Automatic Translation of Biomedical Terms by Supervised Machine Learning

Vincent Claveau

IRISA-CNRS
campus de Beaulieu
35042 Rennes, France
Vincent.Claveau@irisa.fr

Abstract

In this paper, we present a simple yet efficient automatic system to translate biomedical terms. It mainly relies on a machine learning approach able to infer rewriting rules from pair of terms in two languages. Given a new term, these rules are then used to transform the initial term into its translation. Since conflicting rules may produce different translations, we also use language modeling to single out the best candidate. We report experiments on different language pairs (including Czech, English, French, Italian, German, Portuguese, Spanish and even Russian); our approach yields good results (varying according to the considered languages) and outperforms existing ones for the French-English pair.

1. Introduction

In the biomedical domain, the international research framework makes knowledge resources such as multilingual terminologies and thesauri essential to carry out many researches. Such resources have indeed proved extremely useful for applications such as international collections of epidemiological data, machine translation (Langlais and Carl, 2004), and for cross-language access to medical publication. This last application has become an essential tool for the biomedical community. For instance, the well-known PubMed document retrieval system gathers over 17 millions citations and processes about 3 millions queries a day (Herskovic et al., 2007)!

Unfortunately, up to now, little is offered to non-English speaking users. Most of the existing terminologies and document collections are in English, and the foreign or multilingual resources are far from being complete. For example, there are over 4 millions English entries in the 2006 UMLS Metathesaurus (Bodenreider, 2004), 1.2 million Spanish ones, 98 178 for German, 79 586 for French, 49 307 for Russian, and only 722 entries for Norwegian. Moreover, due to fast knowledge update, even well-developed multilingual resources need constant translation support. All these facts point up the need for automatic techniques to produce, manage and update these multilingual resources.

Within this context, we propose to present in this paper an original method to translate biomedical terms from one language to another. This method aims at getting rid of the bottleneck caused by the incompleteness of multilingual resources in most real-world applications. The new word-to-word translation approach we propose makes it possible to translate automatically a large class of simple terms (*i.e.*, composed of one word) in the biomedical domain from one language to another. It is tested and evaluated on translations within various language pairs (including Czech, English, French, German, Italian Portuguese, Russian, Spanish). Our approach relies on two major hypotheses:

- a large class of terms from one language to another are morphologically related;

- differences between such terms are regular enough to be automatically learned.

These two hypotheses, only stated for the biomedical domain, make the most of the fact that, most of the time, biomedical terms share a common Greek or Latin basis in many languages, and that their morphological derivations are very regular (Deléger et al., 2007). These regularities appear clearly in the following French-English examples: ophtalmorragie/ophthalmorrhagia, ophtalmoplastie/ophthalmoplasty, leucorragie/leukorrhagia...

The main idea of our work is that these regularities can be learnt automatically with well suited machine learning techniques, and then can be used them to translate new or unknown biomedical terms. We thus proposed a simple machine learning approach allowing us to infer a set of rewriting rules; it only necessitates examples of pairs of terms that are translation of each other (different languages can be considered as source or target). The inferred rules operate at the letter level; once they are learnt, they can be used to translate new and unseen terms into the target language. It is worth noting that neither external data nor knowledge is required besides the gathering of examples of paired terms for the languages under consideration. Moreover, these examples are simply taken from the multilingual terminologies that we aim at completing; thus, this is a fully automatic process.

After the description of related studies, Section 3. is dedicated to the description of the translation method and Section 4. gives some of its results for a pure translation task.

2. Related work

Few researches aim at translating terms directly from one language to another. One close work is the one of S. Schulz *et al.* (2004) about the translation of biomedical terms from Portuguese into Spanish with rewriting rules which are further used for biomedical information retrieval (Markó et al., 2005). Unfortunately, contrary to our work, these rules are hand-coded making this approach not portable. In a previous work (Claveau and Zweigenbaum, 2005a), an automatic technique relying on inference of transducers (finite-state machines allowing rewriting while analyzing a string)

was proposed. The main drawback of this approach was that it could only handle language pairs sharing the same alphabet and produced less reliable results than the one presented in this paper (see Section 4.3.). More recently, Langlais *et al.* (Langlais and Patry, 2007) proposed a very interesting approach to translate unknown words based on analogical learning. This technique seems promising and its use to translate biomedical terms is under study (Langlais et al., 2007).

Apart from these studies, related problems are often addressed in the domain of automatic corpus translation. Indeed, the cognate detection task (cognates are pairs of words with close forms) (Fluhr et al., 2000, *inter alia*) relies on morphological operations (edit distance, longest common sub-string...) sometimes very close to the rewriting rules we infer. Other studies are based on corpus-based analysis using statistical techniques or lexical clues (punctuation marks, digits...) in order to discover alignments —thus, possible translation relations— between terms in aligned corpora (Ahrenberg et al., 2000; Gale and Church, 1991; Tiedemann, 2004) or comparable corpora (Fung and McKeown, 1997b; Fung and McKeown, 1997a). Besides the problem of the lack of such specialized corpora, these approaches differ from ours in that their goal is to exhibit the translation of a word in a text (relationship problem) whereas we are addressing the problem of producing the translation of a term without other information (generation problem). Moreover, most of the times, these alignment techniques actually need pairs of terms that are translation of each other as a starting point (Véronis, 2000, for a state-of-the-art).

More generally, statistical machine translation (Brown et al., 1990) addresses a close problem; of course, in our case, the sequence to translate is composed of letters and not words. Yet, the method we propose bears many similarities with the standard statistical approach that uses a translation model and a language model (Brown et al., 1993). Nonetheless, the kind of data we manipulate implies important differences. First, we are not concerned with the problem of reordering words, taken into account in the IBM models through the distortion parameter: indeed, the morpheme order (and thus the letter order) of our terms hardly varies from one language to another. Similarly, the fertility parameters or null words in these models, which are used to tackle the word-to-word translation exceptions, are not suited for our data. Such problems are indeed naturally handled by our inference technique which allows us to generate rewriting rules translating not only letter to letter but also from a string of letter to another string of letter of different length.

Studies on transliteration, for instance for Japanese (katakana) (Qu et al., 2003; Tsuji et al., 2002; Knight and Graehl, 1998, for example) or for Arabic (Al-Onaizan and Knight, 2002; AbdulJaleel and Larkey, 2003) bear lot of similarities with our approach. Indeed, some of the techniques used in this domain are very close to one we detail hereafter, but usually only concern the representation of foreign words (mainly named entities) in languages having a different alphabet than the source words. Most of these techniques include a step aiming at transforming the term as a sequence of phonemes; they are said *phonetic-based*. Such techniques thus require external knowledge (letters-to-phonemes table, source-language phonemes to target-language phonemes table...) which makes the approach efficient but not portable to other pairs of languages. The technique we present is in fact similar to some existing approaches developed in the *spelling-based* framework (Al-Onaizan and Knight, 2002). In the existing studies about named-entity transliteration, the two translation directions are not considered as equivalent: one speaks about *forward transliteration* (for example, transliteration of an Arabic name into the Latin alphabet) and *backward transliteration* (retrieving the original Arabic name from its Latin transliteration). This distinction often implies differences in the techniques used (vowelization in Arabic for example); it is not relevant to our approach. Our technique is fully symmetric even if the performances may vary from one translation direction to the other.

Finally, let us mention the studies on computational morphology in which machine learning approaches have been successfully used to lemmatize (Erjavec and Džeroski, 2004), to discover morphologic relations (Gaussier, 1999; Moreau et al., 2007) or to perform morphographemic analysis (Oflazer and Nirenburg, 1999). The technique of rewriting rule inference presented in the next section falls within the scope of such studies.

3. Translation technique

The translation technique we propose here works in two steps. First, rewriting rules (see below for examples) are learnt from examples of terms that are translations of each other in the considered language pair. This set of rules is then used to translate any new term, but conflicting rules may produce several rivaling translations. In order to choose only one translation, we use a language model, learnt on the training data, and keep the most probable translation.

3.1. Rewriting rules

Our translation technique aims at learning rewriting rules (that can also be seen as transliteration rules). These rules, inferred from lists of bilingual term pairs (*cf.* Section 4.1.), have the following form:

$$\langle input\ string \rangle \rightarrow \langle output\ string \rangle$$

In the remaining of this paper, we note r a rewriting rule; $\mathcal{R}$ is the list of every rule inferred during an experiment, $input(r)$ and $output(r)$ respectively refer to the input and output strings of the rule r.

Algorithm 1 gives an overview of our machine learning approach. The first step is performed by the software DPalign (http://www.cnts.ua.ac.be/~decadt/?section=dpalign). It is used to align two sequences by minimizing their edit distance with the dynamic programming approach proposed by Wagner & Fischer (1974); the necessary costs of substituting characters are computed on the whole set of pair to be aligned. Thus, this software does not rely on a formal similarity between characters; it makes it possible to align terms that do not share the same alphabet.

$\mathcal{L}$ Portuguese-English	$\mathcal{L}$ English-Russian
#cetosteróides#	#adenosinetriphosphatase#
#ketosteroid_s#	#аденозин_триф_осф_атаза#
#electroporação_#	#hydrox_ypregnenolone#
#electroporation#	#гидроксипрегненолон_#
#encef_alograf_ia#	#keratoplasty__#
#encephalography_#	#кератопластика#

Table 1: Examples of alignment produced for two language pairs

---

Algorithm 1 Inferring rewriting rules
1: align term pairs at the letter level, put the results in $\mathcal{L}$
2: **for all** term pair W1 in $\mathcal{L}$ **do**
3:    **for all** letter alignment of W1 in which the 2 letters differ **do**
4:       find the best hypothesis of rule r in the search space $\mathcal{E}$
5:       add r to the set of rules $\mathcal{R}$
6:    **end for**
7: **end for**

A list of paired terms is provided in input of DPalign; to each term, we add two characters # to represent the beginning and the end of the string of letters. The output list $\mathcal{L}$ then contains the paired terms aligned at a letter-level; Table 1 presents some examples for two language pairs ('_' means *no character*). Hereafter, the source term of pair p (respectively the target term of p) is written $input(p)$ (resp. $output(p)$); moreover, $align(x, y)$ means that the sub-string x is aligned with sub-string y in the considered term pair.

For each difference between two aligned letters, our algorithm has to generate the best rewriting rule. Many rules are eligible: consider for example the difference i/y in the French-English word pair #opht_almologie#/#ophthalmology_#; some of the rewriting rules our algorithm could generate in this context are i $\rightarrow$ y, gi $\rightarrow$ gy, ie $\rightarrow$ y (note that we do not write the _ character), ologie# $\rightarrow$ ology#, and so on.

The score of a rule is computed from the list $\mathcal{L}$; it is defined as the ratio between the number of times the rule can actually be applied and the number of times the premise of the rule matches a source term from the example list. Thus, formally, it is defined as:

$$score(r) = \frac{|\{p \in \mathcal{L} \mid input(r) \subseteq input(p) \wedge output(r) \subseteq align(input(r), p)\}|}{|\{s \in \mathcal{L}_{input} \mid input(r) \subseteq s\}|}$$

where $\subseteq$ represents the inclusion of character string (for example, $abc \subseteq aabca$).

3.2. Lattice of rules

In order to efficiently choose the best rule among these possibilities, we define a hierarchical relation between rules.

Definition 1 (Hierarchical relation) *Let r_1 and r_2 be two rules, then $r_1 \succeq r_2 \Leftrightarrow (input(r_1) \subseteq input(r_2) \wedge output(r_1) \subseteq output(r_2))$.*

If $r_1 \succeq r_2$, then r_1 is said more general than r_2. This hierarchical relation defines a partial order in the search space $\mathcal{E}$; thus, it makes it possible to order rules hierarchically in $\mathcal{E}$, resulting in a lattice of rules.

Proof ──────────
Reflexivity. For any rule r, we obviously have $input(r) \subseteq input(r) \wedge output(r) \subseteq output(r)$, thus $r \succeq r$.
Transitivity. Let r_1, r_2 and r_3 be three rules such that $r_1 \succeq r_2$ and $r_2 \succeq r_3$. We have $input(r_1) \subseteq input(r_2)$, $input(r_2) \subseteq input(r_3)$, thus $input(r_1) \subseteq input(r_3)$, and similarly we have $output(r_1) \subseteq output(r_3)$. Finally, we have $r_1 \succeq r_3$.
Anti-symmetry. Let r_1 and r_2 be two rules such that $r_1 \succeq r_2$ and $r_2 \succeq r_1$. We have $input(r_1) \subseteq input(r_2)$, $input(r_2) \subseteq input(r_1)$, thus $input(r_1) = input(r_2)$, and similarly we have $output(r_1) = output(r_2)$. Finally, we have $r_1 = r_2$.
Thus, this relation defines a partial order. It is not a total order since we can have r_1 and r_2 such that $r_1 \not\succeq r_2$ and $r_2 \not\succeq r_1$. □

Figure 1 presents such a search space built from the difference i/y in the alignment #opht_almologie#/#ophthalmology_#.
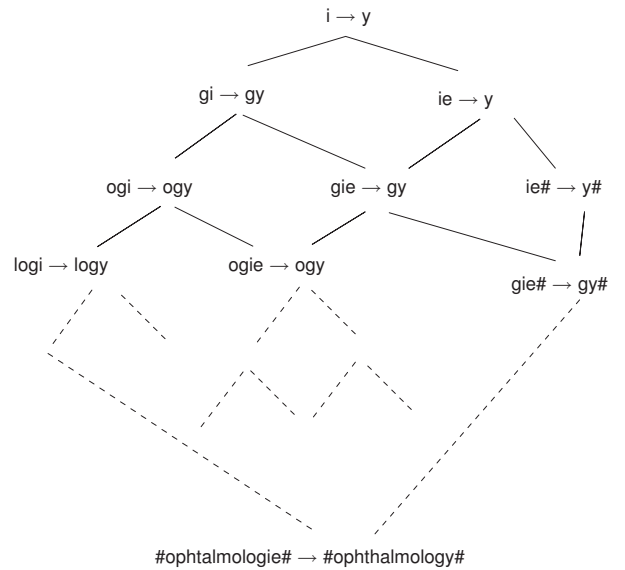

Figure 1: Search lattice $\mathcal{E}$ from the example i/y in #opht_almologie#/#ophthalmology_#

In practice, these lattices of rules are explored top-down; the rules are generated on the fly with a very simple oper-

ator that generates more specialized rules from an existing one. Consider the rule $r_1 = i \rightarrow y$ in the previous example. This is the most general rule for the difference i/y in the alignment #opht_almologie#/#ophthalmology_#. After the computing of its score the algorithm will generate and score every rule that is immediately more specific, that is:

$$\{r_2 \mid r_1 \succeq r_2 \wedge \nexists r_3 \text{ s.t. } r_1 \succ r_3 \succ r_2\}$$

The generation of these specific rules is simply done by adding the letter on the right (respectively on the left) from the input word of the paired term used as example to the input of r_1 and adding the corresponding aligned letter to the right (resp. to the left) of its output. Thus, by applying this to r_1 we have: $g\, input(r_1) \rightarrow g\, output(r_1)$ and $input(r_1)\, e \rightarrow output(r_1)\, \_$, that is $gi \rightarrow gy$ and $ie \rightarrow y$. The inheritance properties of the lattices and this specialization operator make it possible to choose quickly the best rewriting rules for each example according to the score function which is consistent with the specialization operator. Indeed, computing the score is the most time-consuming task of our algorithm because it necessitates analyzing every word in the training set $\mathcal{L}$. However, by considering the hierarchical relation and the way hypothesis are generated, a big amount of time can be saved: for a term pair used as an example, consider two rules r_1 and r_2 such that $r_1 \succeq r_2$. When computing the score of r_2, we have for any word pair p: $input(r_1) \subseteq input(r_2) \subseteq input(p)$, that is, if p is such that $input(r_2) \subseteq input(p)$ then necessarily p was analyzed when computing the score of r_1. Therefore, we do not need to examine every word of $\mathcal{L}$ to compute the score of r_2 but only those that were covered by the denominator of r_1.

3.3. Using the rules and language modeling to translate

Every difference between two aligned letters in every term pair thus ends up with one rewriting rule chosen in the corresponding lattice. All the rules are collected in $\mathcal{R}$. Given a new term to translate, every rewriting rule of $\mathcal{R}$ that can be applied (*i.e.* rules inferred in the training set in which the input string corresponds to a sub-string of the term) is indeed applied. In case of conflicting rules (rules with the same or overlapping premise), all possibilities are generated. Thus, at this stage, a word can receive several concurrent translations. Therefore, the second step of our approach consists in a post-processing technique in order, on the one hand, to select only one of these proposed translations, and on the other hand, to give the user a confidence factor for the result.

These two tasks are conjointly performed by assigning a probability to each possible translation according to a language model (LM). That is, with standard notations, for a word w composed of the letters $l_1, l_2, ..., l_m$:

$$P(w) = \prod_{i=1}^{m} P(l_i \mid l_1, ..., l_{i-1})$$

In practice, the probabilities $P(l_i|...)$ are estimated from the list of output words used as examples in the first step, decomposed in n-grams of letters. As usual with language modeling, to prevent the problem of unseen sequences, the probabilities are actually computed with a limited history, that is we only consider the $n - 1$ previous letters:

$$P(w) = \prod_{i=1}^{m} P(l_i \mid l_{i-n+1}, ..., l_{i-1})$$

In the experiments presented below, we use $n = 7$ letters. A simple smoothing technique is also used to provide more reliable estimations.

Intuitively, the LM aims at favoring translations that "look like" correct words of the output language. Thus, among all the proposed translations, we only keep the one with the better LM score. Contrary to similar work in machine translation or transliteration, we do not include the score of the rewriting rules in this final score. Indeed, preliminary experiments tended to show that it performed worse than using only the LM score, especially when few examples were used.

This language modeling approach also enables to avoid some problems. As it was underlined by Claveau & Zweigenbaum (2005b), some words have similar forms but different Part-of-Speech or semantic role. If available, these additional pieces of information may avoid translation errors. For example, a word in -ique in French may be translated in English in -ic if it is an adjective (*e.g.* dynamique/dynamic) or in -ics if it is a noun (*e.g.* linguistique/linguistics). Similarly, a word in -logie in French may be translated in -logy if it concerns a science (biologie/biology) or in -logia if it concerns a language disorder (dyslogie/dyslogia). It is worth noting that this part-of-speech and semantic information, if available in the data, can easily be used with the language model: the probabilities estimated from the training data are simply conditioned to the information ($Info$ hereafter) we want to consider, that is:

$$P(w, Info) = \prod_{i=1}^{m} P(l_i \mid l_{i-n+1}, ..., l_{i-1}, Info)$$

4. Translation Experiments

This section presents some of the experiments we made with the translation technique previously described. We describe the data, the experimental framework we used and finally the results obtained at this translation task.

4.1. Data

Two different kinds of data are used for these experiments. First, in order to compare our translation approach with previous work, we use the same French-English pairs of terms used in (Claveau and Zweigenbaum, 2005a), that is a list of terms taken from the Masson medical dictionary (http://www.atmedica.com). The second set of data used in our experiments is the UMLS Metathesaurus (Tuttle et al., 1990; Bodenreider, 2004). This collection of thesauri brings together biomedical terms from 17 languages with a language-independent identifier allowing us to form the necessary bilingual pairs of terms. For these two sets, we only consider simple terms (*i.e.* one-word terms) in both studied languages, and we disregard acronyms.

4.2. Experimental framework

In order to evaluate our results, we follow a standard protocol. The word pair list is split into two parts: the first one is used for the learning process as described above (rule inference and language modeling), and the second one, set to contain 1000 pairs, is used as a testing set. Once the rules have been inferred and language modeling has been done, we apply them to every input word of the testing set. We then compare the generated translation with the expected output word; if the two strings exactly match, the translation is considered as correct, in every other case, it is considered as wrong.

The results are evaluated in terms of precision (percentage of correct translations generated). Nonetheless, since the LM gives a confidence factor to each translation, we can decide to keep only those with a LM score greater than a certain threshold. If this threshold is set high the precision may be high, but the number of words actually translated may be low, and conversely. Thus, in order to represent all the possibilities, results below are presented as graphs where each point corresponds to the precision and the percentage of words translated for a certain LM score threshold.

4.3. Results

4.3.1. Translation between French and English

As a first experiment, we focus our attention on the French-English language pair with the help of the Masson data in order to compare these results to those of Claveau & Zweigenbaum (2005a). Figure 2 and 3 respectively present the precision graph of the French into English and English into French translation experiments. In close languages such as French and English, many specialized words are exactly the same. Thus, as a simple baseline, we compute the precision that would be obtained by a system systematically proposing the input term as its own translation. We also indicate the best precision obtained by the transducer based technique exposed in (Claveau and Zweigenbaum, 2005a) within the same experimental framework and data. Whatever the translation direction, the two graphs show that our approach performs very well: for French into English translations, it yields a precision of 85.4% when every word is translated, and 84.8% for English into French. In both cases, it represents a 10% improvement over the transducer-based approach (Claveau and Zweigenbaum, 2005a). Our technique clearly outperforms the baseline results, but it is also worth noting that about 25% of the biomedical terms are identical in French and English, which indicates that the two languages are close enough to make the learning task relatively easy.

Concerning the use of the language modeling, several things are noteworthy. First, without including the Part-of-speech information, the precision rates are a bit lower (82.6% for French to English and 84.8% for English to French). Secondly, if we choose the translation at random among all the generated ones instead of choosing the one with the best LM score, the precision rate falls to about 50% for both translation directions. Finally, if the good translation were always chosen (when it appears in the list of potential translations produced by the rewriting rules),

the precision would reach 90%. It means that the language model makes very few mistakes at choosing the final translation among the different proposals. These facts clearly show the interest of using language modeling and, if available, to include the Part-of-Speech information in it.

4.3.2. Computation time and performances vs. number of examples

In the previous experiments all the available examples (*i.e.* all the paired terms but those kept for the test set) were used to infer the rewriting rules. Let us now examine the results and the computation time when this number varies. Table 2 displays the results we obtain when we keep different number of examples to infer the rewriting rules and to learn the language model probabilities. In this table, we indicate the precision rate in the worst case (*i.e.* every translation is kept), the number of rules that are inferred, as well as the computing time due to the alignment step and the total inference time (including the alignment time). The experiments were carried out on a 1.5MHz Centrino Laptop running Linux, and the algorithm presented in Section 3.1. was entirely implemented in Perl.

Number of term pairs	Alignment time	Total execution time	Number of rules	Precision
5400	132s	146s	727	85.4%
3600	73s	84s	537	83.5%
2800	54s	62s	406	82.0%
1800	36s	42s	309	82.8%
1400	21s	28s	249	82.3%
660	10s	13s	164	80.4%
320	6s	9s	77	77.3%
130	3s	8s	39	76.3%

Table 2: Computation time and precision as a function of the number of term pairs used as examples

One can notice that the precision rates remain very good, even with very few examples ending up with few rules. This is particularly interesting since gathering such pairs of terms could be difficult for certain language pairs due to the lack of multilingual resources. Concerning the computation time, the inference process is fast enough to process several language pairs in a minimal amount of time, thanks to our efficient search in the rewriting rule lattices. Yet, the whole process is slowed down by the alignment step for which the dynamic programming algorithm clearly constitutes a bottleneck.

4.3.3. Other language pairs

The same experiment can be carried out with different language pairs from the UMLS Metathesaurus. We only exhibit some results from many possible combinations; contrary to the previous experiments, we do not include any part-of-speech information in the language modeling.

Figures 4 and 5 present the results obtained with two languages known to be close: Spanish and Portuguese. These results are very good: in the worst case (*i.e.* no LM threshold is set: every term is proposed a translation), 87.9% of Portuguese terms are correctly translated into Spanish and
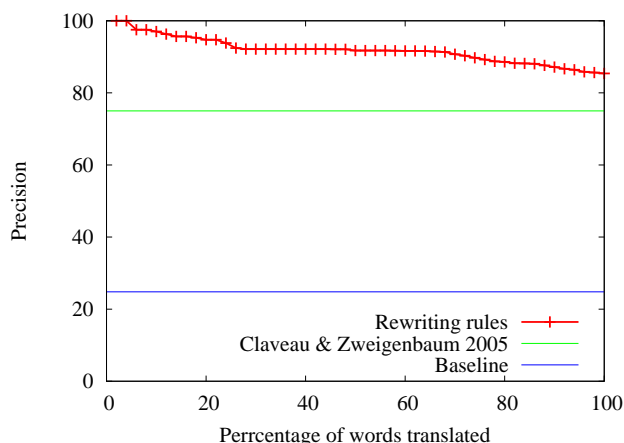
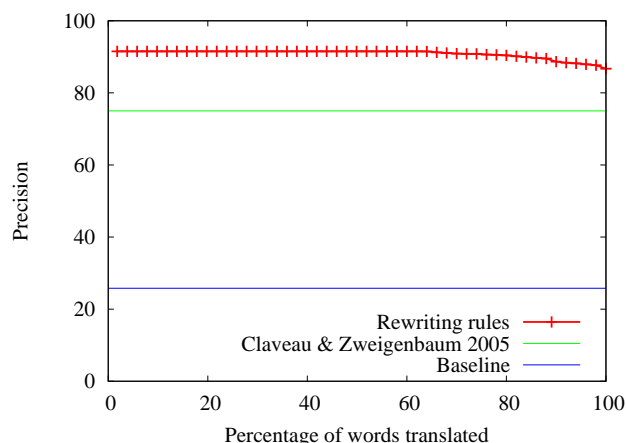Figure 2: Performances of French into English translation

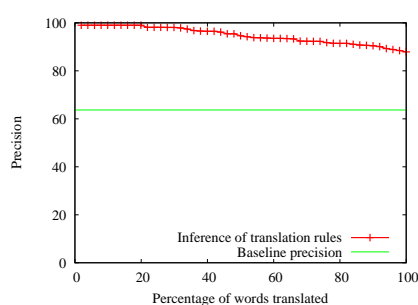Figure 3: Performances of English into French translation

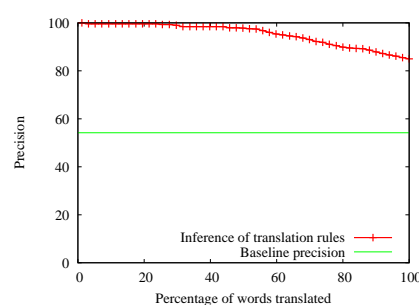Figure 4: Performances of translation from Portuguese into Spanish

Figure 5: Performances of translation from Spanish into Portuguese
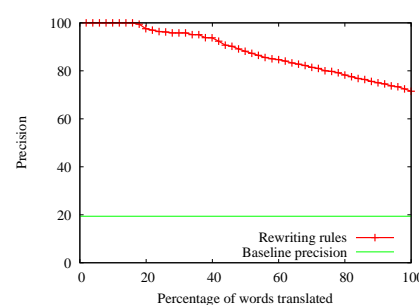

Figure 6: Performances of translation from Portuguese into English
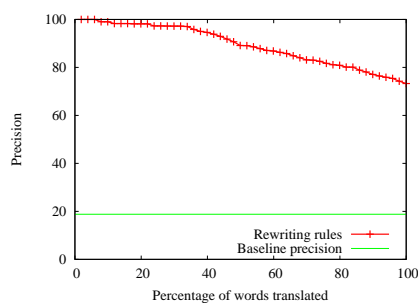

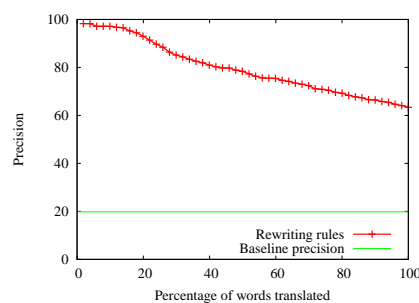Figure 7: Performances of translation from Spanish into English

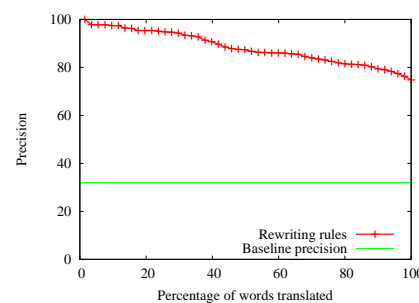Figure 8: Performances of Italian into English translation

Figure 9: Performances of Czech into English translation

85% for Spanish into Portuguese. This is not surprising given the closeness of the two languages, a closeness which further appears in the very high baseline precision. We now focus on translation into English, as it is the way which would be favored in real-world application such as cross-lingual information retrieval. As shown in Figures 6 and 7, translation from Spanish into English provides 71.7% of terms correctly translated; translation from Portuguese into English gives 75.5% of precision. Here again, the results are quite good; they also are in accordance with the proximity of Spanish and Portuguese since both languages perform similarly when translated into English.

Italian or Czech to English translations yield almost similar results as illustrated in Figures 8 and 9, even if these languages are not reputed to be specially close: in the worst case, 70% of Italian terms and 75.5% of Czech terms are correctly translated.

A more surprising result is obtained for German; although these two languages share strong historical links, Figure 10 clearly shows that German and English biomedical terms do not exhibit enough regularities to achieve the same precision rates than the previous languages. Nonetheless, in the worst case, our technique still yields 68.8% of correctly translated terms.

4.3.4. Language pairs with different alphabets
Let us now examine the translation performances of the Russian-English language pair. As we previously said, contrary to the technique of (Claveau and Zweigenbaum, 2005a), the approach described in this paper can be easily used with languages that do not share the same alphabet but show some regularities that can be learnt. Figures 11 and 12 present the results we obtain. Due to the different alphabets, the baseline is 0 in this case. The minimal precision rates (*i.e.* , when every word is translated) are 57.5% for English into Russian and 64.5% for Russian into English.
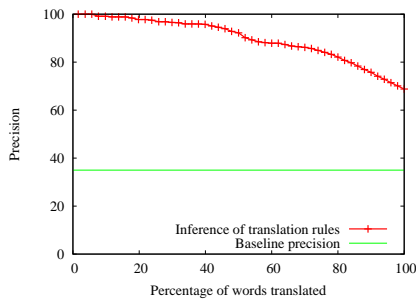
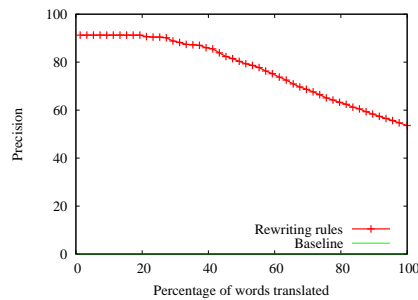Figure 10: Performances of German into English translation

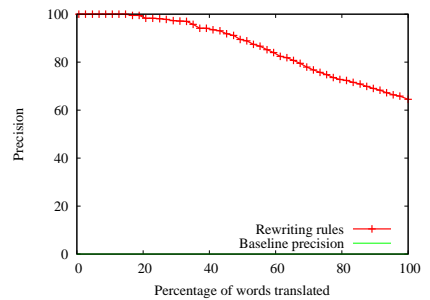Figure 11: Performances of translation from Russian into English

Figure 12: Performances of translation from English into Russian

These translation performances are surprisingly good given the apparent difficulty of the task. Of course, they are a bit lower than those of the other language pairs we examined but could be useful enough for many applications. It also proves if needed that most of the biomedical terms in Russian are built from Cyrilic transliterations of the same Latin and Greek morphemes used in English, French or Italian...

4.3.5. Common causes of errors

Our translation technique automatically captures existing regularities between biomedical terms in different languages. For this reason and unsurprisingly, when examining the results in detail, it appears that the main cause of error is due to the lack of morphological links between the source and the target term. Obviously, this is more often the case for the Russian-English language pair, but still occurs for languages known to be close (*e.g.* asimiento/grip for Spanish-Portuguese translation or embrochage/pinning for French-English). Besides these unavoidable errors, as already discussed, some errors are also due to similar forms with different part-of-speech or semantic features; as it was shown for the French/English experiments, most of these errors could be avoided if we had at our disposal the Part-of-Speech of semantic information. After all, the experiments tend to show that these cases are rare enough to make our approach yield good precision rates (though they are variable according to the considered languages).

5. Concluding remarks and perspectives

The method presented in this paper makes it possible to translate efficiently simple biomedical terms between various languages. It relies on a machine-learning technique inferring rewriting rules from examples of a list of bilingual term pairs and on a letter-based language modeling. These examples can be found easily in the existing –yet incomplete– multilingual terminologies; no other external knowledge or human intervention is needed. The approach is efficient and successful for translating unseen terms with a high precision, depending on the languages, and can thus be used to overcome problems due to incomplete multilingual language resources.

Many perspectives are foreseen for this work including technical enhancements and applications of our translation approach and its use in a cross-language framework. For instance, the translation of complex terms (terms composed of more than one word) is currently closely examined.

These terms are widely used in the biomedical domain (for instance, 50% of the MeSH terminology are complex terms), and some of them are not compositional, meaning that they cannot undergo the word-by-word translation our approach proposes. Moreover, even compositional terms would certainly necessitate a syntactic analysis to identify the head-modifier relations and thus translate it accordingly to these dependency relations. Lastly, our translation system bears numerous similarities with the standard statistical machine translation approach based on a translation model and a language model (Brown et al., 1993). The parallel between the two approaches could lead to interesting insights. Concerning the applications, a cross-lingual information retrieval system in the biomedical domain using this translation approach to translate queries is being studied; first results are promising.

6. References

Nasreen AbdulJaleel and Leah S. Larkey. 2003. Statistical transliteration for English-Arabic cross language information retrieval. In *Proceedings of the 12th International Conference on Information and Knowledge Management, CKIM'03*, pages 139–146, New Orleans, United-States of America.

Lars Ahrenberg, Mikael Andersson, and Magnus Merkel, 2000. *A knowledge-lite approach to word alignment*, chapter 5, pages 97–138. In Véronis (Véronis, 2000).

Yaser Al-Onaizan and Kevin Knight. 2002. Machine transliteration of names in arabic text. In *Proceedings of ACL Workshop on Computational Approaches to Semitic Languages*, Philadelphia, United-States of America.

Olivier Bodenreider. 2004. The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Research*, 32(D267-D270).

Peter F. Brown, John Cocke, Vincent J. Della Pietra Stephen A. Della Pietra, Fredrick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2).

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2).

Vincent Claveau and Pierre Zweigenbaum. 2005a. Automatic translation of biomedical terms by supervised transducer inference. In *Proceedings of the 10th Con-

ference on Artificial Intelligence in Medicine, AIME 05, Lecture Notes of Computer Science, Aberdeen, Scotland, UK. Springer.

Vincent Claveau and Pierre Zweigenbaum. 2005b. Traduction de termes biomédicaux par inférence de transducteurs. In *Actes de la conférence Traitement automatique des langues naturelles, TALN'05*, Dourdan, France.

Louise Deléger, Fiammetta Namer, and Pierre Zweigenbaum. 2007. Defining medical words: Transposing morphosemantic analysis from French to English. In *Proceedings of MEDINFO 2007*, volume 129 of *Studies in Health Technology and Informatics*, Amsterdam, Netherlands.

Tomaž Erjavec and Sašo Džeroski. 2004. Machine learning of morphosyntactic structure: Lemmatizing unknown Slovene words. *Applied Artificial Intelligence*, 18(1):17–41.

Christian Fluhr, Frédérique Bisson, and F. Elkateb, 2000. *Parallel text alignment using crosslingual information retrieval techniques*, chapter 9. In Véronis (Véronis, 2000).

Pascale Fung and Kathleen McKeown. 1997a. Finding terminology translations from non-parallel corpora. In *Proceedings of the 5th Annual Workshop on Very Large Corpora*, Hong Kong.

Pascale Fung and Kathleen McKeown. 1997b. A technical word and term translation aid using noisy parallel corpora across language groups. *Machine Translation*, 12(1/2):53–87.

William A. Gale and Kenneth W. Church. 1991. Identifying word correspondences in parallel texts. In *Proceedings of the 4th Darpa Workshop on Speech and Natural Language*, pages 152–157, Pacific Grove, CA, United-States of America.

Eric Gaussier. 1999. Unsupervised Learning of Derivational Morphology from Inflectional Corpora. In *Proceedings of Workshop on Unsupervised Methods in Natural Language Learning, 37th Annual Meeting of the Association for Computational Linguistics, ACL 99*, pages 24–30, Maryland, United-States of America.

Jorge R. Herskovic, Len Y. Tanaka, William Hersh, and Elmer V. Bernstam. 2007. A Day in the Life of PubMed: Analysis of a Typical Day's Query Log. *Journal of American Medical Informatics Association*, 14:212–220.

Kevin Knight and Jonathan Graehl. 1998. Machine transliteration. *Computational Linguistics*, 24(4):599–612.

Philippe Langlais and Michael Carl. 2004. General-purpose statistical translation engine and domain specific texts: Would it work? *Terminology*, 10(1):131–152.

Philippe Langlais and Alexandre Patry. 2007. Translating unknown words by analogical learning. In *Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLP-CoNLL)*, Prague, Czech Republic.

Philippe Langlais, François Yvon, and Pierre Zweigenbaum. 2007. What analogical learning can do for terminology translation? In *Proceedings of Computa-tional Linguisitics in Netherlands, CLIN'07*, Nijmegen, Netherlands.

Kornél Markó, Olena Medelyan Stefan Schulz, and Udo Hahn. 2005. Bootstrapping dictionaries for cross-language information retrieval. In *Proceedings of the 28th International Conference on Research and Development in Information Retrieval, SIGIR 05*, pages 528–535, Salvador, Brasil.

Fabienne Moreau, Vincent Claveau, and Pascale Sébillot. 2007. Automatic morphological query expansion using analogy-based machine learning. In *Proceedings of the 29th European Conference on Information Retrieval, ECIR 2007*, Roma, Italy.

Kemal Oflazer and Sergei Nirenburg. 1999. Practical bootstrapping of morphological analyzers. In *Proceedings of EACL Workshop on Computational Natural Language Learning, CONLL 99*, Bergen, Norway.

Yan Qu, Gregory Grefenstette, and David A. Evans. 2003. Automatic transliteration for Japanese-to-English text retrieval. In *Proceedings of the 26th International Conference on Research and Development in information Retrieval, SIGIR 03*, Toronto, Canada.

Stefan Schulz, Kornel Markó, Eduardo Sbrissia, Percy Nohama, and Udo Hahn. 2004. Cognate Mapping - A Heuristic Strategy for the Semi-Supervised Acquisition of a Spanish Lexicon from a Portuguese Seed Lexicon. In *Proceedings of the 20th International Conference on Computational Linguistics, COLING'04*, pages 813–819, Geneva, Switzerland, août.

Jörg Tiedemann. 2004. Word to word alignment strategies. In *Proceedings of the 20th International Conference on Computational Linguistics, COLING 04*, pages 212–218, Geneva, Switzerland.

Keita Tsuji, Béatrice Daille, and Kyo Kageura. 2002. Extracting French-Japanese word pairs from bilingual corpora based on transliteration rules. In *Proceedings of the Third International Conference on Language Resources and Evaluation LREC'02*, pages 499–502, Las Palmas de Gran Canaria, Spain, mai.

Mark S. Tuttle, David D. Sherertz, Nels E. Olson, Mark S. Erlbaum, D. Sperzel, Lauren Fuller, and Stuart J. Nelson. 1990. Using Meta-1 – the 1st version of the UMLS metathesaurus. In *Proceedings of the 14th annual Symposium on Computer Applications in Medical Care (SCAMC)*, Washington, D.C.

Jean Véronis, editor. 2000. *Parallel Text Processing*. Kluwer Academic Publishers, Dordrecht.

Robert A. Wagner and Mickael J. Fischer. 1974. The string-to-string correction problem. *Journal of the Association for Computing Machinery*, 21(1):168–173.