Integrating corpus-based and rule-based approaches in an open-source machine translation system

Felipe Sánchez-Martínez, Juan Antonio Pérez-Ortiz, Mikel L. Forcada
Transducens Group – Departament de Llenguatges i Sistemes Informàtics
Universitat d'Alacant, E-03071 Alacant, Spain
{fsanchez,japerez,mlf}@dlsi.ua.es

Abstract

Most current taxonomies of machine translation (MT) systems start by contrasting rule-based (RB) systems with corpus-based (CB) ones. These two approaches are much more than theoretical boundaries since many working MT systems fall within one of them. However, hybrid MT systems integrating RB and CB approaches are receiving increasing attention. In this paper we show our current research on using CB methods to extend a MT system primarily designed following the RB approach. Specifically, the open-source MT system Apertium is being extended with a set of CB tools to be also released under an open-source license, therefore allowing third parties to freely use or modify them. We present CB extensions for Apertium allowing (a) to improve its part-of-speech tagger, (b) to automatically infer the set of transfer rules, and (c) to tackle the problem of the translation of polysemous words. A common feature of these CB methods is the use of unsupervised corpora in the target language of the MT system. The resulting hybrid system preserves most of the advantages of the RB approach while reducing the need for human intervention.

1 Introduction

In the last two decades, with the growing availability of machine-readable (monolingual and parallel) corpora, corpus-based (CB) approaches to machine translation (MT) have strengthened and become useful for real applications. Older approaches, specially rule-based (RB) methods, however, have not fallen into oblivion since they still keep some advantages.

On the one hand, rule-based MT (RBMT) may attain high general performance but at the expense of the large costs needed for building the necessary linguistic resources (Arnold, 2003). On the other hand, corpus-based MT (CBMT), such as statistical MT (SMT) or example-based MT (EBMT), heavily rely upon the availability of parallel corpora, and usually produce ungrammatical outputs; however, they allow for fast prototyping. Some *minor* languages (Forcada, 2006) may not easily benefit from CB approaches since the available parallel corpora are not large enough.

Hybrid MT systems integrating both approaches seem a plausible alternative to the *pure* ones. There are different ways in which a MT system may be considered hybrid; these include but are not limited to:

- the use of automatically obtained monolingual or bilingual dictionaries which are post-edited (or not) by humans;
- the use of statistical parsers with additional handcrafted rules to overwrite the default behaviour where needed;
- the design of hybrid modular systems containing RB modules and CB modules; or
- the use of CB methods to infer rules that are then used by RB modules.

The hybridization discussed in this paper falls between the last two cases. We present three different approaches that may help to reduce both the time and knowledge needed for building a complete RBMT system while still maintaining most

of its advantages and adding interesting ones, such as allowing the co-existence of automatically inferred and handcrafted linguistic data within the same module. More precisely we focus on: (a) the unsupervised training of (statistical) part-of-speech (PoS) taggers by using monolingual target-language (TL) and source-language (SL) corpora; (b) the unsupervised training of models to tackle the problem of the translation of polysemous words without using parallel corpora; and (c) the automatic inference of structural transfer rules by using a small parallel corpus.

Although all methods being presented in this paper may prove useful in many RBMT systems we focus on the development of tools to be used within the open-source MT engine Apertium. [1] Apertium (see section 2) is an open-source shallow-transfer MT engine initially intended for related languages, but currently being extended to deal with less related-language pairs. Moreover, all the approaches introduced in this paper have been, or will shortly be, released under an open-source license. This benefits not only the research community and people interested in building MT systems, but also *minorized* language communities. How the availability of open-source tools for MT benefits the whole community, and more concretely how this may help to the *de-minorization* of minority languages has been discussed by Forcada (2006).

Apertium extended with CB methods is by no means the only RB–CB hybrid MT system. For example, a different hybrid approach is followed by the METIS-II MT system. The METIS-II system (Dirix et al., 2005) is an EBMT system (as already mentioned, a particular case of the CBMT approach) which avoids the usual need for a bilingual parallel corpus. Instead of using a parallel corpus to extract bilingual information, METIS-II uses a bilingual dictionary (similar to that in Apertium) and a monolingual corpus in the TL. Besides that, it requires morphological analyzers and part-of-speech (PoS) taggers for both the SL and the TL, and a shallow parser for the SL (these tools are not part of the METIS-II system; a set of already existing tools can be fitted into the whole system if their input-output formats are made uniform). METIS-II can also query translation memories, if available, to improve the quality of the translations.

[1] `http://www.apertium.org`

METIS-II is a hybrid MT system since shallow parsing is performed by means of a RB method. The main difference between METIS-II and Apertium is in the transfer module: whereas the former uses the chunks (which may be considered as translation units) to query a TL database (built from the monolingual TL corpus, and subsequently enlarged with post-edited translations) for the translation, the latter performs additional operations (reorderings, agreement, etc.) in order to translate every detected chunk. Another difference is the way in which both systems use TL monolingual corpora; while METIS-II uses them during translation, Apertium uses them only to train SL modules not to produce translations.

The rest of the paper is organized as follows: Section 2 overviews the open-source shallow-transfer MT engine Apertium. In section 3 we explain how to use two monolingual corpora, one for the SL and another for the TL, to train certain modules of the Apertium MT system. Section 4 explains how to use a small parallel corpus to extract transfer rules to be used by the Apertium MT engine. Finally in section 5 the presented approaches are discussed.

2 Overview of Apertium

Apertium (Armentano-Oller et al., 2006; Corbí-Bellot et al., 2005)[2] is an open-source shallow-transfer MT engine initially intended for related-language pairs. This MT engine follows a shallow transfer approach and consists of the following pipelined modules:

- A *de-formatter* which separates the text to be translated from the format information (RTF and HTML tags, whitespace, etc.). Format information is encapsulated so that the rest of the modules treat it as blanks between words.

- A *morphological analyzer* which tokenizes the text in surface forms and delivers, for each surface form, one or more *lexical forms* consisting of *lemma*, *lexical category* and morphological inflection information.

- A *part-of-speech (PoS) tagger* which chooses, using a first-order hidden Markov model (Cutting et al., 1992) (HMM), one

[2] The MT engine, documentation, and linguistic data for up to 5 different language pairs (as of December 1, 2006) can be downloaded from `http://apertium.sf.net`.

of the lexical forms corresponding to an ambiguous surface form.

- A *lexical transfer* module which reads each SL lexical form and delivers the corresponding TL lexical form by looking it up in a bilingual dictionary.
- A *structural transfer* module (parallel to the lexical transfer) which uses a finite-state chunker to detect patterns of lexical forms which need to be processed for word reorderings, agreement, etc., and then performs these operations.
- A *morphological generator* which delivers a TL surface form for each TL lexical form, by suitably inflecting it.
- A *post-generator* which performs orthographic operations such as contractions (e.g. Spanish *del*=*de*+*el*) and apostrophations (e.g. Catalan *l'institut*=*el*+*institut*).
- A *re-formatter* which restores the format information encapsulated by the de-formatter into the translated text.

2.1 Linguistic data

The Apertium MT engine is completely independent from the linguistic data used while translating between a concrete pair of languages.

Linguistic data is coded using XML-based formats;[3] this allows for interoperability, and for easy data transformation and maintenance. In particular, files coding linguistic data can be automatically generated by third-party tools; for instance, in section 4 a method to automatically generate the set of rules used by the structural transfer module is introduced.

Apertium provides compilers to convert the linguistic data into the corresponding efficient form used by each module. Two main compilers are used: one for the four lexical processing modules (morphological analyzer, lexical transfer, morphological generator, and post-generator) and another one for the structural transfer. The first one generates finite-state letter transducers (Garrido-Alenda et al., 2002) which efficiently code the lexical data; the last one uses finite-state machines to speed up pattern matching. The use of such efficient compiled data formats renders the engine able to translate tens of thousands of words per second in a current desktop computer.

2.2 Enhancement in progress

Apertium's development team has recently been funded to enhance the system architecture in order to translate between less related language pairs like Catalan–English.

This enhancement implies, among other things, tackling the problem of *lexical selection*, that is, the problem of choosing the correct translation for those words that, according to the bilingual dictionary, can be translated in more than one way.[4] It must be noticed that this problem also arises in the case of related-language pairs; however, in that case the problem can be mostly addressed by introducing multi-word expressions in the dictionaries. Nevertheless, from the point of view of translation quality, the lexical selection problem is more important when the translation involves non-related languages.

The module that will perform the lexical selection (whose training is discussed in section 3.2) will be placed after the PoS tagger, just before the structural transfer module; as a consequence of that it will only use information from the SL to perform the lexical selection while translating.

As already commented, Apertium needs to be provided with linguistic data to carry out the translation. Some of this linguistic information can be unsupervisedly inferred from monolingual or parallel corpora. In the following section the problem of unsupervisedly training the HMM-based PoS tagger and the lexical selector modules is addressed. In both cases only monolingual corpora are used. Then, in section 4 parallel corpora are used to obtain the set of transfer rules to be used by the structural transfer module.

3 Exploiting target-language monolingual corpora to train source-language models

Usually, when a statistical model is trained for a specific language only corpora in this language are used. But when the model to be trained will be

[3] The XML formats (`http://www.w3.org/XML/`) for each type of linguistic data are defined through conveniently-designed XML document-type definitions (DTDs) which may be found inside the `apertium` package.

[4] We prefer the term *lexical selection* instead of *word-sense disambiguation* (WSD) because while WSD deals with *senses*, lexical selection deals with translations; therefore, a word with more than one sense but only one translation is not problematic in a MT context.

used within a MT system, the use of information coming from the other end of the MT system, that is, from the TL, may prove useful for the translation task. Note that the use of TL information does not discriminate among SL analyses leading to the same translation.

In this section we discuss two training methods along the lines sketched above: the training of a SL HMM-based PoS tagger using information not only from the SL, but also from the TL; and the training of lexical selection models using information from both SL and TL. The former has proved to be a successful approach (Sánchez-Martínez et al., 2004a; Sánchez-Martínez et al., 2004b; Sánchez-Martínez et al., 2006); the latter is still an ongoing work.

3.1 Target-language-driven part-of-speech tagger training

This section overviews the TL-driven training method that can be used to unsupervisedly train the HMM-based PoS taggers used within the Apertium MT engine. For a deeper description we refer to papers by Sánchez-Martínez et al. (2004a; 2004b).

Typically, the training of HMM-based PoS taggers is done using the *maximum-likelihood estimate* (MLE) method (Gale and Church, 1990) when tagged corpora[5] are available (supervised method), or using the Baum-Welch algorithm (Baum, 1972; Cutting et al., 1992) with untagged corpora[6] (unsupervised method). However, if the PoS tagger is to be embedded as a module of a MT system, as is the case, HMM training can be done in an unsupervised manner by using some modules of the MT system and information from both SL and TL.

The main idea behind the use of TL information is that the correct disambiguation (tag assignment) of a given SL segment will produce a more likely TL translation than any (or most) of the remaining wrong disambiguations. In order to apply this method these steps are followed:

- first the SL text is split into adequate segments (so that they are small and independently translated by the rest of the MT engine); then,
- the set of all possible disambiguations for each text segment are generated and translated into the TL;
- a statistical TL model is used to compute the likelihood of the translation of each disambiguation; and,
- these likelihoods are used to adjust the parameters of the SL HMM: the higher the likelihood, the higher the probability of the original SL tag sequence in the HMM being trained.

As expected, the number of possible disambiguations of a text segment grows exponentially with its length, the translation task being the most time-consuming one. This problem has been successfully addressed (Sánchez-Martínez et al., 2006) by using a very simple pruning method that avoids performing more than 80% of the translations without loss in tagging accuracy.

The way this training method works can be illustrated with the following example, in which the pruning technique is not used for simplicity. Consider the following segment in English, $s =$*"He books the room"*, and that an indirect MT system translating between English and Spanish is available. The first step is to use a morphological analyzer to obtain the set of all possible part-of-speech tags for each word. Suppose that the morphological analysis of the previous segment according to the lexicon is: *He* (pronoun), *books* (verb or noun), *the* (article), and *room* (verb or noun). As there are two ambiguous words (*books* and *room*) we have, for the given segment, four disambiguation *paths* or PoS combinations, that is to say:

- $\boldsymbol{g_1} =$ (pronoun, verb, article, noun),
- $\boldsymbol{g_2} =$ (pronoun, verb, article, verb),
- $\boldsymbol{g_3} =$ (pronoun, noun, article, noun), and
- $\boldsymbol{g_4} =$ (pronoun, noun, article, verb).

Let τ be the function representing the translation task. The next step is to translate the SL segment into the TL according to each disambiguation path $\boldsymbol{g_i}$:

- $\tau(\boldsymbol{g_1}, s) =$ "*Él reserva la habitación*",
- $\tau(\boldsymbol{g_2}, s) =$"*Él reserva la aloja*",
- $\tau(\boldsymbol{g_3}, s) =$"*Él libros la habitación*", and

[5] In a *tagged corpus* each occurrence of each word (ambiguous or not) has been assigned the correct PoS tag.

[6] In an *untagged corpus* all words are assigned (using, for instance, a morphological analyzer) the set of all possible PoS tags independently of context without choosing one of them.

- $\tau(\boldsymbol{g_4}, s)$ =“*Él libros la aloja*”.

It is expected that a Spanish language model will assign a higher likelihood to translation $\tau(\boldsymbol{g_1}, s)$ than to the other ones, which make little sense in Spanish. So the tag sequence $\boldsymbol{g_1}$ will have a higher probability than the other ones.

To estimate the HMM parameters, the calculated probabilities are used as if fractional counts were available to a supervised training method based on the MLE method in conjunction with a smoothing technique.

The method described in this section can be downloaded from the Apertium project web page,[7] and may simplify the initial building of Apertium-based MT systems for new language pairs, yielding better tagging results than the Baum-Welch algorithm (Sánchez-Martínez et al., 2004b). Our latest experiments on the TL-driven training method, when pruning unlikely disambiguations for each text segment (Sánchez-Martínez et al., 2006), give tagging error rates around 25%, while the Baum-Welch (unsupervised) algorithm provides error rates around 31% and supervised methods (using hand-tagged corpora) provide error rates around 11%. These error rates were calculated (for a Spanish PoS tagger) over ambiguous words only, not over all words. When using the TL-driven training method the TL was Catalan.

Finally, it must be noticed that the HMM-based PoS tagger needs to be provided with a file defining how to group the fine tags delivered by the morphological analyzer, which consist of lexical category and inflection information (such as verb, present, 3rd person, plural) into coarser tags. Sánchez-Martínez et al. (2005) propose a method to automatically obtain the set of coarse tags to be used by the PoS tagger. That method is based on a bottom-up agglomerative clustering algorithm performed over the states of a HMM previously trained following the TL-driven training method explained in this section. By grouping fine-grained tags into coarse ones the HMM complexity is reduced and more accurate PoS taggers are obtained.

[7] `http://apertium.sourceforge.net`. The method is implemented inside package `apertium-tagger-training-tools` which is licensed under the GNU GPL license.

3.2 Target-language driven lexical selector trainer

In natural language processing, word-sense disambiguation (WSD) is a commonly studied problem which consists of selecting a *sense* (or interpretation) from a set of possible senses for a given word in a particular context. As already commented in section 2, for MT purposes the problem is relevant only in the case of words having more than one possible translation into TL (Hutchins and Somers, 1992); because of this, we refer to that problem as *lexical selection*.

As the lexical selection problem is a translation problem one can naturally expect TL information to be valuable in solving that problem in an unsupervised manner. Consider, for example, the word *gato* in the Spanish sentence “El gato y un perro se están peleando”.[8] The noun *gato* may be translated into English as *jack* or *cat* depending on whether it refers to the tool or to the feline mammal, respectively. We can use a TL corpus (English in this case) to collect statistics of co-occurrences of lemmas in English; the resulting model will give higher scores to *cat* if the latter appears in the surrounding context of *dog* (the translation of *perro*) more frequently than *jack*.

The aim of the Apertium module that will address the lexical selection problem is to use only information from the SL when performing the lexical selection in order to translate a text, but information collected from both monolingual TL and SL corpora when training the module.

A number of methods to tackle the more general problem of WSD have been proposed in the literature (Stevenson and Wilks, 2003). Most of them are based on the concept of *bag of words*, where an algorithm uses a set of relevant surrounding words (usually lemmas) to assign the correct sense to a particular lemma. Moreover, most of the proposed methods are application-independent and rely on monolingual contexts. However, in a recent work, Specia and Nunes (2006) focus on the lexical selection problem and explore the use of the translation context as a knowledge source. Nevertheless, they focus on the translation of a few highly ambiguous English verbs into Portuguese; the information provided by the TL is only used to reorder the set of disambiguation rules previously produced by a machine learning approach on SL-

[8] Translated into English as “The cat and a dog are fighting”.

related knowledge sources (Specia, 2006).

In the case of developing a lexical selector for Apertium an important requirement should be met: it should be fast enough to avoid introducing a significant delay in overall translation speed.

Disambiguation method

With the aim of performing lexical selection we consider the input text (after PoS disambiguation) as a sequence of lemmas.

Let T be a function returning for a given SL lemma a set of *"translation sense" marks* in the SL. For example, for the Spanish lemma *gato*, and considering Spanish-to-English translation, that function would return a set with two translation senses, $T(\text{gato}) = \{\text{gato}^{\text{cat}}, \text{gato}^{\text{jack}}\}$, meaning that the Spanish lemma *gato* can be translated into English in two different ways. The lexical selection task consists of selecting one translation sense for each lemma according to the SL context.

In order to choose the right translation sense for a given SL word with lemma s_i and translation senses $T(s_i) = \{s_i^1, s_i^2\}$, a sliding context window of relevant[9] lemmas surrounding s_i in the input SL text can be used, $C(s_i) = (\ldots, s_{i-2}, s_{i-1}, s_i, s_{i+1}, s_{i+2}, \ldots)$.

Let $S(s_i^t, s_j)$ be a function that for a given translation sense $s_i^t \in T(s_i)$, and a given SL lemma s_j returns a score n representing how often the SL lemma s_j co-appears with the SL lemma s_i in such a context in which the correct translation sense for s_i should be s_i^t. For example, $S(\text{gato}^{\text{cat}}, \text{perro}) = 300$ while $S(\text{gato}^{\text{cat}}, \text{rueda}) = 23$, meaning that, as expected, the translation sense gato^{cat} is more likely to appear with SL lemma *perro* (translated into English as *dog*) that with SL lemma *rueda* (translated into English as *wheel*). How these scores are learned from corpora is explained below.

The ambiguity of SL lemma s_i is solved by selecting the optimal translation sense s_i^* by means of the following equation:

$$s_i^* = \arg \max_{s_i^t \in T(s_i)} \sum_{\forall s_j \in C(s_i): s_i \neq s_j} S(s_i^t, s_j). \quad (1)$$

The following example illustrates the disambiguation procedure. Consider the input Spanish sentence "El gato está en un árbol mientras el perro ladra" translated into English as "The cat is in a tree while the dog barks", and suppose that *gato* is the only word that can be translated into English in more than one way, being $T(\text{gato}) = \{\text{gato}^{\text{cat}}, \text{gato}^{\text{jack}}\}$. To solve the ambiguity all relevant lemmas in the surrounding context, $C(\text{gato}) = \{\text{arbol}, \text{perro}, \text{ladrar}\}$, are evaluated through the S function together with each possible translation sense for *gato*:

$$\begin{aligned} S(\text{gato}^{\text{cat}}, \text{arbol}) &= 100, \\ S(\text{gato}^{\text{cat}}, \text{perro}) &= 300, \\ S(\text{gato}^{\text{cat}}, \text{ladrar}) &= 3, \\ S(\text{gato}^{\text{jack}}, \text{arbol}) &= 12, \\ S(\text{gato}^{\text{jack}}, \text{perro}) &= 23, \text{ and} \\ S(\text{gato}^{\text{jack}}, \text{ladrar}) &= 0; \end{aligned}$$

after adding up all the scores for each translation sense, gato^{cat} (403) is chosen as the correct one, and $\text{gato}^{\text{jack}}$ (35) is discarded.

Training method

The training phase consists of building a co-occurrence model of SL lemmas (with the corresponding scores) for each translation sense that will be managed by the MT system. To this end two monolingual corpora (one for the SL and another for the TL) and a bilingual dictionary (most likely the bilingual dictionary already used by the MT system) are used. For the training purpose the corpora are considered, after the PoS disambiguation, as a sequences of lemmas.

Before training SL co-occurrence models, TL co-occurrence models must be built. Each SL translation sense is translated into TL and a co-occurrence model for it is built using a TL monolingual corpus. The way in which these TL co-occurrence models are built is straightforward: the TL is processed using a sliding context window of relevant lemmas, and counts of co-appearing lemmas are collected.

The SL co-occurrences models are built in an analogous way. A SL corpus is processed using a sliding context window of relevant words. For each SL lemma with more than one translation sense, the surrounding SL lemmas are translated by looking them up in a bilingual dictionary; then, the scores provided by the TL co-occurrence model for the translation of each translation sense and translated context are transferred to the SL co-occurrence model.

After this training process, SL co-occurrence models are obtained in an unsupervised way and

[9] A list of stopwords is used to discard words that may not help in the lexical selection task, usually words appearing very often, and in multiple contexts, in SL corpora.

can be used to perform lexical selection as explained above.

4 Use of a small parallel corpus to extract shallow-transfer rules

As has been mentioned in section 2.1, the Apertium MT engine is designed in such a way that algorithms are decoupled from the linguistic data. As the linguistic data are coded using well-defined XML formats, such files can be automatically generated by third-party tools. In this section we overview a possible method that can be used to infer the transfer rules used by the Apertium MT system.

This approach uses a comparatively small parallel corpus to automatically extract shallow-transfer MT rules. The amount of parallel corpora may be considered small compared to the huge amount of parallel corpora (tens of millions of running words) needed to train state-of-the-art statistical machine translation (SMT) systems. This method has been presented elsewhere (Sánchez-Martínez and Ney, 2006); we overview here the approach and outline ongoing research work on this topic.

The method used to obtain a set of transfer rules to deal with the grammatical and lexical divergences between SL and TL is based on the *alignment templates* (AT) (Och and Ney, 2004) approach already used in SMT. An AT represents a generalization performed over aligned phrase[10] pairs using word classes.

The ATs are learned in a three-stage procedure: first, word alignments are computed, then aligned phrase pairs are extracted; and finally, a generalization over the extracted aligned phrase pairs is performed using word classes instead of the words themselves. The use of word classes allows for generalization, to model word reordering, preposition changes and other divergences between SL and TL.

The transformations to apply are mainly based on the PoS of SL and TL words; to adapt the ATs to a shallow-transfer MT system the following linguistic information needs to be provided:

- The set of *closed lexical categories* in both source and target languages. Closed lexical categories are those categories that cannot easily grow by adding new words to the dictionaries: articles, auxiliary verbs, pronouns, etc. Hereafter, we will refer as *closed-class words* to those words whose PoS is in the set of closed lexical categories; analogously, we will refer as *open-class words* to those words whose PoS does not belong to the set of closed lexical categories.

- The set of *dominant categories* in the target language. A dominant category is a lexical category which usually propagates its inflection information (such as gender or number) to neighboring (modifying) lexical categories in the TL. Usually the only dominant category is the noun, which propagates its gender and number to articles and adjectives.

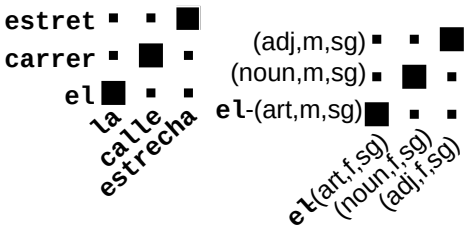

Figure 1: Example of an alignment that can be found in a Spanish–Catalan parallel corpus (on the left) and the alignment template (AT) extracted from it (on the right).

Extraction of alignment templates

To extract the ATs, the PoS tag (including all the inflection information such as gender, number or verb tense) is used to assign a word class to each open-class word. For closed-class words, the lemma is also used to define the word class, therefore each closed-class word is in its own single class. For example, the English nouns *book* and *house* would be in the same word class, but the prepositions *to* and *for* would be in different classes even if they have the same PoS. In this way the method is allowed to learn transformations such as preposition changes or auxiliary verb usage in the TL.

The extraction of an AT is illustrated in figure 1, which shows (on the left) an aligned phrase pair extracted from a Spanish–Catalan parallel corpus, and (on the right) the AT extracted from it. To extract the AT the PoS is used as word class; note, however, that in the case of the article *el* the lemma has also been used to define the word class because it is a closed-class word. The AT shown in figure 1 generalizes the rule to apply when translat-

[10] In this paper with "phrase" we mean any sequence of consecutive words, not necessarily whole constituents or syntactic units.

ing Spanish into Catalan (and vice versa) in order to propagate the gender from the noun (a dominant category) to the article and the adjective.

Application of alignment templates

An AT generalizes a transformation to be applied over a SL phrase while translating. To apply an AT two conditions must hold:

- The SL phrase to which the AT will be applied must match exactly the SL side of the AT; after calculating the word class for each word being translated, all word classes must be the same and in the same order;
- TL inflection information provided by the bilingual dictionary for the dominant words being translated must be preserved in the TL side of the AT.

If the two conditions are met, the AT can be said to be applicable. Notice that for a given SL phrase more than one AT could be applicable, in that case the AT finally chosen for application is the one covering the longest sequence of SL words, and in case of equal number of covered words, the most frequent one.

The application of an AT is done by translating each open-class word by looking it up in a bilingual dictionary, and replacing the morphological information provided by the bilingual dictionary by the morphological information provided by the TL part of the AT. The alignment information is used to put each word in their correct place in the TL. Moreover, closed-class words are not translated using the bilingual dictionary, but instead they are taken from the TL part of the AT.

In the work reported by Sánchez-Martínez and Ney (2006) this approach has been tested using an existing shallow-transfer MT system for the Spanish–Catalan language pair,[11] and using a parallel corpus with around 300 000 running words for AT extraction.[12] The translation performance has been compared to that of word-for-word translation (when no structural transformations are applied) and that of handcrafted rules application using the same MT engine. In both translation directions there has been a significant improvement in the translation quality as compared to word-for-word translation. Furthermore, the translation quality is very close to that achieved when using handcrafted transfer rules. If the best translation quality that can be achieved is assumed to be that of handcrafted rules, the relative improvement, compared to that of word-for-word translation, is about 70% for the Spanish→Catalan translation, and around 60% for the Catalan→Spanish translation.

As further research on this topic, we are working on the use of a bilingual dictionary during the AT extraction procedure. The use of a bilingual dictionary will allow to extract more accurate ATs and to avoid the need of being provided with the set of dominant categories. The result of all this work will be released under an open-source license by the end of 2006.

5 Discussion

In this paper we have presented three corpus-based (CB) approaches capable of inferring linguistic data to be used by the open-source shallow-transfer MT engine Apertium.

On the one hand, the first two CB approaches make use of monolingual corpora in the TL and some parts of the MT itself to train models to be used on the SL. It must be stressed that after training, both models (PoS tagger and lexical selector) only use information from the SL when performing their respective tasks as a part of the whole translation procedure.

On the other hand, the third approach discussed in this paper uses a small amount of parallel corpora to automatically infer structural transfer rules. By the end of 2006, a tool able to produce the corresponding XML file coding the inferred transfer rules will be released for Apertium. Notice that the final generated rules will be as human-readable as handcrafted rules; therefore, human beings will be able to correct them where necessary or to introduce new ones. From our point of view this is a great advantage over other CB approaches to MT, such as SMT. Our approach allows for coexistence of handcrafted and automatically generated rules.

Finally, it must be pointed out that all tools described in this paper have been, or will shortly be, released under an open-source license. Open access to them will benefit, on the one hand, the research community and, on the other hand, peo-

[11] The MT system used (Canals-Marote et al., 2001) follows the same translation procedure that Apertium follows (see section 2).

[12] To compute word alignments a larger corpus was used. However, recent experiments (unpublished) show that using an small corpus to compute word alignments give comparable results.

ple interested in building Apertium-based MT systems.

Acknowledgements

Work funded by the Spanish Ministry of Science and Technology through project TIC2003-08681-C02-01 and by the Spanish Ministry of Education and Science and the European Social Fund through grant BES-2004-4711. The development of the Apertium MT engine was initially funded by the Spanish Ministry of Industry, Tourism and Commerce through grants FIT-340101-2004-3 and FIT-340001-2005-2. The enhancement of Apertium is being funded by the Generalitat de Catalunya.

References

C. Armentano-Oller, R. C. Carrasco, A. M. Corbí-Bellot, M. L. Forcada, M. Ginestí-Rosell, S. Ortiz-Rojas, J. A. Pérez-Ortiz, G. Ramírez-Sánchez, F. Sánchez-Martínez, and M. A. Scalco. 2006. Open-source Portuguese-Spanish machine translation. In *Computational Processing of the Portuguese Language, Proceedings of the 7th International Workshop on Computational Processing of Written and Spoken Portuguese, PROPOR 2006*, volume 3960 of *Lecture Notes in Computer Science*, pages 50–59. Springer-Verlag. (`http://www.dlsi.ua.es/~japerez/pub/pdf/propor2006.pdf`).

Doug Arnold, 2003. *Computers and Translation: A translator's guide*, chapter Why translation is difficult for computers, pages 119–142. Benjamins Translation Library. Edited by H. Somers.

L. E. Baum. 1972. An inequality and associated maximization technique in statistical estimation of probabilistic functions of a Markov process. *Inequalities*, 3:1–8.

R. Canals-Marote, A. Esteve-Guillen, A. Garrido-Alenda, M. Guardiola-Savall, A. Iturraspe-Bellver, S. Montserrat-Buendia, S. Ortiz-Rojas, H. Pastor-Pina, P. M. Perez-Antón, and M. L. Forcada. 2001. The Spanish-Catalan machine translation system interNOSTRUM. In *Proceedings of MT Summit VIII: Machine Translation in the Information Age*, pages 73–76. (`http://www.dlsi.ua.es/~mlf/docum/canals01p.pdf`).

A. M. Corbí-Bellot, M. L. Forcada, S. Ortiz-Rojas, J. A. Pérez-Ortiz, G. Ramírez-Sánchez, F. Sánchez-Martínez, I. Alegria, A. Mayor, and K. Sarasola. 2005. An open-source shallow-transfer machine translation engine for the Romance languages of Spain. In *Proceedings of the 10th European Association for Machine Translation Conference*, pages 79–86, Budapest, Hungary. (`http://www.dlsi.ua.es/~mlf/docum/corbibellot05p.pdf`).

D. Cutting, J. Kupiec, J. Pedersen, and P. Sibun. 1992. A practical part-of-speech tagger. In *Third Conference on Applied Natural Language Processing. Association for Computational Linguistics. Proceedings of the Conference.*, pages 133–140, Trento, Italy.

P. Dirix, I. Schuurman, and V. Vandeghinste. 2005. Metis II: Example-based machine translation using monolingual corpora - system description. In *Proceedings of the Example-Based Machine Translation Workshop held in conjunction with the 10th Machine Translation Summit*, pages 43–50.

M. L. Forcada. 2006. Open-source machine translation: an opportunity for minor languages. In *Proceedings of Strategies for developing machine translation for minority languages (5th SALTMIL workshop on Minority Languages)*. (`http://www.dlsi.ua.es/~mlf/docum/forcada06p2.pdf`).

William A. Gale and Kenneth W. Church. 1990. Poor estimates of context are worse than none. In *Proceedings of a workshop on Speech and natural language*, pages 283–287. Morgan Kaufmann Publishers Inc.

A. Garrido-Alenda, M. L. Forcada, and R. C. Carrasco. 2002. Incremental construction and maintenance of morphological analysers based on augmented letter transducers. In *Proceedings of TMI 2002 (Theoretical and Methodological Issues in Machine Translation)*, pages 53–62.

W.J. Hutchins and H.L. Somers. 1992. *An Introduction to Machine Translation*. Academic Press, London, United Kingdom.

F. J. Och and H. Ney. 2004. The alignment template approach to statistical machine translation. *Computational Linguistics*, 30(4):417–449.

F. Sánchez-Martínez and H. Ney. 2006. Using alignment templates to infer shallow-transfer machine translation rules. In *Advances in Natural Language Processing, Proceedings of 5th International Conference on Natural Language Processing FinTAL*, volume 4139 of *Lecture Notes in Computer Science*, pages 756–767. Springer-Verlag. (`http://www.dlsi.ua.es/~fsanchez/pub/pdf/sanchez06a.pdf`).

F. Sánchez-Martínez, J. A. Pérez-Ortiz, and M. L. Forcada. 2004a. Cooperative unsupervised training of the part-of-speech taggers in a bidirectional machine translation system. In *Proceedings of TMI, The Tenth Conference on Theoretical and Methodological Issues in Machine Translation*, pages 135–144, October. (`http://www.dlsi.ua.es/~fsanchez/pub/pdf/sanchez04b.pdf`).

F. Sánchez-Martínez, J. A. Pérez-Ortiz, and M. L. Forcada. 2004b. Exploring the use of target-language information to train the part-of-speech tagger of machine translation systems. In *Advances in Natural Language Processing, Proceedings of 4th International Conference EsTAL*, volume 3230 of *Lecture Notes in Computer Science*, pages 137–148. Springer-Verlag. (`http://www.dlsi.ua.es/~fsanchez/pub/pdf/sanchez04a.pdf`).

F. Sánchez-Martínez, J. A. Pérez-Ortiz, and M. L. Forcada. 2005. Target-language-driven agglomerative part-of-speech tag clustering for machine translation. In *Proceedings of the International Conference RANLP - 2005 (Recent Advances in Natural Language Processing)*, pages 471–477. (`http://www.dlsi.ua.es/~fsanchez/pub/pdf/sanchez05.pdf`).

F. Sánchez-Martínez, J. A. Pérez-Ortiz, and M. L. Forcada. 2006. Speeding up target-language driven part-of-speech tagger training for machine translation. In *Advances in Artificial Intelligence, Proceedings of the 5th Mexican International Conference on Artificial Intelligence*, volume 4293 of *Lecture Notes in Computer Science*, pages 844–854. Springer-Verlag. (`http://www.dlsi.ua.es/~fsanchez/pub/pdf/sanchez06b.pdf`).

L. Specia and M. G. Volpe Nunes. 2006. Exploiting the translation context for multilingual WSD. In Petr Sojka, Ivan Kopecek, and Karel Pala, editors, *Text, Speech and Dialogue, Processing of 9th International Conference, TSD 2006*, volume 4188 of *Lecture Notes in Computer Science*, pages 269–276. Springer-Verlag.

L. Specia. 2006. A hybrid relational approach for WSD - first results. In *Proceedings of COLING/ACL 06 Student Research Workshop*, pages 55–60.

M. Stevenson and Y. Wilks, 2003. *Oxford Handbook of Computational Linguistics*, chapter Word Sense Disambiguation, pages 249–265. Oxford University Press. Edited by R. Mitkov.