Combining Tools to Improve Automatic Translation

Terence Lewis
Hook & Hatton Ltd
1 St Giles Terrace
Northampton NN1 2BN
United Kingdom
Hook_Hatton@compuserve.com

Abstract

This paper takes a practical look at ways of combining language engineering tools to produce more accurate, “more human” automatic translations. Whilst specific products are discussed, the author believes that the methodology could be successfully implemented with different sets of tools.

Keywords

Analysis, combination, procedure, separation, tools.

Introduction

This paper takes a practical look at ways of combining language engineering tools to produce more accurate, “more human” automatic translations. The tools involved are machine translation software, a translation memory application and alignment software, but also small tools or utilities written to perform simple yet very important tasks. The MT program discussed is the author's own Dutch-English translation software, which has been entirely rewritten in Java. The translation memory software used is Trados Translator's Workbench and WinAlign. All the utilities were written in Java by the author. However, the paper is concerned with presenting an approach or methodology which could conceivably be implemented with a totally different set of tools.

Background

Translation buyers will always prefer the “look and feel” of human translations but can be tempted by the considerable cost savings and speed offered by machine translation (with or without some post-editing). The approach presented here provides a way of increasing the “human look and feel” of automatically generated documents. The use of the word "human" with reference to computer-generated translations is, of course, paradoxical. Can the product of a translation program ever look and feel human? Some members of the translation community would answer this question with a resounding "No". And, if we were to confine our discussion to translation engines that provide no more than word replacement and some basic rearrangement, they would broadly be right. At the level of very simple instructions and technical descriptions, the computer and the human translator may sometimes produce an identical translation. But when it comes to the choice of adjectives, the positioning of adverbs, converting the passive voice to the active voice and making some passives active, or the use of an adverb instead of an adverbial phrase, computer and human will invariably part company. While it is possible to hard-code idiomatic translations and write rules for converting the passive to the active voice, human translations of complex sentences are nearly always more readable.

The wide availability of relatively low-cost translation memory applications has certainly led some members of the translation community to prematurely engage in wild shenanigans on the freshly filled graves of some poor MT packages. But translation memory software alone cannot meet the rapidly growing demand for "information-level" or draft translations. Translation memories still have to be created and stocked by humans, and yet many documents contain parts that could be adequately drafted by "professional-level" translation software. The realities of the marketplace led to author to experiment with ways of using the best bits of the two key language engineering technologies.

Our fundamental principle in combining these technologies is that the machine translation engine should only translate sentences not stored in the translation memory database, and that everything entered in translation memory should be both grammatically and terminologically correct. This means that the document has to be divided into “known” and “unknown” sentences (or segments) in such a way that only the unknown sentences are sent to the machine translation engine. In our workflow this separation is made using the various tools available within the Trados Translator’s Workbench. There are other approaches and solutions.

Process

In the authors' experience, successful implementation of this approach requires a rigorous observance of a carefully designed procedure. Completion of each step in the process is essential; omission of a step will certainly impoverish the results of the next step. For example, the MT engine will attempt to assign a part of speech to an "unknown" Dutch word, i.e. a word not in the core dictionary. However, even if the software gets this task right, the translation engine certainly cannot use its semantic and stylistic tools to full effect without having an English translation to work with. For this reason, it is important to provide English translations against the entries in the list of unknown words and import these into the dictionary. The same logic applies to other steps in the process.

The process looks like this:

- Analysis
- Export unknown segments to base file
- Terminology work on base file

- Send base file to MT
- Validate/post-edit MT output
- Align MT output and base file
- Import alignment project into translation memory
- Generate final document from translation memory

Analysis

We analyse each document using the “Analyse” feature in Translator’s Workbench. The analysis results contain statistical information on the number of complete and partial or “fuzzy” matches between the sentences in the source file and the sentences stored in the database. We decide how we are going to process the document on the basis of this analysis. For example, a document containing less than 10% known segments will be tackled primarily as a machine translation project, although the post-edited MT output will always end up in the translation memory. If, on the other hand, more than 95% of the sentences have matches in the translation memory database, Translator’s Workbench will be used as the principal tool for generating an automatic translation of the source document, and MT may not even be used at all. At this stage it is sometimes useful to run the "Translate" option in Translator's Workbench simply to assess the quality of the translations in the translation memory. If necessary, these translations can be revised 'on the fly'.

If we decide to use the MT engine, we export the unknown sentences into our base file. Most of the process is concerned solely with the translation of this base file:

Base File = Source File - Known Segments

Terminology work

For obvious reasons the terminology work is done on the base file, not on the full source file. Various tools written in Java by the author are used to identify words and abbreviations not in the core dictionary of the MT application. The "NotFoundWords" tool simply identifies the words in the base file that do not have any corresponding entry in the core dictionary. "ListWords" generates a list of all the words in the file. It is useful to run both tools as a way of identifying terms which are in the core dictionary but not with the meaning required by the subject matter of the document being translated. Where possible, we e-mail the "NotFoundWords" output file to customers with a request to them to add their preferred translations against the Dutch words in the list. If customers are unable (or unwilling) to provide terminology, we complete the "NotFoundWords" list with feasible translations, unless the customer is prepared to pay extra for terminology research. It has been suggested that the use of two different sources of translation could lead to terminological inconsistencies. This is a valid observation and we have found that the MT dictionary and the Translation Memory will sometimes even come up with alternative spellings for the same word. For most of our customers, who tend to use our output as a first draft this is not a major problem. However, where terminological consistency is critical, we find that the Translator's Workbench concordance provides a useful way of checking key terms.

The "lsf" (LongSentenceFinder) utility creates a file containing sentences of a user-defined length contained in the document. Running this little program is a good way of identifying potential problem sentences (lengthy sentences, lists without punctuation) for the MT engine. A decision can then be made to pre-process these sentences (by entering them directly in the translation memory or, if appropriate, in the MT application's semantic unit database); in the case of a lengthy series of bullet points or lists without punctuation, simply adding some punctuation in the source file may do the job.

In our experience the "Analysis" and "Terminology" stages are the most important parts of the process, particularly on large projects involving the translation of many files on related subjects. Projects totalling more than 30,000 words may contain no more than 3,000 words in a multiplicity of combinations. It clearly pays dividends to make sure that the translations of these words are correct. Even poorly formed sentences will be intelligible to the person skilled in the art if all the individual terms are properly translated. Although entering terms in the dictionary takes longer than running Search & Replace routines in the output file, most MT engines will generate ill-formed sentences if they have to work with unknown terms left in the source language.

Sending the base file to MT/Validating or post-editing the MT output

In the environment discussed in this paper, the base file is sent to the MT engine by simply clicking an icon in a toolbar. The next stage in the process is to handle the MT output. The Logos and Systran MT applications will produce files which in format, if not in content, are ready to be imported into the translation memory database. Our own program produces a text file. Before this file can be imported into translation memory it needs to be aligned with the base file. For this we use the Trados WinAlign alignment tool, which produces reasonably accurate alignment results.

It is at this stage that any required human intervention in the text occurs. Since the MT output will be imported into the translation memory, it is validated or post-edited. Validation involves checking that each sentence in the output actually matches up with the corresponding sentence in the target file. Bits of long sentences may go missing, words may be repeated; we only want to import perfect matches into the translation memory. Post-editing may go no further than correcting glaring errors made by the MT engine without making stylistic improvements. However, sentences that previous analysis has shown to occur time and again throughout the document are edited more carefully. For instance,

MT output : "The engine is switched on by pressing the red button"

becomes

TM entry: "Press the red button to switch on the engine" (or something similar)

This exercise is really the key to enhancing the "human look and feel" of the final automatic translation.

The “Align” attribute after the creator tag tells the Trados Workbench not to necessarily accept the translation of the segment as a 100% match. This means that as the translation memory engine processes the source file it will stop every time it reaches such a segment. As we want to generate the final document automatically we change the "Align" attribute to something else. Having done this we import the alignment file into the translation memory.

Final document generation

Next, the complete document is generated or “translated” using the “Translate” option in Translator’s Workbench. The formatting of the translated document is handled by the translation memory program. After running the "Clean up" option, the file is e-mailed to our customer without any further intervention.

While this process might appear to be cumbersome for short jobs, it can be smoothly managed from a series of buttons on a custom toolbar from within MS-Word, although it is possible - though more "fiddly" - to work with RTF files or HTML files outside MS-Word in a text editor. It is certainly cost-effective to combine MT and TM on large projects involving the translation of technical documents. Our own experience is that between 10 and 30% of all sentences are repeated at least once in a large collection of files. A more important gain, however, is the steady improvement in the quality of automatic translations.

As already mentioned, this is achieved by only using machine translation for sentences not in the translation memory and by post-editing machine-translated sentences before importing them into translation memory This quality gain has been boosted by importing high-quality human translations and by the selective use of bilingual material available on the Web. Many texts published in Dutch and English are in the public domain; by selecting and aligning texts in the fields in which our key customers work we increase the chances of achieving a high percentage of matches in our analysis, thereby reducing dependence on the MT engine.

The author regularly uses this process to translate patent applications in the fields of polymer chemistry and biotechnology and sets of analytical procedures for a major corporation. Although the typical sentence length in patents might suggest that they are unsuitable for automatic processing, this is not so. They are written in a structured, logical way which our own rule-based MT system can handle quite effectively. Since patents often utilise the language of the prior art and of other patents dealing with similar inventions, much of a new document is found by running the translation memory analysis routine to be already in the translation memory. In some cases, less than 65% of the patent application text needs to be sent to the MT engine. If we are translating a series of patents that are slight variations on a theme, this figure drops dramatically, and in the case of modifications to draft documents to only a few per cent.

The analytical procedures describe laboratory experiments designed to determine properties of chemicals used in our Dutch customer's production plants all over the world. These documents are per se suitable for automatic processing. The order of the laboratory activities that make up a procedure is virtually identical in every document. The headings are the same; the weighing and measuring instructions are standard. In some cases, only the quantities of chemicals used and the values of results varies from one document to another. All procedures contain general safety instructions and specifically applicable Risk and Safety (R & S) phrases: we have entered all our customer's bilingual safety instructions and all the R&S phases in Dutch and English in the translation memory database. After using automatic translation to generate English versions of these procedures for about two years, we now find that only about 10 - 15% of such documents needs to be sent to MT.

Reference has already been made to the LongSentenceFinder utility. It is often less time-consuming to pre-translate lengthy sentences via the translation memory application than to post-edit linguistic jumble (occasionally!) produced by our MT engine. Translator's Workbench also has a feature that allows the user to identify frequently recurring sentences. On a large project it may be useful to increase the "human look and feel" of the translation by entering human translations of such sentences in the translation memory. It is clear that the model outlined in this paper bears little resemblance to the so-called "gisting" model, where low-quality off-the-cuff translation is acceptable in the interest of rapid access to information.

Conclusion

Some may argue that this approach is not really automatic translation at all, but a form of interactive translation using a variety of tools. Insofar as the process requires the involvement of a linguistically competent user who can make informed choices, that is true. However, the ultimate object of this model is to provide automatic translations with a more "human look and feel", which are, at most, "human assisted" computer translations. The degree of human intervention will vary but for the typical documents of regular customers it will decline markedly - even over a period of months. Until such time as somebody somewhere develops the killer language engineering application, combining language tools is the best hope for anyone who aims to produce automatic translations that are readable, intelligible and generally useful. Who wants "quick and dirty" when you can have "quick and clean" for a few bucks more?