[From: *Unity in diversity*? ed. L.Bowker et al., 1998]

Translation Memories: Insights and Prospects

MATTHIAS HEYN
TRADOS Benelux S.A., Brussels, Belgium

Translation memory systems are a type of computer-aided translation tool that allow previous translations to be recycled in the course of new translation jobs. The use of this technology in the software localization industry has received some attention, but relatively little is known about the growing body of users outside the software sector. And while the basic principle of translation memories is easily understood, state-of-art interfaces to such systems often belie the complexity of the technology beneath. This article aims to give an overview of the many types of user of translation memories, and to link user profiles with different functional extensions of the technology. Some of the more technical aspects of translation memories are then discussed. Finally, we flag a number of issues that are beginning to emerge with the growing use of translation memories and that affect translators and technical writers alike.

Introduction

In the past, automation of the professional translation process was usually associated with the use of machine translation (MT), but the situation has changed significantly in the last few years. Today, the keywords are computer-aided translation **tools (CAT tools)** and, especially, **translation memories.** O'Brien (this volume) explains the basic concepts associated with CAT tools. Such tools, in most cases the integration of several functions in one workbench, are becoming standard in professional translation. CAT tools are now used in almost every type of translation work: political, administrative, technical, advertising and biographical, to name just a few.

Whereas the general idea of a translation memory is fairly simple, the practical realization of a functioning product is rather complex. This has mainly to do with the subtasks that such a system has to perform. Translation memories involve many aspects of information science and linguistics, including database design, retrieval technology, mapping of complex data and text structures, client-server architecture, networking, support of language-dependent phenomena (character sets, tokenization, morphology, syntax), and software ergonomics. They represent an interesting type of application, one which appears to users as a rather simple interface, but which has underneath a very complex internal functioning.

Up to now, little attention has been paid to the needs of the various

users of CAT tools. We can distinguish between a kernel set of functions in a translation memory and user-specific functional extensions. As the application area of translation memories continues to broaden, the functional extensions of such systems become more and more varied. In what follows, we first identify the various factors that make the use of CAT tools attractive, and we describe different user profiles and needs. We then discuss technical aspects of CAT tools against this background. We conclude by addressing briefly a number of emerging non-technical issues in the area of CAT tools, as well as the prospects for future development.

The benefits of using CAT tools

Three benefits of CAT tools are usually cited by tool vendors (and see also O'Brien's (this volume) section on 'advantages for the client'): large quantities of texts can be translated faster (Quantity Argument); the quality of translation is increased (Quality Argument); and subsequent similar translation projects can benefit from earlier work (Re-usability Argument). In what follows, we elaborate on these rather general statements and describe in more detail some factors that play a major role in the application of CAT tools. These factors can then be used to distinguish between different user needs.

By their very nature, translation memories find their main application in the translation of repetitive text material. It is important to distinguish between internal repetitions in a document itself and external repetitions where the repetitions are inherent to a family of documents, as happens with updated translations. We will call this the **repetition factor.**

Translation memories enforce greater consistency in translation especially when they are integrated with a terminology database system. We will call this the **consistency factor.** Every translation unit can be accompanied by several types of information, for instance: creation user, creation time, update user, update time, subject code, notes, etc. This leads to an improvement in the quality of translation because revised and approved wordings are re-used. It is like using a translation from an authorized reference, and, as such, leads to standardization across translations. This factor will be called the **reference factor.**

From a linguistic point of view, a translation memory can be described as a bilingual parallel corpus. In the case of systems that allow more than one source or target language, we can speak of multilingual parallel corpora. Such corpora can be used to retrieve a translation unit, by searching for one or several keywords. This function is commonly referred to as **concordancing,** more precisely bilingual concordancing. Translation memories can be seen as a rich source of implicit terminology (in contrast to the explicit terminology stored in term banks). In this sense,

translation memories can, and do, compete with term banks. This factor will be called the **concordance factor.**

Terminology recognition, that is the automatic searching in an associated term bank for terminology in a source translation unit, plays a key role in CAT tools. Terminology recognition should not be confused with terminology extraction, which means the automatic extraction of terminology from text material. Terminology recognition obviates the need for manual searches in databases; the system automatically draws the translator's attention to the relevant terms. CAT tool users thus benefit in two ways: they can keep track of specialized terms; and the retrieval of terminology can be manual or automatic. This factor will be called the **terminology factor.**

CAT tools can create resources automatically in three ways: firstly by creating a translation memory out of existing parallel texts in a process known as sentence alignment; secondly, by creating of a list of term candidates in one language to be introduced into the term bank system (monolingual terminology extraction); and thirdly, by creating a list of term-pair candidates from source and target texts to be introduced into a term bank system (word alignment or bilingual terminology extraction). We will refer to this factor as the **resource creation factor.**

Profile of CAT tool users

As has already been mentioned, the general market for CAT tools is broadening. In particular in countries where there is more than one national language or where translation costs are high, there is greater acceptance of software intended to rationalize the translation process. At the level of individual industries, it is well known that CAT tools have been used extensively in software localization (see O'Brien, this volume), but this is not the only sector where CAT tools find application: international marketing strategies and product liability laws that call for proper localized documentation for targeted markets mean that such tools are now in use in areas as diverse as the pharmaceutical and aeronautics industries. While the repetition and consistency factors are of particular importance in localization and other industries, banks, insurance companies and legal firms tend to place special emphasis on the terminology, reference and consistency factors. In military applications, there is the added requirement of document confidentiality, which has implications for the way in which translation units can be stored. In the multimedia sector, the terminology and concordancing factors, as well as the ability to handle HTML documents, are of utmost importance. Terminology is particularly important in the context of European and other international organizations, but traditional approaches to term bank creation and maintenance are complex and very costly. Here the concordance factor, in

effect a by-product of the use of CAT tools, comes into play: during a test phase of Trados's *Translator's Workbench for Windows* at the European Commission it turned out that concordance access to a translation memory with only 28,000 translation units was in many cases more helpful for retrieving terminology than access to Eurodicautom, the world's largest multilingual term bank, with over 1,300,000 entries. In all international institutions, but especially those with standardized documents, the repetition factor plays an important role, but subphrase repetition, i.e. repetition within a translation unit, remains a problem for commercial translation memory systems.

Translation agencies that handle several clients are highly dependent on their clients when it comes to the format and type of document they have to translate. In some cases, current CAT tools do not yet support the format in question; in other cases, documents are not even available in machine readable form. Whereas in other areas the CAT tool can be 'tuned' to the text type in question, in translation agencies CAT tools must be more flexible. The management of CAT tools that this entails has led in the recent past to a new professional profile in translation agencies, that of the IT Manager. In the case of agencies that do not have in-house revisers, functions like pretranslation and off-line updating are required of a CAT tool, and users of term banks must have access to printing or electronic publishing facilities.

Freelance translators, traditionally conservative when it comes to capital investment and thus less likely to use CAT tools, will find over the coming years that they will have to work with documents that have been pre-processed using these tools. They may even be temporarily forced into the use of a CAT tool by their work suppliers.

Finally, two other user groups may begin to emerge in the near future: terminologists (primarily interested in the concordance and advanced resource creation factors), and non-professional users.

Technical issues in translation memories

A translation memory is a database that stores translation units. This simple definition is complicated by a number of issues that are discussed below, starting with the thorny issue of similarity.

Coping with similarity

When are two sentences similar? This is a very tricky question. There could be misspellings, differences in the formatting, differences in the use of punctuation marks, morphosyntactic or syntactic differences, or differences in embedded elements such as index-markers. For a human being it takes only a short time to say that two sentences are similar.

Spotting similarity using a computer, however, is another story. Certainly, classical computation based on binary oppositions cannot help us.

Computation that does not rely on Boolean binary logic has traditionally been called **fuzzy.** The problem with this term though, is that (in non-mathematical contexts at least) it is used very vaguely. Modern computer science does, however, offer some workable solutions to similarity problems using fuzzy processing. These approaches include the use of neural networks and sparsely coded matrices. Whereas the first generation of the Trados translation memory system, for example, was based on a classical binary approach, and (linguistically motivated) substring-operations on classical database indices, the current generation employs sparsely coded matrices. The advantages are obvious: phenomena like misspellings and complicated syntactic deviations are now manageable and access time has been reduced significantly. Once a suitable technique for error-tolerant retrieval was introduced, additional functions like concordancing became possible.

For the sake of simplicity, the term **fuzzy-match** is used in the CAT tools world to indicate the measure of similarity between two source translation units. It is important to understand that this is only a relative notion whereby a higher fuzzy-match value means more similarity.

Interactive access

When users are accessing a translation memory in interactive mode, acceptable response times are in the range of up to one second. Response time depends, naturally enough, on the power of the computer, but also on the size of the translation memory, the type and number of processes running in addition to translation memory access, and the time spent in exchanging translation unit information between the translation memory system and the front-end (word processor).

Translation memory size

The size of translation memories is a real problem in systems with traditional data-access, i.e., those that do not use error-tolerant retrieval technology. The standard solution here is to extract from the **master translation memory** a smaller **working translation memory** which is made up of all stored translation units whose source segments are similar to the text to be translated. This is done in two stages. First the source text is segmented into translation units, according to the segmentation strategies of the product. Then the translation units are retrieved from the translation memory and all matches above a certain threshold value are put into the working translation memory. The user then accesses this smaller translation memory interactively.

Working translation memories, however, have a number of disadvantages. For one, they necessitate an additional pre-processing phase. For another, the threshold similarity value is based on heuristics and users have no access to translation units below this value, even in cases where such units could be helpful. Also, if a translator changes the segmentation interactively, e.g. in case of segmentation errors, the system can no longer retrieve translation units. Interactivity is also compromised: the user is restricted to a buffer-translation memory and cannot work interactively on the real translation memory. This means that changes to the working translation memory cannot be accessed by other users. Finally, the changes to the working translation memory must be resynchronized with the original master translation memory once the translation process is complete. This is again an additional processing phase and working translation memories only make sense if a versatile update mechanism is available to carry out resynchronization.

In recent sparsely-coded-matrix based systems, real interactive work on 'big' master translation memories is possible. Big translation memories are typically in the order of 100,000 translation units, although memories in the range of 500,000 to 1,000,000 translation units are envisaged by the end of 1997. According to current research estimates, translation memories could be made up to 40% bigger without any increase in constant access times.

The ideal situation for all user groups is when a translation memory system is based on modern technology and at the same time allows for both the creation of a working translation memory and direct access to the master translation memory.

Additional processes

As already mentioned, response times can also degrade, sometimes significantly, if additional processes, such as term recognition or the passing of translation units to an attached machine translation system, are running on the system alongside the main translation memory process. In order to avoid response time degradation, additional processes can be controlled by the translator or else performed in the background, so that the translation memory process always has the highest priority.

Data exchange with the front end

Normally, data exchange between the word processor and translation memory system is sufficiently fast.

Batch Processing

Batch processes are those that are carried out in non-interactive mode.

The preparation of a working translation memory as discussed above is an example of such a process. Other batch processes are described below.

Pretranslation

Pretranslation can be carried out off-line. It involves the replacement of source text segments with all 100% matches and all fuzzy-matches up to a certain threshold, found in the translation memory. It may also involve the replacement of any source language terms detected by the terminology recognition process. For ergonomic reasons, the system should highlight the results of the pretranslation process using a different colour.

The advantages of pretranslation are that if a text contains many 100% matches, the translation can be performed much more quickly, since the translator can skip over the parts already translated, and that texts can be pre-processed and then be translated off-line by external users. As is the case with working translation memories, off-line translation requires sophisticated updating facilities in the translation memory system.

Pretranslation is a process that is needed by nearly all user groups, in particular, translation agencies supplying work to freelancers.

Repetition analysis

Repetition analysis is a process that compares a document with a translation memory and computes statistics regarding how many 100% matches, fuzzy-matches and internal repetitions are encountered. In addition, word counts and translation unit counts as well as the overall statistical distribution of items in a document can be output. Text segmentation must be sufficiently powerful to do proper word and translation unit segmentation and it must be able to cope with **placeables** correctly. Placeables are non-translatable items such as graphics and automatic field codes (automatic numbering, dates, etc.) or tags, which are normally not translated, but simply *placed* in the target translation unit by the translator.

Repetition analysis is playing an increasing role in the negotiation of prices for translation projects. This means word, translation unit, and repetition counts must be correct. Where the repetitive character of documents is not easily measured (e.g. for users at the institutional level), delta computing, which allows the similarity of a set of documents to be gauged, offers a way of making estimates objective.

Analysis of frequent occurrences of translation units

The detection of all translation units occurring more than a certain number of times in a document can be very helpful. A list of these source translation units can be translated in isolation, thereby pre-filling the translation memory with the highly repetitive parts of the document. This

is possible only if translation is feasible out of context, as is frequently the case for technical documentation. It may also be possible to export frequently occurring source translation units for which no matches above a certain threshold value can be found in the translation memory, to a machine translation system, in order to speed up the translation project.

Post-translation processes

Once translation has taken place, a final phase of revision and updating of the translation memory may be necessary. The translated text itself may also have to be cleaned up, especially in the case of source-preserving systems (see below).

Updating and revisions

The revision of translation memories is very important to all user groups, but especially when the reference and consistency factors play a major role. In interactive translation memory systems, updating is done automatically by accepting a translation unit from the user. Simply re-opening a translation unit allows revisions to be done easily. This is the ideal situation since the updated translation unit is immediately visible to all users of the translation memory system. Users should also be able to update translation memories without using the front-end. This can be achieved by concordancing and editing the concordance results, that is by editing the translation memory directly.

In non-interactive translation memory systems, and when using working translation memories, an explicit update has to take place in the form of a batch process when the translation project has been finished.

Source-preserving systems

A distinction should be made between systems that keep the source translation unit in the document in a hidden form and systems that do not. The first type of system, a **source-preserving system,** creates a bilingual document in which the original source translation units are hidden. Such documents themselves contain a translation memory, and giving one away is like giving away a translation memory. Thus, once a translation project has been completed, all source translation units are usually deleted from the translated document. This is normally done by an update procedure. Although both source-preserving and non-source preserving systems allow working translation memories to be updated, only source-preserving systems are flexible enough for off-line revision. In off-line revision, revisers have access to source and target translation units in a document without using the CAT tool. This means that global replace operations and other facilities offered by word processors can be used

completely independently of the translation memory system. In source-preserving systems, it is even possible to update a document from a translation memory. In cases where a translation memory is more up to date than a text, this can be very useful.

Search-engine and data storage

There is often confusion about the distinction between the search engine and data storage in translation memory systems. The search engine is responsible for the retrieval of similar translation units and the data storage is responsible for the physical storage of translation units. Physical storage can be done with any kind of database system. The architecture of the search engine is a more important factor in the performance of a translation memory system. As has already been pointed out, sparsely coded matrix approaches (a subtype of neural network) are currently state of the art, and significant improvements are not expected from traditional search engines, characterized in the case of translation memories by linguistically motivated string operations on data-storage indices.

Networking

The network capabilities of current translation memory systems also give rise to misunderstandings. Ideally, the following client-server scenario would prevail: if a large number of users were searching for different source translation units at the same time, a translation memory server would provide them with the required set of target translation units almost in real-time. So far there is no such system on the market. The only solution to this problem given a client-server architecture is to create a temporary working translation memory for each user, which is then copied to the user's workstation.

File-sharing remains the only viable option for networked users. If a big group of users has to share a translation memory, a good choice is a system that fits major needs and allows development in the client-server direction.

Additional information stored in translation memory systems

To facilitate the interpretation of target translation units, they should be accompanied by additional formatting and administrative information. Administrative information must be user-definable: fixed field approaches are unacceptable. There must also be automatically maintained fields. Typically, these are fields such as creation date, creation user, change date, change user, used date, etc. Since these fields enlarge a translation memory, there should be a means of selecting or deselecting them. An automatically updated usage counter, which keeps track of the use of

translation units allows for subsequent reduction of a translation memory to all the translation units that have been used at least once over a certain period. Users should also be able to select target translation units on the basis of a subject field code. Other possible fields include those used to install and enforce security mechanisms.

Front-end integration

The term **front-end** refers to the application with which the translator controls the translation memory system. This is normally a standard word processor, but there are still some older systems on the market that come with their own editor. Such systems are, however, less than ideal, for a variety of reasons: firstly, the document has first to be converted into the internal editor format and later, after translation, it must be converted back into the original format. This can cause formatting errors and requires additional work to be done. Secondly, idiosyncratic editors are not as user-friendly as standard word processors. They are character based and often lack automatic reformatting capabilities, multilevel undo/redo operations, spelling checkers, thesauri, revision handling, auto-text entries, macro-languages and hyphenation facilities. Thirdly, translators are already familiar with their standard word processor, and it is a natural solution to integrate the translation memory into the environment translators are already familiar with. State-of-the-art translation memory systems are integrated into standard word processor systems like Microsoft Word for Windows or WordPerfect for Windows.

Integration into an existing word processor can be done in several ways: the first distinction that has to be drawn concerns the dependence of the translation memory system on the word processor environment. **Internal integration** means that the translation memory system is completely integrated into the word processor using only the means for displaying and manipulation that the word processor offers. **External integration** means that the translation memory runs as an application independent of the word processor using its own windows to display retrieval results and its own menus for the manipulation of the translation memory system.

Internal integration

The advantage of internal integration is that the translation memory system appears to end-users as a functional extension of the word processor. There are, however, several disadvantages associated with this approach: internal integration can use only the display facilities provided by the word processor. This means that there are clashes if a certain type of formatting is used within the document itself to mean one thing and by

the translation memory system to mean another. For this reason, internally integrated solutions often avoid direct display, which means that users have to open and close windows in order to consult system information. From an ergonomic point of view this is a major disadvantage. Internal integration also means a higher dependency on the word processor, which makes it more difficult - from a software engineering point of view - to integrate a translation memory system into new word processors re-using existing functionalities. Therefore, if internal integration is used, quick responses to new platforms or updates of word processors cannot be expected.

External integration

Externally integrated systems appear to users as applications in their own right. This has the disadvantage that users are forced to use a tool other than their tried-and-trusted word processor. On the other hand, translation memory systems integrate a number of functions, so that bundling all functions into one running application seems to be more natural than overburdening the already fully packed menu structures of modern word processors. External integration allows faster upgrades to new platforms, since only the part consisting of the communication with the word processor has to be re-implemented. This is an important point to be considered when purchasing a system. All in all, and especially from an ergonomic point of view, external integration seems preferable.

Indirect integration

In some cases the front-end used for the creation of documents seems rather complicated to translators. This is the case especially with desktop publishing systems such as FrameMaker, PageMaker, Quark XPress or Interleaf. It thus often makes sense to convert from the desktop publishing system to the standard word processor translators are familiar with. This type of integration is called **indirect integration.** Powerful conversion tools have recently been developed, which smooth the complex format provided by desktop publishing systems into a format consumable by translators.[1] The advantages of staying in the normal word processor environment outweigh the work involved in the two conversion steps.

Depth of integration

Systems differ with regard to the depth and sophistication of integration

[1] See especially the S-Tagger developed by itp-Ireland. This tool converts a FrameMaker file into a Microsoft Word treatable format, which translators can work on very easily.

they allow. As already mentioned, a translation memory system identifies a source translation unit and retrieves a similar or identical target translation unit. But *what* does the system identify as a source translation unit? A proper translation memory system should support all constructions possible in the word-processing system, for example, tables, footnotes, endnotes, field-codes, frames, columns, embedded objects, pictures, indices, and revision codes.

The segmentation capabilities of a translation memory system are also very important. Experience in computational linguistics has shown that segmentation is not at all a trivial task. There are ambiguities that cannot be resolved exactly (e.g. punctuation marks after numbers), language-dependent phenomena (e.g. semicolon in Greek, abbreviations in Finnish, word boundaries in the languages of the Far East), and document- and user-type dependent phenomena (e.g. treatment of tabulars, semicolon, etc.), all of which cause segmentation problems. The only way to overcome these problems is to allow users to define their own segmentation rules, as well as lists of abbreviations, ordinal followers, etc. Some segmentation errors are, however, unavoidable. Therefore a system should allow the interactive **shrinking and expanding** of source translation units in order to specify exactly the size of a translation unit. If this is not possible, users will soon become dissatisfied.

Automatic exchange of numbers and other invariable constructions is another useful function, especially in the area of banking, but users must be able to deactivate this function, if it does not apply to a certain document type.

Segmentation must also foresee a means of **exclusion.** This means that parts of the document can be marked so that they can be excluded by the translation memory system. This must be possible at the paragraph level (e.g. to exclude foreign language citations or programming language code) and at the character ievel (e.g. to exclude invariable elements like proper names in biographical documents or tags in tagged file formats).

Front-end independence

The storage of translation units in a translation memory should be independent of the front-end. This means that, in an extreme case, a user can translate part of a document using, for example, Microsoft Word and the rest within WordPerfect, operating on the same translation memory. This is an important feature for the use of translation memories if different front-ends are used by different users, the user is planning a change of front-end, or translation memories are exchanged between different user groups. In principle this feature is valuable for all users.

Front-end independence appears to be simple. Its technical realization, however, is complex, because the formatting conventions of different

front-ends have to be mapped into one single representation in a translation memory system. If a translation memory system offers front-end independence, this indicates highly sophisticated format management.

A special front-end: concordance access

As has been mentioned before, concordancing allows access to translation memories. In this sense concordancing is a front-end of its own, enabling tasks like terminology searches and the maintenance of translation memories. We can expect to see many improvements in this area, for example, the use of filters for concordancing, the displaying of selective parts of a translation memory via concordance windows, and concordance access to more than one translation memory. The next generation of the Translator's Workbench, for example, will allow browser-like access to translation memories. Concordance access is important for all users, but especially for large institutions and terminologists. Only modern systems with sparsely-coded matrix technology allow for concordancing.

Emerging Issues

Authoring

The use of translation memory technology can influence overall document production quite dramatically, often leading to a certain stream-lining in formatting. Unlike with machine translation, there is no need for a controlled language and the effect on overall text-flow organization is rather positive.

The way translators see themselves

Mastering a technology can significantly improve the way translators see themselves. The use of modern computer technology influences nearly all professions. Being able to use new technologies and especially translation memory technology represents an additional professional skill for translators, one that is already highly appreciated on the market.

Peep-hole translations

The existence of translation memory technology may also influence the way translators formulate texts. For example, since retrieved translation units normally require fewer changes if they do not contain anaphoric and cataphoric references, translators are tending to avoid the use of such devices. The effect is a more technical style, and sometimes a less readable text. In the end it is up to the translator to decide whether text cohesion should be compromised in order to facilitate the translation memory.

Translation rates

As has already been mentioned, the pricing of translations has been affected by translation memory systems. The localization industry already pays different rates for 100% matches, different types of fuzzy-matches and no-matches, and other user groups are likely to follow suit in the near future.

Copyright issues

As with term banks or dictionaries, translation memories can give rise to copyright problems. A translation memory can be extremely valuable as a source of terminology or a resource in retranslation. Ownership of a translation memory can guarantee an individual translator's independence. Translation memories are thus valuable resources whose monetary value is very difficult to estimate. Copyright problems arise when it is unclear to whom a translation memory belongs, the supplier of a translation service or the client who commissions that service. In many cases this is subject to negotiation.

Bilingual corpus collection

Translation memories are beginning to provide specialized multilingual corpus material that is superior to automatically aligned corpora on two counts: its quality, because texts are translated manually and proof-read by specialists; and its domain and text-type specificity. Given fast (error-tolerant) concordancing, bilingual corpora can also compete with term banks.

The future

Given that CAT tools providers are now seeking to support more languages, especially those of East Asia, we can expect to see developments involving UNICODE in the near future. Furthermore, although simplistic approaches to the retrieval of subsegments of translation units using pattern recognition are showing astonishingly good results, such approaches can be applied only to certain types of texts. Future developments here will involve the integration of more linguistic knowledge and will therefore be restricted in the range of supported languages and the quality of retrieval for individual languages. As has been indicated above, work is also ongoing in the area of interactive client-server architectures for translation memories. One final area where things are changing rapidly is that of access to terminological information over the Internet. First applications in this area, such as MultiTerm for the WEB have already been released.