Information Retrieval in Translation Memory Systems: Assessment of CurrentLimitations and Possibilities for Future Development

Lynne Bowker

School of Translation and Interpretation, University of Ottawa, P.O. Box 450, Station A, Ottawa, Ontario K1N 6N5, CANADA, Email: lbowker@uottawa.ca

Lynne Bowker holds a PhD in Language Engineering from the University of Manchester Institute of Science and Technology in the United Kingdom. She is currently an Assistant Professor at the School of Translation and Interpretation of the University of Ottawa, Canada, where she teaches and conducts research in the areas of translation technology, terminology and corpus linguistics. Her recent publications include Computer-Aided Translation Technology: A Practical Introduction (University of Ottawa Press, 2002) and Working with Specialized Language: A Practical Guide to Using Corpora (Routledge, 2002, co-authored with Jennifer Pearson).

L. Bowker (2002). **Information Retrieval in Translation Memory Systems: Assessment of CurrentLimitations and Possibilities for Future Development.** *Knowledge Organization, 29(3/4).* 198-203. 11 refs.

ABSTRACT: A translation memory system is a new type of human language technology (HLT) tool that is gaining popularity among translators. Such tools allow translators to store previously translated texts in a type of aligned bilingual database, and to recycle relevant parts of these texts when producing new translations. Currently, these tools retrieve information from the database using superficial character string matching, which often results in poor precision and recall. This paper explains how translation memory systems work, and it considers some possible ways for introducing more sophisticated information retrieval techniques into such systems by taking syntactic and semantic similarity into account. Some of the suggested techniques are inspired by those used in other areas of HLT, and some by techniques used in information science.

KEY WORDS: translation memory, computer-aided translation tools, information access

1. Introduction

It is clear that in this "information age," the volume of documentation that is being produced is increasing at a rapid rate. As more and more companies begin selling their products on the global market, there is a need for them to translate the accompanying documentation in very short turnaround times (Sprung, 2000). Products cannot be sold in foreign markets until translated documentation has been prepared, and each day that a product is not on the shelves means a loss of potential sales for a company. Many companies aim to achieve simultaneous shipment or "simship," which means that they want to make their product available in a variety of languages at the same time, rather than releasing the English-language version one month, then the French-language version three months later, followed by the German-language version six months later, and so forth. Document translation is often the last stage of preparing a product for the global market. Consequently, if a company is aiming for simship, translators are under enormous pressure to produce this documentation as quickly as possible, while continuing to maintain a

high standard of quality. To help them with this task, many translators are turning to a new type of human language technology (HLT) tool, known as translation memory (Bowker, 2002).

Translation memory tools are computer-aided tools, which means they are designed to help (rather than replace) human translators. Although they were conceived of as early as the 1970s (Melby, 1995), such tools have only been widely commercially available since the late 1990s. The aim of this paper is to evaluate the current and potential usefulness of these tools for allowing translators to access relevant information. The paper begins by explaining how translation memories work. It then goes on to assess some of their limitations, specifically with regard to information access and retrieval, and ends by considering possibilities for future developments that could help to optimize the usefulness of the information retrieved by these tools.

2. What are "translation memories"?

A translation memory is essentially a database that contains texts that have been previously translated. It is based on the principle of "recycling" previously translated documents – a translator should be able to re-use parts of texts that have been previously translated, and should never have to re-translate a portion of text that has already been translated (O'Brien, 1998). A translation memory can be created by a single translator, or it can be networked so that a group of translators can contribute to the same memory.

2.1 Data organization in a translation memory

The data contained in a translation memory are organized in a very precise way. There are two main types of texts stored in a translation memory: 1) source texts, which are the original texts in language A, and 2) target texts, which are the texts that have been translated into language B.

In an initial step, the translation memory tool divides each text into small units known as segments. These segments usually correspond to sentences or sentence-like units (e.g., these could include titles or headings, items in a bulleted list, cells in a table). The segments from the source texts are linked to their corresponding segments in the target texts. This process is called alignment, and an aligned pair of segments is known as a translation unit. Table 1 illustrates translation units that consist of English-language segments aligned with their corresponding French translations.

Translation Unit 1	EN: Please insert the diskette. FR: Insérez la disquette.
Translation Unit 2	EN: You need to double-click on an icon to open or start it. FR: Vous devez double-cliquer sur une icône pour l'ouvrir ou le démarrer.

Table 1. English and French segments are aligned to created translation units.

2.2 Working with a translation memory

When a translator receives a new text to translate s/he begins by opening this text in the translation memory environment. The translation memory tool proceeds to divide this new text into segments. Once this has been accomplished, the tool starts at the beginning of the document and compares each segment to the contents of the translation memory database. If it finds a segment that it "remembers" (i.e., a segment that matches one that has been previously translated and stored in the translation memory database), it retrieves the corresponding translation unit from the database and shows it to the translator, as illustrated in Table 2. Now the translator can refer to the previous translation and adopt or modify it for use in the new translation. This saves the translator time as s/he will not need to do as much research to come up with an appropriate translation.

Segment from new source text	This computer program is protected by copyright law and international treaties.
Matching translation unit retrieved from translation memory	EN: This computer program is protected by copyright law and international treaties. FR: Ce logiciel est protégé par les lois et les traités internationaux sur le droit d'auteur.

Table 2. An exact match located in a translation memory.

Of course, language is dynamic, which means that the same idea can be expressed in a number of different ways (e.g., 'The filename is invalid' / 'This file does not have a valid name'). Consequently, a translator cannot reasonably expect to find many exact lexical matches for previously translated segments in the translation memory. However, it is highly likely that there will be segments in a new source text that are similar to, but not exactly the same as, segments that are stored in the translation memory. For this reason, translation memory tools also employ a powerful fea-

ture known as fuzzy matching. As shown in Table 3, a fuzzy match is able to locate segments in the memory that are an approximate or partial match for the segment in the new source text. These types of matches are very useful for translators because at least part of the previous translation may be reusable.

Segment from new source text	The specified operation was interrupted by the system.
Fuzzy match retrieved from translation memory	EN: The operation was interrupted by the application. FR: L'opération a été interrompue par l'application.

Table 3. A fuzzy match retrieved from the translation memory (differences between new segment and previously translated segment are underlined).

When using fuzzy matching techniques, the translator can set the sensitivity threshold of the match; in other words, the translator can decide how similar the two segments must be in order for a translation unit to be retrieved and displayed. Setting the appropriate sensitivity threshold can actually be quite tricky: if the threshold is set too high (e.g., 95% similarity), then potentially useful matches may be overlooked (silence = poor recall), but if it is set too low (e.g., 30% similarity), then irrelevant segments may be erroneously retrieved (noise = poor precision).

3. Limitations of current matching techniques

Although translation memories are gaining popularity with translators, these tools do have a number of limitations. The principal limitation has to do with the way the matching is carried out. At the present time, the majority of translation memory systems employ matching techniques that are based solely on superficial character string matching. For instance, for two segments to qualify as an exact match, they must be identical in every way (e.g., in terms of spelling, inflection, punctuation). Table 4 provides some examples of stored segments that would not be retrieved as exact matches because they differ slightly from the new segment.

In each of the cases shown in Table 4, the differences between the new segment and those stored in the database would have very little impact on the translation of the segment. Rather, each of these would provide the translator with very useful information.

It is precisely to overcome these types of situations that fuzzy matching techniques were developed. Each of the examples shown in Table 4 could be retrieved by using the fuzzy matching feature; however, as previously mentioned, the sensitivity threshold must be carefully set to optimise both precision and recall. Fuzzy matching alone does not completely resolve this type of problem because even fuzzy matching is executed by means of superficial character string matching. This means, for instance, that "dish" is considered to be a closer match to "disc" than is "diskette" because there are fewer superficial differences between "dish" and "disc". This approach to matching can have an impact in terms of both noise and silence because segments that are superficially similar may not be semantically related (e.g., 'File the form' / 'Fill the dorm'), and segments that are semantically related may not be highly similar in terms of physical appearance (e.g., 'File the form' / 'He is refiling those forms'). A translator who is looking for an equivalent of a given segment would find the translation of a semantically-related segment to be more useful than that of a segment which bears only a superficial resemblance to the source text segment.

New segment to be matched	Store figure 1-1 on disk.
Not retrieved as an exact match because of a difference in spelling.	Store figure 1-1 on disc.
Not retrieved as an exact match because of a difference in punctuation.	Store figure 1.1 on disk.
Not retrieved as an exact match because of a difference in numerals.	Store figure 1-2 on disk.
Not retrieved as an exact match because of a difference in inflection.	Storing figure 1-1 on disk.

Table 4. Examples of segments that would not be retrieved as exact matches.

As pointed out by Rapp (2002), using character string similarities between segments facilitates implementation and allows for the construction of fast search engines; however, as outlined above, it may lead to poor search results since character string similarity does not necessarily mean semantic similarity. Translators certainly appreciate having access to fast tools, but this cannot come at the expense of high-quality results. In order for translation memories to be optimally useful, the search techniques will need to be modified to take into account syntactic and semantic information, such as inflection, derivation and synonymy.

4. Possibilities for future development

This section will explore some possibilities for augmenting the search techniques used by translation memory tools in order to allow them to retrieve

more meaningful information. Some of these techniques have been inspired by work carried out in other areas of human language technology (HLT), such as machine translation, while others have been inspired by research and applications in information science.

4.1 Part-of-speech tagging

One possibility for improving the syntactic processing capabilities of translation memories comes from work done by Rapp (2002) in the context of example-based machine translation (EBMT). This approach would involve annotating the texts in the translation memory database, as well as the new source text to be translated, with tags that indicate the part-of-speech of each word in the texts (Garside et al., 1997). There are software packages, known as part-of-speech taggers, which can do this automatically, and although they do not necessarily produce perfect results, they are typically able to tag texts with upwards of 97% accuracy.

Rapp suggests that in addition to comparing the character string similarities of the two segments, a system could also compute the syntactic similarity of the two segments by comparing the part-of-speech tags. If the two segments had similar tags, this could be considered an additional measure of overall similarity. It should be noted, however, that Rapp's work was done with English and German – two languages that are from the same family and that exhibit a high degree of structural similarity. While this approach may work in some cases, it would not work with all languages (e.g., French often uses noun phrases in places where English uses verb phrases (Vinay & Darbelnet, 1995)). Furthermore, while syntactic similarity may represent a stronger measure of overall segment similarity than character string matching, it is still not enough to guarantee semantic similarity.

Somers (2003) warns of another potential drawback to adding part-of-speech information to a translation memory, noting that currently, translation memory systems remain largely independent of the source language and wholly independent of the target language. If language-specific information were to be added, developers would need to create different matching engines for different languages; however, any resulting gains in match suitability may not be significant enough to merit this extra effort on the part of the developers.

4.2 Lemmatization

Another type of syntactic processing that could be used to enhance the retrieval capabilities of a translation memory system is related to lemmatization. A lemma is a sort of head word that is used to represent all related syntactic forms (e.g., the lemma 'go' includes the forms 'going', 'gone', 'went'). If the texts in a translation memory were lemmatized, it would be possible for the system to make connections between different forms of the word that have been inflected (e.g., conjugated or made plural) or derived (e.g., changed from a noun to a verb), as shown in the following example:

He manages to eliminate viruses successfully.
He managed a successful elimination of the virus.

Macklovitch and Russell (2000) emphasize that this type of matching capability is greatly needed for translation memory systems to be considered truly useful.

4.3 Controlled language and automatic paraphrasing

Another approach that can be integrated into translation memory systems is the use of controlled language. Controlled language has been successfully used in both machine translation and in information science (e.g., indexing). The basic idea behind controlled language is that only a restricted set of terms and syntactic constructions can be used (Nyberg et al., 2003). If all the texts used with a translation memory system were written in a type of controlled language, then the number of matches would be increased since there would be a high degree of character string similarity between the segments.

The main problem with this approach is that it is not always possible for the translators to control the style of the texts they receive. These texts come from a wide variety of clients, and these clients may or may not agree to produce their texts in a controlled language. Furthermore, controlled language is not necessarily suitable for all text types. It may be useful for technical or instructional texts, but it is not appropriate for journalistic or advertising texts.

A possible variation of this approach could be automatic paraphrasing, as described by Shimohata and Sumita (2002) in the context of machine translation. In this approach, a computer program is used to replace synonymous expressions with one standard expression. If this were to be applied to translation

memory systems, both the new text to be translated and the texts stored in the database would need to be run through the automatic paraphrasing system in order to facilitate matching.

As noted above, not all texts can ideally be expressed using a controlled language. In such cases, the translator could use the matched expressions found in the controlled language translation memory database as inspiration, and s/he could then make appropriate stylistic or terminological adjustments. While the gain in productivity would not be as significant in such cases, it may still save the translator some research time.

Another significant drawback to this approach is the fact that there is not one single controlled language that is appropriate for all domains or text types. Different controlled languages would need to be developed for the different domains and text types encountered by the translator. The features and vocabularies of these controlled languages would then need to be programmed into the automatic paraphrasing system.

4.4 Thesauri

A related strategy that would not entail the rephrasing of texts could be the integration of a thesaurus into a translation memory system. With the help of a thesaurus, it may be possible to determine the semantic similarity of two segments by comparing the semantic similarity of the individual words they contain. For example, if a thesaurus could be used to show that words such as 'find' / 'detect' / 'identify' are semantically linked, and that words such as 'eliminate' / 'destroy' / 'remove' are semantically linked, then a translation memory system should be able to propose the following segments as potential matches:

> If a virus is found, it will be eliminated.
> Whenever a virus was detected, it was destroyed.
> When viruses are identified, they are removed.

As was the case with controlled languages, however, it will likely be necessary to customize thesauri to account for the semantic relationships that are appropriate to different specialized domains.

5. Concluding remarks

Translation memory systems currently use the relatively unsophisticated technique of character string matching in order to retrieve information from a database. The advantages of such an approach are that it is easy to implement, it allows information to be processed rapidly, and it is relatively language independent. This type of tool is gaining popularity among translators, who need to be able to work quickly in this era of globalization. However, quality should not come at the expense of speed, and a retrieval system that has poor precision or poor recall could actually cause a translator to waste time since s/he may be required to weed through irrelevant material or undertake unnecessary research.

In order for translation memory systems to be maximally useful, the search techniques need to be more sophisticated. This includes taking into account syntactic and semantic similarities between segments. If features such as lemmatization and thesauri can be incorporated into translation memories, both the amount and type of information retrieved can be optimized. It is important, however, that such features actually be encoded into translation memory systems in a user-friendly way, and not presented as accompanying tools or additional steps. Learning how to use a conventional translation memory system effectively already requires a considerable investment of time, and since translators are already being pressured to work more quickly, they may not be willing to take even more time to learn or use additional or complicated programs.

6. References

Bowker, L. (2002). *Computer-Aided Translation Technology: A Practical Introduction*. Ottawa, Ontario, Canada: University of Ottawa Press.

Garside, R., Leech, G., & McEnery, T. (Eds.). (1997). *Corpus Annotation: Linguistic Information from Computer Text Corpora*. London/New York: Longman.

Macklovitch, E., & Russell, G. (2000). What's been forgotten in translation memory? In J.S. White (Ed.), *Envisioning Machine Translation in the Information Future: Proceedings of the 4th Conference of the American Machine Translation Association (AMTA)* (pp.137-146). Berlin: Springer.

Melby, A. (with Warner, T.). (1995). *The Possibility of Language: A discussion of the nature of language, with implications for human and machine translation*. Amsterdam/Philadelphia: John Benjamins.

Nyberg, E., Mitamura, T., & Huijsen, W.-O. (2003). Controlled Language for Authoring and Translation. In H. Somers (Ed.), *Computers and Transla-*

tion: A Translator's Guide (pp. 237-273). Amsterdam/Philadelphia: John Benjamins.

O'Brien, S. (1998). Practical Experience of Computer-Aided Translation Tools in the Software Localization Industry. In L. Bowker, M. Cronin, D. Kenny, & J. Pearson (Eds.), *Unity in Diversity? Current Trends in Translation Studies* (pp. 115-122). Manchester: St. Jerome Publishing.

Rapp, R. (2002). A Part-of-Speech-Based Search Algorithm for Translation Memories. In M. González Rodriguez & C. Paz Suarez Araujo (Eds.), *Proceedings of the 3rd International Conference on Language Resources and Evaluation (vol. 2) Las Palmas de Gran Canaria, Spain, May 2002,* 466-472.

Shimohata, M., & Sumita, E. (2002). Automatic Paraphrasing Based on Parallel Corpus for Normalization. In M. González Rodriguez & C. Paz Suarez Araujo (Eds.), *Proceedings of the 3rd International Conference on Language Resources and Evaluation (vol. 2) Las Palmas de Gran Canaria, Spain, May 2002*, 453-457.

Somers, H. (2003). Translation Memory Systems. In H. Somers (Ed.), *Computers and Translation: A Translator's Guide* (pp. 31-46). Amsterdam/Philadelphia: John Benjamins.

Sprung, R.C. (2000). Introduction. In R.C. Sprung (Ed.), *Translating into Success: Cutting-edge strategies for going multilingual in a global age* (pp. ix-xxii). Amsterdam/Philadelphia: John Benjamins.

Vinay, J.P., & Darbelnet, J. (1995). *Comparative Stylistics of French and English*. (Translated and edited by J.C. Sager & M.J. Hamel). Amsterdam/Philadelphia: John Benjamins.