[from: *Survey of the state of the art in human language technology*, ed. Giovanni Battista Varile & Antonio Zampolli (Pisa: Giardini, 1997); pp. 257-261]

8.4 Machine-aided Human Translation

Christian Boitet
Université Joseph Fourier, Grenoble, France

Section 8.3 has covered Machine Translation (MT), where translation proper is performed by a computer, even if the human helps by preediting, postediting, or answering questions to disambiguate the source text. In computer-aided translation, or more precisely Machine-Aided Human Translation (MAHT), by contrast, translation is performed by a human, and the computer offers supporting tools.

8.4.1 State of the Art

We can distinguish three types of MAHT systems, corresponding to three types of users, and offering different sets of functionalities.

Specific Software Environments Designed for Professional Translators Working in Teams

Existing products now are those of Trados (MultiTerm), IBM (Translation Manager), and SITE-EuroLang (EuroLang Optimizer). They are available on PC/Windows, PS/OS2, or Unix-based workstations.

The intended users are competent translators working in teams and linked through a local network. Each translator's workstation offers tools to:

access a bilingual terminology.
access a *translation memory*.
submit parts ot the text to an MT server.

These tools have to be completely integrated in the text processor. The software automatically analyzes the source text, and attaches keyboard shortcuts to the terms and sentences found in the terminogical data base and in the translation memory. One very important design decision is whether to offer a specific text processor, as in IBM's Translation Manager, or whether to use directly one or more text processors produced by third parties, as in EuroLang Optimizer.

The server supports tools to:

manage the common multilingual lexical data base (MLDB), often a multilingual terminological data base (MTDB), and the common translation memory, where previous translations are recorded. Here, concurrent access and strict validation procedures are crucial.
manage the translation tasks (not always offered).

Let us take the case of the most recent product, EuroLang Optimizer. One instance is available on Sun workstations under Unix. The server uses a standard DBMS (data base management system) (Oracle or Sybase) to support the terminological data base and the translation memory. The translator's workstations use Interleaf or Framemaker as text processors, while their data base functions are degraded versions of those of the servers, and are implemented directly in C++. In the other instance, the server runs on a PC under Windows NT, again with Oracle or Sybase, while the translator's workstations use Word 6 on PCs under Windows 3. Source languages currently include English, French, German, Italian and Spanish. There are 17 target languages (almost all languages written with the Latin character set).

When a document has to be translated, it is preprocessed on the server, and sent to a translator's workstation with an associated *kit*, which contains the corresponding subsets of the dictionary and of the translation memory, as well as (optionally) translation proposals coming from a batch MT system. MAHT-related functionalities are accessible through a supplementary menu (in the case of Word 6) and keyboard shortcuts dynamically associated with terms or full sentences. The translator may enrich the kit's lexicon. When translation is completed, the document is sent back to the server with its updated kit. On the server, the new translation pairs are added to the translation memory, and updates or additions to the dictionary are handled by the (human) manager of the MTDB. The overall productivity of the translators is said to be increased by up to 30% or 40%.

Environments for Independent Professional Translators

These environments are usually less powerful, quite cheaper, and callable from all or at least many commercial text and document processors. This is because free lance translators are usually required to deliver their translations in the same formats as the source documents, and those vary from one customer to the next.

As far as dictionaries are concerned, the situation is different from the preceding case. There is no central MLDB to manage, but it is very important for independent translators to be able to easily create, access, modify, export and import terminological files.

Examples are Mercury/Termex (Melby, 1982) by LinguaTech, a resident program for PCs, and WinTool (Winsoft, 1987), a desk accessory for Macintoshes. In 1992, MicroMATER, an SGML-based standard for PC-oriented terminological dictionaries, has been adopted in relation with ongoing efforts to devise standards for the encoding of more complex dictionary structures within the TEI initiative and in cooperation with InfoTerm (Vienna) and other organizations working on terminology.

Tools for Occasional Translators

An occasional translator may be competent in both languages, or only in the source language! As a matter of fact, there exist tools to help monolinguals produce parametrizable *canned* text in two languages. For example, Ambassador by Language Engineering runs on Macintosh and PC, is available in English-Japanese, English-French, English-Spanish and French-Japanese, and offers about 200 *templates* of letters and forms, and 450 textual *forms* (of sentence or paragraph size).

In the other context, the translator is at least bilingual, but is not a professional, and does not necessarily translate into his native tongue. Even if s/he does, s/he often does not know certain specific terms s/he has learned in the source language (take for example English-Malay or French-Arabic). Tools for bilinguals, such as SISKEP (Tong, 1987), are designed for such users. All are implemented on micros.

These tools offer different functionalities from those for professionals:

There is no translation memory.

The dictionaries must contain general terms, and there are usually three dictionary levels: personal and temporary terms, terminology, general vocabulary.

There are aids concerning the target language (thesaurus, conjugator, style checker, etc.).

Again, it is possible to propose a specific editor, with filters to and from standard word processors, as is done in SISKEP, or to interface the tools directly with one or several word processors. That second course was impractical until a recent past, because developers had to obtain access to the source code of the text processors. This has changed since 1991, when Apple launched version 7 of MacOS, which offers the possibility to let applications communicate through special *events*. The PC world is following with Windows.

8.4.2 Limitations in Current Technology

Serious problems in current technology concern the unavailability of truly multilingual support tools, the engineering of multilingual lexical data bases, the sacred character of the source text, and the limitation to handling only one language pair at a time.

Unavailability of Truly Multilingual Support Tools

MacOS 7.1, available since mid-1992, is still the only operating system supporting any number of writing systems at the same time. With a text processor based on Apple's Script Manager, such as WinText, it is possible to include English, Arabic, Chinese, Japanese, Thai, Russian, etc., in the same document, and to use the writing system as a distinctive feature in search-and-replace actions, or for checking the spelling or the grammar. But, in practice, the size of the OS grows considerably, because it is necessary to include a variety of fonts and input methods. With the languages above, MacOS 7.1 takes 4 to 5 Mbytes of RAM. Input methods and fonts must also often be purchased from third parties.

For other environments, the situation is still very unsatisfactory. At best, it is possible to find localized versions, which handle one *exotic* writing system besides the English one.

Engineering of Multilingual Lexical Data Bases

The multilingual lexical data bases (MLDB) found on MAHT servers are often nothing more than collections of bilingual dictionaries. In the case of terminology proper, MTDBs do exist, but are not yet integrated with MAHT environments. Such MTDBs include, for example, EuroDicautom at the EU (European Commission), Aquila by SITE-Sonovision, and MultiTerm by Trados. In the current state of EuroLang Optimizer, the MTDBs are planned to be monosource and multitarget, but are still bilingual, although the same company continues to market the fully multilingual Aquila on PC LANs.

As far as MLTBs are concerned, then, the problems concern more the management of the data bases than their design. That is because the MTDBs have to evolve constantly, taking into account possibly contradictory or incomplete contributions by many translators. In the case of MLDBs of general terms, there are still many design problems, and available solutions, such as that of EDR in Tokyo, are still too heavy to be used in MAHT systems.

“Sacred” Character of the Source Text and Limitation to Handling One Language Pair at a Time

Very often, translation is more difficult than it should be because the source text is not well written. If translation has to be performed into several languages, which is often the case, for example for technical manuals, it would make sense to prepare the source text, possibly annotating or rewriting parts of it. That possibility is however not offered in current MAHT systems, and the source texts remain *sacred*.

8.4.3 Future Directions

Current tools will no doubt be improved, in terms of speed, ergonomy and functionalities. Key research issues concern ergonomy, progress in Example-Based MT (EBMT), and integration with Dialogue-Based MT (DBMT).

Ergonomy

It must be realized that accessing large MLDBs and translation memories are very computer intensive operations. To identify complex terms requires full morphological analysis and partial syntactic analysis. Matching a sentence against a large set of sentences and producing a meaningful set of exact or *near* matches is not feasible in real time. The current answers to that problem is to preprocess the documents on a server (or on the workstations, in the background), or, in the case of PC-oriented stand-alone tools for occasional translators, where real time behavior is required, to simplify the morphological analysis and to suppress the translation memory.

The increase of computing power and the object orientation of future operating systems should make it possible to drastically improve the ergonomy and power of MAHT tools, by searching the terminological data base and the translation memory in the background, and dynamically updating MAHT suggestions for the current part of the document being translated, and possibly modified in the source form. These suggestions might appear in MAHT windows logically attached to the windows of the main applications (text processor, spreadsheet, etc.), or, if tighter integration is possible, in its application windows themselves. The main point here is that it would not be necessary to modify the code of the main applications.

Progress in EBMT

Example-Based MT (EBMT) goes one step further than the retrieval of identical or similar sentences. It aims at producing translation proposals by combining the translations of similar chunks of texts making up the sentence and previously identified as possible translation units in the translation memory. It is not yet clear whether the intensive efforts going into that direction will succeed to the point where EBMT could be included in MAHT tools in a cost-effective way.

References

Melby, A. K. (1982). Multi-level translation aids in a distributed system. In *Proceedings of the 9th International Conference on Computational Linguistics*, volume 1 of *Ling. series 47*, pages 215-220, Prague. ACL.

Tong, L. C. (1987). The engineering of a translator workstation. *Computers and Translation*, 2(4):263-273.

Winsoft (1987). *Manuel d'utilisation de WinTool*. Winsoft Inc., Grenoble. Version 1.1.