

Towards a Dynamic Linkage of Example-Based and Rule-Based Machine Translation

Michael **Carl** (IAI), Leonid L. **Iomdin** (IPPI), and Oliver **Streiter** (IAI)

IAI
Institute for Applied Information Sciences
Martin-Luther Str. 14
66111 Saarbrücken, Germany
`{oliver,carl}@iai.uni-sb.de`

IPPI
Institute for Information Transmission
Problems
Bol'shoi Karetny Pereulok 19,
Moscow, 101447, Russia
`iomdin@ippi.ras.ru`

1 Introduction

As a result of a long-standing co-operation between IAI and IPPI in the area of MT (Machine Translation), IAI and IPPI started to develop and implement an advanced plug-in software module, a **Case-Based Analysis and Generation Module (CBAG)**, which serves as a front-end for conventional rule-based machine translation systems (RBMT). The principal idea of CBAG is to combine the advantages of two machine translation ideologies: RBMT, on the one hand, and Translation Memories (TM), on the other hand, which, in our opinion, will drastically improve the quality of machine translation, ensure a stronger performance, and contribute to better maintainability and adaptability of the MT systems involved.

An experimental and methodological objective of this activity is to investigate the consequences of the introduction of an application-oriented CBAG component into a theoretically-based MT paradigm and to determine exactly what kind of linguistic entities (syntactic constructions, lexicographic types, collocations etc.) to be translated are suitable for simple uniform processing without entailing additional translation errors. We believe that an extensive research in this area will contribute to a better understanding of translation as a sort of human activity and help to optimise the general paradigm of machine translation (by converging to a model of the human activity).

The paper is structured as follows. In the next section, we shall outline the need for dynamic linkage of different MT paradigms. The third section discusses the benefits of such a linkage. In the fourth section, we propose adequate strategies and solutions to achieve this purpose. The fifth section describes the overall system architecture.

2 Background and Problem Formulation

The expertise gained by the world's leading NLP producers over the last 20 years has demonstrated beyond any doubt that all of the individual approaches to the task of MT/NLP which so far have been resorted to have their strong and weak sides. It is very unlikely that an entirely new, "ideal" approach may be proposed and implemented on a sizeable scale in the foreseeable future. It is our firm belief that considerable progress in the field can only be achieved by combining the strong sides of different approaches.

Versatile NLP systems for a variety of languages are already available: some of them are commercial products to be found on the market, others have been primarily designed for scientific purposes. NLP systems currently include terminological data banks, mono- and multilingual on-line dictionaries, Translation Memories, computer-assisted translation systems, and “classical“ (=automatic) MT systems. Until recently, most of these systems have been “island solutions“ which only offer one application mode each. In the best case they could be integrated with a word processor or a generic toolbox. In such a scenario one may have access to a dictionary tool in one window, to a spelling checker in another window etc. Logically, the components remain separated, and more often than not they are based on different linguistic sources, strategies and approaches. With competing solutions to one problem (e.g. Machine Translation vs. Electronic Dictionary vs. Translation Memory) the user has to decide case by case when to use which option, without any chance of making them interact with each other. Attempts to run different translation engines in parallel (e.g. Pangloss, see Brown 1996) actually offer little help. On the other hand, it has been shown that an integration of different MT approaches may yield better results than those achieved by an individual system. An instructive example is the experience of Verbmobil where the exploitation of complementary strengths of various MT approaches in one framework (deep analysis vs. shallow dialogue-act based approach vs. simple translation memory technology) improves the performance of the system (see e.g. Nübel 1997). This fact corroborates the opinion that the current situation must and can be changed.

2.1. Converse Machine Translation Strategies

Under the label of Machine Translation very different strategies and approaches have been suggested. In a sense, RBMT on the one hand and TM on the other hand represent diametrically opposed approaches.

None of the approaches has so far given a satisfactory response to the world's increasing need for automatic translation. The matter is that both have, beside their obvious advantages, a number of serious drawbacks. To list but a few, a TM, even a very large one, is unlikely to translate correctly a completely new sentence, let alone a new text. In their turn, RBMT Systems generally do not learn (i.e. do not store translation results to be re-used later) and are difficult to adapt to new domains. In a way, both approaches are cumbersome and lack flexibility: they can hardly be expected to switch strategies in compliance with the changing requirements.

Moreover, what happens to be an advantage in one approach may turn into a disadvantage in the other. So, if texts to be processed show great variance, the generally higher recall of RBMT systems is of advantage, whereas a high recall of a TM in this case may play a negative role as it is certain to generate extra noise. On the other hand, a TM may be preferred to an RBMT system if the scope of texts is limited because in this case they generally show greater reliability with respect to the produced translation. This means that the advantages must be combined depending on the user's needs and the sorts of texts that are to be translated.

2.2. Need for Dynamic Linkage

In an attempt to combine the advantages of RBMT and TM, an intermediate approach, Example-Based Machine Translation (EBMT) has recently been proposed (see e.g.

Furuse & Iida 1992, Nirenburg *et al.* 1994, Brown 1996, Nagao 1997). Significant progress in the field of MT, however, cannot be achieved by a static combination of two (or more) MT approaches. Rather, different approaches have to be combined dynamically, so that, with each sentence processed, the system makes full use of the modules' respective advantages in order to improve the overall performance of the whole system and to reduce the working load of each module. The research activities undertaken by IAI and IPPI represent a step towards the development of an MT system that **integrates example-based and rule-based techniques in one framework**. This paper describes an example-based component, CBAG, and its integration into two concrete rule-based machine translation systems, CAT2 and ETAP-3. The CBAG component is designed in such a way that a dynamic interaction with RBMT systems is possible.

3 Benefits of Linkage

The described MT system architecture fulfils to a high degree the requirements with which a good MT system must comply. An outline of these requirements and the way in which the two MT paradigms concerned - RBMT and TM - cope with them now is given below.

3.1. Recall and Precision

Recall and precision refer to the degree to which units of the source text match translation units stored in the system. Within TM tools currently available, the matching of these units relies on graphemic similarity when retrieving examples from the database. The advantage of such a matching algorithm is that it can detect similarities in the case of misspellings, slight variations in word order, or presence of certain types of specially listed inserted elements. However, in the case of substantial variations in word order or stronger morphological dissimilarities, including alternations and suppletivisms of the kind *good-better*, *mouse - mice* or *go - went*, the quality of the result deteriorates rapidly.

3.2. Adaptability

Adaptability refers to the ability of the system to embed the target side of a translation unit properly into its target context. In TMs, this ability is limited, especially if the stored cases cover sub-sentential chunks. A concatenation of a number of sub-sentence translations is not necessarily a valid translation. To give a simple example, if the German/English CB contains three cases: (1) *die Brille - the eyeglasses*, (2) *ist billiger - is cheaper*, (3) *in Rußland - in Russia*, which completely covers a sentence like *Die Brille ist billiger in Rußland*, this will still not be sufficient to obtain an adequate translation as the concatenation will yield an ungrammatical string **The eyeglasses is cheaper in Russia*. For a correct adaptation, target language peculiarities as agreement, control, word order etc. must be taken into account.

3.3. Coverage and Reliability

The coverage of an MT system is the extent to which various types of source texts can be translated into a target language more or less successfully (in other words, coverage may be viewed as the ratio of translation instances in which the system does not fail at some intermediate stage to the whole set of translation instances). Coverage

is sometimes opposed to reliability, which can be defined as the extent to which the MT system yields translations acceptable for the user.

Currently good coverage and high reliability happen to be mutually exclusive features of MT systems. This is due to the fact that ample coverage can only be achieved through complicated and hence often unreliable analysis and generation mechanisms typical of RBMT. On the other hand, TMs are the more reliable the longer the bulk of the matching examples are. However, with longer examples the coverage of a TM will decrease because longer examples are less likely to be found in the TM.

4. Strategies for Linkage

The goal is to improve the MT performance by optimizing the above features to the maximum extent possible. To this end, the CBAG module to be incorporated into the RBMT paradigm, even though it is generally based on the TM approach, includes important innovations, which are summarized below.

4.1 Recall and Precision

Higher recall and precision can be achieved if the measure of similarity in a TM is supplemented by relatively simple linguistic knowledge. The cases to be stored in the CB must include the results of **morphological analysis** rather than be mere surface strings. E.g. the Russian word *let* ('year', genitive plural) bears no surface string resemblance to *god* ('year', singular) and will be lost in a conventional TM. However, if it is analyzed morphologically and stored, we will be able to identify all expressions with *god/let* if it is the only difference they have. The recall of the tool will obviously be higher. Further, accurate morphological analysis of the CB cases will also augment the precision of the tool as compared to grapheme-based similarity approach. So, the same Russian word form *let* will no more be confused in the retrieval process with a graphematically similar but different word *leto* 'summer', which (practically) has no plural at all.

4.2. Adaptability and Reliability

In order to enhance the TM adaptability and the RBMT reliability, sequences of subsententially decomposed chunks are passed through the RBMT module, which will check and eventually correct agreement features, word order and the like. On the other hand, as the chunks obtained from CBAG remain "hermetically sealed" for the RBMT (i.e. the RBMT considers them as single nodes disregarding their internal structures), it is bound to operate faster and in a more robust way, if for no other reason than simply because it has fewer units to handle.

4.3. Recall and Coverage

Better recall and coverage is achieved by means of **case induction**, which will take place during the case compilation phase. Case induction is seen as a sort of generalization derived from specific cases. It is presumed that each case may be viewed as a set of features that can be divided into two subsets: **fixed features** which are case-specific (e.g. lexical instantiations) and **variable features** (e.g. information such as grammatical case and number) which are typical for a whole range of similar cases. Case induction disregards the fixed features while keeping track of the variable

ones. For instance, from French/English translation examples (1) and (2) below a generalized case (3) can be inferred. Case (3) will match a number of chunks such as *station de sport*, *station de taxi*, *station de métro*, *station de terre* etc. where the instantiations of the slot *X* are constrained by a set of features which can be shared with *ski*. These sequences would be translated in the absence of full matching cases into *sport station*, *taxi station*, *metro station* and *ground station*, respectively. Such generalizations are similar to parsing trees in a conventional NLP environment. The difference is that generalizations are generated from examples and no explicit grammar rules are specified.

case	French expression	English expression
(1)	<i>ski</i>	<i>ski</i>
(2)	<i>station de ski</i>	<i>ski station</i>
(3)	<i>station de X</i>	<i>X station</i>

4.4. Example

The introduction of fixed/variable features in the TM, even if it does not induce a generalized case, is expected to bring good results as compared to “unassisted” TM or RBMT operation.

Consider an English sentence which must be translated by an MT into German and/or Russian: *The United States of America has sent an official invitation to the Union of Soviet Socialist Republics and the Federal Republic of Germany*. It is quite obvious that such a sentence, simple as it is, is very hard to correctly translate within an RBMT system alone.

To mention just a few difficulties, in order to yield acceptable translations like ger. *Die Vereinigten Staaten von Amerika haben eine offizielle Einladung an die Union der sozialistischen Sowjetrepubliken und die Bundesrepublik Deutschland geschickt* or russ. *Soedinennye Štaty Ameriki napravili oficial’noe priglašenje Sojuzu Sovetskix Socialisticeskix Respublik i Federativnoj Respublike Germanii*, the system will have to disambiguate highly ambiguous nouns *union* and, especially, *states*, choose appropriate and far from evident equivalents for *Socialist* and *Federal* (respectively, *sozialistisch/socialisticeskij* rather than *Sozialist/socialist* and *Bundes-/federativnyj* rather than *föderalistisch/federal’nyj*), and replace an attributive construction with an appositive one (*Republic of Germany* - *(Bundes)republik Deutschland*). On the other hand, such a sentence is impossible to translate with the help of TM (unless the TM happens to store exactly this example) for the reasons that have already been stated.

However, in a combined system using fixed/variable feature opposition we can obtain a good result quite easily. So, if the database includes the following translation examples in which some of the nodes are tagged with fixed morphological markers (underlined> and other ones with variable markers which ensure that both example-internal agreement features and external agreement and control features are correctly defined, the translation of the source sentence by the RBMT is actually reduced to the translation of the sentence *X has sent an official invitation to Y and Z* - with *X*, *Y* and *Z* already translated!

case	English expression	German expression	Russian expression
(1)	<u>The United States of America</u>	die Vereinigten Staaten von <u>Amerika</u>	Soedinennye Štaty <u>Ameriki</u>
(2)	<u>The Union of Soviet Socialist Republics</u>	die Union der sozialistischen <u>Sowjetrepubliken</u>	Sojuz <u>Sovetskix Socialisticeskix Respublik</u>
(3)	<u>The Federal Republic of Germany</u>	die <u>Bundesrepublik Deutschland</u>	Federativnaja Respublika <u>Germanija</u>

5. Overall System Architecture

Two MT Systems, ETAP-3 (see e.g. Apresjan et.al. 1989, 1993) and CAT2 (Streiter 1996) play the role of RBMT. The morphological processor MPRO (Maas 1996) and the morphological module of the ETAP-3 system which ensure an almost 100% coverage of Russian, English and German will be used as the morphological analysis (MA) and the morphological generation (MG) modules. KURD (Carl and Schmidt-Wigger 1998) and EDGA (Carl 1998) serve as a basis for the example based component. The CBAG consists of three parts: The Case Base (CB), the Case Based Analysis module (CBA), and the Case Based Generation module (CBG). The interaction between the CBAG module and the RBMT system is shown in Fig. 1.

The CB contains a set of pre-compiled translation examples that we will henceforth refer to as **cases**. The content of the CB may range from isolated multiword expressions over formulas, proper names, or complete terminological data banks to phrases and whole sentences.

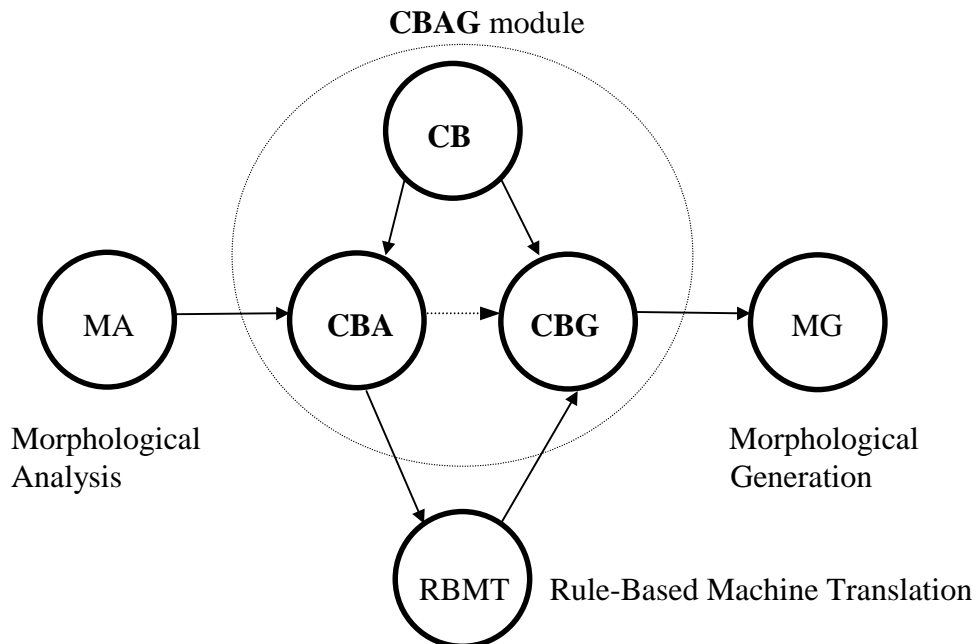


Fig. 1: Overall System Architecture

The CBA module matches the (morphologically analyzed) input text in the source language against the CB, whereby those parts of the input text that fit a CB case are reduced (\approx presented as single nodes), annotated and tagged with type information of the matching case. These matching parts of the input text will be referred to as

chunks. The task of the CBA module is thus to find all possible chunks in the input text.

CBA module operation may have one of the following three outcomes:

- (1) The entire input text is segmented into chunks. If this is the case, the (reduced) chunks need not pass through the whole RBMT process at all. They are sent directly to the CBG module (note the dotted line in Fig. 1), which re-generates the appropriate target language chunks using the assigned tags.
- (2) No chunks could be found in the input text. In this case, the source text is transmitted to the RBMT system to be processed as usual.
- (3) The input text matches the CB partially. In this case, both the identified chunks and the remaining unrecognized text elements are transmitted to the RBMT. The CBG module only re-generates those target language parts that have previously been reduced by CBA.

Since the output of the CBAG module is fully determined by the cases stored in the CB, the RBMT component is either simply assisted with the recognition of multiword expressions, proper names and the like, or the recognition and translation of terminology, or else partially or completely circumvented when large parts of the input text are matched in the CB. This means that an MT System in such a configuration operates in an adaptive manner, adjusting itself to the data which the user enters into the CB and the texts encountered: while a complete match of cases in a sentence converts the system into a TM, in the next sentence the system may return to a purely rule-based treatment, or combine the two approaches.

The design of CBAG constituents is discussed in more detail below.

5.1. Case Base

The Case Base module includes (1) the Case Base proper and (2) the Case Base Compilation module (CBC). The task of the latter is to create a high-quality CB from a set of translation examples and later to maintain and update it. The CBC module, which is of primary importance for the whole system, will operate as follows.

First, the morphologically analyzed translation examples are passed to the **Example Disambiguation and Labelling** sub-module (EL), which filters out ambiguous examples as well as those which cannot be tagged properly so that only unambiguous and appropriately labelled cases can be sent to the CB. Further, the disambiguated cases are transmitted to the **Case Induction** sub-module (CI). EDGA serves as an induction module, with the aim to create Case Generalizations. During the compilation of the CB new generalized Cases will be created from the Cases already present in the CB. Through such “learning” of generalized cases, the coverage of phenomena for which the RBMT system will not need to develop special treatment procedures can be substantially enlarged.

All cases are sent to the **Example Chunking** sub-module (EC), which decomposes the examples into a sequence of chunks using the cases already stored in the CB. The resulting sequence of chunks is then passed to the **Example Reduction** sub-module (ER) and, according to certain criteria, which will have to be elaborated in detail, stored in the CB. The general arrangement of the CB and CBC module is shown in Fig. 2.

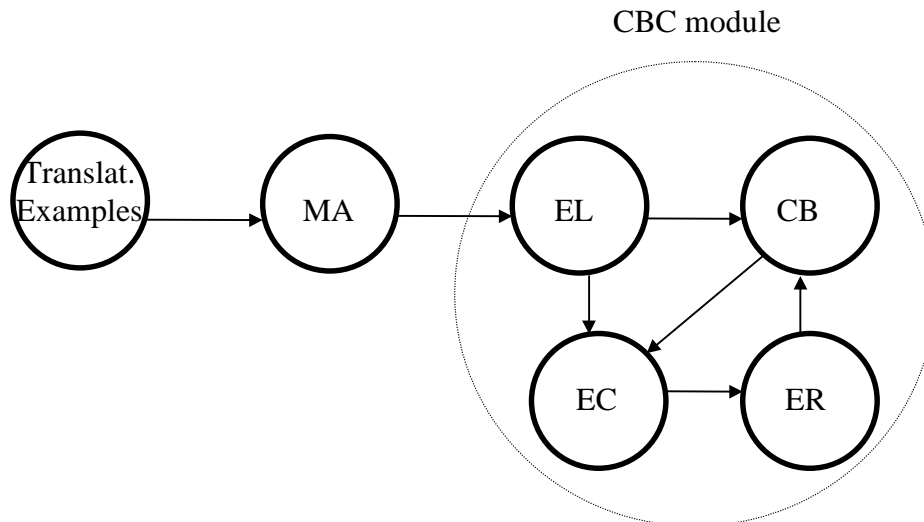


Fig. 2. Case Base and Case Base Compilation module

The CB is a resource to be shared by CBA and the CBG modules. In the former module, the CB will serve as a basis for the decomposition and reduction of the source text into a number of chunks and in the CBG module as a basis for re-generation of the appropriate target language chunks from the reduced chunk labels.

5.2. Case Base Analysis Module

The operation of the CBA module consists of two sub-modules which operate according to a set of rules outlined above; (1) text chunking (TC) sub-module, which decomposes the rearranged text into a sequence of chunks using a matching algorithm that compares the text with the CB; (2) chunk reduction (CR) during which the obtained chunks are reduced and tagged.

The compilation of translation examples annotated with labels into the CB is carried out with the help of MPRO and KURD on the one hand (for CAT2) and the ETAP-3 morphology and KURD (for ETAP-3). During CB compilation, KURD is used to disambiguate the chunk to be stored whenever possible (e.g. chunks like *red wine* whose elements are ambiguous (Verb/Adj/Noun+Verb/Noun) will be disambiguated to Adj+Noun. and to collect variable features (e.g. grammatical case or number of noun phrases) in order to transmit them to the RBMT.

If the entire input text is segmented into a number of chunks equal to or bigger than the entities treated by the RBMT, these chunks need not pass through the RBMT at all and the Chunk labels are directly passed to the Chunk Generator. If the text is segmented into chunks smaller than the entities treated by the RBMT, all chunks are passed through the RBMT which will generate appropriate variable features of the reduced chunks when necessary and supply translations for the unreduced chunks.

5.3. Case Base Generation Module.

The CBG module consists of two sub-modules. The Chunk Generator (CG) re-generates the parts of the target texts that have previously been reduced by the chunking mechanism of the CBA module. The Chunk Merger (CM) gathers together the variable features of the chunks as provided by the RBMT and the fixed features

provided by the CB thus creating objects which are ready to be sent to the Morphological Generation module (MG).

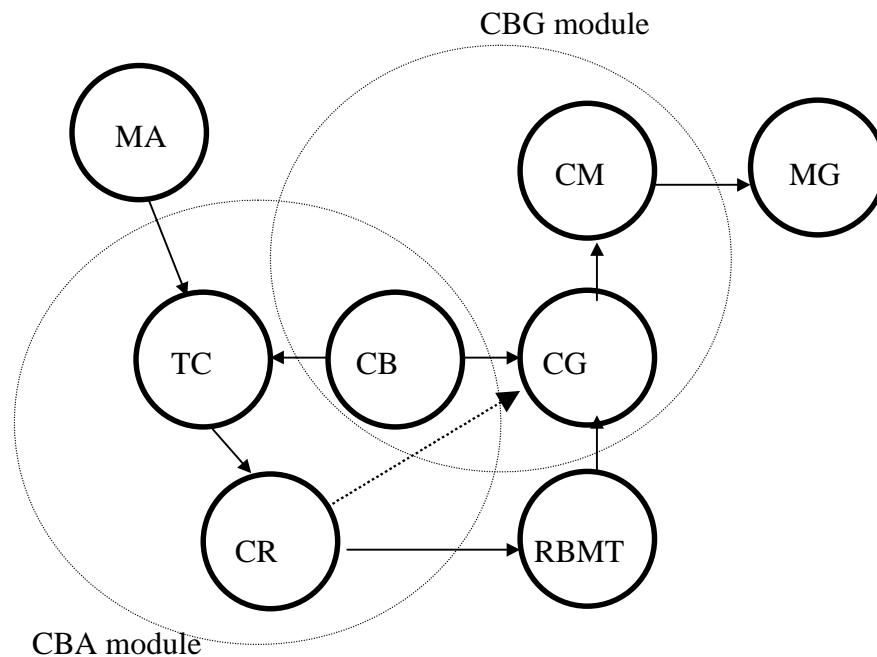


Fig. 3. Layout of the CBA and the CBG modules

The generation of texts from CB entries will be done with the help of MPRO and KURD (for CAT2) and the ETAP-3 morphology and KURD (for ETAP-3). KURD rules are employed to re-distribute values of grammatical features in accordance with the descriptions found in the CB, including the Case Labels.

6. Summary and Further Outlook

The paper described the development and implementation of an advanced NLP application called Case-Based Analysis and Generation Module (CBAG), which is expected to operate within a conventional rule-based machine translation system (RBMT) and drastically improve the performance of the latter. The main idea under the CBAG module is to introduce into mainstream machine translation paradigms, which are based on sophisticated models of language and in-depth linguistic research, a significant share of human translation experience accumulated in Translation Memories (TMs), which are, after all, relatively simple but very large and accurate collections of bilingual texts.

While most of the described architecture has already been implemented, the evaluation and fine-tuning of the described components are still under discussion. Expected results of this activity are:

- 1) The efficiency of a CBAG module integrated into conventional MT systems will be determined. It will become clear to which extent the performance and translation quality can be improved by this measure.
- 2) It will become clear what types of word combinations, or chunks, if introduced into a Case base, have a positive and sizeable effect on the translation quality and performance.

- 3) It will be experimentally found out what induction mechanisms can be used to extend the CB without creating additional noise. It is known that while rule-based CB extensions are a conservative means that does not deteriorate the translation quality, free induction is more powerful but may increase the risk of producing bad translations. A reasonable compromise may only be achieved experimentally.

References

- Apresjan et al. 1989: Jurij Apresjan, Igor Boguslavsky, Leonid Iomdin et al. *Lingvisticskoe obespechenie sistemy ETAP-2. (The linguistics of the ETAP-2 MT system.)* Moskva, Nauka.
- Apresjan et al. 1993: Jurij Apresjan, Igor Boguslavsky, Leonid Iomdin et al. . *Le système de traduction automatique "ETAP"*. In: *La Traductique*. P.Bouillon et A.Clas, ed. Montréal, Les Presses de l'Université de Montréal,-AUPELF/UREF, 1993, p. 377-391.
- Brown 1996: D. Ralf Brown. *Example-Based Machine Translation in the Pangloss System*. Proceedings of the COLING-96, Vol. 1, pp. 169-174., Copenhagen, 1996.
- Carl 1998: Michael Carl. *A Constructivist Approach to Machine Translation*. International Conference on New Methods in Language Processing (NeMLaP) 98, pp. 247-256, Sydney, 1998.
- Carl & Schmidt-Wigger 1998: Michael Carl and Antje Schmidt-Wigger. *Shallow Post Morphological Processing with KURD. Translation*. International Conference on New Methods in Language Processing (NeMLaP) 98, pp.257-265, Sydney, 1998.
- Furuse & Iida 1992: O. Furuse, O. and H. Iida. *An Example-Based Method for Transfer-Driven Machine Translation*. The Third International Conference on Theoretical and Methodological Issues, Empiristic vs. Rationalist Methods in MT. Montréal, 1992.
- Maas 1996: Heinz-Dieter Maas. *MPRO - Ein System zur Analyse und Synthese deutscher Wörter*. In: Roland Hausser, ed., *Linguistische Verifikation, Sprache und Information*. Max Niemeyer Verlag, Tübingen 1996.
- Nagao 1997: Makoto Nagao. *Machine Translation through Language Understanding*. Proceedings of MT Summit VI, pp. 41-49, San Diego, 1997.
- Nirenburg *et al.* 1994: Sergei Nirenburg, Stephen Beale and Constantine Domashnev. *A Full-Text Experiment in Example-Based Machine Translation*. International Conference on New Methods in Language Processing (NeMLaP) 94, pp. 78-87, Manchester, 1994.
- Nübel 1997: Rita Nübel. *End-to-End evaluation in VERBMOBIL I*. Proceedings of MT Summit VI, San Diego, pp. 232-240, 1997.
- Streiter 1996: Oliver Streiter. *Linguistic Modeling for Multilingual Machine Translation*, Shaker Verlag, Aachen.