Combinatory Examples Extraction for Machine Translation[0]

Alberto Simões and José João Almeida

Universidade do Minho, Campus de Gualtar, 4710-057 Braga

Abstract

One of the bottlenecks of example-based machine translation (EBMT) is to be able to amass automatically quantities of good examples. In our work in EBMT, we are investigating how far one can go by performing example extraction from parallel corpora using Probabilistic Translation Dictionaries to obtain example segmentation points. In fact, the success of EBMT highly depends on examples quality and quantity, but also in their length. Thus, we give special importance on methods to extract different size examples from the same translation unit. With this article we show that it is possible to extract quantities for examples from parallel corpora just using probabilistic translation dictionaries extracted from the same corpora.

1 Introduction

Recent research on Machine Translation (MT) have been focused on corpus based translation, and two specific trends: Statistic Based Machine Translation (SBMT) and Example Based Machine Translation (EBMT — Somers, 1999; Hutchins, 2005).

In this second approach to MT, parallel corpora is used as a base of examples of previously done translations. The translation process is defined as:

$$
\begin{aligned}
&translate : sentence \times tmdb \longrightarrow sentence \\
&translate(s, db) \stackrel{\text{def}}{=} \\
&\quad \underline{let} \quad l1 = split(s) \\
&\qquad\quad\; l2 = <match(db, x) \mid x \in l1> \\
&\quad \underline{in} \quad\; recombine(l2)
\end{aligned}
$$

Basically, we will first split the sentence to be translated in segments, try to translate them using a database (*db*) of translations, and then recombine the translations in a final sentence. To split into segments is important because it is highly improbable we find the full sentence we want to translate in the examples database.

This approach to translation is being one of the more promising, mainly because it just depends on corpora. Traditional translation approaches used language specific grammars that were difficult to write and error prone[1].

So, EBMT quality highly depends on the used corpora quality and the number and length of the examples available[2]. In fact, the granularity of the examples is very important during EBMT. While a full-sentence example if very liable, it is not flexible (not easy to reuse). In the other hand, a single word example is very flexible, but not liable for translation (because it lacks context). So, we need examples with different sizes between these two extremes.

This paper describes an algorithm to extract segments from parallel corpora, using different example granularities and combinations. This algorithm is based on a translation matrix where translation probabilities between words are set, and the relationship between words and segments are extracted.

[0]This work has been partially funded by Fundação para a Ciência e Tecnologia of Portugal through grant POSI/PLP/43931/2001, and co-financed by POSI, within Linguateca.

[1]In this paper we will not discuss the differences between SBMT and EBMT systems. For some discussion on the subject please refer to Hutchins, 2005

[2]EBMT depends as well on the type of corpora used, as it should be the same kind of the text we are trying to translate.

These segments are then consolidated in a database with occurrence count for probabilities measures.

To create the matrix, we need probabilities between words, which are extracted from Probabilistic Translation Dictionaries (PTDs).

PTDs are one kind of the results obtained from a word aligner. They associate to each word a set of translation hypothesis with a probability. By analyzing large number of parallel texts in English and Portuguese and in Spanish and Portuguese, we have got sizable dictionaries for the two language pairs (in the two directions).

We use PTDs instead of traditional bilingual dictionaries for two main reasons:

- PTDs are easier to obtain, as they are automatically extracted from parallel corpora, and they are highly dependent from the corpus they are based on. Thus, the translations obtained from PTDs should be more adequate to segment the corpus they were extracted from than the translations from traditional bilingual dictionaries;
- PTDs include probabilistic information which will be crucial for the examples extraction algorithm we present bellow. Without this kind of information it would be quite hard to find suitable points where to cut examples.

Section 2 describes NATools, the scalable PTDs extractor used. Follows the main section with our algorithm. In section 4 we propose a simple way to evaluate the extracted examples.

2 Probabilistic Translation Dictionaries extraction

This section presents a brief explanation of the structure of NATools (Hiemstra, August 1996; Simões, 2004; Simões & Almeida, 2003), the Probabilistic Translation Dictionary extractor we used.

NATools is composed of different modules, which work as a pipeline: a corpora splitter, a corpora encoder, a co-occurrence counter, the EM-Algorithm, the dictionary creation, and junction at the end.

The extractor processes two sentence aligned texts (a sequence of translation units), and creates a probabilistic translation dictionary with the following structure:

$$w_\alpha \rightharpoonup (occur \times w_\beta \rightharpoonup P\left(\mathcal{T}\left(w_\alpha\right) = w_\beta\right))$$

That is, we map to each word on the source language ($\mathcal{L}_\alpha$) a pair: the occurrence counter of that word on the corpora ($occur$), and another map, from possible translations from target language (w_β) to its respective probability of being a translation.

The following example is from EuroParl (Koehn, 2002) with more than a million translation units, and 30 million words in each language. The resulting PTD include about 100 000 entries, each with 1 to 8 possible translations.

```
** Word: europe
** OccurrenceCount: 42853

       europa: 94.71 %
     europeus:  3.39 %
      europeu:  0.81 %
     europeia:  0.11 %

** Word: stupid
** OccurrenceCount: 180

    estúpido:  17.55 %
    estúpida:  10.99 %
   estúpidos:   7.41 %
     avisada:   5.65 %
     direita:   5.58 %
     impasse:   4.48 %
     ocupado:   3.75 %
```

While it is possible to perform translation using only PTDs, it will end being a word-by-word translation, not respecting the grammar of the target language, but preserving the order of the source-language unit.

3 Example Extraction Algorithm

The examples extraction algorithm can be applied both to translation units present in

the corpus from where we extracted PTDs, as well as to other translations units. Quality will depend on the knowledge of PTDs regarding the words used in the translation units.

With each translation unit, we create a matrix where the relationship between words will be marked, and then the examples extracted. This is an approach similar to Carl, 2001 and Melamed, 1999.

3.1 Alignment Matrix Creation

For each translation unit, a matrix is created with the probabilities of translations between words, and a translation diagonal is searched. This diagonal (normally near the main matrix diagonal) will include the cells where words are translations. This is explained bellow:

1. create the translation matrix where each row is a word in the source segment (w_{SL}), and each column is a word in the target segment (w_{TL}). For each cell in the matrix add the mean of the translations probabilities from the source language to the target language and from the target language to the source language:

$$\frac{\mathcal{P}(\mathcal{T}(w_{TL}) = w_{SL}) + \mathcal{P}(\mathcal{T}(w_{SL}) = w_{TL})}{2}$$

   While the PTDs extraction algorithm creates a matrix with all translation units in the corpus and fills each cell with the co-occurrence count for each word pair, this algorithm uses those obtained values to extract segments.

2. for words that do not appear in the corpus used to extract the PTDs, we do not have their probabilities of being a translation of any other word. This happens a lot with proper nouns and numbers.

   Thus, we use a filter to find words (or numbers) that are written in the same way in both languages and, for these words relationships we force a probability value of 80%.

   Table 1 shows the translation matrix[3] after the first two steps of the algorithm. Translations probabilities were added, and probabilities for words written in the same way were set to 80%.

	el	perro	ladró	al	gato	.
o	60.6	0.0	0.0	0.0	0.0	0.0
cão	0.0	74.5	0.0	0.0	0.0	0.0
ladrou	0.0	0.0	71.5	0.0	0.0	0.0
ao	2.2	0.0	0.0	45.9	0.0	0.0
gato	0.0	0.0	0.0	0.0	80.0	0.0
.	0.0	0.0	0.0	0.0	0.0	80.0

Table 1: Alignment matrix

3. we are working primarily with Portuguese, Spanish and English. While translations between these languages can change words in the sentences, as well as some phrases, it is known that most of the translations will have a similar order in the words (if we consider only Indo-European languages). This means that the correct translation relations will appear in a diagonal near the matrix main diagonal.

   Thus, we use a smoothing algorithm in order to lower the cells' values according to their distance to the main diagonal (we just multiply the values by a distance factor).

4. while in the previous example we used Portuguese and Spanish, whose word order in the sentence are normally the same, this does not happen for languages pairs as Portuguese and English.

   With this in mind, we define a set of patterns for words' common order change in the sentence, like the name/adjective order[4]. Table 2 shows two translation matrices where patterns would be found. Note that these patters are language specific.

[3] This is a simple example just to illustrate the translation matrix.

[4] These patterns are described using a simple Domain Specific Language that will not be detailed in this article

	big	dog
cão	0.0	35.5
grande	24.2	0.0

	natural	language	processing
processamento	0.0	0.0	23.3
de	0.0	0.0	0.0
linguagem	0.0	39.1	0.0
natural	25.0	0.0	0.0

Table 2: Alignment matrix examples where patterns would be applied

Patterns found are automatically marked with all cells in the rectangular area defined by the pattern cells. These are marked with a special mark so we can identify patterns later.

5. the most important bit of the algorithm is the diagonal-finder, constructing the translation diagonal for each translation unit. This translation diagonal passes by the cells with more probability in the matrix.

   This algorithm relies on anchor points. A point $x_{i,j}$ is an anchor point if its value is 20% bigger than all elements in the i row and 20% bigger than all elements in the j column. Pattern blocks (rectangles with more than 1×1 of size) are also anchor points.

   When no near anchor points are found, the algorithm proceeds by enlarging a rectangle step by step, until it finds one. These blocks include at two of their corners (top left, and bottom right) an anchor point, or pattern block. Notice that pattern blocks are totally included (if no anchor point was found, which is highly improbable).

3.2 Examples Extraction

As soon as the translation diagonal is computed, we extract a tree of examples:

1. step the translation diagonal, finding blocks. These blocks can have any size, from the one unit square (word to word segments), and getting bigger to (in extreme) the size of the translation unit

2. extract single blocks. These will result in the following kind of dictionary:

```
todos  /  todos
os  /  los
dias  /  días
...
poder mandar os seus / poder en-
viar a sus
filhos para a / hijos a la
escola / escuela
```

   Notice that there are one-word entries (the ones found in one-unit squares) and bigger examples. The examples extracted are often not traditional linguistic constituents like phrases or multiword expressions, as is the case in most of the EBMT literature as well.

3. The we concatenate two and three contiguous examples extracted from the same translation unit creating bigger examples.

   For instance, from the previous example, if we join two examples, we get:

```
todos os / todos los
os dias / los días
   ...
poder mandar os seus fil-
hos para a / poder en-
viar a sus hijos a la
filhos para a escola / hi-
jos a la escuela
```

   And joining three examples:

```
todos os dias / todos los días
 ...
poder mandar os seus fil-
hos para a escola / poder en-
viar a sus hijos a la escuela
```

4. Finally, examples are joined under the same head entry and counted. The more frequently a given example is extracted from the corpus, the higher probability of being commonly used. See two entries in the examples database:

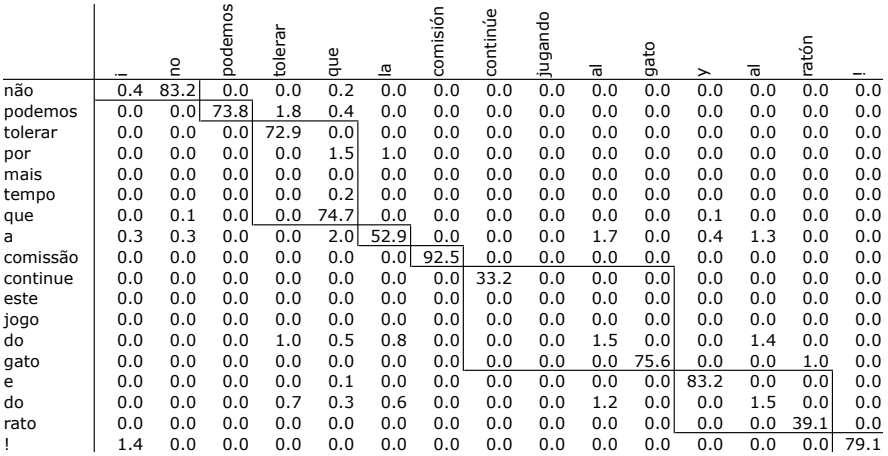

Figure 1: Alignment matrix with translation diagonal

```
todos os
   642 todos los
     3 cada año
     2 todas las
     2 los de todos

todos os dias
     2 todos los días
     1 recordemos todos los
```

While it should be possible to extract a quality measure from the alignment matrix, at the moment we are just using the number of times an example was actually extracted from the corpus.

3.3 Dynamic Examples Extraction

Because we are having big problems on storing efficiently examples, and because the granularity of the examples needed highly depend on the sentence we are trying to translate, we are developing a method for dynamic examples extraction from parallel corpora.

Instead of simply perform the examples extraction from the alignment matrix in batch mode, we are preparing to store a bit-compressed version of all translation unit matrices, and consult them at run-time.

For each segment we are trying to translate, we search the corpus for translation units where it occurs. We retrieve the alignment matrix and calculate on-the-fly the example translation for that segment.

Notice that the extracted translation using this methodology is not necessarily just the translation of the source segment we had to translate. This happens because we will fit the source segment in the matrix, and cut the smaller example that contains it.

4 Evaluation

Although these examples may be useful for tasks other than EBMT, we are primarily interested in evaluating their quality for MT.

A primarily evaluation was done extracting random examples from the ones extracted, before grouping them under the same head entry. We skipped 1000 examples, and then extracted 100 examples: taking one, skipping 100 taking other, and so on. We did this both for Portuguese/English and Portuguese/Spanish. This is shown on table 3, in the column "Single".

The second evaluation was done using the examples grouped under the same head entry. We joined equal examples, counting the number of times they appear in our database. Then, sorted from more occurrence to less occurrence. Done the same process as before to select 100 unities. Then, we did the same for 100 examples of pairs (glued examples). The result of evaluating these 200 units is shown in the column "Accumulated."

The selection of good examples was done just in case they mean exactly the same. For instance, pairs like:

```
tudo para / todo lo posible para
```

were considered bad pairs. While they can

	Single			Accumulated		
	Nr of exs	Good	%	Nr of exs	Good	%
Portuguese/Spanish	100	66	66%	200	191	95%
Portuguese/English	100	45	45%	200	156	78%

Table 3: Portuguese/Spanish and Portuguese/English examples evaluation

be used in the same context and without major problems they do not mean really the same.

Also, we should note that the quality of the examples (the ones in the second column) raise with the number of translation units we process. For instance, the portion processed from the Portuguese/English corpus is 20% of the portion already processed for the Portuguese/Spanish corpus.

5 Conclusions

Examples extracted so far are of reasonable quality as our first experience shown. Although we need to prepare an expedite way evaluate them, we are continuing extracting examples. The process is not as fast as we would like (takes about 12 hours to extract examples from 40000 units in a Pentium IV 3GHz) but we are continuously extracting more and more examples from our corpora.

Examples are not useful just for MT. They can be used for anything parallel corpora is used, as they roughly consist on translation units as well. Also, many of the tasks that usually deal with parallel corpora can gain using an example-based parallel corpora.

We are working on a characterization of the PTDs and the examples extracted, in order to analyse the kinds of material that are possible to extract with this algorithm, and investigate which if any linguistic clues (like part of speech, lemmatization or canonical order) can help us in both reducing the number of examples and increase their quality and generality.

Acknowledgment

We would like to thank Diana Santos for the help reviewing the article and for relevant comments.

References

Carl, M. (2001). Inducing probablilistic invertible translation grammars from aligned texts. In M. Carl & A. Way (Eds.), *Workshop on example-based machine translation* (pp. 12–22). Santiago de Compostella, Spain.

Hiemstra, D. (August 1996). *Using statistical methods to create a bilingual dictionary.* Unpublished master's thesis, Department of Computer Science, University of Twente.

Hutchins, J. (2005). Towards a definition of example-based machine translation. In *Mt summit x workshop on example-based machine translation.* Phuket, Thailand.

Koehn, P. (2002). *Europarl: A multilingual corpus for evaluation of machine translation.* (Draft, Unpublished, `http://people.csail.mit.edu/~koehn/publications/europarl.ps`)

Melamed, I. D. (1999). Bitext maps and alignment via pattern recognition. *Computational Linguistics*, *25*(1), 107–130.

Simões, A. M., & Almeida, J. J. (2003). Natools – a statistical word aligner workbench. *SEPLN*.

Simões, A. M. B. (2004). *Parallel corpora word alignment and applications.* Unpublished master's thesis, Escola de Engenharia - Universidade do Minho.

Somers, H. (1999). Review article: Example based machine translation. *Machine Translation*, *14*(2), 113–157.