Is Machine Translation actually Translation?

by
R. Lee
Humphreys

People have been playing with Machine Translation ever since computers were first thought of. Historically, the performance of MT systems has never been particularly good (some would say it's downright bad) and acceptable translations could only be produced by extensively post-editing the output. Although post-editing is time consuming - and not always popular with translators - in some applications the use of MT as a factor in the production of translations can be cost-effective.

In recent years the number of MT systems on the market has increased and (inevitably) the cost of suitable platforms to run them has dropped. As was clear from the substantial international attendance at the first ever MT Evaluators' forum in Switzerland last year[1], language centre directors in a wide variety of manufacturing, service and financial corporations are now wondering whether MT could help to contain ever-rising translation costs. Translators are expensive individuals - and they are not getting any cheaper.

Operational Evaluation

The most obvious way for a potential user to determine whether MT is going to cut costs is to carry out an operational evaluation on site comparing MT+PostEdit costs with those of pure human translation. However, this is both time-consuming and complex. A variety of problems must be addressed:

Source Material:
Even now, many potential MT users have limited quantities of readily accessible text in electronic form. Moreover, an available electronic form may not be suitable for the MT system and may require some sort of conversion process. In the near future we can expect corporate text processing systems to handle (at least) ISO SGML markup. However, even then it is probable that MT engines will expect documents to be conformant to a strictly limited set of Document Type Definitions since certain aspects of translation behaviour may be keyed off this.

Translator Training:
Post-editing machine-translated text is very different from post-editing human text. Like all skills, it takes time to acquire - hence the post-editing performance of translators will inevitably be poor in the initial phases of the evaluation. The MT system may embed or presuppose some particular text-editing environment with which the translators are unfamiliar: again, some training element is required.

Quality Control:
It is sometimes suspected that post-edited MT translations tend to be of inferior quality to pure human

translations (even for the same person) because there is some temptation to post-edit only up to that point where an acceptable (rather than good) translation is realised. Hence some independent quality control of output is required to ensure that translations from either source (HT or MT+PostEdit) are of the same standard.

Dictionary Updating:

A significant part of the cost-effectiveness of simple MT systems resides in their ability to handle the translation of specialist terminology consistently (for single-word terms translation is largely a matter of removing number/gender inflections in the source language, performing dictionary lookup, and returning suitably inflected target language forms). In order to get a representative performance in the evaluation, it will be necessary to ensure that the system dictionaries are updated with at least some of the terms in the users' particular business or technical areas. Dictionary updating is generally time consuming and (again) requires experience and/or training.

During the evaluation period it will only be possible to put a fraction of the terms in frequent use in the dictionary. It will therefore be necessary to try to infer what the expected performance of the system might be when some much larger portion of the terms are put in the dictionary during actual use.

Similar considerations apply to dictionary updating for multi-word terms (eg "one-to-one mapping" as used in software documentation).

The net result of these complications is that an operational evaluation conducted by a user will be extremely expensive. Several presentations at the Evaluators' Forum indicated that this sort of evaluation was taking anything up to 12 person months and more of translator time. A published study by Vasconcellos[2] was similarly lengthy.

Quality Assessment

From a user perspective, it would be nice if there was some reliable way of measuring and describing the performance of MT systems; that would open up the possibility of having comparison tests of MT systems conducted and published by a third-party (or by the suppliers themselves).

Traditional MT systems have a "transformer" architecture: they work by taking input sentences in one language and actually transforming those sentences in various ways - including dictionary lookup - to make the output sentence look more like a sentence in the target language. These quasi-sentences can be scored for the degree to which they look like plausible sentences in the target language (Intelligibility) and for the degree to which they preserve the meaning of the source language sentence they translate (Accuracy or Fidelity). (For discussion see eg.[3,4,5,6,7]). Collecting scores for a large sample of sentences gives system performance profiles with respect to each quality dimension. Figure 1 shows a schematic Intelligibility profile for a small commercial MT system:

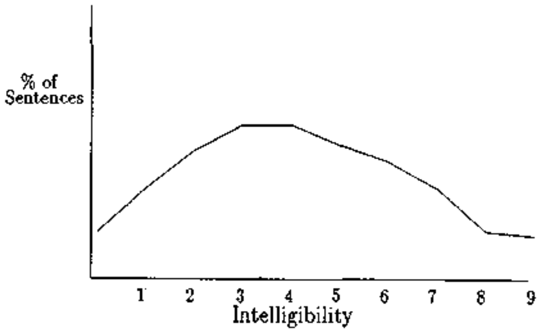


Figure 1: Schematic Quality Profile for an MT System

Working out scoring principles in detail is not easy. Should the persons doing the scoring be required to identify a precise (and lengthy) list of errors which are then weighted to derive an overall sentence score? Or should they just try to score on the basis of a "global impression" of the quality of the sentence? If scorers are required to identify precise errors, then the scoring principles have to provide strong guidance as to what counts as an instance of a particular error (e.g. what counts as an error in tense or aspect in a given context) and where one error ends and another starts (not easy with poor quality quasi-sentences). If the scorers are required to score on the basis of a global impression, there is enormous scope for inter and intra scorer variation and inconsistency. In short there is no one obvious quality scoring procedure that should be adopted for all cases or used for general benchmarking purposes.

A second problem is that of interpretation. It may be possible to show that one MT system is consistently better than another in the sense that it has the same sort of profile as the others centred on a higher mean score. However, it is also possible that the profiles for different MT systems differ in shape, as schematically exemplified in Figure 2.

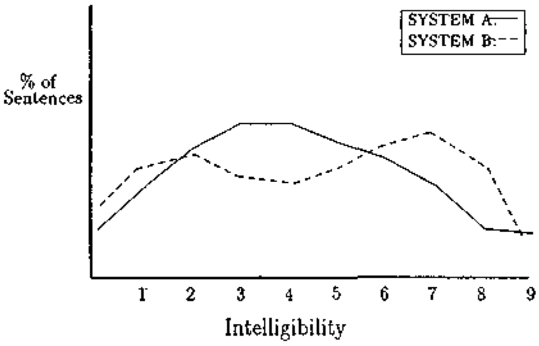


Figure 2: Schematic Intelligibility Profiles for 2 MT Systems

Should the potential user prefer system A to system B? The former has more very good translations, but it also has rather more very bad translations.

Assessing MT quality is particulary difficult because both the input and output are intended to be in natural language - and characterising natural language is difficult (so difficult that theoretical linguists spend their entire lives on the task). We can compare the situation with another domain of NLP - automatic text analysis. Although the input to such systems will be in natural language (e.g. English language news stories on terrorist activities), their output can be expressed in a much more restricted formal language (e.g. as news-item-specific values in a component template for terrorist incidents in general). As a result, in a recent evaluation of text-analysis systems[8], participants were able to reach agreement on what the templates should be for a news story and on a rather simple error classification. Thus scoring of text-analyser outputs against a specimen answer template for each news story could be done automatically. The MT analogue of this - automatic scoring system translations against a specimen translation for each sentence - is more or less out of the question.

Coverage and specification

For some years the trend (at least in research circles) has been towards systems with substantial linguistic knowledge in the form of rule-based grammars. These systems have a different performance profile; typically, either they deliver a well-structured output sentence, albeit stylistically inappropriate, or they deliver nothing at all. Output will tend to contain rather fewer badly degraded sentences.

This behaviour reflects that fact that, at the core of such systems, either a structure is recognised by at least one linguistic rule at each conceptual level or it is not; if it is, the processes of "informed" analysis and synthesis can continue; and if it is not, the system simply halts in an error state. In the older "transformer" architecture, rules have a much more permissive character; if a structure is recognised by a rule at some particular point in the process, that's fine; if it is not recognised by *any* rule at that particular point, that's fine too. In the very worst case, transformer architecture can return a completely unchanged input string. In the very best case, it can return something which looks astonishingly like a well-formed target language sentence. In the average case, it returns neither one thing nor the other.

With linguistic knowledge architecture, attention to some degree shifts from "Does it translate at all" to "Precisely what sort of things does it translate"? The system developer will have embedded rules in the system which cope with a number of phenomena - but this rule set will always be incomplete with respect to the vast variety of grammatical phenomena that are manifest in the language as a whole. Both developer and user need some instrument for checking whether the system really does cover a certain set of phenomena in all their combinations.

Rather than churning through increasingly large "natural" text corpora, the R&D community has recently turned its attention to the use of suites of specially constructed test sentences[9]. Each sentence in the suite contains either one linguistic construction of interest or a combination thereof. Thus part of an English test suite might look as follows:

John runs.
John has run. *aspectual auxiliary*
John will run. *modal auxiliaries*
John can run.
John may run.
John should run.
John will have run. *modal and aspectual auxiliaries*
John may have run.
John should have run.
John can have run.

John does not run. *negation (with do-support)*
John not run.
John has not run. *negation and aspectual auxiliaries*
John will not run. *negation and modal auxiliaries*
John may not run.
John should not run.
John could not run.
John will not have run. *negation, modals and aspectuals*
...

This fragment just chops and changes its way through all combinations of "can", "might" and the like together with optional "not". In practice, one would expect test suites to run to very many thousands of sentences. Suites may include grammatically unacceptable sentences (e.g. *John not run*) which ought not to be translated. In systems which use the same linguistic knowledge for both analysing and synthesising text, the fact that an ill-formed sentence is rejected in analysis suggests that it is unlikely to be constructed in synthesis either. There will be a test suite for both the source and the target language (if the system runs in both directions) and some specialised sentences in each language designed to probe particular translation problems.

Clearly the test suite will be an important tool in MT system development. How useful will it be for the user?

It is perfectly possible for the user to run an MT system on a test suite of his/her own devising and, in some cases, this may be perfectly appropriate. However, apart from the difficulty of knowing how to design a test suite (this is still a research problem), and the cost of actually constructing it, there remains a familiar problem: how are the results to be interpreted? Suppose System A and System B both produce acceptable translations for 40% of the test sentences and that they actually fail on different, or only partially overlapping, sentence subsets. Which one is better? If System B (but not System A) fails on test sentences which embody phenomena with very low historical frequencies in the user's type of text material, then clearly System B is the better choice. But users typically do not have reliable information on the relative frequencies of various types of construction in their corpora - and this can only be obtained by using automatic tools and techniques which are not yet widely available.

The same problem of interpretability holds when MT systems are evaluated by an independent agency using some sort of standard set of test suites. Published test suite information certainly gives a much better insight into expected performance than the vague promissory notes offered with current systems; but it doesn't immediately translate into cost information.

Controlled language and specification

A central problem of LK architecture is that it is brittle. If an incoming sentence contains some construction which is not covered by the system's rules (either because the system linguists did not anticipate it or because they failed to supply a complete and correct rule encoding of it) then it will crash into some error state. Commercial LK systems must be provided with various fail-soft mechanisms to cope with these problems, such as "parse-fitting" which attempts to figure out the best possible parse of most of the sentence given that the system cannot actually come up with a correct one. But resort to coping mechanisms will inevitably result in severely degraded output (similar to the efforts of the old transformer architecture).

One way to live with a brittle system is to make sure it never has to take the strain. Suppose instead of writing technical manuals (for example) in ordinary technical English we wrote them in a specially controlled sub-set of technical English. That is, as well as precisely specifying the set of general-purpose words and terms (for a given domain) to be used, we would also specify which particular grammatical constructions of English could be used to construct sentences in our controlled language ("No Passives", for example). Suppose we also specified a controlled subset of French for the same purposes following the same general principles (although the particular rules would, of course, be different). The linguistic knowledge of an MT system could then be tailored for translating between these two controlled languages. For a restricted application domain of this sort it would be much easier to construct readily interpretable measures of system performance.

It might be argued that to insist on controlled input is to merely replace extensive post-editing with extensive pre-editing. This would be true if documents to be translated were originated in free text. However, in many corporate contexts it is now quite normal to write technical documentation (for example) in controlled subsets of English to improve readability for both native and non-native English speakers. For example, European and US aerospace manufacturers now write Service Manuals in an industry standard Simplified English[10]. In this circumstance no specific pre-editing for MT would be required. Moreover, controlled language input could improve the performance of other NLP tools that might be used in document processing and maintenance eg intelligent indexing systems, intelligent analysers and so on.

Producing text that is truly conformant to some specification will require that the author is provided with support tools which check the text for conformity and suggest possible conformant continuations whilst the author is actually writing the text. These support tools would use the same linguistic knowledge as the MT engine itself.

Conclusion

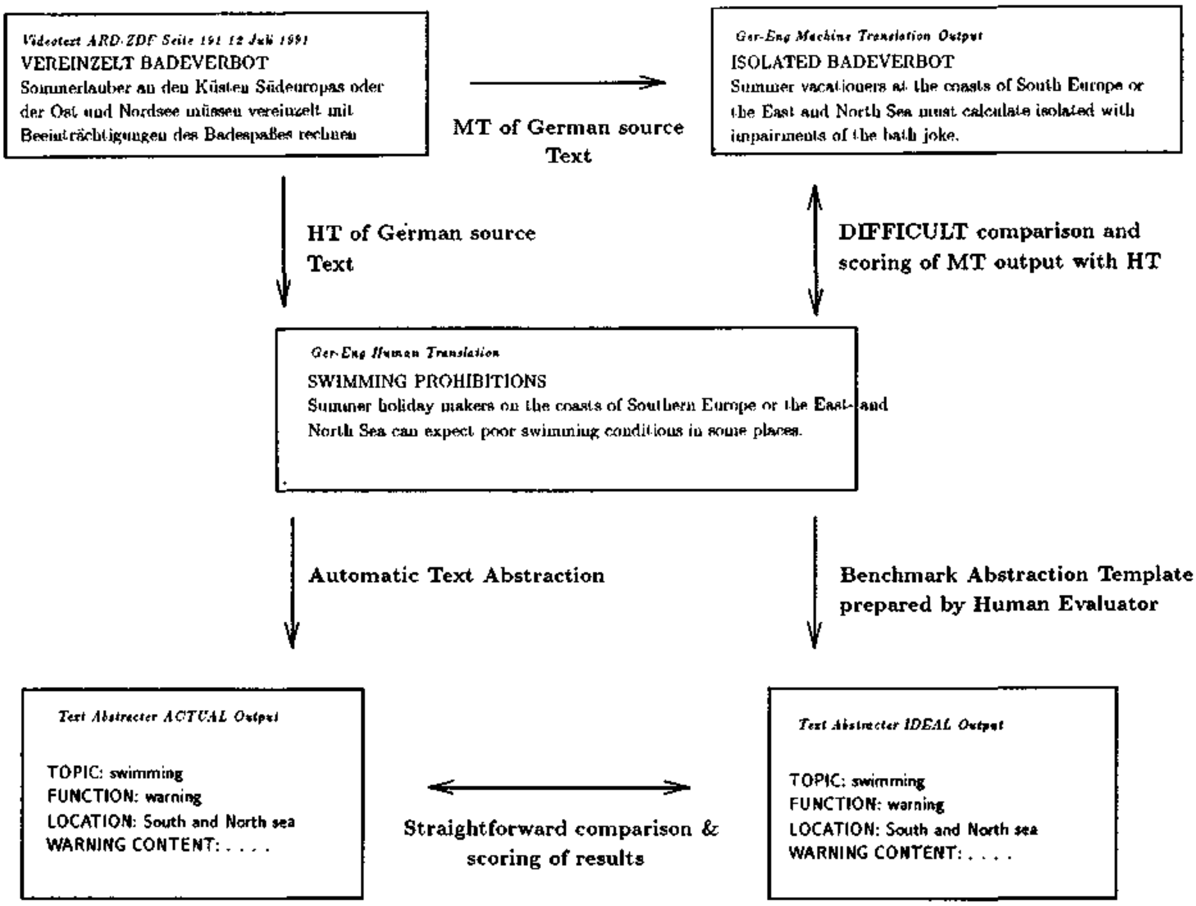
Evaluating a traditional MT system is likely to be expensive and time-consuming.

Currently, users feel they are obliged to evaluate because there are no acceptable performance specifications for MT systems. Providing specifications for MT systems has historically been difficult because output is of poor quality and is likely to have a difficult-to-characterise relationship to real translations. The increasing use of genuine linguistic knowledge of both source and target languages in MT is likely to result in better quality output. A further problem is that a specification of system coverage in terms of the varieties of linguistic construction it handles may be difficult to interpret in terms of expected performance on actual text. One way of resolving the specification issue is for suppliers to produce MT systems tailored to particular controlled language subsets; this strategy is also likely to produce the highest possible performance.

R. Lee Humphreys is a Senior Research Assistant in the Department of Language and Linguistics, University of Essex, Colchester, UK.

References

1. *Evaluators' Forum*, Ste Croix, Switzerland, 20-24 April, 1991.
2. Muriel Vasconcellos (1989), Long-term Data for an MT Policy *Literary and Linguistic Computing* 4(3), OUP, 203-2133.
3. L Balkan, M Jaäschke, L Humphreys, S Meijer & A Way (1991) Declarative Evaluation of an MT System: Practical Experiences *Applied Computer Translation*, 1(3), 49-59.
4. R Pierce & John B Carroll (1966) *Language and Machines - Computers in Translation and Linguistics (Alpac Report)*, Washington DC.
5. John Lehrberger & Laurent Bourbeau (1987) *Machine Translation: Linguistic Characteristics of MT Systems and General Methodology of Evaluation*, Amsterdam, John Benjamins.
6. G. van Slype (1982) Conception d'une méthodologie générale d'évaluation de la traduction automatique, *Multilingua*, 1(4), 221-237
7. M King & K Falkedal (1990), Using Test Suites in Evaluation of Machine Translation Systems *13th International Conference on Computational Linguistics*, Helsinki, 211-216
8. Wendy Lehnert & Beth Sundheim (1991) A Performance Evaluation of Text-Analysis Technologies *AI Magazine*, 81-94
9. Dan Flickinger, John Nerbonne, Ivan Sag & Tom Wasow (1987), Toward Evaluation of NLP Systems, *Ms delivered at Session of 25th Annual Meeting of the Association for Computational Linguistics.*
10. *AECMA/AIA Simplified English: A Guide for the Preparation of Aircraft Maintenance Documentation in the International Aerospace Maintenance Language*, AECMA document: PSC-85-16598, Change 5, AECMA, Paris, 1989.



Videotext ARD-ZDF Seite 191 12 Juli 1991

VEREINZELT BADEVERBOT

Sommerlauber an den Küsten Südeuropas oder der Ost und Nordsee müssen vereinzelt mit Beeinträchtigungen des Badespaßes rechnen

Ger-Eng Machine Translation Output

ISOLATED BADEVERBOT

Summer vacationers at the coasts of South Europe or the East and North Sea must calculate isolated with impairments of the bath joke.

Ger-Eng Human Translation

SWIMMING PROHIBITIONS

Summer holiday makers on the coasts of Southern Europe or the East- and North Sea can expect poor swimming conditions in some places.

Text Abstracter ACTUAL Output

TOPIC: swimming
FUNCTION: warning
LOCATION: South and North sea
WARNING CONTENT: . . . .

Text Abstracter IDEAL Output

TOPIC: swimming
FUNCTION: warning
LOCATION: South and North sea
WARNING CONTENT: . . . .

MT of German source Text
HT of German source Text
DIFFICULT comparison and scoring of MT output with HT
Automatic Text Abstraction
Benchmark Abstraction Template prepared by Human Evaluator
Straightforward comparison & scoring of results

Machine-translated text has to be evaluated by comparison with both the source text and some "high quality" human translation of it. Because the MT does not necessarily belong to any well-defined language, this sort of comparison is very difficult. By contrast, the output of other NLP systems such as Automatic Text Abstraction systems is well defined and expressible in some formal language (here shown as simple tables). Comparison of the Automatic abstractor's output with some model or benchmark (prepared by a human expert) is quite straightforward.

Figure 3: Evaluating Natural Language Processing Systems: MT vs Text Abstraction