

# Scaling up a hybrid MT system: From low to full resources

**Vincent Vandeghinste**

K.U.Leuven

*This article describes a hybrid approach to machine translation (MT) that is inspired by the rule-based, statistical, example-based, and other hybrid machine translation approaches currently used or described in academic literature. It describes how the approach was implemented for language pairs using only limited monolingual resources and hardly any parallel resources (the METIS-II system), and how it is currently implemented with rich resources on both the source and target side as well as rich parallel data (the PaCo-MT system). We aim to illustrate that a similar paradigm can be used, irrespectively of the resources available, but of course with an impact on translation quality.*

## 1. Introduction

There are myriad approaches to machine translation, but none have shown acceptable levels of translation quality from an end-user's perspective. MT systems that exist today reach at best a level of translation quality that might speed up the work of a human translator. The most widespread use of MT systems are *online translation services*, which are available through many Web sites and provide a gist translation of the source language text. MT systems in limited domains are occasionally sufficiently accurate to be useful for *real* translation tasks.

In *rule-based machine translation*, the development of a new language pair, especially with so-called 'smaller' languages, is rather rare due to high costs and long development times. In *statistical machine translation*, these expenses depend on the availability of parallel corpora containing aligned sentences in both the source and target language.

In order to develop MT systems for new language pairs more efficiently, we developed a new methodology which allows reuse of existing tools and corpora for both the source and target language. Since deep syntactic parsers and parallel corpora are unavailable for many language pairs, we implemented this new methodology with low resource source and target languages in the METIS-II system (Carl et al., 2008), limiting ourselves to using only the kind of very basic resources that are available for many languages or that can be built relatively easily. When more tools and resources are available, we can still apply similar methodology. We are now scaling up to more sophisticated tools and large parallel corpora: Parse and Corpus-based Machine Translation (PaCo-MT).

This article will first describe the used approach in general (Section 2), then the METIS-II approach using low resources (Section 3), and then the PaCo-MT approach using full resources (Section 4).

## 2. A Hybrid Approach toward MT reusing existing resources

This section describes the common ideas behind both the METIS-II system, for which the implementation of a Dutch-English translation system is described in Vandeghinste (2008), and the PaCo-MT system, which is currently implemented and which is partially described in Vandeghinste (2007; 2009). Figure 1 shows where both the METIS and PaCo approach can be situated on the Vauquois triangle (1968) and this paper aims to illustrate how to climb the Vauquois triangle within the presented approach.

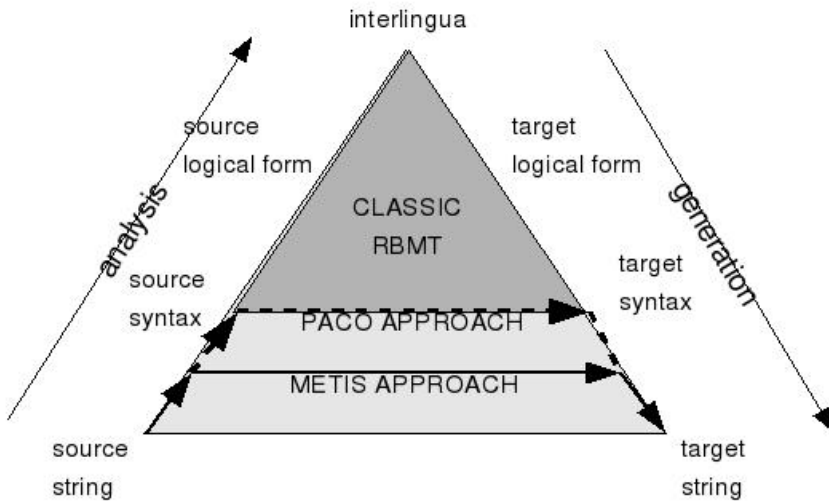


Figure 1. METIS and PaCo situated on the Vauquois triangle.

We describe how both approaches borrow from different MT paradigms, including rule-based MT, statistical MT, and example-based MT.

### 2.1. Rule-based Machine Translation (RBMT)

RBMT is characterised by use of linguistic rules in translation. It consists of source language syntactic and semantic analysis, a series of structural conversions, and target language generation. There are two approaches toward RBMT: the interlingua approach and the transfer approach.

In the *interlingua* approach, the source language analysis leads to an interlingual representation of the sentence. This is an abstract (in principle *language-independent*) representation from which a target

language string is generated. For the interlingual treatment, abstraction is applied by the monolingual modules so that the content or function of all lexical items is recoded in terms of semantic universals. An example interlingua system is described by Rosetta (1994). Some disadvantages of interlingua systems are described in Van Eynde (1993).

In *transfer* systems, the source sentence is analysed, most often by a rule-based parser; and transfer rules convert the source sentence structure into the target sentence structure, from which the target sentence is generated by a language generator, using target language generation rules.

Although in academia most current approaches are no longer rule-based, many of the industrial MT engines still are. For instance, most of the translation pairs available at free online MT engines, including Babelfish and Microsoft, are transfer systems.

## 2.2. Statistical Machine Translation (SMT)

SMT systems implement a theory of probability distribution and probability estimation. They learn a *translation model* from a parallel corpus, which contains aligned source and target language information, and a *language model* from a target language corpus. The best translation is searched for by maximising the probability according to these models. Using statistics in MT has had a major impact on translation accuracy (Ney, 2005).

One *advantage* of statistics and probability distributions is that they offer a formalism for expressing and combining scores for each translation hypothesis: The probabilities can be used as scores, and it is obvious how to combine scores. Nuances and shades of difference can best be expressed in values between 0 and 1. There are ways to estimate these probabilities without human intervention (Ney, 2005).

There are also *disadvantages* to SMT. One major disadvantage is the need for a large parallel corpus. This is often unavailable, and when available is often limited to specific domains. Another disadvantage is that SMT systems are often like a *black-box*: it is very hard to improve results after a basic system has been built (except by enlarging the corpora). Due to the models that are used, SMT systems are known to have, among other things, problems with capturing information about long-distance dependencies, and hence produce incorrect translations in such cases. SMT also seems to suffer from *ceiling* effects in performance (Lønning et al., 2004). To break through these ceilings, we see increasing use of linguistic features within the SMT paradigm.

Another approach to improve SMT is to move from the word level to the *phrase* level, using a set of heuristics to determine phrase boundaries (Koehn, et al., 2003). The term *phrase* is not used in the linguistic sense, but denotes *any sequence of words*. This seems to be the most used approach in current SMT systems.

### 2.3. Example-based Machine Translation (EBMT)

EBMT can be located somewhere between RBMT and SMT, as many EBMT approaches integrate both rule-based and data-driven techniques (Carl and Way, 2003).

EBMT is sometimes confused with the related technique of *translation memory* (TM). Although both have the idea of reusing already existing translations, they differ in the sense that a TM is an interactive tool for the *human* translator, whereas EBMT is an *automatic* translation technique (Somers, 2003).

The idea for EBMT dates back to Nagao (1984). He identified the three main components of EBMT as matching fragments of text against a database of real examples, identifying the corresponding translation fragments, and recombining these to give the target text.

An EBMT system is developed on the basis of a parallel, aligned corpus. These corpora, however, are often only available for limited domains and a limited set of languages, but for general translation purposes they are not as easy to acquire. In this respect, EBMT suffers from the same drawback as SMT. A related issue is the required size of the database of translated text fragments. Although Mima et al. (1998) reported that the MT improvement was more or less linear with the number of examples, it is assumed (Somers, 2003) that there is some limit after which adding more examples no longer improves (and even worsens) the quality of the output, as examples might contradict each other.

Other problematic issues in EBMT are how examples are stored and which information is stored with them; how source language strings are matched with the corpus; extraction of appropriate fragments from the translation; and recombination of these fragments into a grammatical target language output (Somers, 2003).

### 2.4. A hybrid approach

The general approach behind the METIS-II and PaCo-MT systems draws on these three paradigms and seeks to combine their strengths and avoid their weaknesses. Figure 2 presents the general architecture of our approach.

The first processing step consists of *source language analysis*, which results in one or more parse trees representing the syntactic structure of the sentence. This parse tree can be very shallow or it can be a full parse tree, depending on the tools and resources available for the source language. It can be a phrase structure tree or a dependency tree, or it can simply contain chunked data, with a depth of only 1. The use of a (full) parser for linguistic analysis is common in RBMT systems, as well as performing a source language analysis independently of the target language.

The second processing step consists of *converting the source language tree into one or more target language bags of bags*. A bag is an

unordered list of words or phrases, so a bag of bags is a tree-like structure but the daughters of each node in the bag of bags are unordered. All terminal nodes in the source language analysis tree can be converted to target language equivalents by looking up the node's *lemma* or *word form* and *part-of-speech* in the dictionary.

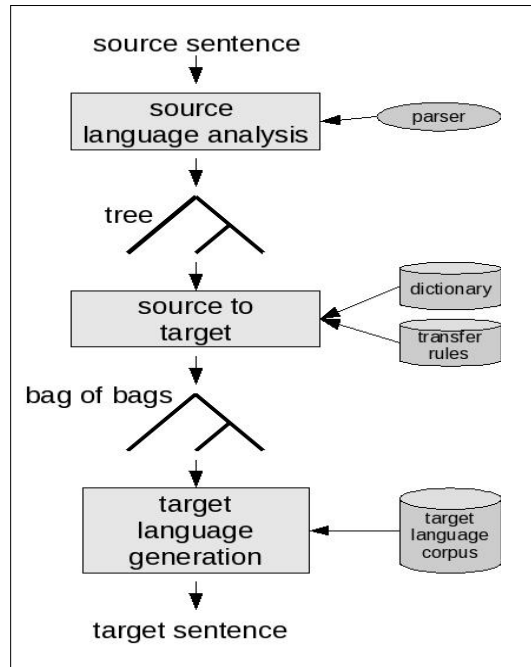


Figure 2. General Architecture

Using dictionaries consisting of lemmas or stems has the advantage of greatly improving its coverage over using a dictionary containing all surface word forms as they appear in text. Terminal nodes (lemma/word form + part-of-speech) which are not in the dictionary are left untranslated by default. As shown in figure 3, apart from single words, the dictionary can also contain more complex, structured items, both on the source and the target language side, covering more complex cases than simple word-by-word translations. This means that:

- the part-of-speech tag sets for source and target language need not be the same, as the tags are *translated* via dictionary look-up;
- the syntactic structure for source and target language can be different, as the structure is also *translated* via dictionary look-up;
- non-terminal nodes can be found in the dictionary, and can lead to translations which are fragments of syntactic trees in the target language. For these nodes, the order of the daughters in the target language can already be fixed.

There are often structural changes between source and target language which are not word-specific but more general and are thus not covered in the dictionary. Therefore, we introduce *transfer rules* which model these structural differences, and bring the bag of bags closer to the desired target language structure.



Figure 3. Examples of dictionary entries

An example transfer rule is, when translating from Dutch to English, *Verb Group Treatment* (Vandeghinste, 2008): In Dutch, the auxiliary and past participle can be separated, but in English they tend to stay together, except in certain cases, which we ignore for now. We detect if, within the same clause, we find an auxiliary and a past participle. If so, we put them under the same mother node, so they stay close together in target language modelling, as words belonging to the same mother node will not be separated by the target language model. This is illustrated in figure 4.

The use of a dictionary combined with a set of transfer rules is similar to what is done in transfer-based RBMT. The difference with our approach is that, depending on the available parallel resources, we can both use manual dictionary entries and automatically derived entries, each with a weight representing its confidence.

The final step in the core MT engine is the *generation of target language strings* from the bags of bags, using a target language corpus. Therefore, the target language corpus needs to be *pre-processed* similarly to how the source language is analysed. The daughters of each of the bags and sub-bags are looked up in the target language corpus in order to retrieve the frequency of occurrence for each permutation of the order of the daughters, and to determine the most probable target language string. For instance, if we have the target language noun-phrase bag containing the words *big*, *the*, *black* and *dog*, what is the most likely permutation of these four words? Two permutations yield a grammatical surface string: *the big black dog* and *the black big dog*, but the former is most likely to appear in real English text. Using a target language model to order the bags of bags allows for a very light transfer model, as it defers a great part of the reordering modelisation onto the target language model.

The target language corpus is also used to perform *lexical selection* between several translation alternatives by looking at which translation alternatives are most likely to co-occur in the target language corpus. For instance, the Dutch word *zwart* can have the English translations *black* and *gloomy*. When we want to translate the phrase *de zwarte hond*, the target

language corpus tells us that the word *black* is far more likely to co-occur with *dog* than the word *gloomy*.

This is somewhat similar to what is done in traditional EBMT, albeit EBMT tries to find these nodes in a parallel corpus, whereas we try to find them in a pre-processed target language corpus. The use of probabilities and weights at every step in the translation process is borrowed from statistical NLP and SMT.

### 3. Using only low resources

In this section we first give a system description of METIS-II for Dutch to English, and end with a description of how the evaluation of this system was performed at several stages in its development. With low resources we essentially mean that neither full parsers nor parallel corpora were used.

#### 3.1. System description

In the METIS-II project (Carl et al, 2008; Vandeghinste et al., 2008) this approach was tested using only limited resources on different language pairs: Greek to English, German to English, Spanish to English and Dutch to English. We briefly describe the approach, which is used for the latter language pair (Vandeghinste, 2008). figure 4 presents an example sentence.

*Source language analysis* is performed using a tokeniser, a part-of-speech tagger (Brants, 2000), a lemmatiser, a shallow parser (NP and PP detection, head detection) and a clause detector (relative phrases and subordinate clauses). The system does *not* use a full syntactic parser.

To *translate nodes* in the shallow parse tree, a manually compiled *dictionary* (gathered from several internet sources plus further manual editing) is used together with a limited set (<20) of manually defined transfer rules. *Part-of-Speech Tag mapping rules* which convert the source language tags (Van Eynde, 2005) into target language tags<sup>1</sup> are used to *translate* the non-lemma features of the source language tags (*singular* vs. *plural*, *present* vs. *past*, etc) into features of the target language tag (for instance, the Dutch tag the Dutch tag WW(pv,tgw,ev) is converted into VVB).

As described in the previous section, every node is looked up in the dictionary, and the structure of the bag of bags is converted by the transfer rules in a structure more similar to English sentence structure. These rules can concern word and chunk ordering information. For instance, as shown in figure 4, there is a rule in English (see also Huddleston & Pullum, 2002) that puts auxiliaries and past participles together under one node except in the case of inversion, frequency adverbs and some other adjuncts. In Dutch, however, they are separated. Other rules concern mappings of tense and aspect.

Note that *not* using a parallel corpus is one of the key properties of METIS-II, as parallel corpora are not available, not large enough or too domain specific for most language pairs. It is what makes METIS-II different from most data-driven approaches of MT.

From all the previous processing steps, we have a ranked set of bags of bags each representing a translation alternative. They are ranked according to their weight, which is a combination of the weight generated by the different statistical source language analysis modules. These weights estimate the probability of an analysis, and the lower the weight, the less trustworthy the analysis.

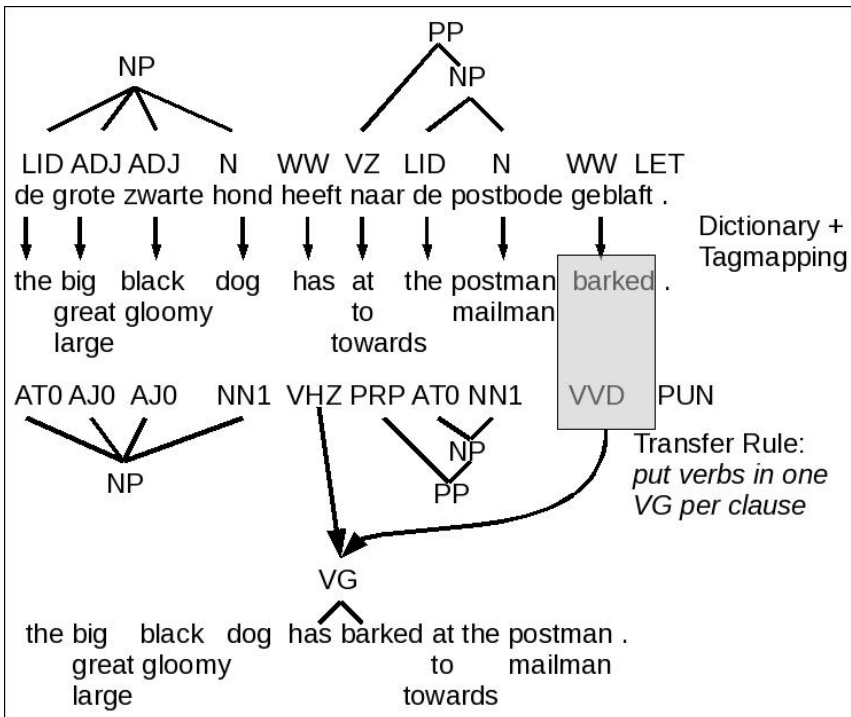


Figure 4. Example of the conversion from source to target language in the METIS-II engine.

For each of these bags of bags, the order of the daughters of each of the bags and daughter-bags needs to be determined, so the bags of bags are converted into conventional tree representations of the target language sentence, each with a weight. This is done by looking up each bag (and daughter-bag) in the pre-processed *target language corpus*.

The target language corpus in this case is the British National Corpus (BNC), a balanced collection of samples of written and spoken language from a wide range of sources, which is already tagged. Pre-processing consisted of lemmatisation (done by the *Reversible Lemmatiser* (Carl et al., 2005)), chunking, and clause detection.



Matching a bag with the BNC results in a number of permutations of the bag elements each receiving a matching score, because they match with corpus chunks. The closer they match with what is found in the corpus, the higher they score. Since not all elements of each bag are leaf nodes, the lemmas of the heads of the translation candidates are used to perform matching. A bag element is matching a corpus element when the lemma (or the lemma of the head daughter) matches. The accuracy of matching ( $a_m$ ) is calculated according to this formula:

$$a_m = m / (n + p^2)$$

where  $m$  is the number of matching bag elements,  $n$  is the total number of bag elements, and  $p$  is the number of elements in the corpus chunk which are not in the bag, and which cannot be replaced by one of the elements in the bag. We take the square of  $p$  to make it a more important factor. Experiments showed that this improved translation accuracy.

This matching process allows us to retrieve word order information from the target language corpus, by using word order information from the matching corpus fragment. In addition, this process performs lexical selection because not every bag alternative matches with the same accuracy, leading to translation candidate selection when a certain combination of words occurs in the corpus.

Apart from the matching accuracy we also take into account the relative frequency of the corpus chunk with respect to the total frequency of all corpus chunks in which the same or a higher number of elements match ( $m$ ).

Permutations which do not match with any corpus fragments are no longer considered, allowing us to move from a *bag* representation to several conventional tree representations. For a more detailed description of this system and some examples, we refer to Vandeghinste (2008).

After lexical selection and word order have been determined, a final step remains: the target language tree contains lemmas and part-of-speech tags, and these need to be converted into the appropriate tokens. For this purpose we again use the *Reversible Lemmatiser* (Carl et al., 2005) in reverse mode.

### 3.2. Evaluation

When building a prototype, it is of utmost importance to test and evaluate the prototype at different stages in its implementation. In Vandeghinste et al. (2005), we described an experiment on the first version of the prototype, in which we validate the general idea behind the approach, viz. noun phrase translation. For this evaluation, we translated 685 NPs, which resulted in a number of translation alternatives, ranked by their weight. Humans judged whether the first translation alternative was correct (57.7%), or amongst any of the translation alternatives (13.6%). This implies that, by only changing

the weighing mechanism, we were able to get a maximum of 71% correct NPs. The moderate results can be explained by the limited coverage of the lexicon (80% word coverage) and bugs in this early version (no output for 12%).

Another partial evaluation, described by Dirix et al. (2006) was a source language independent evaluation. For this evaluation, a set of 150 bags of words was generated, having chunk structures derived from original Dutch, Greek or Spanish sentences. All words were manually translated into English. An average BLEU score (Papinemi et al., 2002) of 21.17% was reached and the error analysis led to the observation that the dictionary was clearly not sufficient to bridge the gap between source and target language. This led to the introduction of a transfer mechanism in the next version of the system.

In Vandeghinste et al. (2007) we tested the effect of adding a limited set of transfer rules, leading to a clear improvement in both BLEU and NIST scores (Dodington, 2002). The evaluation of this final METIS using automatic MT metrics showed that the BLEU score was not that different from a standard unoptimised statistical MT system trained on the Europarl corpus (Zwarts & Dras, 2007), as shown in table 1.

Table 1. Evaluation Results

	<b>METIS-II</b>	<b>Zwarts&amp;Dras</b>
BLEU	19.79%	20.70%
NIST	6.06	
TER	59.33%	

In other words, the performance of the METIS-II system, without using parallel data other than a dictionary, reaches a performance level almost similar to that of an (unoptimised) SMT system, but without using a parallel corpus.

#### 4. Scaling up to full resources

In the PaCo-MT system, we scale up this approach, using far more tools and resources. We implement the translation pairs Dutch-English and Dutch-French in both directions. In this section, we describe the approach for translation from Dutch to English, and compare it with the low resources approach from the previous section.

Instead of a shallow source language analysis, we now use *full parsers*, giving us a detailed analysis of the source language sentence. For Dutch, we use the Alpino parser (Van Noord, 2006), resulting in a

dependency tree representation combined with a phrase structure tree of the source sentence, as shown in figure 5. Not only do we know the NPs and PPs, we also know for instance what the *subject* or *direct object* of a sentence or any of its clauses is.

Instead of using only a hand-made dictionary, we derive dictionary entries from publicly available parallel corpora. As described in Vandeghinste (2007) we parse the Dutch side of Europarl (Koehn, 2005) (and other parallel data) with Alpino and the aligned English side with the Stanford parser (Klein & Manning, 2003). This is a stochastic parser, trained on the Penn treebank<sup>2</sup> and yielding a phrase structure tree and a dependency tree as output.

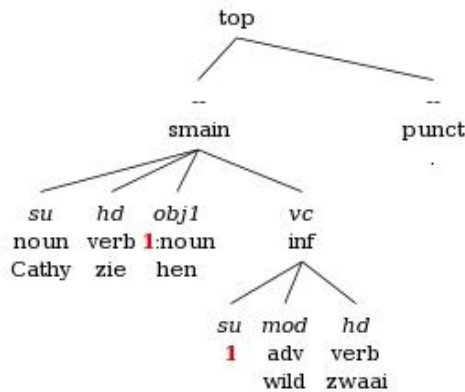


Figure 5. Alpino parse tree for the sentence “Cathy zag hen wild zwaaien.” (Cathy saw them wave wildly.)<sup>3</sup>

Parsing both sides results in a parallel Treebank, in which all sentences are aligned. We also align at the word level, using GIZA++ (Och & Ney, 2003), a tool designed for SMT. Word and sentence alignments are put in the dictionary, together with their alignment frequency in order to obtain a dictionary containing full sentences and single words, each with a weight.

In addition to this, we align at the sub-sentential level, meaning that we align non-terminal non-root nodes in both source and target language trees, so that for instance subject noun phrases are aligned. This is similar to what is done by Hearne (2005) in what is called Data-oriented Translation (DOT), but she applies it on a small parallel corpus only.

We put the resulting alignments in our *dictionary*, together with weights based on the alignment and parser confidence and the frequency of occurrence, leading to a dictionary that contains all sorts of entries: single words, phrases and constituents, clauses, and full sentences. Note that deriving dictionary entries from a large parallel corpus is one of the major differences (together with the use of full linguistic parsers) between this approach and the low resources approach used in the METIS-II project.

Returning to the translation processing chain of PaCo-MT, we try to match every node of the input parse tree with the source language side of the dictionary entries, retrieving, when possible, the full sentence (and in this way functioning like a translation memory). If the full sentence cannot be retrieved, we seek for lower level matches, recursively descending down the input parse tree, resulting in target language fragments that need to be recombined into one target language sentence, much like in EBMT. This dictionary matching process leads to a number of bags of bags, each representing an alternative translation hypothesis for the target sentence.

The structure of these bags of bags can be altered by the automatically derived *transfer rules*. When nodes in the parallel treebank are aligned, we do not only extract dictionary entries from these alignments, but also transfer rules, making abstraction of the concrete words and tokens which align, and only taking into account categories (constituents) and relations (dependency labels). Using the relative frequencies of occurrence of these alignments gives us weighting information, which allows us to prefer one transfer rule over the others.

Using automatically derived transfer rules is another difference with the low resource approach, in which we used manually edited transfer rules. Of course, transfer rules in PaCo-MT can also be manually edited.

The final step, before outputting a target language sentence, consists of generating a string from the bag of bags. For each unresolved bag (and recursively for the whole tree), we try to find the most probable order and combination for the daughter nodes. All permutations of the daughter nodes are looked up in the pre-processed target language corpus in order to retrieve their frequency of occurrence. This is done at different levels of abstraction, beginning with the most concrete level in which we try to find the exact same words in the exact same functions. When we do not find these, we abstract over the words and so on, until we find some information allowing us to prefer one ordering over the others (for more details we refer to Vandeghinste, 2009). Our approach is somewhat similar to the *feature templates* approach used by Velldal (2007), although we only derive *context free* information, by only linking the node with its immediate daughters, whereas Velldal's extracted information expands over several levels in the tree.

Apart from the order of the daughter nodes we need to select which of the translation alternatives will be used in the output. The alternatives that co-occur within linguistically motivated corpus fragments in the target language corpus are identified, and the resulting relative frequencies are used to estimate the weights of the translation alternatives.

As the PaCo-MT system is currently under development no results for the full MT processing chain are available yet, but first results for the target language generation, both for Dutch and for English are promising. As described in Vandeghinste (2009), we set up an experiment in which we compare our tree-based target language modeling with a standard trigram model. We performed a source language independent evaluation, in which

we used the parse trees of the test sentences as input, but with all surface order information removed. It is then up to the target language generator to generate a surface sentence from this bag of bags. Figure 6 compares tree-based language modeling with a standard backoff trigram model with a branch and bound approach.<sup>4</sup> Results were consistent for a set of different MT metrics (WER, NIST, TER (Snover et al., 2006)).

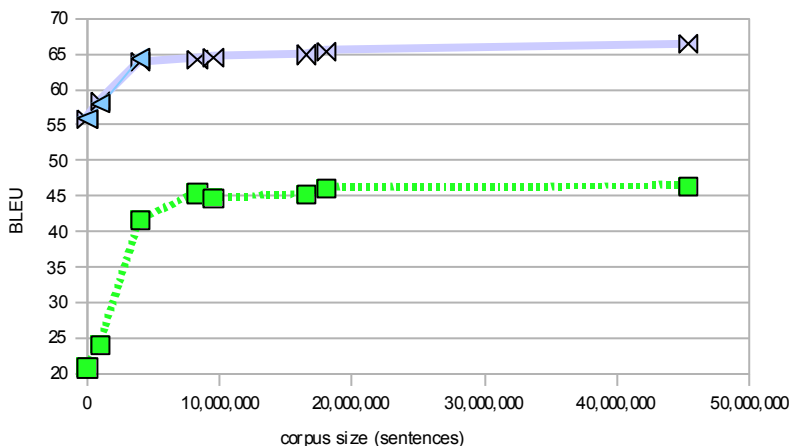


Figure 6. BLEU scores for target language generation for Dutch. Comparing tree-based language modeling (continuous line) with trigram language modeling (dotted line).

## 5. Conclusions

In this paper we described a hybrid approach towards machine translation, seeking to combine the strengths and avoid the weaknesses of the classic approaches towards MT.

The difficulty in developing an RBMT system resides in the huge cost and effort of rule design; especially for the transfer rules, which are language pair dependent. This becomes unfeasible when the commercial potential of the language pair is low. However, even when language pairs have a high commercial potential, and rules have been designed and improved for more than 30 years, the results are often disappointing. Therefore, most commercial RBMT systems are starting to use corpora in order to give weights to their rule-sets and to allow for rule ranking when more than one rule applies.

The difficulty in developing an SMT system (and an EBMT system) resides in the need for large parallel and monolingual corpora to feed the translation and target language model. In SMT systems, the use of *n-grams* with a low *n* leads to weak models in the case of long distance

dependencies and other long distance phenomena. For languages such as Dutch the subject and verb can be very far apart in sub-ordinate clauses, which is problematic for subject-verb agreement in SMT systems. Apart from using even larger corpora, there is a tendency in SMT to extend the models used to include more and more syntactic features.

Consequently, the RBMT and SMT worlds are moving closer to each other. The linguists of the RBMT world are starting to use statistics, while the engineers of the SMT world are starting to use linguistic features.

The hybrid approach described here currently has two instantiations. The first one is the METIS-II system, a system designed to minimise the use of tools and resources, especially language pair specific resources, by avoiding the use of parallel corpora. The only parallel data required for this system is its dictionary. Other tools used in METIS-II are monolingual analysis tools, which are available for many more languages or which can be easily built or trained. While translation quality of this system is not very good, it is not much worse than translation quality of an SMT system, which does require a parallel corpus, but does not require any language specific tools.

The second system is the PaCo-MT system. Although it is hard to draw firm conclusions about a system which is still in an early development stage, we can focus on the design of the system and why certain design choices have been made in order to overcome weaknesses of other approaches toward MT.

In the PaCo-MT project we use a rule-based architecture, but avoid the high development time for the rules by automatically deriving them from parallel treebanks. The parsers we use can be rule-based (like Alpino) with a stochastic component for disambiguation and speed, or they can be purely stochastic (like the Stanford parser), trained on a linguistically annotated treebank like the Penn treebank. We use already existing parsers so that we do not need to develop monolingual grammars. From the alignment between fragments we want to derive language pair specific translation grammars. For alignment we primarily look at techniques coming from the SMT world, but these might be improved using some linguistic features.

## **Acknowledgements**

This research is made possible by the STEVIN-programme of the Dutch Language Union, Project PaCo-MT (STE-07007), which is sponsored by the Flemish and Dutch Governments, and by the SBO-programme of the Flemish IWT, Project AMASS++, Project Nr. 060051.

The PaCo-MT system is built in cooperation with the University of Groningen, which provides us with the treebanks and the alignments, and

with the translation company Oneliner, which provides us with translation memories and test sets.

## Bibliography

- Brants, T. (2000). TrN - a statistical part-of-speech tagger. In Proceedings of 6th Applied Natural Language Processing Conference (ANLP) (pp.224-231). Seattle, Washington.
- Carl, M., Melero, B., Badia, T., Vandeghinste, V., Dirix, P., Schuurman, I., Markantonatou, S., Sofianopoulos, S., Vassiliou, M. & Yannoutsou, O. (2008). METIS-II: Low Resources Machine Translation. *Machine Translation* 22 (1), 67-99.
- Carl, M. & Way, A. (2003). *Recent advances in Example-based Translation*. Mechelen: Kluwer Academic Publishers.
- Carl, M., Schmidt, P. & Schütz, J. (2005). Reversible Template-based Shake & Bake Generation. In M. Carl & A. Way (Eds) Proceedings of the 2nd workshop on Example-based Machine Translation at the Tenth MT Summit (pp.17-25). Phuket, Thailand.
- Dirix, P., Vandeghinste, V. & Schuurman, I. (2006). A new hybrid approach enabling MT for languages with little resources. In K.S. de Rijke, R. Scha, & R. van Son (Eds.), *Proceedings of the Sixteenth Computational Linguistics in the Netherlands (CLIN)* (pp.117-132). Universiteit van Amsterdam. Amsterdam, The Netherlands.
- Doddington, G. (2002). Automatic Evaluation of Machine Translation Quality using N-gram Co-occurrence statistics. In Proceedings of HLT-2 (pp.138-145). San Diego, California.
- Hearne, M. (2005). Data-Oriented Models of Parsing and Translation. PhD Thesis. Dublin City University.
- Huddleston, R. & Pullum, G. (2002). *The Cambridge Grammar of the English Language*. Cambridge: Cambridge University Press.
- Klein, D. & Manning, C. (2003). Fast Exact Inference with a Factored Model for Natural Language Parsing. In *Advances in Neural Information Processing Systems 15 (NIPS 2002)*, Cambridge, MA: MIT Press, 3-10.
- Koehn, P., Och, F. & Marcu, D. (2003). Statistical Phrase-Based Translation. In Proceedings of HLT-2003 (pp.127-133). Edmonton, Canada.
- Koehn, P. (2005). Europarl: a parallel corpus for statistical machine translation. In Proceedings of the Tenth Machine Translation Summit. (pp.79-87). Phuket, Thailand.
- Lønning, J. T., Oepen, S., Beermann, D., Hellan, L., Carroll, J., Dyvik, H., Flickinger, D., Johannsen, J.B., Meurer, P., Nordgård, T., Rosén, V. & Velldal, E. (2004). LOGON. A Norwegian MT effort. In Proceedings of the Workshop in Recent Advances in Scandinavian Machine Translation. Uppsala, Sweden.
- Mima, H., Iida, H. & Furuse, O. (1998). Simultaneous Interpretation Utilizing Example-based Incremental Transfer. In Proceedings of COLING/ACL-1998 (pp.855-861). Montreal, Canada.
- Nagao, M. (1984). A Framework of a Mechanical Translation between Japanese and English by Analogy Principle. In A. Elithorn, A. and R. Banerji, R. (Eds) *Artificial and Human Intelligence* (pp.173-180). Amsterdam. The Netherlands.
- Ney, H. (2005). One Decade of Statistical Machine Translation: 1996-2005. In Proceedings of MT Summit X. (pp.i-12-17). Phuket, Thailand.
- Och, F. & Hermann, N. (2003). A Systematic Comparison of Various Statistical Alignment Models, *Computational Linguistics* 29(1). 19-51.
- Papinemi, K., Roukos, S., Ward, T., Henderson, J., & Reeder, F. (2002). BLEU: a method for automatic evaluation of Machine Translation. In Proceedings of ACL2002 (pp.311-318). Philadelphia, USA.
- Rosetta, M.T. (1994). *Compositional Translation*. Kluwer International Series in Engineering and Computer Science, Volume 273, Dordrecht: Kluwer.
- Snover, M., Dorr, B., Schwartz, R., Micciulla, L., & Makhoul, J. (2006). A study of translation edit rate with targeted human annotation. In Proceedings of AMTA-7. (pp.223-231). Cambridge, Mass..
- Somers, H. (2003). An overview of EBMT. In M. Carl and A. Way (Eds) *Recent advances in Example-Based Machine Translation*. Mechelen: Kluwer Academic Publishers.
- Vandeghinste, V., Dirix, P., Schuurman, I., Markantonatou, S., Sofianopoulos, S., Vassiliou, M., Yannoutsou, O., Badia, T., Melero, M., Boleda, G., Carl, M. & Schmidt, P. (2008).

- Evaluation of a Machine Translation System for Low Resource Languages: METIS-II. In *Proceedings of LREC-2008*. Marakesh, Morocco.
- Vandeghinste, V., Dirix, P. & Schuurman, I. (2005). Example-based Translation without Parallel Corpora: First experiments on a prototype. Proceedings of the Workshop on Example-Based Machine Translation. Hosted by MT SUMMIT X. Phuket, Thailand.
- Vandeghinste, V., Dirix, P. & Schuurman, I. (2005). The effect of a feq rules on a data-driven MT system. In F. Van Eynde, I. Schuurman, & V. Vandeghinste (Eds.), *New Approaches to Machine Translation*. Proceedings of the METIS-II Workshop (pp.27-34). Centre for Computational Linguistics. KULeuven. Leuven, Belgium..
- Vandeghinste, V.(2007). Removing the distinction between a translation memory, a bilingual dictionary and a parallel corpus. In *Proceedings of Translation and the Computer* (29). London, UK.
- Vandeghinste, V. (2008). A Hybrid Modular Machine Translation System. PhD Thesis. LOT. Utrecht. The Netherlands.
- Vandeghinste, V.(2009). Tree-based Target Language Modeling. In Proceedings of the 13th Annual Meeting of the European Association for Machine Translation (pp.125-159).Barcelona.
- Van Eynde, F. (1993). Machine Translation and linguistic motivation. In F. Van Eynde (Ed.), *Linguistic Issues in Machine Translation*.. Communication in Artificial Intelligence Series. London: Pinter Publishers.
- Van Eynde, F.(2005). *Part of Speech Tagging en Lemmatisering van het D-Coi Corpus* . Annotation Protocol. Centrum voor Computerlinguïstiek. Leuven. Belgium.
- Van Noord, G. (2006). At Last Parsing Is Now Operational. In *Proceedings of TALN06* (pp.20-42). Leuven,.
- Vauquois, B.(1968). Structures profondes et traduction automatique. Le système du CETA. In *Revue Roumaine de Linguistique*, 13(2), 105-130.
- Velldal, E. (2007). Empirical Realisation Ranking. PhD Thesis. University of Oslo. Norway
- Zwarts, S. & Dras, M. (2007). Syntax-based Word Reordering in Phrase-Based Statistical Machine Translation: Why Does it Work? In B.Maegaard (Ed.), *Proceedings of the Eleventh Machine Translation Summit* (pp.559-566).Copenhagen, Denmark.

---

<sup>1</sup> CLAWS 5 tag set at <http://www.comp.lancs.ac.uk/ucrel/claws5tags.html>

<sup>2</sup> <http://www.cis.upenn.edu/~treebank/>

<sup>3</sup> Taken from [http://www.let.rug.nl/~van Noord/bin/alpinods\\_act2svg?lot\\_test\\_suite1/1](http://www.let.rug.nl/~van Noord/bin/alpinods_act2svg?lot_test_suite1/1)

<sup>4</sup> Note that we achieved higher BLEU scores than in Vandeghinste (2009) because of a bug that was found, which led to reduced quality.