# Combining MT Systems Effectively*

**Petr Homola**
homola@ufal.mff.cuni.cz
Institute of Formal and Applied Linguistics
Charles University, Prague, Czech Republic

**Jernej Vičič**
jernej.vicic@upr.si
University of Primorska
Koper, Slovenia

**Abstract**

The paper describes a sophisticated method of combining two MT systems to obtain a new translation pair. Instead of a simple pipe, we use a complex data structure to pass the data from the first MT system to the second one. Evaluation results are reported for the language triplet Czech-Slovenian-Slovak.

## 1. Introduction

Machine translation (MT) is very important in multilingual societies such as the European Union with its more than twenty official languages and a plenty of regional idioms. It is obvious that the development of an MT system is extremely costly in terms of time and manpower so every method that simplifies the creation of a new translation pair can save valuable resources.

The rule-based shallow-transfer approach to MT has a long tradition and has been successfully used for automatic translation between closely related languages; the most notable such system is Apertium (Corbi-Bellot et al. 2005). Shallow-transfer systems usually use a morphological disambiguator before the transfer phase which typically works deterministically. This is obviously a huge restriction, especially for lexical transfer, since in most language pairs, many words have more translations depending on the syntactic and/or semantic context. The parser and structural transfer also produce ambiguous output relatively often. Even if a shallow-transfer MT system would be designed for a narrow domain which significantly simplifies the lexicon and reduces lexical ambiguity in translated texts, a crucial problem is the morphological disambiguation which is mostly performed by a statistical tagger. Even if we had enough morphologically annotated data to train the tagger, the state-of-the-art taggers have a high error rate. Since the morphological disambiguation is the first module in the core of the system, errors are introduced into the processed data at the very beginning of the translation process and it becomes impossible for the subsequent modules to work properly.

We have implemented an MT framework that uses rule-based partial parser and shallow transfer. The tagger was omitted and finally, we have added a ranker based on a trigram language model as the last module in the translation pipeline. Our experiment with translation from Czech into Slovenian and subsequently from Slovenian into Slovak shows that a sophisticated combination of two MT systems can be used to obtain a new translation pair with reasonable quality.

The paper is organized as follows: Section 2 contains a brief description of related research. In Section 3, we describe a modification of the commonly used shallow-transfer approach that leads to higher translation quality. In Section 4, we explain the implementation of the transfer. Section 5 describes the statical ranking module. In Section 6, we evaluate our MT experiments and finally, we discuss the novel method in Section 7 and conclude in Section 8.

## 2. Recent research

### Shallow-transfer MT

Translation between closely related languages has been explored in detail by (Dyvik 1995) for Scandinavian languages. The shallow-transfer approach to rule-based MT has been first proposed in (Hajič, Hric, and Kuboň 2000) for translation from Czech into Slovak. As there are practically no syntactic nor semantic differences between the two languages, their system uses a direct lemma-to-lemma lexical transfer with a one-to-one dictionary. Later, the system was extended to the language pair Czech-Polish (Debowski, Hajič, and Kuboň 2002) and finally, a partial parser has been added to the system's architecture for the language pair Czech-Lithuanian (Hajič, Homola, and Kuboň 2003).

Czech is a language with rich inflection, i.e., a word usually has many different endings that express various morphological categories. The morphological analyzer assigns a set of lemmas and tags to each word. As it was necessary to have only one tag for each word in the transfer phase, a statistical tagger was used with an accuracy of approx. 94% (Hajič and Kuboň 2003).

The bilingual glossaries contained lemmas of the source language and their counterparts in the target language. It is an inherent problem of dictionaries that a source lemma often corresponds to several lemmas in the target language and the correct translation depends on the semantic context, the style of the text etc. Even for very closely related languages

*The presented research has been supported by the grant No. 1ET100300517 of the GAAV ČR.

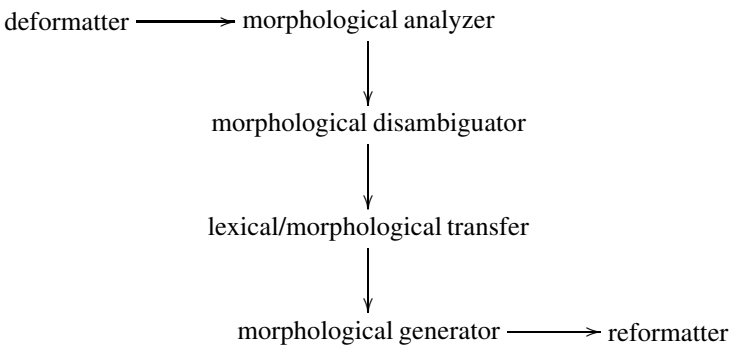



Figure 1: The shallow-transfer MT architecture as proposed in (Hajič, Hric, and Kuboň 2000)

such as Czech and Slovak, there may occur semantically relevant discrepancies. This problem has been partially solved by the division of the glossary into a domain-specific part and a general part. During the lexical transfer, the domain-specific glossary is used first and the general glossary is used only if no translation has been found.

The final phase generates word forms in the target language which is comparatively simple. It may happen that a lemma is unknown in the morphological module of the target language because it has not been translated at all or simply because the module does not contain it. In such a case, the lemma is left unchanged in the target sentence.

### Combining two MT systems

The idea of using a natural language as interlingua is not novel. The main motivation of such an approach is to exploit existing resources to construct an MT system for a new language pair. For example, Google Translate[1] uses English as interlingua for many language pairs. In the context of closely related languages, (Babych, Hartley, and Sharoff 2007) argue that using a pivot language does not negatively influence translation quality. In this subsection, we briefly present existing research in the area of Scandinavian MT.

(Bick and Nygaard 2007) present an MT system that translates from Norwegian (Bokmål) into English using Danish as an interlingua. The translation from Norwegian into Danish uses the shallow-transfer approach.

As there are almost no syntactic differences between the two Scandinavian languages and there is a widely corresponding polysemy, they generate the Danish translation from the output of a Norwegian tagger by substituting lemmas using a one-to-one dictionary. The output of the newly constructed Norwegian-to-Danish MT system is piped into an existing Danish parser and further processed. This approach exploits the fact that "the polysemy spectrum of many Bokmål words closely matches the semantics of the corresponding Danish word, so different English translation equivalents can be chosen using Danish context-based discriminators".

The first step in the system is the disambiguation of lemmas and PoS tagging. The subsequently used Norwegian-Danish one-to-one lexicon was built widely automatically by creating a monolingual automatically lemmatized Norwegian corpus and regarding Norwegian as 'misspelled Danish', using a Danish spell checker on the lemma candidates. Furthermore, phonetic transmutations for Norwegian and Danish were produced to generate hypothetical Danish words from Norwegian words. The presented approach resulted in a list of 226,000 lemmas with Danish translation candidates.

After tagging, Norwegian lemmas are substituted by Danish ones. Additionally, there is a special handling of compound nouns based on partial translation of words. The morphology of the two languages is not completely isomorphic and there are also some structural differences that are handled by a CG grammar (for example, double definiteness in Norwegian which is solved by substitution rules).

## 3. Increasing the accuracy of the shallow-transfer approach

As has been already mentioned, the statistical tagger used to disambiguate the input text at the beginning of the translation process introduces too many errors into the processed data. Unfortunately, the only way to avoid these errors is to omit the tagger from the system and work with ambiguous input. Obviously, the exclusion of the tagger from the system has to be compensated somewhere else in the translation process.

Let us have a look at an example. We would like to translate the following Czech phrase into German:

(1) *auta* *jezdila*
cars-NEUT,NOM,PL went-PAST,NEUT,PL.
"the cars moved"

If we would use a tagger, and if its result would be correct, the output would be as follows:

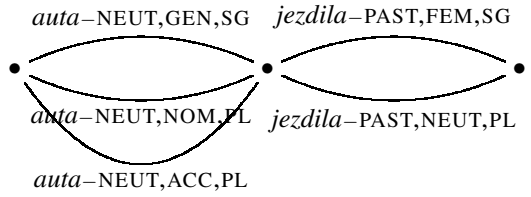
(2) *auta*-NEUT,NOM,PL *jezdila*-PAST,NEUT,PL

and a word-to-word translation into German would give a correct translation. However, both words are morphologically ambiguous and if we omit the tagger, each input word form would split in several morphologically distinct lemma-tag pairs. For example, some Czech adjectival word forms

[1] http://translate.google.com

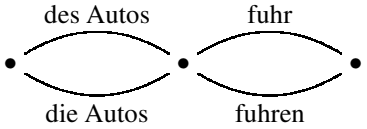
can have up to 27 distinct morphological meanings. The following structure would be the input of the subsequent modules:

(3)




Without a parser or another module which would resolve the ambiguity, the system would output the following German representation after the morphological synthesis:

(4)




We see that two edges have been merged into one due to morphological syncretism but there are still four possible outputs if one would consider all paths through the multigraph from the initial node to the end node.

We decided to add a module to the system that would find the 'best' path through the multigraph. We suggest to use a language model for the target language. In our experiments, a trigram model based on word forms and trained on about 20 million words from the Wikipedia has been used.

In the resulting German representation (in the above example), the correct path through the multigraph would be found correctly. Nevertheless, there is another problem — for longer sentences, this approach leads to a combinatorial explosion. Fortunately, the solution is comparatively simple: we have added a non-deterministic partial parser based on LFG (Bresnan 2002) and our experiments show that even if we parse only noun and prepositional phrases, the morphological ambiguity gets reduced significantly even for languages with rich inflection, such as Czech. Syntactic analysis is needed anyway to mark local dependencies that will be used in the structural transfer. The improved architecture is given in Figure 2.

## 4. Transfer

Transfer and syntactic synthesis are performed jointly in one module. The task of the transfer module is to adapt complex structures created by the parser to the target language lexically, morphologically and syntactically. In the following subsections, we describe lexical transfer and structural transfer separately.

### Lexical transfer

The aim of the lexical transfer is to 'adapt a syntactic structure lexically', i.e., the lemmas associated with nodes are translated. Morphological features may be adapted as well where appropriate.

The following is a fragment of the dictionary used in lexical transfer (Czech-Slovenian):

(5)
```
hvězda|zvezda
dodat|dodati
kůň|konj
strom|drevo|gender=neut;
```

Let us have a brief look on the last line of the example. The Czech noun *strom* "tree" is masculine while its Slovenian counterpart *drevo* is neuter, therefore there is the additional information *gender=neut* which instructs the transfer module to adapt the feature *gender* of the corresponding syntactic structure so it can be correctly synthesized morphologically.

### Structural transfer

The task of the structural transfer is to adapt the syntactic structures of the source language (their properties and mutual relationship) so the synthesis generates a grammatically well-formed sentence with the meaning of the source sentence. It is to note that the well-formedness can generally be guaranteed only locally for the part of the sentence a syntactic tree covers (this is one of the flaws of partial parsing).

When changing the structure, one may do one of the following:

1. Change values of atomic features in the corresponding feature structure, add atomic features with a specific value or delete some atomic features.
2. Add a node to the syntactic tree.
3. Remove a node from the syntactic tree.

## 5. The ranker

An essential part of the whole MT system is the statistical post-processor. The main problem with our simple MT process described in the previous sections is that both the morphological analyzer and transfer introduce a huge number of ambiguities into the translation. It would be very complicated (if possible at all) to resolve this kind of ambiguity by hand-written rules. Therefore we have implemented a stochastic post-processor which aims to select one particular sentence that suits best the given context.

We use a simple language model based on trigrams (trained on word forms without any morphological annotation) which is intended to sort out "wrong" target sentences (these include grammatically ill-formed sentences as well as inappropriate lexical mapping). The language model for Slovak has been trained on a corpus of approx. 20 million words which have been randomly chosen from the Slovak Wikipedia[2].

Let us present an example of how the ranker works. In the source text, the following Czech segment occurred as a matrix sentence:

(6) *Společnost* *ve* *zprávě*
company-FEM,SG,NOM in report-FEM,SG,LOC
*uvedla*
inform-LPART,FEM,SG
"The company informed in the report..."

[2] http://sk.wikipedia.org

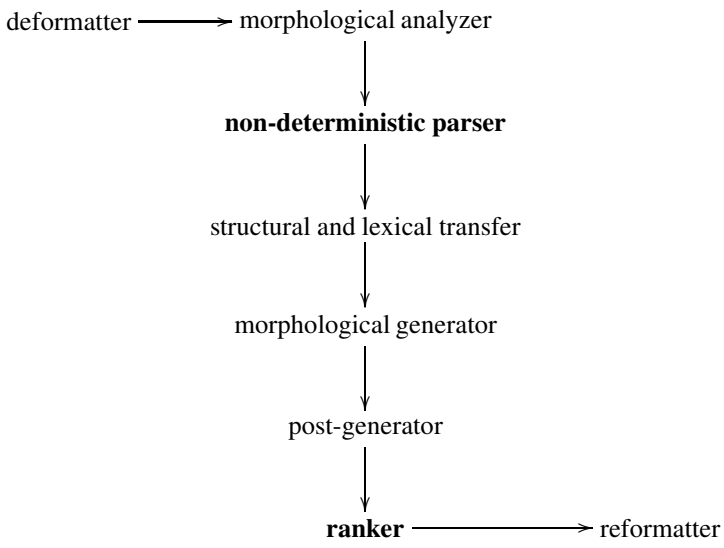



Figure 2: Improved shallow-transfer approach

The rule-based part of the system is supposed to generate (if there were no rules for VPs) four target segments that collapse to the following two ones after morphological synthesis: 1. *Spoločnosť vo správe uviedli*, 2. *Spoločnosť vo správe uviedla.* The Czech word *uvedla* is ambiguous (fem.sg and neu.pl). According to the language model, the ranker is supposed to choose the second sentence as the more probable result.

There are also many homonymic word forms that result in different lemmas in the target language. For example, the word *pak* means both "then" and "fool-pl.gen", the word *tři* means "three" and the imperative of "to scrub", *ženu* means "wife-sg.acc" and "(I'm) hurrying out" etc. The ranker is supposed to sort out the contextually wrong meaning in all these cases if it has not been resolved by the parser.

Let us define the trigram language model formally. For a given word sequence $W = \{w_1, \ldots, w_n\}$ of $n$ words we define its probability as:

$$p(W) = p(w_1, \ldots, w_n) = \; = \prod_{i=1}^{n} p(w_i | w_0, \ldots, w_{i-1}) \tag{7}$$

where $w_0$ is chosen appropriately to handle the initial condition.

As it is computationally not viable to work with unlimited history, we use a mapping $\phi$ that approximates the history (in our case by trigrams):

$$p(W) \approx \prod_{i=1}^{n} p(w_i | w_{i-2}, w_{i-1}) \tag{8}$$

To estimate the trigram probabilities, we use a large training corpus:

$$f(w_3 | w_1, w_2) = \frac{c_{123}}{c_{12}} \tag{9}$$

where $c_{123}$ is the number of times the sequence of words $(w_1, w_2, w_3)$ is observed and, analogically, $c_{12}$ is the number of times the sequence $(w_1, w_2)$ is observed.

Due to the well-known problem of sparse data, we have to use smoothing. A common smoothing method is the linear interpolation of trigram, bigram and unigram frequencies and a uniform distribution on the vocabulary:

$$\begin{aligned} p(w_3 | w_1, w_2) &= \lambda_3 f_3(w_3 | w_1, w_2) \\ &+ \lambda_2 f_2(w_3 | w_2) + \lambda_1 f_1(w_3) + \lambda_0 \frac{1}{V} \end{aligned} \tag{10}$$

Finally, we modify the formula used to find the word sequence with maximal probability. Multiplying many small numbers on a computer may result in zero so we operate with logarithms of the probabilities and use the fact the the logarithm of a product is equal to the sum of logarithms.

$$\begin{aligned} \text{argmax}_W p(W) &= \text{argmin}_W - \log p(W) = \\ &= \text{argmin}_W \sum_{i=1}^{n} - \log p(w_i | w_{i-2}, w_{i-1}) \end{aligned} \tag{11}$$

## 6. Combining two MT systems

We did two experiments with coupled MT systems translating from Czech to Slovak through Slovenian as the intermediary language. The first system simply pipes the output of the Czech-to-Slovenian MT system into the Slovenian-to-Slovak one. The other experiment couples both MT systems at a higher level, omitting morphological synthesis and statistical ranker in the first language pair. As our experiments have shown, the latter approach produces significantly better translation.

The high-level pipeline is schematized in Figure 3. The dotted arrow denotes the 'shortcut' which has been taken in the system architecture to omit morphological synthesis and ranker in the first language pair.

| | BLEU | NIST |
|---|---|---|
| simple pipe | 0.1690 | 3.5916 |
| high-level pipe | 0.2303 | 4.1737 |

Table 1: Evaluation of the coupled MT systems

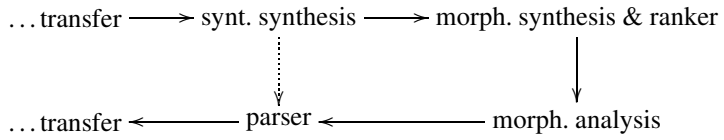



Figure 3: Combining two MT systems

## Evaluation

To evaluate the difference in translation quality, we have used the MT evaluation metrics BLEU (Papineni et al. 2001) and NIST (Doddington 2002) although they are based on words which is a crucial problem for languages with rich inflection (a comparatively small difference in an ending of a word is penalized as if the translation was completely wrong; see (Callison-Burch, Osborne, and Koehn 2006) for a detailed discussion of the deficits of BLEU). Nevertheless if we compare the scores given by both metrics, we see that they correlate in expressing which method gives better results.

The evaluation of our experiments with MT from Czech to Slovak through Slovenian clearly shows that we get better results if we couple the two MT systems at a higher level. The main point is that the statistical ranker is not used in the first MT system, postponing the selection of one translation hypothesis to a later stage. At the first sight, this strategy may seem to cause huge ambiguity in the translation process. However, if we do not use morphological synthesis in the first MT system, we do not need morphological analysis in the second system either, thus what we can avoid is the morphological ambiguity of the input in the second MT system (which is extremely important for languages with rich inflection, such as Slovenian). In other words, the parser in the second MT system deals with ambiguity of a different type, namely structural and semantic, which resulted from the first system and could not have been resolved prior to the ranking.

The comparison of both systems (on the same input text) has brought an interesting observation: the MT system with the more sophisticated coupling works faster, most probably due to the fact that morphological ambiguity of the intermediary representation (which is the input of the MT for the second language pair) is widely reduced.

The results are summarized in Table 1.

## 7. Advantages of the improved approach

The proposed improvement of the shallow-transfer approach has a very advantageous side-effect. Since the disambiguating module is placed at the end of the translation process (unlike the original architecture where it was the first module), all modules can generate ambiguous output. Thus in the dictionary, not only one-to-one entries are allowed which would be too restrictive for most language pairs. Furthermore, the parser can be non-deterministic, i.e., rules can be applied in any order and give possibly more than one syntactic representation. This property has also been used in the combined translation pair that consists of two MT systems — the ranker is used at the and of the whole pipeline.

The adapted shallow-transfer architecture also has a practical advantage. A tagger has to be trained on a morphologically annotated corpus, whereas an n-gram language model can be trained on word forms, i.e., no annotation is needed. It is well known that manual annotation is a very time-consuming task and for many small languages, there are no such corpora available, hence it is really a huge advantage. For the n-gram language model, one can use any unannotated corpus, such as the Wikipedia which is available in many languages including the small ones.

## 8. Conclusions and future work

We have presented a modification of the commonly used shallow-transfer MT approach and a sophisticated method of combining two MT systems to obtain a new translation pair. Experiments performed on the language pairs Czech-Slovenian and Slovenian-Slovak clearly show that the translation quality is better as compared to a simple pipe of the two MT systems.

In our further research, we would like to examine how a more complicated statistical language model for the target language will influence the quality of the shallow-transfer approach.

## References

Babych, B.; Hartley, A.; and Sharoff, S. 2007. Translating from under-resourced languages: comparing direct transfer against pivot translation. In *Proceedings of MT Summit XI*, 29–35.

Bick, E., and Nygaard, L. 2007. Using Danish as a CG Interlingua: A Wide-Coverage Norwegian-English Machine Translation System. In *Proceedings of NODALIDA*.

Bresnan, J. 2002. *Lexical-functional syntax*. New York: Blackwell Textbooks in Linguistics.

Callison-Burch, C.; Osborne, M.; and Koehn, P. 2006. Re-evaluating the Role of BLEU in Machine Translation Research. In *Proceedings of the EACL'06*.

Corbi-Bellot, A.; Forcada, M.; Prtiz-Rojas, S.; Perez/Ortiz, J. A.; Remirez-Sanchez, G.; Martinez, F. S.; Alegria, I.; Mayor, A.; and Sarasola, K. 2005. An Open-Source Shallow-Transfer Machine Translation Engine for the Romance Languages of Spain. In *Proceedings of the 10th Conference of the European Association for Machine Translation*.

Debowski, L.; Hajič, J.; and Kuboň, V. 2002. Testing the limits — adding a new language to an MT system. *Prague Bulletin of Mathematical Linguistics* 78.

Doddington, G. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the ARPA Workshop on Human Language Technology*.

Dyvik, H. 1995. Exploiting Structural Similarities in Machine Translation. *Computers and Humanities* 28:225–245.

Hajič, J., and Kuboň, V. 2003. Tagging as a Key to Successful MT. In *Proceedings of the Malý informatický seminář*.

Hajič, J.; Hric, J.; and Kuboň, V. 2000. Machine translation of very close languages. In *Proceedings of the 6th Applied Natural Language Processing Conference*, 7–12.

Hajič, J.; Homola, P.; and Kuboň, V. 2003. A simple multilingual machine translation system. In *Proceedings of the MT Summit IX*.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2001. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, 311–318.