TAGGING AS A KEY TO SUCCESSFUL MT

Jan Hajič, Vladislav Kuboň
ÚFAL MFF UK
Malostranské nám. 25
118 00 Praha 1
Czech Republic
{hajic,vk}@ufal.mff.cuni.cz

Abstract

This paper describes the key role of a stochastic morphological tagger in an MT system between very closely related languages. The MT system Česílko exploits the close relatedness of both natural languages in question (Czech and Slovak), which allows substantial simplification of the translation method used. It also uses to a great advantage the possibilities of combination of a human translation and MT through a concept of translation memory. The paper also discusses general issues concerning the best tagging method used for languages with relatively rich morphology.

1. INTRODUCTION

One of the most widely used techniques of machine-aided human translation of the last decade or so is without doubts a method of human translation supported by a translation memory. This technique can substantially speed up the translation process especially when it concerns the translation and localization of various kinds of technical documentation.

The main idea of the translation memory is very simple. It takes an advantage from the fact that it is often the case (especially when localizing technical documentation) that for the currently translated document there is at least one document with similar content that had already been translated. Such a document may for example be a part of the previous version of the documentation to a particular software or hardware. The translation memory in fact contains both the source and target text divided into pairs of segments. These segments are typically sentences. When a human translator starts translating a new sentence, the system tries to match the source sentence with sentences already stored in the translation memory. If it is successful, it suggests the translation and the human translator decides whether to use it, to modify it or to reject it.

In our previous papers, e.g. [Hajič, Hric and Kuboň 2000], we have demonstrated that the use of translation memory (TM) has also some side effects, which can be exploited for making the translation process more automatic. In the same paper we have also described a method of "triangular" translation for a group of closely related languages through a pivot language using both human and machine translation.

In this paper we would like to concentrate on one part of our system, namely the module of automatic translation between very closely related languages, which is based on the stochastic morphological disambiguation of the source text.

2. THE USE OF THE TRANSLATION MEMORY IN THE SYSTEM ČESÍLKO

It is quite clear that the localization of the same source into several typologically similar target languages individually, one language pair after another, is a waste of money and effort. In the translation process it is necessary to solve very similar problems for each source-target language pair. The use of one language from the target group as a pivot and to perform the translation and localization through this language seems to be quite natural solution for these problems. It is of course much easier to translate texts from Czech to Polish or from Russian to Bulgarian than from English or German to any of these languages.

The system Česílko was designed as a tool allowing to automatically construct translation memories for human translators between very closely related languages (such Czech and Slovak). Such translation "memory" would then be used as if created by humans, but appropriately marked for the human translators.

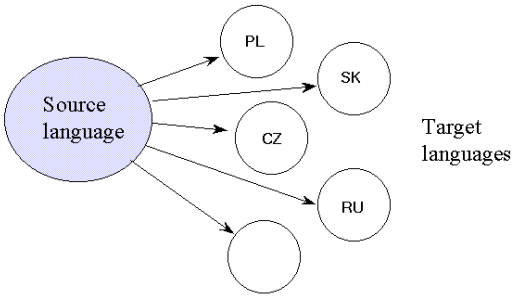

Fig.1 A traditional model for localization

If we have at our disposal two translation memories – one human made for the source/pivot language pair (say, English/Czech) and the other one created by an MT system for the pivot/target language pair (Czech/Slovak or Czech/Polish), the substitution of segments of a pivot language (Czech) by the segments of a target language (Slovak or Polish) is then only a routine procedure. The human translator translating from the source (English) to the target language (Slovak or Polish) then gets a translation memory for the required pair (source/target).

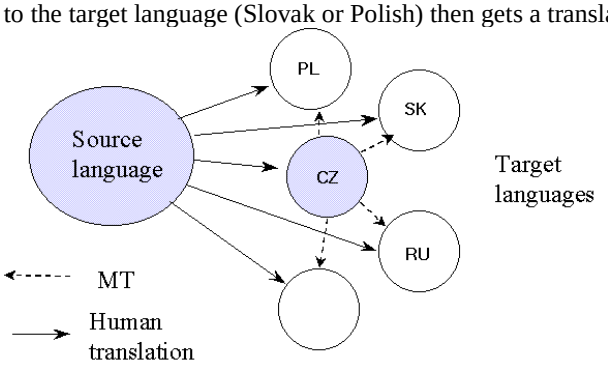

Fig.2 Our model for localization

The system of penalties which is normally applied to results of MT in translation memory based systems guarantees that if there is already a human-made translation present in the memory, it gets higher priority than the translation obtained as a result of MT.

3. BASIC PROPERTIES OF THE SYSTEM

In the group of Slavic languages there are more closely related languages than Czech and Russian. Apart from the pair of Serbian and Croatian languages, which are almost identical and were considered one language few years ago, the most closely related languages in this group are Czech and Slovak.

This fact has led us to the idea that the core of the system of a Czech to Slovak MT should use as simple method of analysis and transfer as possible. Our experience from an existing MT system RUSLAN (Czech-to-Russian MT system) aimed at the translation of software manuals for operating systems of mainframes – cf. [Oliva 89]) led us to the idea that a full-fledged analysis of Czech is not necessary. According to this experience, full syntactic analysis would be too unreliable. The other reason why the syntactic analysis of the source text was omitted was the fact that such an approach would not profit from the closeness of both languages as much as a less complicated method. The system therefore uses the method of direct word-for-word translation, the use of which is justified by the similarity (even though not identity) of syntactic constructions of both languages.

The system has already been tested on texts from the domain of documentation to corporate information systems. It is, however, not limited to any specific domain. Currently it is being tested on texts of a Czech general encyclopedia. Its primary task is, however, to provide support for translation and localization of various technical texts.

3.1 PROBLEMS OF MACHINE TRANSLATION BETWEEN CZECH AND SLOVAK

The greatest problem of the word-for-word translation approach is the problem of ambiguity of individual word forms. The type of ambiguity differs slightly between the group of languages with a rich inflection (majority of Slavic languages) and the group of languages that do not have such a wide variety of forms derived from a single lemma. For example, in Czech there are only rare cases of part-of-speech ambiguities (stát [to stay/the state], žena [woman/chasing] or tři [three/rub(imper.)]), however, the ambiguity of gender, number and case is very high (for example, the form of the adjective *jarní* [spring] is 27-times ambiguous). The main problem is

that even though several Slavic languages have the same property as Czech, the ambiguity is not preserved at all or it is preserved only partially, it is distributed in a different manner and the “form-for-form” translation is not applicable.
Without the analysis of at least nominal groups it is often very difficult to solve this problem, because for example the actual morphemic categories of adjectives are in Czech distinguishable only on the basis of gender, number and case agreement between an adjective and its governing noun. An alternative way to the solution of this problem was the application of a stochastically based morphological disambiguator for Czech whose success rate is relatively very high.
The basic problems of automatic translation between Czech and Slovak may be also demonstrated on the following example:

Example 1:
Source: *Při* zakládání *třídy výkazů se* třídě *nejprve přidělí* označení *a přiřadí se skupině uživatelů.*

Target: *Pri* zakladaní *triedy výkazov sa* triede *najprv pridelí* označenie *a priradí sa skupine užívateľov.*
[When a report class is founded, the class first receives a label and it is assigned to a group of users.]

The sample sentence contains two interesting phenomena – the translation of similar Czech word forms *zakládání* [founding] and *označení* [label] (both are nouns regularly derived from verbs) into Slovak forms *zakládaní* and *označenie* and the translation of the Czech word-form *třídě* [class/sorting].
The translation of the pair of similar words illustrates the fact that even though both languages are really very similar, a „full size” bilingual dictionary is necessary. The translation of similar words is irregular to the extent that prevents the use of some simpler mechanism (direct transcription).
The word form *třídě* may be translated into Slovak either as *triede* (if the original word form represents a noun) or as the form *triediac* (if the original form is a transgressive derived from the verb *třídit* [to sort]). This word form is another ilustration the need of a reliable tagger capable of high quality morphological disambiguation of the input.
Taken these facts into account, we came to the following composition of the system:

1. Import of the source (Czech input) sentence (a segment from an “empty” translation memory”)
2. Morphological analysis of Czech
3. Morphological disambiguation of Czech
4. Domain-related bilingual glossaries
5. General bilingual dictionary
6. Morphological synthesis of Slovak
7. Export of the output to the original translation memory (Slovak target sentence)

Let us now look in more detail on individual modules of the system:

Import of the source (Czech input) sentence

The input text is extracted out of a translation memory previously exported into an external file. The exported translation memory of TRADOS has a SGML-like markup with a relatively simple structure (cf. the following example):

Example 1. – A sample of the exported translation memory

<RTF Preamble>
...
</RTF Preamble>
...
<TrU>
<CrD>23051999
<CrU>VK
<Seg L=CS_01>Pomocí výkazů ad-hoc můžete rychle a jednoduše vytvářet rešerše.
<Seg L=SK_01>n/a
</TrU>

Our system uses only the segments marked by <Seg L=CS_01>, i.e. those which contain a source language sentence, and <Seg L=SK_01>, which is empty and which will later contain the same sentence translated into the target language.

Morphological analysis of Czech

The morphological analysis of Czech is based on the morphological dictionary developed by Jan Hajič and Hana Skoumalová in 1988-99 (for the tagset description, see [Hajič 1998]). The dictionary covers over 700,000 lemmas and it is able to recognize more than 15 mil. word forms. The morphological analysis uses a system of positional tags (each morphological category has a fixed place in the tag) with 15 positions.

Example 2 – tags assigned to the word form "pomocí" (help/by means of)

pomocí:

NFP2------A----
NFS7------A----
R--2----------

where :
N – noun; R – preposition
F – feminine gender
S – singular, P – plural
7, 2 – case (7 – instrumental, 2 – genitive)
A – affirmative (non-negative form)

The morphological analyzer is written in C and can effectively process about 5000 tokens per second (sustainable rate, including file compression/decompression, network file sharing, etc.).

Morphological disambiguation of Czech

The module of morphological disambiguation is a key to the success of the translation. It currently gets an average number of 4.29 tags per unit of text (word) on input (it used to be less in the recent past, but the average number of tags per token is growing due to the continuing expansion of the dictionary, the process of which creates new homonyms). The tagging system is based on an exponential probabilistic model (for the model definition and motivation, end evaluation results see [Hajič 1998]. The learning is based on a manually tagged corpus of Czech texts, containing roughly 1.2 mil. tokens. The system learns contextual rules (features) automatically and also automatically determines feature weights. The average accuracy of tagging is now over 94% (measured on tokens of running text).

Training of the tagger is based on mostly newspaper text, manually annotated within the Prague Dependency Treebank project (see Hajič 1998; for tagger training , we are using only the level 1 (morphological) annotation, not the level 2 (syntactic) annotation, of course.). Training took about 4 days of CPU time, when all the 15 morphological categories have been trained separately due to memory limitation reasons.

For feature selection, all combinations of up to 3 simple context constraints have been allowed, including individual morphological category ambiguity classes (both left and right), (full) tags as well as individual morphological category values (only in the left context, of course), individual morphological category membership in an ambiguity class (left and right context) and full word forms (again, both left and right). The maximum width of the context for feature selection was +-1 for fixed position context constraints, and +-4 for variable distance position context constraints. In variable distance context constraints, the position of a relevant token is determined by a particular major part-of-speech category value.

The number of possible features in a feature pool varies from several thousand (for categories with a very light ambiguity, such as possessive gender and number) to several million (for the most difficult morphological categories from the tagging point of view, such as case, number or gender). The resulting tagger has over 11 thousand rules total for all morphological categories (feature batches, in the terminology of [Hajič 1998] selected and weighted during the training process. These rules are stored in a SGML format and used by the runtime module of the tagger for tagging both the Czech side of the dictionaries during the preprocessing stage and for tagging the Czech input sentence during the translation proper.

Even though in theory it is not necessary, the exponential model does use smoothing. It is based on the ambiguity class of the morphological category in question, and it applies whenever no feature can be applied for a given token in context (remember, the rules are being selected, contrary to e.g. Ratnaparkhi's maximum entropy tagging system, which uses all predefined features and only determine their weights). In other words, if no rule can be applied in a given context, the most probable value of the morphological category in question wins: if, for example, the choice for case is genitive and locative, genitive wins.

As an example of a rule, let's present the rule for distinguishing gender of adjectives of the "soft" type, which are in most of their forms typically 4-way gender ambiguous (among all Czech genders, which are four: masculine animate (M) and inanimate (I), feminine (F), and neuter (N)):

Current ambiguity class for gender: FIMN, *and*

Gender of the closest unambiguous noun to the right (+4 positions max.) is unambiguously I (masc. inanim.) *and*

Current ambiguity class for part of speech is A (i.e., unambiguous adjective)

⇨ probability of current value of the gender category: I 98.3%, F 0.8%, M 0.1%, N 0.8%

Such a rule is in fact one the first rules learned by the system, and it corresponds nicely to the grammatical rule of gender (and number and case) agreement between a noun and a modifying adjective, which typically precedes its head noun in Czech. Not all the rules are such straightforwardly interpretable; some of the rules that use ambiguity classes heavily are not transparent at all, sometimes they "simulate" the lack of more detailed context constraint (we have limited the number of simple context constraints to three – see above – but we are aware of the fact that sometimes we would need as many as 10 context constraints should the rules be linguistically plausible). Nevertheless, the learning algorithm uses best what it can, namely, the features from in the pool of possible features (however, the search for features is based on a "greedy" algorithm, and therefore possibly in general not optimal).

It is interesting to note that the tagger does not overtrain even when very low thresholds are set for the number of occurrences of a feature to be still considered for selection. We have found experimentally that even the threshold of 2 (i.e., when we consider features with frequency 2 in the 1.2 mil. token training corpus) improves accuracy on independent test data, and only the threshold of 1 (meaning that we consider every feature found at least once in the training data) slightly overtrains (and also adds a huge number of rules, not surprisingly). The difference between threshold 4 and 2 is not large, though, and it might be well worth investigating whether it makes any difference in MT between Czech and Slovak to use the more detailed rule set, since the "threshold 4" rule set is significantly smaller, making the system over 50% faster.

In any case, the tagger is reasonably fast, with the full set of 11 thousand rules it can tag at a sustainable rate of 200 tokens per second; we have found experimentally that even though the tagger can run in a "full" (wide-beam) Viterbi mode, the improvement in doing so is negligible (undoubtedly due to the fact that we use some unambiguous right-context information in the rules) and thus we use the built-in Viterbi search with beam set to 1, effectively making decisions at every token immediately. This simplification does speed up the tagging process significantly.

Lemmatization immediately follows tagging; it chooses the first lemma with a possible corresponding tag and works with an accuracy close to 98%. This works well for homonymy among lemmas with a different part of speech, but it fails completely for true polysemy resolution (word sense disambiguation for words with the same part of speech). We plan to add "real" word sense disambiguation in the near future, using the methodology described in (Hajič, Hladká 1999).

Domain-related bilingual glossaries

The domain related bilingual glossaries contain pairs of individual words and pairs of multiple-word terms. The glossaries are organized into a hierarchy specified by the user; typically, the glossaries for the most specific domain are applied first. There is one general matching rule for all levels of glossaries – the longest match wins. The multiple-word terms are a sequence of lemmas (not word forms). This structure has several advantages, among others it allows to minimize the size of the dictionary. However, it entails preprocessing of the terminological dictionary by the same tools (morphology and tagger) since typically words in terminological phrases are inflected, too, and usually there is no external indication which word is the headword. (In fact, this means we have to have a morphological analyzer and a tagger available for the target language as well, or at least an approximation of a tagger suitable for noun phrase handling.) On the other hand, this greatly simplifies

the terminological dictionary handling by the end users: in general, it does not require any special involvement on their part – the linguistic experts responsible for terminology simply maintain the terminological dictionaries as if they are to be used by humans. We believe that this approach might prove to be very important part of our system design, since it eliminates the well-known high cost factor for MT dictionary maintenance. In fact, this idea might prove valuable also for MT systems for ordinary pairs of languages, not only those so closely related. Tags are "translated" too (see below in the "main dictionary" section for reasons). Currently, the system handles well *n:n* term translation, uses heuristic guessing for asymmetric cases (*m:n*) and a more sophisticated system for handling the tags correctly in an *n:m* translation case is und development.

General bilingual dictionary

The main bilingual dictionary contains data necessary for the translation of both lemmas and tags. The translation of tags (from the Czech into the Slovak system) is necessary, because both systems use close, but slightly different tag sets. Also, the tags do not always correspond exactly: for example, there are some Slovak nouns which have different gender, or tags with variants which do not exist in the other language. Therefore, a Czech tag is not translated into a single tag, but into a priority-ordered list of tags.

Morphological synthesis of Slovak

The morphological synthesis of Slovak is based on a monolingual dictionary of Slovak, developed by J.Hric (1991-99), covering more than 100,000 lemmas. The coverage of the dictionary is still growing. It aims at a similar coverage of Slovak as has currently been achieved for Czech.

Export of the output to the original translation memory

The export of the output of the system ČESÍLKO into the translation memory of TRADOS Translator's Workbench basically means that the output is cleaned of all irrelevant SGML markup and the whole resulting Slovak sentence is inserted in the relevant location of the original translation memory file. The following example also shows that the marker <CrU> contains the information that the target language sentence was created by an MT system.

Example 3. – A sample of the translation memory containing the results of MT

<RTF Preamble>
...
</RTF Preamble>
...
<TrU>
<CrD>23051999
<CrU>MT!
<Seg L=CS_01>Pomocí výkazů ad-hoc můžete rychle a jednoduše vytvářet rešerše.
<Seg L=SK_01>Pomocí výkazov ad-hoc môžete rýchlo a jednoducho vytvárať rešerše.
</TrU>

3.2 EVALUATION OF RESULTS

The problem how to evaluate results of automatic translation is very difficult. For the evaluation of our system we have exploited the close connection between our system and the TRADOS Translator's Workbench. The method is simple – the human translator receives the translation memory created by our system and translates the text using this memory. The translator is free to make any changes to the text proposed by the translation memory. The target text created by a human translator is then compared with the text created by the mechanical application of translation memory to the source text. TRADOS then evaluates the percentage of matching in the same manner as it normally evaluates the percentage of matching of source text with sentences in translation memory. In the first testing on relatively large texts (tens of thousands words) the translation created by our system achieved about 90% match (as defined by the TRADOS match module) with the results of human translation.

4. CONCLUSION

The role of tagging in a MT system between closely related languages is crucial. We have achieved 90% match (as measured by TRADOS technical matching tools), and subjectively, the translations seem even better than that. The tagger (together with the corresponding morphological analyzer) is used at three different places in the

system: in the preprocessing stage, for the tagging of both the source and target dictionaries, and at runtime, for tagging the input (source) sentence. In all three cases, we also use the results of tagging for lemmatization, due to relatively high degree of lexical homonymy in Czech and Slovak (even though the lexical homonymy, as opposed to morphological homonymy, is lower than, say, in English), since lemmatization amounts basically to major part of speech disambiguation.

The success ratio of the translation achieved by our system justifies the hypothesis that word-for-word translation might be a solution for MT of really closely related languages. The remaining problems to be solved are those of one-to-many or many-to-many translation, where the lack of certain information in glossaries and dictionaries (and our current inability to get it out of it automatically) sometimes causes an unnecessary translation error.

The success of the system ČESÍLKO has encouraged the investigation of the possibility to use the same method for other pairs of Slavic languages, namely for Czech-to-Polish translation. Although these languages are not so similar as Czech and Slovak, we hope that an addition of a simple partial noun phrase parsing might provide similar results. The first results of Czech-to Polish translation are quite encouraging in this respect.

Acknowledgements

This project was supported by the grant of MŠMT ČR No. LN00A063, and partially supported by the grant GAČR 405/03/0914.

References

Hajič, Jan (1999). Word Sense Disambiguation for Czech Texts. In: Proceedings of Text, Speech, Dialogue. Brno 1999. p. 109-114

Hajič, Jan (1998). *Building and Using a Syntactically Annotated Corpus: The Prague Dependency Treebank*. In: Festschrift for Jarmila Panevová, Karolinum Press, Charles University, Prague. pp. 106—132.

Hajič, Jan and Hladká, Barbora (1998). *Tagging Inflective Languages. Prediction of Morphological Categories for a Rich, Structured Tagset*. ACL-Coling'98, Montreal, Canada, August 1998, pp. 483-490.

Hajič, Jan, Hric, Jan and Kuboň, Vladislav (2000): Machine Translation of Very Close Languages. In: Proceedings of the 6th Applied Natural Language Processing Conference, Seattle, Washington, USA, April 2000, pp. 7-12

Oliva, Karel (1989). *A Parser for Czech Implemented in Systems Q*; Explizite Beschreibung der Sprache und automatische Textbearbeitung XVI, MFF UK Prague