Automatic induction of shallow-transfer rules for open-source machine translation

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

Abstract

This paper focuses on the inference of structural transfer rules for shallow-transfer machine translation (MT). Transfer rules are generated from alignment templates, like those used in statistical MT, that have been extracted from parallel corpora and extended with a set of restrictions that control their application. The experiments conducted using the open-source MT platform Apertium show that there is a clear improvement in translation quality as compared to word-for-word translation (when no transfer rules are used), and that the resulting translation quality is very close to the one obtained using hand-coded transfer rules. The method we present is entirely unsupervised and benefits from information in the rest of modules of the MT system in which the inferred rules are applied.

1 Introduction

The increasing availability of machine-readable parallel corpora has given rise to the development of corpus-based machine translation (MT) approaches such as statistical MT (SMT) or example-based MT (EBMT). However, corpus-based approaches usually require a very large parallel corpus (with tens of millions of words) that is not always available. On the other hand, rule-based MT (RBMT) attains high performance but at the expense of the large effort needed to build the necessary linguistic resources (Arnold, 2003) such as structural transfer rules.

In this paper we focus on the automatic inference of structural transfer rules from parallel corpora, which are small compared to the size of corpora commonly used to build SMT or (some) EBMT systems. The approach we present is tested on the shallow transfer MT platform Apertium for which structural transfer rules are generated.

Overview. In rule-based MT, transfer rules are needed to perform syntactic and lexical changes. The approach we present in this paper to infer shallow-transfer MT rules is based on the alignment templates approach (Och and Ney, 2004) already used in SMT (see section 3). An alignment template (AT) can be defined as a generalization performed over aligned phrase[1] pairs (or *translation units*) by using word classes.

The method we present is entirely unsupervised and needs, in addition to the linguistic data used by the MT system in which the inferred rules are used, only a (comparatively) small parallel corpus and a file defining a reduced set of lexical categories usually involved in lexical changes.

Sanchez-Martınez and Ney (2006) use ATs to infer shallow-transfer rules to be used in

[1]For the purpose of this paper, with *phrase* we mean any sequence of consecutive words, not necessarily whole syntactic constituents.

MT. The work reported in this paper can be seen as a reformulation and improvement of that work. Sanchez-Martınez and Ney (2006) use ad-hoc linguistic information, in addition to that already present in the rest of modules of the MT system, to define the priorities used to establish agreement restrictions. This additional linguistic information is not necessary here, as restrictions may be easily derived from the bilingual dictionary using a general approach.

Transfer rules are generated for use with the open-source shallow-transfer MT platform Apertium; however, the approach we present is suitable for any other shallow-transfer-based MT system. The generated transfer rules (see section 2.1) are coded in a well-defined XML format, and can be edited by human experts or even co-exist with handcrafted ones.

The method we present[2] has been tested with an Apertium-based MT system for the Spanish–Catalan language pair; the experimental results show that the use of AT-based shallow-transfer rules drastically improves the translation quality as compared to word-for-word translation, i.e. when no transfer rules are used, and is comparable to the quality achieved when using handcrafted rules.

Background. There have been other attempts to learn automatically or semi-automatically the structural transformations needed to produce correct translations into the target language (TL). Those approaches can be classified according to the translation framework to which the learned rules are applied. On the one hand, some approaches learn transfer rules to be used in rule-based MT (Probst et al., 2002; Lavie et al., 2004). Probst et al. (2002) and Lavie et al. (2004) infer transfer rules for MT involving "minor" languages (e.g. Quechua) with very limited resources. To this end, a small parallel corpus (of a few thousand sentences) is built with the help of a small set of bilingual speakers of the two languages. The parallel corpus is obtained by translating a controlled corpus from a "major" language (English or Spanish) to a "minor" language by means of an elicitation tool. This tool is also used to graphically annotate the word alignments between the two sentences. Finally, hierarchical syntactic rules, that can be seen as constituting a context-free transfer grammar, are inferred from the aligned parallel corpus.

On the other hand, in the EBMT framework, some researchers deal with the problem of inferring the kinds of translation rules called *translation templates* (Kaji et al., 1992; Brown, 1999; Cicekli and Guvenir, 2001). A translation template can be defined as a bilingual pair of sentences in which corresponding units (words or phrases) are coupled and replaced by variables. Liu and Zong (2004) provide an interesting review of the different research works dealing with translation templates. Brown (1999) uses a parallel corpus and some linguistic knowledge in the form of equivalence classes (both syntactic and semantic) to perform a generalization over the bilingual examples collected. The method works by replacing each word by its corresponding equivalence class and then using a set of grammar rules to replace patterns of words and tokens by more general tokens. Cicekli and Guvenir (2001) formulate the acquisition of translation templates as a machine learning problem, in which the translation templates are learned from the differences and similarities observed in a set of different translation examples, using no morphological information at all. Kaji et al. (1992) use a bilingual dictionary and a syntactic parser to determine the correspondences between translation units while learning the translation templates. In any case, the translation templates used in EBMT differ from the approach presented in this paper, firstly because our approach is largely based on part-of-speech and inflection information, and the inferred translation rules are flatter, less structured and non-hierarchical (because of this, they are suitable for shallow-transfer MT); and secondly, because the way in which the transformations to

[2]The method is implemented inside package `apertium-transfer-tools` and, released under the GNU GPL license, is freely available at `http://sf.net/projects/apertium`.

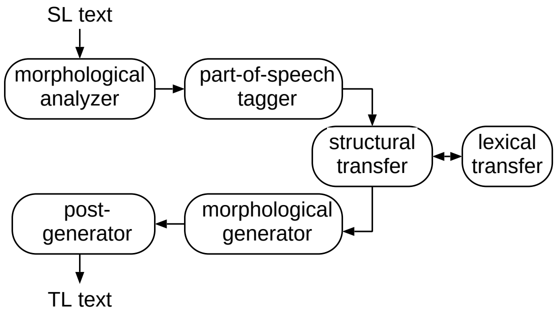


Figure 1: Main modules of the Apertium shallow-transfer MT platform (see section 2). The structural transfer module is the one that applies the inferred transfer rules.

apply are chosen (see section 5) differs from those used in the EBMT framework.

The rest of the paper is organized as follows: the next section overviews the open-source shallow-transfer MT platform Apertium used to test our approach; section 3 overviews the alignment templates (ATs) approach; section 4 explains how to extend the ATs in order to use them to generate (section 5) shallow-transfer rules to be used in MT. Section 6 describes the experiments conducted and the results achieved. Finally, in section 7 we draw some conclusions and outline future work.

2 Overview of Apertium

Apertium[3] (Armentano-Oller et al., 2006) is an open-source platform for developing MT systems, initially intended for related language pairs. The Apertium MT engine follows a shallow transfer approach and may be seen as an assembly line consisting of the following main modules (see figure 1):

> A *morphological analyzer* which tokenizes the source-language (SL) 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 tagger* which chooses, using a first-order hidden Markov model (HMM) (Cutting et al., 1992), one 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 shallow 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. Note that this is the module that applies the transfer rules generated by the method presented here.

> 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 *de+el=del*) and apostrophations (e.g. Catalan *el+institut=l'institut*).

Modules use text to communicate, which makes it easy to diagnose or modify the behavior of the system.

2.1 Linguistic data and compilers

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

Linguistic data is coded using XML-based formats;[4] 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, as is the case of the method we present.

Apertium provides compilers to convert the linguistic data into the corresponding efficient

[3]The MT platform, documentation, and linguistic data for different language pairs can be freely downloaded from `http://apertium.sf.net`.

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

(binary) form used by each module of the engine. 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; both generate finite-state processors which make Apertium capable of translating tens of thousands of words per second in a current desktop computer.

3 The alignment templates approach

Alignment templates (ATs) (Och and Ney, 2004), initially used in SMT, perform a generalization over bilingual phrase pairs using word classes instead of words. An AT $z = (S_m, T_n, A)$ consists of a sequence S_m of m SL word classes, a sequence T_n of n TL word classes, and a set of pairs $A = \{(i, j) : i \in [1, n] \wedge j \in [1, m]\}$ with the alignment information between TL and SL word classes.

Learning a set of ATs from a parallel corpus consists of:

1. the computation of the word alignments,
2. the extraction of bilingual phrase pairs, and
3. the substitution of each word by its corresponding word class.

Word alignments. A variety of methods, statistical (Och and Ney, 2003) or heuristic (Caseli et al., 2005), may be used to compute word alignments from a (sentence aligned) parallel corpus. For our experiments (section 6) we have used the open-source `GIZA++` toolkit[5] in the following way. First, standard `GIZA++` training runs to estimate translation models to translate from language L_1 to language L_2, and vice versa. Then, from the training corpus, Viterbi alignments[6] A_1 and A_2 are obtained (one for each translation

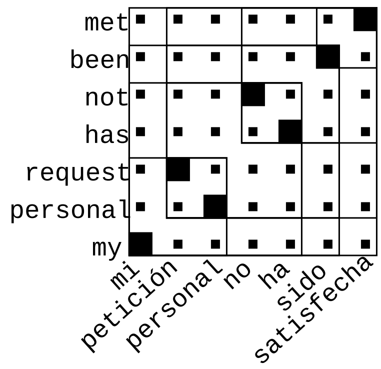


Figure 2: Example of bilingual phrases extracted (see section 3) for a given word-aligned Spanish–English sentence pair in which the alignment information is represented as a binary matrix. Each square corresponds to a bilingual phrase.

direction) and symmetrized via the following method (Och and Ney, 2003, p. 33):[7]

> first the intersection $A = A_1 \cap A_2$ of both alignments is computed, then
>
> the alignment A is iteratively extended with alignments $(i, j) \in A_1$ or $(i, j) \in A_2$ if neither SL word $w_{\mathrm{S}j}$ nor TL word $w_{\mathrm{T}i}$ has an alignment in A, or the following two conditions hold:
>
> 1. One of the following (neighboring) alignments $(i \quad 1, j)$, $(i+1, j)$, $(i, j \quad 1)$, $(i, j+1)$ is already in A.
> 2. The new alignment $A \cup \{(i, j)\}$ does not contain any alignment with both horizontal ($(i \quad 1, j)$, $(i+1, j)$) and vertical ($(i, j \quad 1)$, $(i, j+1)$) neighbors.

Bilingual phrase pairs. The extraction of bilingual phrases (Och et al., 1999) is performed by considering all possible pairs within a certain length and ensuring that (see figure 2):

1. all words are consecutive, and
2. words within the bilingual phrase are not aligned with words from outside.

[5] `http://www.fjoch.com/GIZA++.html`

[6] The Viterbi alignment between source and target sentences is defined as the alignment whose probability is maximal under the translation models previously estimated.

[7] For easier understanding, think about the alignment information as a binary matrix (see figure 2).

The set of bilingual phrases that are extracted from the word-aligned sentence pair $(w_{S1}, \ldots, w_{SJ}), (w_{T1}, \ldots, w_{TI})$ can be formally expressed as follows:

$$
\begin{aligned}
BP(w_{S1}^{J}, w_{T1}^{I}, A) = \{(w_{Sj}^{j+m}, w_{Ti}^{i+n}) : \\
\forall (i', j') \in A : j \leq j' \leq j + m \Leftrightarrow \\
i \leq i' \leq i + n\}.
\end{aligned}
$$

Generalization. The generalization is simply done by replacing each word by its corresponding word class. The use of word classes instead of the words themselves allows the description of word reorderings, preposition changes and other divergences between SL and TL.

4 Alignment templates for shallow-transfer machine translation

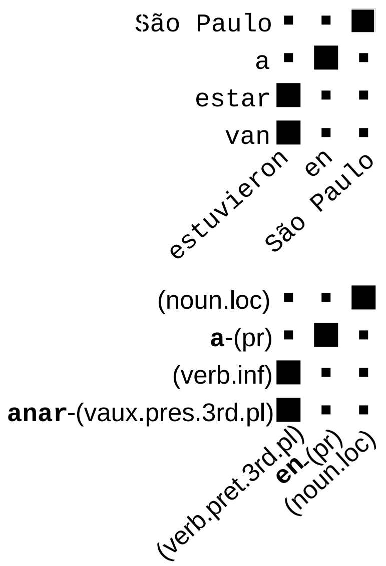
Shallow-transfer MT is an special case of the (indirect) rule-based transfer MT framework. Shallow transfer rules simply detect patterns of lexical forms and apply lexical and syntactic changes to them. Therefore, a simple intermediate representation (IR) consisting of lexical forms is used by the translation engine.

In order for the shallow-transfer MT system to benefit from the AT approach the parallel corpora must be in the same IR used by the translation engine. To that end, the morphological analyzers and part-of-speech taggers of the MT system in which the transfer rules will be applied are used to analyze the parallel corpus before computing the word alignments (see section 3).

4.1 Word-class definition

The transformations to apply are mainly based on the part-of-speech of SL and TL words; therefore, part-of-speech information (including all inflection information such as gender, number or verb tense) is used to define the word class each word belongs to.

Using part-of-speech information to define the set of word classes allows the method to learn syntactic rules such as reordering and agreement rules, and verb tense changes,



$$R = \{w_1 = \texttt{verb.*}, w_3 = \texttt{noun.*}\}$$

Figure 3: Example of Spanish–Catalan bilingual phrases (top), alignment template (bottom) obtained when each word is replaced by its corresponding word class, and TL restrictions (see section 4.2) for the Spanish-to-Catalan translation. Words in bold face correspond to lexicalized categories (see section 4.1). Word classes in the horizontal axis correspond to the SL (Spanish) and in the vertical axis to the TL (Catalan). Alignment information is represented as a binary matrix.

among others. However, in order to learn lexical changes, such as preposition changes or auxiliary verb usage, additional linguistic information, provided by an expert, is needed.

Lexicalized categories. A set of (lexicalized) categories usually involved in lexical changes such as prepositions and auxiliary verbs may be provided.[8] For those words whose part-of-speech is in that set of lexicalized categories (from now on, *lexicalized words*) the lemma is also used when defining the word class they belong to. In this way, lexicalized words are placed in single-word classes. For example, if prepositions are considered lexicalized categories, words *to* and *for* would be in different word classes, even if they have the same part-of-speech and inflection information, while words *book* and *house* would be in the same word class (noun, singular). Figure 3 shows an example of Spanish–

[8] Lexicalized categories are specified through a simple XML file.

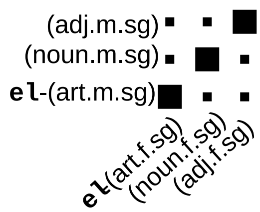
Catalan bilingual phrase and the generalization performed when each word is replaced by its corresponding word class; words in bold face correspond to lexicalized categories. The AT shown in figure 3 generalizes, on the one hand, the use of the auxiliary Catalan verb *anar* to express the past (preterite) tense and, on the other hand, the preposition change when it refers to a place name, such as the name of a city or a country.

4.2 Extending the definition of alignment template

In section 3 an alignment template (AT) was defined as a tuple $z = (S_m, T_n, A)$ in which only the alignment between SL and TL word classes was considered. Here we extend the definition of AT to $z = (S_m, T_n, A, R)$, where a set of restrictions, R, over the TL inflection information of non-lexicalized categories is added.

TL Restrictions. When translating (see next section), that is, when applying ATs, TL inflection information of non-lexicalized words is taken from the corresponding TL word class in the AT being applied, not from the bilingual dictionary; because of this, restrictions are needed in order to prevent an AT to be applied in certain conditions that would produce an incorrect translation. For example, an AT that changes the gender of a noun from masculine to feminine (or vice versa) would produce an incorrect TL word if such a change is not allowed for that noun. Restrictions refer to TL inflection information; therefore, they are obtained for a given translation direction and they change when translating the other way round.

TL restrictions are obtained from the bilingual dictionary. In Apertium bilingual dictionaries, changes in inflection information are explicitly coded. The following two examples show, on the one hand, a Spanish–Catalan bilingual entry and, on the other hand, the restriction over the TL inflection information for the Spanish-to-Catalan translation derived for that bilingual entry:[9]



$$R = \{w_2 = \texttt{noun.m.*}, w_3 = \texttt{adj.*}\}$$

Figure 4: Spanish–Catalan alignment template (AT) and TL restrictions over the inflection information for the Spanish-to-Catalan translation (see section 4.2).

Bilingual entry without any inflection information change

```
<e><p>
 <l>castigo<s n="noun"/></l>
 <r>càstig<s n="noun"/></r>
</p></e>
```

Restriction: w=`noun.*`

Bilingual entry in which the gender changes from feminine (Spanish) to masculine (Catalan)

```
<e><p>
 <l>calle<s n="noun"/>
         <s n="f"/></l>
 <r>carrer<s n="noun"/>
          <s n="m"/></r>
</p></e>
```

Restriction: w=`noun.m.*`

As can be seen, restrictions provide the part-of-speech and inflection information that the lexical form should have at translation time after looking it up in the bilingual dictionary; the star at the end of each restriction means that the rest of inflection information is not restricted. The second bilingual entry would be responsible of the restrictions attached to w_2 in the AT shown in figure 4. That AT generalizes the rule to apply in order to propagate the gender from the noun to the article and the adjective, and can only be applied if the noun (w_2) is masculine in the TL (see next section to know how ATs are applied).

[9]Lemmas between `<l>` and `</l>` XML tags correspond to Spanish words; analogously, lemmas between `<r>` and `</r>` tags correspond to Catalan words. Inflection information is coded through the `<s>` (*symbol*) XML tag, the first one being the part-of-speech.

SL and TL non-lexicalized words are not aligned.

5 Generation of Apertium structural transfer rules

This section describes the generation of Apertium structural transfer rules; note, however, that the generation of transfer rules for other shallow-transfer MT systems would also be feasible by following the approach presented here.

Apertium structural transfer uses finite-state pattern matching to detect, in the usual left-to-right, longest-match way, fixed-length patterns of lexical forms to process and performs the corresponding transformations. A (generic) shallow-transfer rule consists of a sequence of lexical forms to detect and the transformations to apply to them.

Filtering of the alignment templates. To decide which ATs to take into account for the generation of rules the method is provided with a frequency count threshold. ATs whose frequency count is below this threshold are discarded. In the experiments we have tested two different ways of interpreting the frequency count:

- to use directly the frequency count c, and
- to use a modified frequency count $c' = c(1+\log(l))$, where l stands for the length of the SL part of the AT.

The second approach aims at solving the problem caused by the fact that longer ATs have lower frequency counts but may be more accurate as they take more context into account.[10]

Moreover, ATs satisfying one of the following conditions are also discarded:

- the bilingual phrase the AT comes from cannot be reproduced by the MT system in which the transfer rules will be used. This happens when the translation equivalent (in the bilingual dictionary) differs from that in the bilingual phrase extracted from the corpus.

[10]A similar approach was used by Mikheev (1996) in his work on learning part-of-speech guessing rules to prioritize longer suffixes over shorter ones.

Rules generation. In our approach, a rule consists of a set U of ATs with the same sequence of SL word classes, but different sequences of TL word classes, different alignment information or different set of TL restrictions. Formally this may be expressed as follows:

$$U = \{(S_m, T_n, A, R) \in Z : S_m = S^U\},$$

where Z refers to the whole set of extracted ATs and S^U to the sequence of SL word classes all ATs $z \in U$ have in common.

For each set U an Apertium shallow-transfer rule matching the sequence of SL word classes S^U is generated; that rule consists of code applying (see below) always the most frequent AT $z = (S_m, T_n, A, R) \in U$ that satisfies the TL restrictions R. A "default" AT, which translates word for word, is always added with the lowest frequency count. This AT has no TL restrictions and is the one applied when none of the rest can be applied because their TL restrictions are not met.

Code generated for each alignment template. Code is generated by following the order specified by the TL part T_n of the AT. The generated code for each unit in T_n depends on the type of its word class:

- if the word class corresponds to a non-lexicalized word, code is generated to get the translation of the lemma of the aligned SL (non-lexicalized) word by looking it up in the bilingual dictionary, and to attach to the translated lemma the part-of-speech and inflection information provided by the TL word class;
- if the word class corresponds to a lexicalized word, it is introduced as is; remember that word classes belonging to lexicalized words store complete lexical forms consisting of lemma, part-of-speech and inflection information.

Note that the information about SL lexicalized words is not taken into account when generating the code for a given AT.

Lang.	# sent.	# words
`es`	100 834	1 952 317
`ca`	100 834	2 032 925

Table 1: Number of sentences and words in the Spanish–Catalan parallel corpus used for training.

Trans. dir.	Eval. corpus	# words
`es-ca`	post-edit	10 066
	parallel	13 147
`ca-es`	post-edit	10 024
	parallel	13 686

Table 2: Number of words of the two different corpora (see section 6) used for evaluation for each translation direction.

Example of AT application. The following example illustrates how the AT shown in figure 3 would be applied to translate from Spanish to Catalan the input text *vivieron en Francia*.[11] This text segment, after morphological analysis and part-of-speech tagging, is transformed by the MT engine into the intermediate representation *vivir*-`(verb.pret.3rd.pl)` *en*-`(pr)` *Francia*-`(noun.loc)`, which becomes the input to the structural transfer module.

The AT is applied in the order specified in its TL part. For the word classes corresponding to non-lexicalized words, the aligned SL words are translated into TL (Catalan) by looking them up in the bilingual dictionary: *vivir* is translated as *viure* and *Francia* is translated as *Franca*. Then, the inflection information provided by the TL part of the AT (see figure 3) is attached to each translated lemma. Finally, word classes corresponding to lexicalized words are just copied to the output as they appear in the TL part of the AT. For the running example the structural transfer output would be: *anar*-`(vaux.pres.3rd.pl)` *viure*-`(verb.inf)` *a*-`(pr)` *Franca*-`(noun.loc)`, which the generation module would transform into the Catalan phrase *van viure a Franca*.

6 Experiments

Task. We have tested our approach on both translation directions of the Spanish–Catalan (`es-ca`) language pair.[12] Table 1 shows the number of sentences and words in the training parallel corpus; this corpus comes from *El Periodico de Catalunya*,[13] a daily newspaper published both in Catalan and Spanish.

The definition of word classes is performed by considering a small set with around 8 lexicalized categories (see section 4.1) for each language. The most common lexicalized categories are: prepositions, pronouns, determiners, subordinate conjunctions, relatives, modal verbs and auxiliary verbs. Remember from section 4.1 that only categories usually involved in lexical changes are lexicalized.

Evaluation. The performance of the presented approach is compared to that of the same MT system when no transfer rules are used at all (word-for-word MT), and that of using hand-coded transfer rules. To that end we calculate the word error rate (WER) computed as the word-level *edit distance* (Levenshtein, 1965) between the translation performed by the MT system for a given setup and a reference translation divided by the number of words in the evaluated translation.

Table 2 shows the number of words of the different corpora used form evaluation. Note that two different evaluation corpora have been used, one (post-edit) in which the reference translation is a post-edited (corrected) version of the MT performed when using hand-coded transfer rules, and another (parallel) in which the text to translate and the reference translation come from a parallel corpus analogous to the one used for training.

Results. Table 3 shows the results achieved for each translation direction and evaluation corpus. The error rates reported are: (a) the results of a word-for-word translation (when no structural transformations are applied),

[11] Translated into English as *They lived in France*.

[12] All linguistic data used can be freely downloaded from `http://sourceforge.net/projects/apertium`, package `apertium-es-ca-1.0.2` .

[13] `http://www.elperiodico.com`

Trans. dir.	Eval. corpus	No rules	AT count	AT log	Hand
es-ca	post-edit	12.6 %	8.6 %	8.5 %	6.7 %
	parallel	26.6 %	20.4 %	20.4 %	20.8 %
ca-es	post-edit	11.6 %	8.1 %	8.1 %	6.5 %
	parallel	19.3 %	15.0 %	14.9 %	14.5 %

Table 3: Word error rate (WER) for each translation direction and evaluation corpus. The error rates reported are (from left to right): the result when no transfer rules are used, the best result achieved when the count is used directly when discarding infrequent ATs (AT count), the best result achieved when a modified frequency count is used when discarding infrequent ATs (AT log, see section 5), and the results achieved when hand-coded transfer rules are used.

(b) the results when the frequency count is directly used to discard infrequent ATs, (c) the results when a modified frequency count (see section 5) is used to discard infrequent ATs, and (d) the results achieved when using hand-coded transfer rules; in all cases the same linguistic data (morphological and bilingual dictionaries) were used.

As can be seen, when evaluating via a post-edited translation, handcrafted rules perform better than our method; however, they give comparable results when using a evaluation corpus similar to the one used for training. This result suggests, on the one hand, that our training method produces text of the same style of that used for training and, on the other hand, that even though they "learn" the style of the training corpus, the translation quality for other texts is quite good. Note that the post-edited translation used as reference is a corrected version of a MT performed with the same handcrafted rules; therefore, this evaluation is slightly biased towards the system using handcrafted rules.

Finally, note that both criteria used to discard infrequent ATs (see section 5) give comparable results for both translation directions. This may be explained by the fact that, on the one hand, rules that do not apply any AT (because of TL restrictions not being met) perform a word-for-word translation, and on the other hand, rules with longer ATs have more restrictions to check and, therefore, they are more likely to eventually perform a word-for-word translation.

7 Discussion

In this paper the generation of shallow-transfer rules from statistically-inferred alignment templates (ATs) has been tested. To this end, little linguistic information, in addition to the linguistic data used by the MT engine, has been used in order to learn, not only syntactic changes, but also lexical changes to apply when translating SL into TL. This linguistic information consists of a small set of lexical categories involved in lexical changes (prepositions, pronouns, etc.) and can be easily provided.

The method presented has been tested using an existing open-source shallow-transfer MT system. The performance of the system when using the automatically generated rules has been compared to that of a word-for-word translation (when no structural transformations are applied) and that obtained using hand-coded transfer rules. In all cases, there is 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 hand-coded transfer rules, being comparable in some cases.

Finally, we plan to improve the generated rules so that they apply shorter ATs inside the same rule when none of the longer ATs can be applied because of TL restrictions not being met. This gradual "back-off" code in rules would avoid falling back straight into word-for-word translation as it is done now. We also plan to test the presented method with other Apertium-based linguistic packages. Preliminary results on the Spanish–Portuguese language pair show results in agreement to those provided in this paper when evaluating through a parallel corpus.

Acknowledgements

Work funded by the Spanish Government through projects TIC2003-08681-C02-01 and TIN2006-15071-C03-01, and by the Spanish Government and the European Social Fund through research grant BES-2004-4711. We thank G. Ramırez-Sanchez for her help when defining the set of lexicalized categories.

References

C. Armentano-Oller, R.C. Carrasco, A.M. Corbı-Bellot, M.L. Forcada, M. Ginestı-Rosell, S. Ortiz-Rojas, J.A. Perez-Ortiz, G. Ramırez-Sanchez, F. Sanchez-Martınez, 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.

D. 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.

R.D. Brown. 1999. Adding linguistic knowledge to a lexical example-based translation system. In *Proceedings of the Eighth International Conference on Theoretical and Methodological Issues in Machine Translation (TMI-99)*, pages 22–32.

H.M. Caseli, M.G.V. Nunes, and M.L. Forcada. 2005. LIHLA: A lexical aligner based on language-independent heuristics. In *Proceedings of the V Encontro Nacional de Inteligencia Arti cial (ENIA 2005)*, pages 641–650.

I. Cicekli and H.A. Guvenir. 2001. Learning translation templates from bilingual translation examples. *Applied Intelligence*, 15(1):57–76.

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.

H. Kaji, Y. Kida, and Y. Morimoto. 1992. Learning translation templates from bilingual text. In *Proceedings of the 14th Conference on Computational Linguistics*, pages 672–678. Association for Computational Linguistics.

A. Lavie, K. Probst, E. Peterson, S. Vogel, L. Levin, A. Font-Llitjos, and J. Carbonell. 2004. A trainable transfer-based machine translation approach for languages with limited resources. In *Proceedings of Workshop of the European Association for Machine Translation (EAMT-2004)*.

V.I. Levenshtein. 1965. Binary codes capable of correcting deletions, insertions, and reversals. *Doklady Akademii Nauk SSSR*, 163(4):845–848. English translation in Soviet Physics Doklady, 10(8):707-710, 1966.

Y. Liu and C. Zong. 2004. The technical analysis on translation templates. In *Proceedings of the IEEE International Conference on Systems, Man & Cybernetics (SMC)*, pages 4799–4803. IEEE.

A. Mikheev. 1996. Unsupervised learning of word-category guessing rules. In *Proceedings of the Thirty-Fourth Annual Meeting of the Association for Computational Linguistics*, pages 327–333.

F.J. Och and H. Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

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

F.J. Och, C. Tillmann, and H. Ney. 1999. Improved alignment models for statistical machine translation. In *Proceedings of the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, pages 20–28.

K. Probst, L. Levin, E. Peterson, A. Lavie, and J. Carbonell. 2002. MT for minority languages using elicitation-based learning of syntactic transfer rules. *Machine Translation*, 17(4):245–270.

F. Sanchez-Martınez 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.