A Virtual Translation Machine for Hybrid Machine Translation

Oliver Streiter[*], Michael Carl[**], Leonid Iomdin[***]

[*]Academia Sinica, Institute of Information Science, Nankang, Taipei, Taiwan 115
oliver@iis.sinica.edu.tw

[**]Institute for Applied Information Sciences (IAI), Martin Luther-Straße 14, 66111 Saarbrücken, Germany
carl@iai.uni-sb.de

[***]Institute for Information Transmission Problems, RAS, Bol'shoi Karetnyj Pereulok 19, Moscow, 101447, Russia
iomdin@iitp.ru

Abstract

The last decade has seen the rise of a number of new and promising approaches to machine translation. Currently the research in this field is directed to the development of hybrid MT systems which integrate more than one approach to MT, the idea being that an integration will help achieve properties that combine the advantages of the approaches involved. The experiments staged until now have illustrated how different approaches can be technically combined. These experiments equally suggested that improvements in translation quality may be expected from such measures. However, theory framing lacks far behind. As a consequence, no fundamental insight could be formed which could allow to predict system properties of hybrid systems and thus to apply specific types of hybridization to concrete translation settings. Aso, it is difficult to apply equal evaluation criteria to the integrated system as well as to the individual components. This lack of theoretical foundation of the hybridization is mainly due to different architectures and different operations performed by the systems. A detailed description and evaluation of the system operations, however, is required in order to optimize the performance of hybrid approaches, to know which operation should be performed according to which approach, or to assign weights if different approaches run in parallel. For this purpose, we propose to define the translation process in terms of a virtual translation-machine (VTM), which allows to conceptualize and evaluate all MT approaches and their hybridization. A first sketch of a VTM is given here, together with some virtual operations and their evaluation functions.

Виртуальная машина перевода для гибридного машинного перевода

Оливер Штрайтер[*], Микаэль Карл[**], Леонид Иомдин[***]

[*]Академия Синика, Институт Информатики, Нанканг, Тайпэй, Тайвань 115
oliver@iis.sinica.edu.tw

[**]Институт прикладной информатики (ИАИ), Мартин-Лютер-Штрассе 14, 66111, Саарбрюккен, ФРГ
carl@iai.uni-sb.de

[***]Институт проблем передачи информации РАН, 101447, Москва, Большой Каретный пер., 19
iomdin@iitp.ru

Резюме

В последние годы появился ряд новых перспективных подходов к МП. Современные исследования в этой области нацелены на разработку гибридных систем МП, в которых интегрируются разные подходы. Ожидается, что с помощью интегральной архитектуры удастся объединить преимущества этих подходов. Новейшие эксперименты показывают, как различные подходы к МП можно сочетать на практике. Одновременно эти эксперименты подтверждают тот факт, что при интеграции можно рассчитывать на повышение качества перевода. Однако теория гибридного МП существенно отстает от экспериментальной практики. В результате пока не удается достичь такого понимания проблемы, которое позволило бы предсказывать особенности гибридных систем в целом и тем самым применять конкретные схемы интеграции к системам с конкретными параметрами. Трудно также подобрать универсальные критерии оценки интегральной системы и отдельных ее компонентов. Отсутствие теоретической базы для гибридных систем объясняется в первую очередь разнообразием их организации и функционирования. Поэтому для оптимизации гибридного подхода требуется детальное описание и оценка системы, знание того, как распределять ее работу между компонентами, умение приписывать веса этим компонентам, если они включены параллельно. В данной работе мы предлагаем взглянуть на процесс перевода с точки зрения виртуальной переводческой машины (ВПМ), которая позволяла бы осмыслить и оценить разнообразные подходы к МП, применяемые как порознь, так и в сочетании друг с другом. Ниже дается эскиз ВПМ и кратко рассматриваются ее возможные операции.

1. Introduction

In the 70s and the 80s, research and development in the field of machine translation (MT) was dominated by the **rule-based** approach (RBMT). Inspired by linguistic theories, linguists and information scientists jointly developed MT systems, in which the following division of labor was prevalent: the information scientists developed the computational framework and linguists filled the framework with their linguistic knowledge. The performance of these systems was seldom impressive, so that many systems quietly passed away when financial support was no longer available, or for other natural reasons. Companies that experimented with some of the few commercial RBMT systems had to go through a long and expensive period of customization.

At the beginning of the 90s, when more machine readable corpora became available, the unique position of the linguist in the production of translation-related knowledge was questioned. Monolingual and, of course, bilingual corpora could also provide translation knowledge which could be used if the translation framework was changed accordingly. MT systems that make use of knowledge extracted from corpora are called **corpus-based** machine translation (CBMT) systems.

Translation Memories (TM), initially developed as an extension to the translator's personal dictionaries, allow to find pieces of text identical or similar to pieces of text already translated. The newest versions of TMs are able to cope with huge amounts of bilingual texts and have an in-built training function which allows to constantly update translation knowledge that TMs contain. A crucial phase in the treatment of bilingual texts is the alignment, a procedure in which translation units of the two texts (phrases, sentences, paragraphs etc.) are put in correspondence with each other (Somers 1999).

Largely, TMs stay within the CBMT paradigm and as such can be viewed as another variety of MT. In fact, they are sometimes considered as a subset of **example-based** MT (EBMT). However, we believe that a more informative classification could be produced if TMs and EBMT are both viewed as different varieties of **memory based** MT (Sato and Nagao 1990). In this case the label of EBMT proper should be applied to MT paradigms which operate with smaller translation units than TMs usually do (phrases but not sentences or paragraphs) and which, as a consequence, can rely on richer data representation than that just string information. A detailed discussion of this issue can be found in Carl (2000).

The **statistics-based** approach represents a third variety of CBMT. This approach relies on stochastic translational models, trained on aligned corpora in order to optimize translation parameters (Brown *et al*. 1993).

Although the partisans of the "classical" RBMT approach come under pressure due to the advantages that CBMT approaches have over RBMT, they can gain some new support in the newly developing web-based information technology, such as **multilingual information retrieval** (Czuba and Liu 1999), **on-line translation, multilingual visualization** (Hong and Streiter 1999), and **UNL-related activities** (Boguslavsky *et al*. 2000).

It is becoming more and more obvious that all approaches have their individual strong and weak points and that there are no ideal systems of machine translation, unless they are specially designed for a narrowly and strictly defined translation setting. In particular, corpus-based MT systems are easy to customize, while memory-based systems are valued for their translation reliability. RBMT systems have the advantage of being able to produce a raw translation for many text types, a capacity often referred to as coverage. It thus seems only natural to try and combine different approaches in order to offer a more satisfactory MT paradigm.

2. Hybridization

The hybridization of MT approaches attempted so far primarily concentrated on two aspects: the technical aspect and the improved translation quality. A prominent example is a parallel run of three different MT engines and the combination of their output (Frederking and Nirenburg 1994, Frederking *et al*. 1994). Although this research was ground-breaking, not so much could be concluded. First of all, only graphematic outputs of the engines were combined and such types of interaction obviously have a limited effect. Interactions of different systems or approaches in internal and thus richer representations may produce diversified and more impressive effects (for a discussion, see Streiter and Iomdin 2000). A second drawback of this research, as well as of many others that followed, is a pre-scientific flavor. In these experiments A is added to B and a reaction is observed. However the attempts to explain what properties of A and what properties of B caused exactly what kind of reaction have been neglected to a large extend, and a scientific stage in which accurate predictions can be made and verified still seems far ahead.

However, the experience accumulated by hybridization attempts is constantly growing. These attempts include

- integration of statistical information into RBMT systems using various techniques (Nomiyama 1991, Carbonell *et al*. 1992, Doi and Maraki 1992, Chen and Chen 1995, Rayner and Bouillon 1995, Streiter *et al*. 2000, Iomdin and Streiter 2000),
- extraction of new translation units out of bilingual text and their compilation into RBMT systems (Streiter and Iomdin 2000),
- combination of translation memories with RBMT (Heyn 1996, Carl *et al*. 2000),
- combination of EBMT with RBMT (Carl *et al*. 2000, Turcato *et al*. 2000),
- combination of translation memory with EBMT (Carl and Hansen 2000).

In light of the abundance of experimental results and the diversity of techniques used, it is surprising that no theory of hybrid MT has emerged so far. The purpose of the present paper is to bridge the obvious gap.

3. Towards a Virtual Translation Machine

Every MT paradigm and, within one MT paradigm, every concrete MT system performs the translation task differently. A module which is crucial in one system may be entirely absent in another. The description and evaluation of such a module may thus be irrelevant when we compare two systems when contemplating their integration into a hybrid MT-system.

However, whereas internal modules of different MT systems can only be compared with difficulty, the external behavior could be measured and could thus motivate a choice for hybridization. This is the traditional black box approach to MT (see e.g. Nagao 1985) which has already been resorted to in the hybridization experiments mentioned above.

Of course, the problem of objective evaluation in MT has not yet received an adequate solution. On the one hand, translations themselves are difficult to evaluate (see Hutchins 1996). On the other hand, as far as CBMT systems are concerned, they are not necessarily responsible for the quality of the translation, as they mostly reproduce the textual material which has been taught to them (Reinke 2000). For these systems, evaluation tools typically applied to Information Retrieval systems are more appropriate.

To facilitate the task of finding a common basis for comparison and evaluation of different MT systems, we propose to model the translation process as a virtual translation machine (VTM). The VTM[1] decomposes translation into a number of virtual operations which can be tackled independently of each other in order to:

a) describe the most important properties for the current translation setting;
b) describe the weak and strong points of the different approaches;
c) choose an appropriate form of hybridization;
d) test the performance of the integrated system with respect to that of every individual component;
e) test the performance of the integrated system with respect to the requirements of the translation setting.

The definition of virtual operations should be made system independent and language independent to the maximum extent possible. However, the application of the definitions is both system and language dependent. As the basic unit of VTM we propose the translation unit (TU), e.g. a word or an idiomatic expression. The notion of grammar, which has an important status in evaluation approaches that do not follow the black box approach (e.g. Nyberg *et al*. 1994) is ignored together with all stratificational concepts of MT inherited from the RBMT approach.

4. Translation Units

We define a TU as an ordered or unordered set of lexical items of the source language which is associated with an ordered or unordered set of lexical items of the target language, prior to the moment the translation process starts. In an ordered set, the permutation of two lexical items results in another TU.

Lexical items in a TU may be supplemented by a number of constraints. For example, each lexeme may be confined to a specific surface string (e.g. the lexeme *be* may be confined by the surface string *are* for the TU *How are you*). Similar constraints may equally be expressed by a set of restricting morphosyntactic features.

A translation unit may further be restricted by constraints on slots. A slot is a position which is not lexically specified and may be filled by words or phrases. Thus, if a conventional dictionary mentions *rely on* ***someone***, *someone* is not a lexical specification, but a metalinguistic specification of the slot, so that the TU is actually *rely on* plus the constraint expressed by *someone*. Frame descriptions found in advanced linguistic dictionaries, also belong to the TU. Two TUs which only differ in constraints on the slots are counted as two different TUs.

---

[1] In the most recent usage, the term "virtual machine" is used to refer to the software that acts as an interface between a compiled Java binary code and the hardware platform that actually performs the instructions. The Java virtual machine specification defines an abstract rather than a real machine by an instruction set, a set of registers, a stack etc. The physical implementation of this abstract processor is done in a code that is recognized by the real processor or may be built into the microchip processor itself. The "abstract machine" is a more general term describing a technique which allows to bridge high-level programming languages and Von Neumann machine languages. Most prominent instances of such abstract machines are the P-code of Pascal and the Warren Abstract Machine for Prolog, and if one looks back in history, the Turing machine. At even higher levels, an abstract machine has recently been proposed to bridge HPSG grammars to low-level languages (Winter and Francez 1999).

An MT system is free to choose its TUs. Thus, *rely on me* and *rely* may equally be used as TUs, in addition to or as a replacement of *rely on someone.* While translation memories do not use slots, slots can be found both in EBMT and RBMT. What may be a slot in RBMT may be lexically specified in EBMT. If the slot of a TU is instantiated by another TU, e.g. during the syntactic analysis or generation, we do not produce one big TU but retain two separate TUs, as the new correspondence has not been defined prior to the translation process. Of course, this combination of TUs could be stored for further translation processes by way of which they would be transformed into new TUs.

We define a lexical item as a word, a word form or a concept denoted by the word.

5. Parameters of a Virtual Translation Machine

The first implicit attempt to define such a VTM has been made in Carl *et al*. (2000), who argue that all MT systems

1) establish a match between the source text (ST) and translation units (TU) available in the MT system and
2) adapt the target side of the translation units (TUTS) to the new context (i.e. TUTSs exchange information and as a consequence change their form and position).

As can be seen, the notion of grammar of the source language is reduced to the collection of information which guides the ST=>TU matching on the one hand and the adaptation on the other hand. The notion of grammar of the target side is presented as the form and position of words. A more detailed definition of MT submodules is possible:

1) the match between
   1a) the source text (ST) and the source side of the translation unit (TUSS) and
   1b) the source side of the translation unit (TUSS) and the target side of the translation units (TUTS);
2) The exchange of information between TUs related to
   2aa) changes of word forms;
   2ab) the insertion of words, i.e. words which are not specified in the TU;
   2ba) the ordering of words within a TU;
   2bb) the ordering/scrambling of TUTSs.

5.1. Recall

The operations 1a) to 2b) can be evaluated in terms of **recall** since they offer reference solutions allowing to assess the success of the operations. The obtained values are called, respecitively. **recall_ST → TUST** (recall of the source text with respect to translation unit matching), **recall_TUST → TUTT** and **recall_TUTT → TT**.

The **recall** parameter evaluates an operation through the comparison of the outcome of an operation with a reference solution. If, among other things, the reference solution is produced, the operation is considered to be correct. The quotient of the sum of the correct operations and the sum of reference solutions indicates the percentage of good solutions against the solutions expected to be produced:

$$\mathbf{recall} = \Sigma_{\text{correct operations}} / \Sigma_{\text{reference solutions}}$$

5.2. Coverage and Reliability

Following (Streiter 2000), we will define coverage in terms of recall in untrained corpora and reliability as the recall in trained corpora. Coverage reflects the capability of a module to make correct predictions for the system that has not received any special training. Reliability indicates the degree to which a module can be taught and indicates whether learned knowledge can be retrieved. In addition to this, reliability is also required to increase coverage.

A module which achieves a reliability of less than 1 implies that some parts of the trained items cannot be correctly reproduced. Such kind of 'translation errors' are either due to inconsistencies in the training text: i.e. for one TUSS there is more than one TUTT and the system is unable to retrieve the 'correct' TUTT or has decided to use the 'wrong' TUTT.

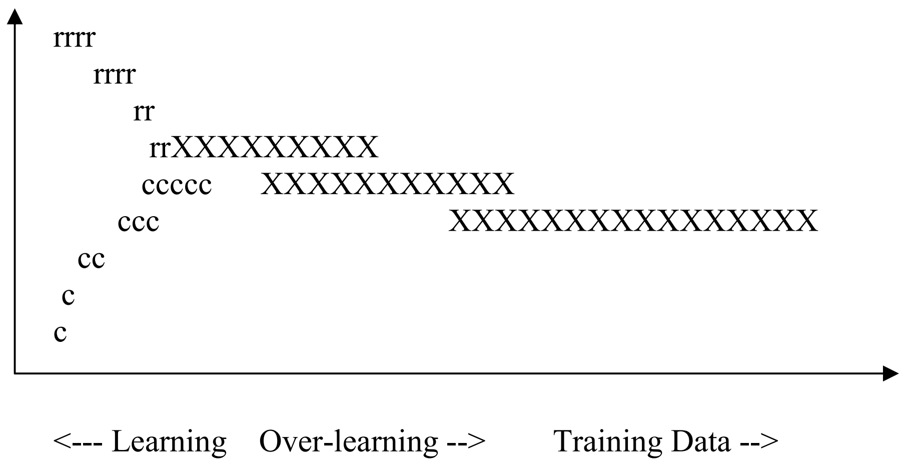
A a module achieving coverage of more than 1 indicates that its output is different from what is expected in the test text. There are two reasons why this might occur: (1) as with reliability, the module has learned inconsistencies of the training text which cannot be resolved in the test text or (2) the items to be treated are unknown to the module.

We believe that one may have high reliability without high coverage. High coverage and low reliability would mean that a module translates the test text better than it translates the training text. It is reasonable to expect that this is only possible if the test text is a particular consistent subset of the training text while the training text contains inconsistencies. A similar conclusion can be drawn from the experimental data reported in (Daelemans et al 1999a, Daelemans et al 1999b).

For any module, we thus expect the scores for reliability to be higher or equal than the score for the coverage. With an accumulation of the training data, the reliability and the coverage converge, but the coverage does not usually exceed the reliability. With more training data, the coverage increases because more items are becoming either known or similar to known. When all necessary items are learned, the difference between coverage and reliability disappears. The performance of the system in that case is limited by its reliability.

Due to inconsistencies usually present in non-treated texts, we further assume that the reliability decreases with more and more training data. With little training data, learned instances may be correctly retrieved. When more instances are learned ambiguities arise in the retrieval. While most experimental data suggest an asymptotic rise of the coverage, we currently have no data to estimate the drop in reliability.

Before the two functions converge, however, the coverage may decrease again under the influence of the decreasing figures of reliability. The best performance a system may ever achieve thus corresponds to the estimated point between the reliability and coverage, marked below as X.



We thus estimate the point where reliability and coverage may converge via the assumption that the space between the current coverage and reliability will be distributed between them with more training data in the same proportion as the space between 0 and 1 is currently distributed between them[2]:

$$K = c + \frac{(r^2 - \sqrt{c}) \cdot \sqrt{c}}{(1 - r^2) + \sqrt{c}}$$

In order to model the asymptotic behavior of the coverage and (maybe) the reliability, we take the square of the reliability (*r*) and the square root of the coverage (*c*), by way of which we can also simulate the over-learning.

For instance, this formula yields for *c*= 50% and *r* = 90%: K = 0.5 + ((0.81 - 0.7) * 0.7 / (1 - 0.81) + 0.7)) = 0.58.

The following table illustrates how the estimated maximum (K) depends on the recall on learned corpora (reliability) and unlearned corpora (coverage) in one model with an increasing number of trained sentences:

Learned Sentences	Reliability	Coverage	Estimated K
4,000	0.9981	0.6927	0.8559
8,000	0.9970	0.7000	0,8562
12,000	0.9958	0.7067	0.8561
16,000	0.9946	0.7071	0.8536

6. Arithmetic Across Modules

Imagine that we conceptualize an MT system as consisting of 2 submodules, e.g. the decomposition module for the mapping ST=>TU and the adaptation module TU=>TT. Imagine further that we can evaluate the ST=>TU mapping in terms of coverage and reliability. We want to replace the module responsible for the ST=>TU mapping by two modules, one responsible for the ST=>TUSS and the other responsible for the TUST=>TUTS. In this section we provide a terminology which one can use in the estimation of the gain of such a modification.

If we know the recall of Module B and know how many correct forms are generated by module B, we can calculate the conditional number of correct forms produced by module A as follows:

correct(A|B) =correct(A,B) * recall(B)

Obviously, if module A depends on the outcome of module B, we will set the joint operation:

correct(A,B) =correct(A)

Else, if module A works independently of B's output, we will have

correct(A,B) =correct(A) / recall(B)

The conditional recall of module A is computed according to the following formula:

---

[2] Boitet (1999) assumes a relation where K reflects a system inherent maximum which can only be achieved with intensive training:

Coverage * Quality = K.

As our concept of coverage is not separated from the quality (we define coverage as recall of the unlearned corpora), we model K by means of coverage and reliability.

recall(A|B) = correct(A|B) / reference(A)

The following is an example which assumes a maximum dependence of A and B. Suppose that module A has 50 correct from 100 in the reference. Module B has 25 correct from 50 in the reference, so the recall for both is 0.5. For the conditional operation of modules A and B we have a reference of 100 and 25 correct forms and a conditional recall of 0.25:

	Module A	Module B	Modules A|B
Correct	50	25	25
Reference	100	50	100
Recall	0.5	0.5	0.25

In the case when the two modules are independent, the conditional recall A|B is 0.5, as shown in the first column of the above table.

Unfortunately, cases of clear dependence or clear independence of the modules are seldom found. A fundamental issue that is NOT solved by the VTM but has to be clarified with the help of the concrete modules is whether their performance is independent or not. More precisely, if the performance is positively correlated (items treated correctly in Module A are also treated correctly in Module B), we can obtain estimated performance values by selecting the minimum of A and B. If the performance is not correlated we can estimate the performance values via the product of the respective values of A and B. If the performance is negatively correlated (items treated correctly in A are treated erroneously in B) we estimate the overall performance value with 0. If two operations work on different units (e.g. ST=>TU works on TUs and Adaptations work on lexemes) we can estimate them as independent. If they work on identical units, the best performances are obtained with a positive correlation and the worst performance with a negative correlation. Of fundamental importance is thus the calculation of their dependence, using, for example, the Spearman's Correlation Coefficient.[3]

We therefore propose a formula which contains the described correlation of –1, 0 and 1 as special cases:

*Total = lim - ((min * max) - lim) * (|corr| -1))*

Here, *total* is the score for the combined module, *min* is the score of the worse performing submodule, *max* is the score of the better performing submodule, *corr* is the Spearman's correlation coefficient, *lim* = 0 for corr < 0 and *min* for corr ≥ 0.

The scores of this function for *min* = 0.2 and *max* = 0.3 are:

corr	Total
-1	0

[3] The Spearman's correlation coefficient is defined as

$$r_{xy} = \frac{\sum_{i=1_n} (x_i - \overline{x}) \cdot (y_i - \overline{y})}{\sqrt{\sum_{i=1_n} (x_i - \overline{x})^2 \cdot \sum_{i=1_n} (y_i - \overline{y})^2}}$$

where x and y are the sets to be correlate and $\overline{x}$ and $\overline{y}$ are the respective arithmetic means. r_{xy} takes values between -1.0 and +1.0. Perfect (inverse) correlation corresponds to +1.0 (-1.0) and no correlation to 0.0.

-0.8	0.012
-0.4	0.036
-0.2	0.048
0|	0.06
0.2	0.116
0.4	0.144
0.8	0.172
1	0.2

Another essentially different type of hybridization involves the parallel run of different modules performing the same operation. Either all systems work on the tasks and the best output is used for further processing (Frederking and Nirenburg 1994), or the task is distributed among the modules according to the expectations where the individual items are treated best (Carl *et al*. 2000). In such an architecture, the recall of two modules can be added if the two modules are negatively related. In case of a positive correlation we simply take the maximum of one of the modules, under the assumption that the best module/best output can always be identified.

*total = max - ((correlation - 1) * min / 2)*

We thus can see that this parallel form of hybridization is generally more promising (in the worst case the maximum determines the performance while in the pipeline form of hybridization at best the minimum determines the performance). The scores of this function for min = 0.2 and max = 0.3 are:

corr	Total
-1	0.5
-0.8	0.48
-0.4	0.44
-0.2	0.42
0|	0.4
0.2	0.38
0.4	0.36
0.8	0.32
1	0.3

7. Relevance of the Proposed Operations

Having defined a principle set of virtual operations and evaluation measures associated with them, we now come to the discussion of some of these operations in order to highlight their relevance and to show how they can be applied to individual and hybrid forms of MT-systems.

7.1. Case A.

The ST ⇒ TU mapping is an operation which could be used in order to evaluate the hybridization presented in Carl *et al*. (2000). In this research an EBMT system is linked to an RBMT system so that phrases stored in the EBMT system are recognized by the EBMT system and do not require an analysis of the RBMT system. The ST ⇒ TU mapping of each individual system can be established, as well as that of the integrated system. The principal hypothesis has been that the high coverage of the RBMT system and the high reliability of the EBMT system could be maintained and that the combined system shares both properties. Given the above definitions, such a proof becomes possible.

In addition, given the correlation coefficient of the two parallel submodules and the empirically obtained data of the joint modules, it becomes possible to evaluate the function which selects the module or the output for each individual item, not only in this tasks, but also in (Frederking and Nirenburg 1994).

In the evaluation of the individual MT systems as well of the combined system, one must also be aware of unpredictable and even irrational distortions that may be produced by some of the TU's processed. For example, the English/German RBMT system wrongly identifies *Washington Post* as two Tus and translates them separately as *Washington* and *Post* (recall=2/2). However, by pure coincidence the ultimate English-German translation is correct (recall=2/2). In another instance, the RBMT system incorrectly recognizes *Secretary of State* as two TUs and in this case the ST ⇒ TU mapping into German becomes incorrect: *Minister des Zustandes* 'minister of situation' instead of *Staatsminister* (recall=1/2). Naturally, these distortions depend on the language pair concerned: so, an English/Russian system would completely fail in the former case but may produce a quasi-correct translation in the latter.

Given the scores of the individual operations and the correlation coefficient, the properties of the combined system can be estimated and according to this estimation, the interfacing of the two systems can be started or not. (Actually with a parallel architecture, we always may expect improved result, unless the modules are highly correlated).

Then, the experimental data obtained with the combined system can be compared to the predictions made by the proposed model. The hypothesis that the size of TUs is positively correlated to the reliability of ST-TU could also be tested. Further hypotheses can be derived from the model, e.g. that bigger TUs have a higher correlation of ST ⇒ TUSS and TUSS ⇒ TUTS mapping.

7.2. Case B.

The ST ⇒ TUSS has been evaluated tentatively in Streiter *et al*. (2000) where monolingual statistical corpora were used to enhance RBMT. It has been suggested that the ST => TUSS can be improved using monolingual statistical information on the source text, irrespective of the TUSS ⇒ TUTS mapping. Monolingual statistical information concerning the target text is used to enhance the TUSS ⇒ TUTS mapping. Given the proposed VTM definition, a much richer evaluation would have been possible, including the conclusion whether better recall values are due to higher coverage or higher precision. Alternatively, a higher correlation coefficient of ST ⇒ TUSS and TUSS ⇒ TUTS could also be responsible for the increased values of recall.

7.3. Case C.

TUSS ⇒ TUTS evaluation could be applied in the evaluation of the research described in (Streiter and Iomdin 2000): Bilingual corpora have been used to specify translation relations of an RBMT system. Here, as in Case A, it would have been interesting to see whether the scores of reliability can be improved by this measure without any tradeoff in coverage. If this is the case, the estimated maximum and, additionally, the system's potentials of translation could become a higher estimate.

For the multi-word expressions extracted from parallel corpora and then compiled into the RBMT lexicon it would have been interesting to see whether they can be adapted; in other words, whether there has been a real gain achieved through hybridization, i.e. adaptation through the RBMT approach (which is not taken for granted in memory-based approaches) and high scores on TUTS ⇒ TUTS mappings through the CBMT approach. Without the model of the VTM machine such evaluations have not been performed.

7.4. Case D.

The adaptation is an explicit topic in Carl *et al*. (2000) where the TUTS coming from the EBMT component can be adapted by the RBMT component, i.e. German *für den alten Mann* ‘for the old man’ can be adapted by the RBMT system as *auf den alten Mann* ‘on the old man’. Thus it would have been interesting to see whether the EBMT system can be added to the RBMT system without any tradeoff in adaptation. From the viewpoint of the EBMT system, it would have been possible to analyze whether the RBMT system can be added in order to increase the scores on adaptation, without any tradeoff in SS ⇒ TU mapping.

Conclusions

We have attempted to show that any progress in the hybridization of MT relies on the definition of a virtual translation machine. This VTM defines a set of virtual operations and corresponding evaluation measures which can be combined across modules in order to estimate the performance of the combined modules. With these utilities at hand the process of hybridization can develop as follows:

- Identify the main system requirements in terms of the evaluation measures of virtual operations.
- Identify the submodule with the lowest scores for this evaluation measure.
- Identify another approach/module/system which probably performs better on this task.
- Calculate correlation coefficients between the old module and its context, the new module and the old context and the new and the old module. If possible, try a parallel architecture of the new and old module, especially with low or negative correlations between them. If you simply add or replace a module, estimate the performance values before implementation work, using the correlation coefficient.
- Evaluate the combined model empirically and improve the VTM definition.

References

Boguslavsky, Igor, Leonid Iomdin, Leonid Krejdlin, Irina Sagalova, Victor Sizov, Nadezhda Frid. Модуль универсального сетевого языка в рамках развитой системы обработки текстов на естественном языке. [Creating a Universal Networking Language Module within an Advanced NLP System.] Submitted to Dialog 2000. [In Russian, English Summary.]

Boitet, Christian. 1999. A research perspective on how to democratize Machine Translation and translation aides aiming at high quality final output. In Proceedings of MT-Summit VII. Singapore.

Brown, Peter F., Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical Machine Translation: Parameter estimation. Computational Linguistics 32:263-311. http://www.clsp.jhu.edu/ws99/projects/mt/ibm-paper.ps

Carbonell, Mitamura and Nyberg. 1992. The KANT Perspective: A Critique of Pure Transfer (and Pure Interlingua, Pure Statistics, ...). In: Proceedings of Theoretical and Methodological Issues in Machine Translation (TMI-92). http://www.lti.cs.cmu.edu/Research/Kant/

Carl, Michael. 2000. Towards a Model of Competence for Corpus-Based Machine Translation. In: Hybrid Approaches to MT. Streiter, Oliver, Michael Carl and Johann Haller (eds.). 2000. IAI Web-based Working Paper. http://www.iai.uni-sb.de.

Carl, Michael, Leonid Iomdin, Catherine Pease and Oliver Streiter. 2000. Towards a Dynamic Linkage of Example-Based and Rule-Based Machine Translation In: Hybrid Approaches to MT. Streiter, Oliver, Michael Carl and Johann Haller (eds.). 2000. IAI Web-based Working Paper. http://www.iai.uni-sb.de.

Carl, Michael and S. Hansen. 2000. Linking Translation Memories with Example-Based Machine Translation. In: Hybrid Approaches to MT. Streiter, Oliver, Michael Carl and Johann Haller (eds.). 2000. IAI Web-based Working Paper. http://www.iai.uni-sb.de.

Chen, K.-H. and H.-H. Chen. 1995. Machine Translation: An Integrated Approach. In: Proceedings of the 6th International Conference on Theoretical and Methodological Issues in Machine Translation, Leuven, Belgium.

Czuba, Krzysztof and Xin Liu. 1999. Experiments in Translingual Information Retrieval Using Web-based MT and WordNet. Proceedings of the 8th International Conference on Theoretical and Methodological Issues in Machine Translation (TMI 99), August 23-25, University College, Chester, England.

Daelemans, Walter, Sabine Buchholz, and Jorn Veenstra. 1999a. Memory-based shallow parsing. In Proceedings of CoNLL-99, Bergen, Norway, June 12. http://ilk.kub.nl/papers.html.

Daelemans, Walter, Antal Van den Bosch, and Jakub Zavrel. 1999b. Forgetting exceptions is harmful in language learning. Machine Learning, special issue on natural language learning . 34. 11-43. http://ilk.kub.nl/papers.html.

Doi, Shinichi, and Kazunori Maraki. 1992. Translation ambiguity resolution based on text corpora of source and target language. In COLING-92.

Frederking, Robert and Sergei Nirenburg. 1994. Three heads are better than one. In: Proceedings of ANLP-94, Stuttgart, Germany.

Frederking, Robert, S.Nirenburg , D. Farwell, S. Helmreich, E. Hovy, K. Knight, S. Beale, C. Domashnev, D. Attardo, D. Grannes, and R. Brown. 1994. Integrated Translation from Multiple Sources within the Pangloss MARK II Machine Translation System. In: Proceedings of Machine Translation of the Americas. 73-80.

Heyn, Matthias. 1996. Integrating machine translation into translation memory systems. In: European Association for Machine Translation - Workshop Proceedings. 111-123, ISSCO, Geneva.

Hong, Munpyo and Oliver Streiter. 1999. Overcoming the Language Barriers in the Web: The UNL Approach. Tagungsband der 11. Jahrestagung der Gesellschaft für linguistische Datenverarbeitung. http://rockey.iis.sinica.edu.tw/oliver/publ.

Hutchins, John. 1996. Evaluation of Machine Translation and Translation Tools. In: Survey of the State of the Art in Human Language Technology. http://cslu.cse.ogi.edu/HLTsurvey/HLTsurvey.html.

Iomdin, Leonid and Oliver Streiter. 1999. Learning from Parallel Corpora: Experiments in Machine Translation. In Proceedings of the Dialogue'99 International Seminar in Computational Linguistics and Applications, Vol.2. 79-88, Tarusa (Russia), June 1-5, 1999. http://proling.iitp.ru/bibitems/.

Nagao, M. 1985. Evaluation of the Quality of Machine-Translated Sentences and the Control of Language. Journal of Information Processing Society of Japan, 26 (10). 1197-1202.

Nomiyama, H. 1991. Lexical selection mechanism using target language knowledge and its learning ability. In IPSJ-WG, NL86-8 (in Japanese, cited in (Doi and Maraki 1992)).

Nyberg, Eric H., Teruko Mitamura and Jaime G. Carbonell. 1994. Evaluation Metrics for Knowledge-Based Machine Translation. In: COLING 94. The 15th International Conference on Computational Linguistics, Proceedings. 95-99. Kyoto, Japan.

Reinke, Uwe. 2000. Evaluierung der linguistischen Leistungsfähigkeit von Translation Memory-Systemen - Ein Erfahrungsbericht. In: Hybrid Approaches to MT. Streiter, Oliver, Michael Carl and Johann Haller (eds.). 2000. IAI Web-based Working Paper. http://www.iai.uni-sb.de.

Rayner, Manny and Pierette Bouillon. 1995. Hybrid Transfer in an English-French Spoken Language Translator. Proceedings of IA '95, Montpellier, France 1995. http://www.cam.sri.com/tr.

Sato, S. and M. Nagao. 1990. Towards memory-based translation. In: COLING-90.

Somers, Harold. 1999. Further Experiments in Bilingual Text Alignment. *International Journal of Corpus Linguistics*, 3, 1. http://solaris3.ids-mannheim.de/ijcl/ijcl-3-1.html#somers.

Streiter, Oliver. 2000. Reliability in Example-Based parsing. Submitted to: TAG+5, International Workshop on Tree Adjoining Grammars and Related Formalisms. http://rockey.iis.sinica.edu.tw/oliver/publ

Streiter, Oliver, Michael Carl and Johann Haller (eds.). 2000. Hybrid Approaches to MT. IAI Web-based Working Paper. http://www.iai.uni-sb.de.

Streiter, Oliver and Leonid L. Iomdin. 2000. Learning Lessons from Bilingual Corpora: Benefits for Machine Translation. To appear in: Journal of Corpus Linguistics.

Streiter, Oliver, Leonid Iomdin, Munpyo Hong and Ute Hauck. 2000. Learning, Forgetting and Remembering: Statistical Support for Rule-Based MT. In: Hybrid Approaches to MT. Streiter, Oliver, Michael Carl and Johann Haller (eds.). 2000. IAI Web-based Working Paper. http://www.iai.uni-sb.de.

Turcato, Davide, Paul Mc. Fetridge, Fred Popowich and Janine Toole. 1999. A unified example-based and lexicalist approach to Machine Translation. In: Hybrid Approaches to MT. Streiter, Oliver, Michael Carl and Johann Haller (eds.). 2000. IAI Web-based Working Paper. http://www.iai.uni-sb.de.

Winter, Shuly and Nussim Francez. 2000. Efficient Implementation of Unification-Based Grammars, Journal of Language and Computation, 1, 1. 53-59. http://www.dcs.kcl.ac.uk/journals/jlac/V11/index.html#wintner.