# The place of automatic evaluation metrics in external quality models for machine translation

Andrei Popescu-Belis
ISSCO / TIM / ETI
University of Geneva
17 September 2007

## 1. Introduction: what is translation evaluation?

Given a sentence $S_n$ in a source language, and a sentence $T_n$ in a target language, determine a score $\mathbf{s}(S_n, T_n)$ such as

- $\mathbf{s} = 1$ iff $T_n$ is a perfect translation of $S_n$
- $\mathbf{s} = 0$ iff $T_n$ is clearly not a translation of $S_n$
- $\mathbf{s}(S_n, T_n) > \mathbf{s}(S_n, T_k)$ iff $T_n$ is a better translation of $S_n$ than $T_k$

There are three issues arising from this definition. First, what does a "better translation" mean? The answer is generally found by asking human subjects (= language users) to grade or rank translations. Second, could **s** be computed automatically, directly from $S_n$ and $T_n$? The answer is "probably not", because this is also the goal of MT, so this presupposes that the task under evaluation is already solved. But, at least, so, could **s** be approximated, and what supplementary knowledge is needed? Finally, obtaining consistently high value of **s** is not the only desirable property of an MT system, as indicated by the variety of quality metrics gathered in the FEMTI guidelines.

This paper has four sections. First, it offers a principled view of MT evaluation, through the FEMTI framework, which includes quality models with quality characteristics, quality attributes, and metrics. Second, the paper shows that there are two types of justifications for automatic MT evaluation metrics, which we call structural reasons (or "glass-box") and empirical reasons (or "black-box"). Third, the paper outlines a model for empirical, distance-based metrics of MT output quality, which clarifies the arguments that can be made for or against them. Finally, the paper puts forward a proposal for automatic task-based evaluation.

## 2. Principled view of MT evaluation: FEMTI

FEMTI is Framework for the evaluation of MT, started within the ISLE project, and available at http://www.issco.unige.ch/femti. FEMTI consists of two classifications (or surveys): one for the characteristics of the context of use of MT systems, and the other for quality characteristics and metrics. FEMTI helps evaluators to define evaluation plans, in particular thanks to its user-friendly support interfaces, which allow users to specify the intended context of use of an MT system, then to retrieve and customize a contextualized quality model.

The problem of MT evaluation is better understood thanks to the following ISO-inspired notions (from ISO/IEC 9126 and 14598, and the SQUARE framework). Quality is "the totality of features and characteristics of a product or service that bear on its ability to satisfy stated or implied needs" (ISO/IEC 9126), and is decomposed into quality characteristics, then into measurable attributes, each with internal/external metrics. There are six categories of quality characteristics: functionality, reliability, usability, efficiency, maintainability, portability. A

metric is “a measurement is the use of a metric to assign a value (i.e., a measure, be it a number or a category) from a scale to an attribute of an entity” (ISO/IEC 14598).

The FEMTI refinement of ISO quality characteristics for MT (Hovy, King & Popescu-Belis, 2002) contains the following list of characteristics:

- 2.1 Functionality
  - *2.1.1 Accuracy*
    - 2.1.1.1 Terminology
    - 2.1.1.2 Fidelity / precision
    - 2.1.1.3 Well-formedness
      - 2.1.1.3.1 Morphology
      - 2.1.1.3.2 Punctuation errors
      - 2.1.1.3.3 Lexis / Lexical choice
      - 2.1.1.3.4 Grammar / Syntax
    - 2.1.1.4 Consistency
  - *2.1.2 Suitability*
    - 2.1.2.1 Target-language suitability
      - 2.1.2.1.1 Readability
      - 2.1.2.1.2 Comprehensibility
      - 2.1.2.1.3 Coherence
      - 2.1.2.1.4 Cohesion
    - 2.1.2.2 Cross-language / Contrastive
      - 2.1.2.2.1 Style
      - 2.1.2.2.2 Coverage of corpus-specific phenomena
    - 2.1.2.3 Translation process models
      - 2.1.2.3.1 Methodology
        - 2.1.2.3.1.1 Rule-based models
        - 2.1.2.3.1.2 Statistically-based models
        - 2.1.2.3.1.3 Example-based models
        - 2.1.2.3.1.4 TM incorporated
      - 2.1.2.3.2 MT Models
        - 2.1.2.3.2.1 Direct MT
        - 2.1.2.3.2.2 Transfer-based MT
        - 2.1.2.3.2.3 Interlingua-based MT
    - 2.1.2.4 Linguistic resources and utilities
      - 2.1.2.4.1 Languages
      - 2.1.2.4.2 Dictionaries
      - 2.1.2.4.3 Word lists or glossaries
      - 2.1.2.4.4 Corpora
      - 2.1.2.4.5 Grammars
    - 2.1.2.5 Characteristics of process flow
      - 2.1.2.5.1 Translation preparation activities
      - 2.1.2.5.2 Post-translation activities
      - 2.1.2.5.3 Interactive translation activities
      - 2.1.2.5.4 Dictionary updating
  - *2.1.3 Interoperability*
  - *2.1.4 Functionality compliance*
  - *2.1.5 Security*
- 2.2 Reliability
  - *2.2.1 Maturity*
  - *2.2.2 Fault tolerance*
  - *2.2.3 Crashing frequency*
  - *2.2.4 Recoverability*
  - *2.2.5 Reliability compliance*
- 2.3 Usability
  - *2.3.1 Understandability*
  - *2.3.2 Learnability*
  - *2.3.3 Operability*
    - 2.3.3.1 Process management
  - *2.3.4 Documentation*
  - *2.3.5 Attractiveness*
  - *2.3.6 Usability compliance*
- 2.4 Efficiency
  - *2.4.1 Time behaviour*
    - 2.4.1.1 Overall Production Time
    - 2.4.1.2 Pre-processing time
    - 2.4.1.3 Input to Output Tr. Speed
    - 2.4.1.4 Post-processing time
      - 2.4.1.4.1 Post-editing time
      - 2.4.1.4.2 Code set conversion
      - 2.4.1.4.3 Update time
  - *2.4.2 Resource utilisation*
    - 2.4.2.1 Memory usage
    - 2.4.2.2 Lexicon size
    - 2.4.2.3 Intermediate file clean-up
    - 2.4.2.4 Program size
- 2.5 Maintainability
  - *2.5.1 Analysability*
  - *2.5.2 Changeability*
    - 2.5.2.1 Ease of upgrading multilingual aspects
    - 2.5.2.2 Improvability
    - 2.5.2.3 Ease of dictionary update
    - 2.5.2.4 Ease of modifying grammar rules
    - 2.5.2.5 Ease of importing data
  - *2.5.3 Stability*
  - *2.5.4 Testability*
  - *2.5.5 Maintainability compliance*
- 2.6 Portability
  - *2.6.1 Adaptability*
  - *2.6.2 Installability*
  - *2.6.3 Portability compliance*
  - *2.6.4 Replaceability*
  - *2.6.5 Co-existence*
- 2.7 Cost (Introduction, Maintenance, Other)

Examples of quality metrics from FEMTI include the following. For <2.1.1.2 Fidelity>, the assessment of the correctness of the information transferred by human judges; for <2.4.1.3 Input to Output Translation Speed>, the number of translated words per unit of time; for <2.1.3.2 Punctuation errors>, the percentage of correct punctuation marks; for <2.5.2.3 Ease of dictionary update>, the time OR effort necessary to update dictionary. These examples show that some metrics require human judges that cannot be replaced with software (#1 above), while others can be applied both by human judges or software (#2), although software is more precise

& cheaper; some other metrics require human judges or complex software (#3), and others require human users of the system (#4).

A discussion of the role of automatic procedures in MT evaluation implies that only automatic metrics for the quality of MT output such as BLEU, WER, etc. will be considered. There metrics belong in FEMTI Part II, under <2.1 Functionality>: in this section, the current metrics require human judges, and it is not at all clear how they could all be automated.

## 3. Place of automatic metrics in FEMTI

If FEMTI is a complete taxonomy, then automatic metrics (proposed independently of FEMTI or ISO) should belong in FEMTI as well, but where? In other words, if a function $\mathbf{s}(S, T)$ : SL x TL → [0; 1] is to be called a quality metric, one should justify that it is quality metric at all, generally by indicating what quality it measures. So, it must be possible to integrate this (external) quality into the ISO/FEMTI classification, most likely under <Functionality>, if not present yet.

There are two types of justifications for automatic MT evaluation metrics. The structural or "glass-box" justifications are based on the analysis of the definition of a score **s**, the structure of which would indicate that it measures the same quality attribute as a recognized metric applied by humans. Hence, its place in FEMTI would be under the same quality attribute. However, this is an infrequent justification, and does not seem acceptable without the second one.

The empirical or "black-box" justifications (which are by far the most frequent), attempt to show that the values of the score **s** on a given test set are statistically correlated with a recognized metric applied by human judges, and then conclude that the two metrics measure the same quality.

The empirical justification of a score offers also a method (by reverse engineering the justification) to construct such a score **s**:

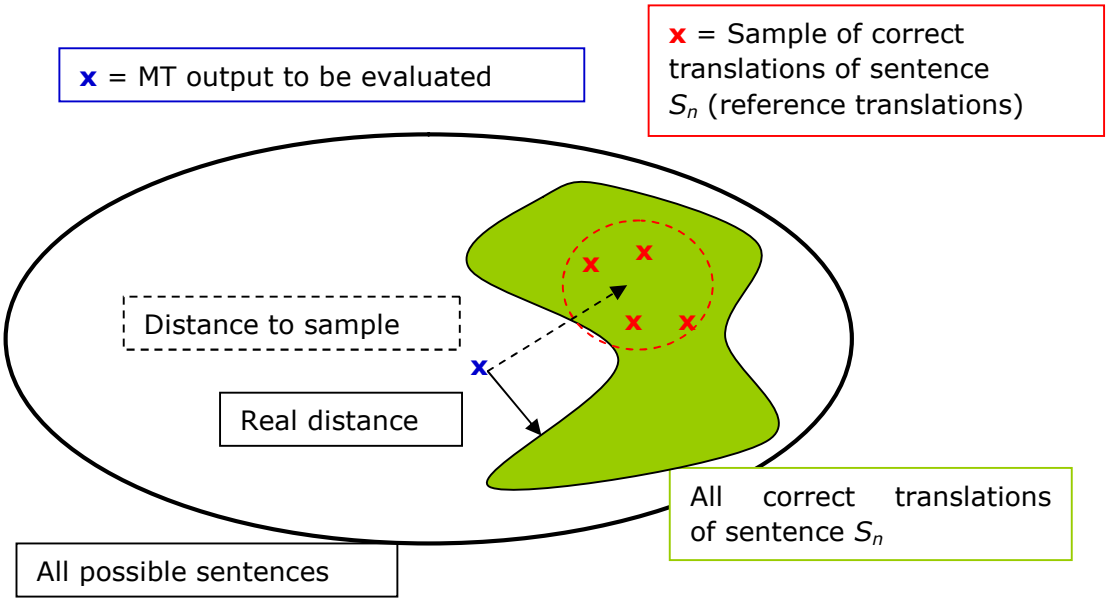
- start with a set of MT sentences that are already scored by humans according to a metric $\mathbf{s_h}$, i.e. start with a large set of triples ($S_n$, $T_n$, $\mathbf{s_h}(n)$)
- train a statistical model to approximate $\mathbf{s_h}$ and then estimate its error using cross-validation. The resulting classifier is the new automatic metric.

However, this approach amounts (again) to solving the same problem as statistical MT! (strictly speaking, the training data for SMT most often has only triples with $\mathbf{s_h} = 1$). So this is clearly too difficult, especially for an evaluation problem. The solution is to use supplementary information about *correct translation(s) of the evaluation data set*, which is perfectly acceptable for evaluation, but not for SMT design.

## 4. A model of trainable distance-based metrics

In many NL tasks, the evaluation metrics are distance-based, i.e. the evaluation data set (the test set) contains the desired output associated to each element of the input data, and the evaluation metrics are defined as distances between a system's output and the desired output, averaged over all items of the input data set. This model cannot apply as is to MT, because there is no unique desired output for an input sentence; however, a frequent proposal is to compute a distance between a system's output and a sample of correct outputs (often up to 4), and

therefore replace the score $\mathbf{s}(S_n, T_n)$ (which presupposes that the MT problem is solved) with an approximation $\mathbf{d}(\{T_{ref(}1_{)}, \ldots, T_{ref(}4_{)}\}, T_n)$. This can be represented as in the Figure below.




To construct (or train) a distance-based automatic metric **d**, one must thus start with a set of machine-translated sentences ($T_n$) that are already scored by humans according to a metric $\mathbf{s_h}$ (and quite often each source sentence is accompanied by reference translation(s)), i.e. start with a large set of t-uples $(\{T_{ref(}1_{)}, \ldots, T_{ref(k)}\}, T_n, \mathbf{s_h}(n))$. The problem of "MT evaluation" is then to find a distance **d** that approximates $\mathbf{s_h}$ (say $\mathbf{d}(\{T_{ref(}1_{)}, \ldots, T_{ref(k)}\}, T_n) \approx \mathbf{s_h}(n)$) . In order to be able to offer the empirical justification, developers generally tune their metric on a test set (e.g. trying several distances $\mathbf{d}_i$ and choosing the one closest to $\mathbf{s_h}$ – e.g. choosing the best parameters for BLEU or for NIST), or sometimes even apply proper machine learning to train a statistical model **d** explicitly to approximate $\mathbf{s_h}$. In both cases, the error of the model was estimated using cross-validation.

This view of how trainable metrics can be constructed also shows the advantages and drawbacks of trainable (empirical) distance-based metrics. Among the advantages, these metrics have a relatively low application cost (once the reference is produced), have high speed, and are reproducible (while human judges may vary). But they also have some drawbacks: their correlation with reference (human) metric holds mainly for data that is similar to the training (or validation data), so their behavior is unknown for different (unseen) types of data (as with every machine-learned classifier). Their correlation with ISO-style qualities is unclear/variable. And of course they need training data, which may have imperfect inter-judge agreement.

## 5. An alternative: task-based evaluation

Many developers of commercial MT systems argue that measuring the utility of MT output for a given task is a more relevant evaluation metric than distance-based approached, or even than metrics of output quality in general. For instance, one can study the performance of human subjects on a task using human *vs*. machine-translated text. These metrics are related to qualities that are closer to ISO's quality in use, but they become increasingly popular as limits of BLEU

become visible (a good example is NIST's / LDC's HTER metric). These metrics seem well-adapted if a system intended for specific application, but they are expensive and time-consuming, and cannot be applied at each design change of the system.

A proposal for a solution is to use automatic task-based evaluation, that is, use MT output for another NLP module for which good automatic metrics are available, for instance reference resolution, or document retrieval.

## 6. Conclusion: research vs. applications?

There seem to be two different views of "quality" in the realm of MT systems. According to the utilitarian view (which is related to the proposal from Section 5), a "better" system means only "better adapted to the users who wish to pay for it" – there are no absolute metrics of quality, independent of the context of use. In this view, task-based metrics are the one that measure MT quality, and as we have mentioned, they do work, and could be automated.

But could the utilitarian view be the whole story? When human translators are trained, they are not taught to translate only in specific contexts, but they are taught a generic "translation capacity", which they can then tune to different contexts. So, there seems to be a fairly generic "translation capacity" which it is legitimate to try to reproduce with NLP-style research. According to this cognitive view, there must be a direct way to measure translation quality independently of its use. The quest for this metric is the object of MT evaluation, but why did this quest become just another NLP problem, complete with machine learning techniques, annotated data, etc.? This is a bit puzzling, and a bit worrying: if MTeval is a new NLP problem, then why not MTeval-eval, etc.?

The reason MT evaluation has become an NLP problem is that the invariants of translation aren't well understood; if they were, and if they could be computed, then they would constitute good candidates for a ground truth representation of the translated text, which should be invariant across all translations. Candidates for such invariants (or at least components there of) are the logical form of sentences, or even the inferences one could draw from them (in a relevance-based account of translation equivalence).