <> j, \\ & w_{i_{last}} w_{j_1} \text{ is not in} \\ & \text{the corpus or} \\ & (w_{i_{last}} w_{j_1}) \in C; \\ \frac{2 * count(w_{i_{last}} w_{j_1})}{count(w_{i_{last}}) + count(w_{j_1})}, & \text{else.} \end{cases} \quad (4)$$

Another Recombination Matrix

$$A = \begin{cases} -3, & \text{if } i = j; \\ -1, & \text{if } i <> j, \\ & w_{i_{last}} w_{j_1} \text{ is not in} \\ & \text{the corpus;} \\ -2, & (w_{i_{last}} w_{j_1}) \in C; \\ \frac{2 * count(w_{i_{last}} w_{j_1})}{count(w_{i_{last}}) + count(w_{j_1})}, & \text{else.} \end{cases} \tag{5}$$

Contents

1. The Framework
2. MT Systems
3. Experiments
4. Conclusions

The Experimental Settings

- 2 EBMT systems: $Lin - EBMT$, $Lin - EBMT^{REC+}$
- 2 language pairs, both directions of translations: English-Romanian, German-Romanian
- 1 corpus: RoGER

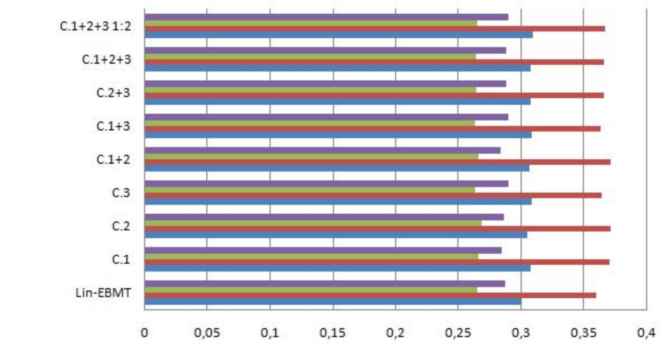
The Corpus: RoGER

- Developed between 2005 and 2006, at the University of Hamburg, NatS, together with Natalia Eliţa
- Romanian, German, English, Russian
- Manual of an electronic device
- 2333 sentences, between 25K and 27K words
- Manually verified
- No diacritics, some data replaced with meta-notations

Experimental Setting

- Training: 2200 sentences, approx 27 K items, 13 words the average sentence length
- Test: 133 sentences, approx 1.6 K items, 12.3 words the average sentence length

BLEU (Papineni et al., 2002) Scores



	Lin-EBMT	C.1	C.2	C.3	C.1+2	C.1+3	C.2+3	C.1+2+3	C.1+2+3 1:2
🟪 Romanian-German	0,2867	0,2842	0,2857	0,2891	0,2836	0,2891	0,2875	0,2875	0,2894
🟩 German-Romanian	0,2643	0,2658	0,2682	0,2627	0,2654	0,2627	0,2633	0,2633	0,2646
🟥 Romanian-English	0,3597	0,3695	0,3711	0,3633	0,3712	0,3632	0,3656	0,3656	0,3668
🟦 English-Romanian	0,2997	0,3067	0,3042	0,3083	0,3062	0,3083	0,3073	0,3073	0,3085

Evaluation

Best Score Differences:

- English-Romanian: 0.0088
- Romanian-English: 0.0115
- German-Romanian: 0.0039
- Romanian-German: 0.0027

Contents

1. The Framework

2. MT Systems

3. Experiments

4. Conclusions

Conclusions & Further Work

- Impact of word-order constraints

Further work:

- Additional constraints;
- Priorities for the constraints are used (weighting);
- Different corpus and languages;
- Manual analysis of the data;
- N-grams of several lengths etc.

Discussions

Thank you for your attention!

Suggestions ... Questions ...