SoSe 06 – Project “Machine Translation” - Part II

Example Based Machine Translation by means of Pattern Extraction – III (Recombination)

Monica Gavrila

gavrila@nats.informatik.uni-hamburg.de

Overview

- Definition of Translation Patterns
- Pattern Extraction
  - Monolingual Phase
  - Bilingual Phase
  - Alignment

Done

- Recombination
  - Pattern Retrieval
  - Core Recombination Method
  - Translation Confidence Score

Today

- Evaluation

Recombination I/O

- **Input:** an SL sentence and a set of translation patterns extracted from the corpus

- **Output:** one or more TL translation strings, ranked according to a translation confidence score

Recombination Algorithm - Overview

- Retrieve patterns that cover the SL input
- Extract, rank and choose ***base patterns***
- Bound unmatched segments to variables
- Retrieve segments from the remaining set of patterns
- Find TL equivalents
- Rank TL equivalents – if needed

Recombination - Example

- **Input:**

SL sentence:

The style of driving must always be adapted to suit traffic situation.

Possible existing patterns:

X1 must always be adapted to suit Y1 – X2 muß stets Y2 angepaßt werden.

the style of driving X1 – die Fahrweise X2

X1 traffic situation – X2 der Verkehrsituation

- **(Wanted) Output:**

Die Fahrweise muß stets der Verkehrsituation angepaßt werden

Recombination - Steps

- **Pattern Retrieval**
- Core Recombination Method
- Recursive Matching
- Partial Translations
- Translation Confidence Score

Base Pattern (BP)

- **Base Patterns** = patterns whose SL side cover the SL input to the greatest extent

Base Pattern Features

- Four conditions to be fulfilled:
  - the SL side of the *BP must be less than or equal in length (number of words) to the SL input*
  - *the SL side of the BP consists only of lexical items of which the SL input is composed*
  - *the lexical items in the SL side of the BP appear with the same frequencies as they do in the SL input*
  - *when the variables of the SL side of the BP have been instantiated with the string values of the unmatched segments of the SL input, that sequence of lexical items must match exactly the SL input*

Base Patterns - Example

- SL sentence:

The style of driving must always be adapted to suit traffic situation.

- Patterns:

X1 must always be adapted to suit Y1 (– X2 muß stets Y2 angepaßt werden).

X1 must always Y1 (– X2 muß stets Y2).

X1 must always be adapted to suit traffic situation and road surface (– X2 muß stets der Verkehrsituation und dem Fahrbahnzustand angepaßt werden).

Pattern Retrieval Algorithm

- Retrieve candidate translation patterns – patterns that share one or more lexical items with the SL input
- Filter the candidate translation patterns according to the four conditions in the definition of the *base pattern*
- *Rank the base patterns (ratio of cover); the longest patterns are preferred.*

$$Cover = \frac{Length(BP)}{Length(Input)}$$

Pattern Retrieval - Example

- SL sentence:

The style of driving must always be adapted to suit traffic situation

- Patterns:

X1 must always be adapted to suit Y1 – .....

X1 must always Y1 – .....

X1 must always be adapted to suit traffic situation and road surface – .....

The style of X1 - .....

X1 must Y1 - .....

X1 should be adapted Y1 - .....

Recombination - Steps

- Pattern Retrieval
- **Core Recombination Method**
- Recursive Matching
- Partial Translations
- Translation Confidence Score

Core Recombination Method

- **Input:** SL input and a base pattern

- **Output:** TL string(s)

Core Recombination Algorithm

- Instantiate the variables on the SL side of the BP with the string values of the unmatched segments in the SL input
- Retrieve the TL equivalents (match not only in terms of string value, but also alignment type)
- Insert them into the appropriate variables on the TL side of the BP
- Form TL string(s)

Core Recombination Algorithm - Observations/Problems

- Translation ambiguity
  - more TL equivalents (-> confidence score)
  - non-bijective alignment (-> matching **successive alignments** of the variables in the BP, rather than matching successive variables)

- Parts of text for which there is
  - No direct matching -> Recursive matching
  - No matching at all -> Partial translations

Core Recombination Algorithm Example

- Input:

  The style of driving must always be adapted to suit traffic situation.

  X1 must always be adapted to suit Y1 – X2 muß stets Y2 angepaßt werden.

- Variable instantiation:

  The style of driving must always be adapted to suit traffic situation.

  X1 must always be adapted to suit Y1 – X2 muß stets Y2 angepaßt werden.

Core Recombination Algorithm Example

- Retrieving TL equivalents:

  The style of driving must always be adapted to suit traffic situation.

  X1 must always be adapted to suit Y1 – X2 muß stets Y2 angepaßt werden.

  the style of driving X1 – die Fahrweise X2

  X1 traffic situation – X2 der Verkehrsituation

- Output:

  Die Fahrweise muß stets der Verkehrsituation angepaßt werden.

Recombination - Steps

- Pattern Retrieval
- Core Recombination Method
- **Recursive Matching**
- Partial Translations
- Translation Confidence Score

Recursive Matching

- In case of no direct matching of SL variables:
  - obtaining more training data, or
  - searching for ‘indirect’ matching: **recursive matching**
- Only bijective (1:1 basis) aligned texts are considered
- Match successively shorter parts of an SL fragment, from left to right, against the SL fragments of a translation pattern.
- Longest match considered
- TL equivalents are concatenated naively, according to the order of the matches with the portions of the SL fragment

Recursive Matching Example

traffic situation and road surface – NO PATTERN

Looking for:

traffic situation and road - NO PATTERN

traffic situation and – NO PATTERN

traffic situation - OK

X1 traffic situation Y1 – X2 der Verkehrsituation Y2

Recursive Matching Example (2)

and road surface NO PATTERN

and road – NO PATTERN

and - OK

X1 and Y1 – X2 und Y2

road surface - OK

X1 road surface Y1 - X1 dem Fahrbahnzustand X2

Result:

der Verkehrsituation und dem Fahrbahnzustand

Recombination - Steps

- Pattern Retrieval
- Core Recombination Method
- Recursive Matching
- **Partial Translations**
- Translation Confidence Score

Partial Translations

- **Partial Translation:** the SL input cannot be fully matched against a set of translation patterns

- **Result:** un-instantiated variables on the TL side of the BP.

- **Example:**

Input: The style of driving must always be adapted to suit traffic situation.

Output: Die Fahrweise muß stets *Y2* angepaßt werden

Recombination - Steps

- Pattern Retrieval
- Core Recombination Method
- Recursive Matching
- Partial Translations
- **Translation Confidence Score**

Translation Confidence Score

- Needed in case of translation ambiguity: more than one translation solution
- Example:

The style of driving must always be adapted to suit traffic situation.

Existing patterns:

X1 must always be adapted to suit Y1 – X2 muß stets Y2 angepaßt werden.

the style of driving X1 – die Fahrweise X2

X1 traffic situation – X2 der Verkehrsituation

X1 traffic situation – X2 die Verkehrsituation

Output(s):

1. Die Fahrweise muß stets *der Verkehrsituation* angepaßt werden.

2. Die Fahrweise muß stets *die Verkehrsituation* angepaßt werden

Translation Confidence Score (2)

- Translations ranked according to a confidence score, that uses:
  - bi-gram model (BG)
  - component that measures the boundary friction (BF)
  - bilingual similarity/probability (BS)
  - penalty, weights

$$Confidence = (w_1(BG) + w_2(BF) + w_3(BS))(1 - p)$$

Bi-gram Model (BG)

- measures the probability that each translation solution is a valid sequence of words in TL
- Computed from the TL sentences, of the corpus used for pattern extraction (monolingual text)
- Probabilistic model -> more reliable as more data is supplied
- Uses Dice's co-efficient (see link given last talk)

BG Model

- **Formula:**

$$BG = \frac{\sum_{i=1}^{n-1} 2 \frac{|w_i, w_{i+1}|}{|w_i| + |w_{i+1}|}}{n-1}$$

- **On-line references:**
  - **www.clsp.jhu.edu/ws99/projects/mt/wkbk.rtf**
  - **http://en.wikipedia.org/wiki/Bigram**
  - **http://acl.ldc.upenn.edu/J/J93/J93-2003.pdf (Brown et al., 1993)**

BG Model Example

- BG (“**Die Fahrweise muß stets der Verkehrsituation angepaßt werden**”) = ?

Dice1=Dice(Die|Fahrweise)
Dice2=Dice(Fahrweise|muß)
Dice3=Dice(muß|stets)
Dice4=Dice(stets|der)
Dice5=Dice(der|Verkehrsituation)
Dice6=Dice(Verkehrsituation|angepaßt)
Dice7=Dice(angepaßt|werden)

BG = (Dice1+...+Dice7) / 7

Boundary Friction Score (BF)

- Measures the amount of boundary friction that appears when TL fragments are connected to TL variables
- Measured considering ***N*** *words of the TL fragments in the BP preceding and being after the inserted TL fragment*
- **Formula:**

$$BF = \frac{1}{n} \sum_{i=1}^{n} \frac{Overlap(F_i, F_{prev}) + Overlap(F_i, F_{subsqt})}{2\mathrm{N}}$$

BF Score Example

Die Fahrweise muß stets der Verkehrsituation angepaßt werden.

Obs: *N=5*

S1. Die Fahrweise muß S2. stets der Verkehrsituation angepaßt werden

1. Look corpus for ***muß***
2. Sequence of words before ***muß*** are recorded

[man muß] *[die fahrweise muß]* *[....]*

3. Each of these are compared with the main sequence S1
4. Maximum overlap is calculated (the word considered is ignored) *(in the example: 2)*
5. Do steps 1-4 for S2 (sequences of words after ***stets***)
6. Calculate using the given formula

Bilingual Similarity/Probability (BS)

- Measure of the probability that the TL fragments retrieved during recombination are translations of the SL fragments bound to the SL variables
- Bilingual similarity metric – used also when aligning the text (Pattern extraction – given in last talk)
- **Formula:**

$$BS_p = \frac{BLD + |Cognates|}{1 + |Cognates|}$$

$$BLD = 2\frac{|S \cap T|}{|S| + |T|}$$

$$BS = \frac{\sum_{i=1}^{n} BS_i}{n}$$

BS Example

The style of driving must always be adapted to suit traffic situation.

Die Fahrweise muß stets der Verkehrsituation angepaßt werden.

BS1=BS(“The style of driving” | “Die Fahrweise”) - as in pattern extraction

BS2=BS(“traffic situation”|”der Verkehrsituation”)

BS=(BS1+BS2) / 2

Other Elements

- Penalty *p*

$$p = \frac{UnInstVarsTLBP}{InstVarsTLBP}$$

  - *1-p never becomes zero. p is scaled by 0.9*

- *Weights -> the translation confidence score lies between 0 and 0.9*

$$w_1 + w_2 + w_3 = 1.0$$

  - *In experiments, all considered 1/3*

Recombination Algorithm - Conclusion

- Retrieve patterns that cover the SL input
- Extract, rank and choose base patterns
- Bound unmatched segments to variables
- Retrieve segments from the remaining set of patterns
- Find TL equivalents
- Rank TL equivalents – if needed

Thank You!

Questions? Problems?