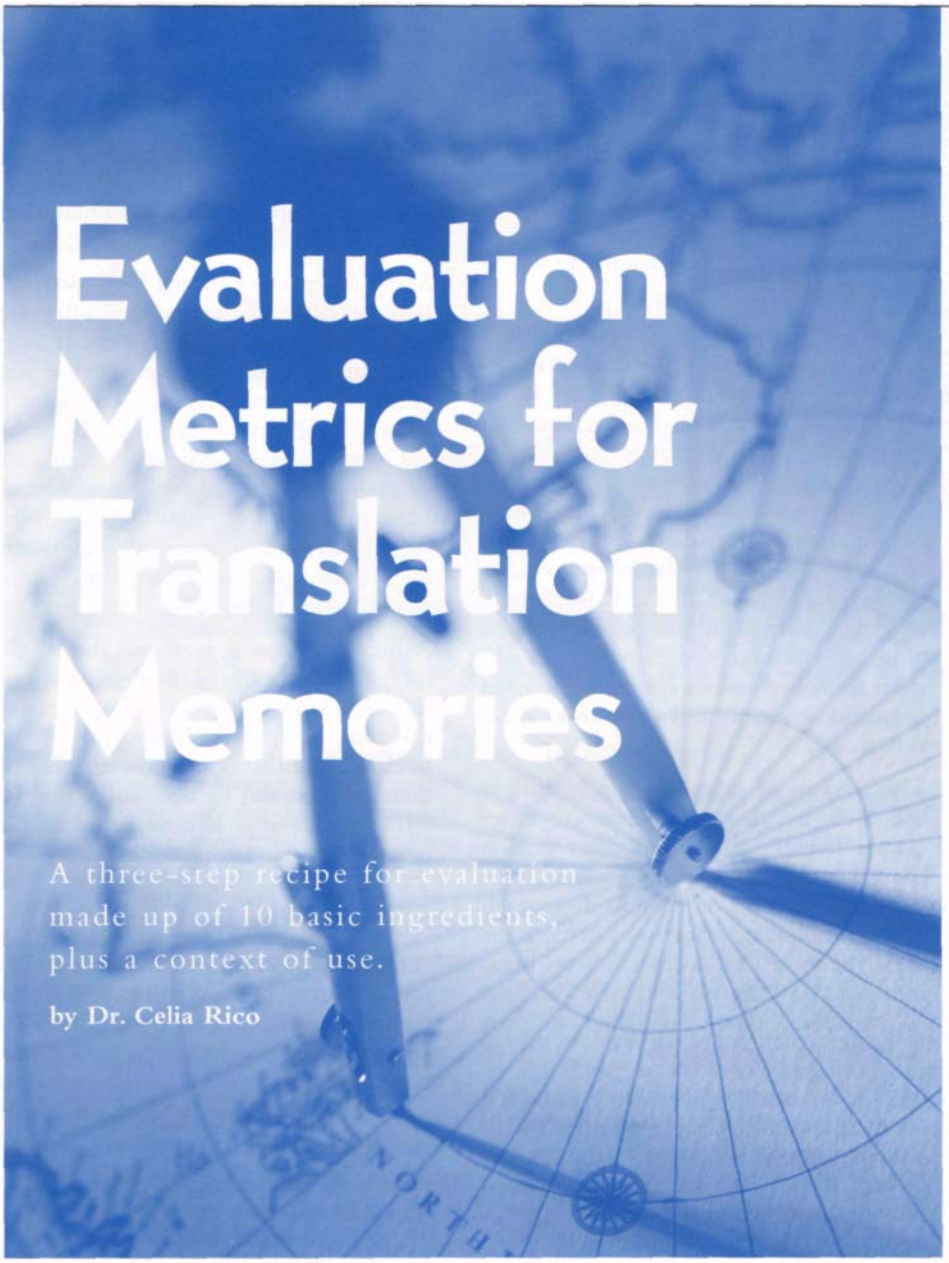

Evaluation Metrics for Translation Memories

A three-step recipe for evaluation made up of 10 basic ingredients, plus a context of use.

by Dr. Celia Rico

So you need a good translation-memory program for your language business? No problem. First, define what you mean by "good." Good price? Good retrieval of previous translations? Ease of use? Speed? Or perhaps the fact that you want the same TM system as your competitors?

Since Martin Kay introduced the concept of the Translator's Amanuensis back in 1980 in his now famous report "The Proper Place of Men and Machines" (Kay, 1997), the idea of an incremental cooperative man-machine system incorporating into a word processor some simple facilities peculiar to translation has evolved, experiencing some transformations, to become what we now know as a translation memory. Translation memory may be defined as "a multilingual text archive containing (segmented, aligned, parsed and classified) multilingual texts, allowing storage and retrieval of aligned text segments against various search conditions" (Eagles, 1995).

The fact is that translation memories are nowadays at the forefront of translation technology with a central position in the translation workflow. They allow an easy leverage of text segments already stored in the TM database and, consequently, lead to an increase in productivity, quality, and consistency.

In this sense, and complemented with other types of technology, TMs provide the translation industry with a powerful tool for resources' optimization (see Alan Melby, "Eight Types of Translation Technology" on www.ttt.org/technology.html). Obviously, deciding what makes a good TM program is, then, a key issue. After all, a freelance translator and a big translation agency may have very different concepts of quality.

One of the main problems in evaluation is how to avoid subjective statements which, in any case, do not contribute to establishing relevant and efficient methods. In a TM, the core task is common, to retrieve translated segments which match new segments to be translated. But methods vary and so do prices, user-interfaces, performance, customization options, and company policy, to name but a few.

Evaluation is always a complex process since it involves taking a series of decisions which ultimately affect your business. And the introduction of a new tool should never be imposed out of sudden urge or chance.

The Evaluation Recipe

Whenever I want to buy a pair of shoes I try them on and consider whether I feel comfortable wearing them, and whether they fit my needs. I will buy different shoes for different purposes: playing professional football, walking, or staying at home. I draw the possible scenario for using my future shoes and I buy accordingly.

A TM program is admittedly not a pair of shoes. It certainly has some greater degree of sophistication, but the test is still valid: does the system fit the user's needs? The scenario test I am about to describe here uses Eagles and the ISO standard 9126 as the point of departure.

It rests on three main interconnected aspects: user profile, tasks, and system performance. They are broken down into a set of features which together provide a practical evaluation framework concentrating on how the system matches the user's requirements and needs. In this sense, the "golden rule" of evaluation might be formulated as follows: first define the context of use and then check whether the system conforms to these specifications. As a matter of fact, the recipe for evaluation can be outlined as three main steps:

1. Design your scenario test as a set of features.
2. Decide how each feature contributes to the final assessment of the system.
3. Execute the evaluation.

The Scenario Test

The set of features that best describes the context of use of a TM can be arranged as the answers to the following questions:

- Translation volume: What is the average volume of translation and how much time is usually allocated for the job?
- Text type and characteristics of the original text: Is the text repetitive, idiomatic, terminology-rich/poor? What file formats do you usually work with? Is consistency among translations a must?
  The answers to these questions would help you decide whether your texts are candidates for TM processing.
- What languages are involved?
- Translation environment: Are you using other translation aids and do you need the TM to interact with them?
- What degree of reusability of previous translations is required?
- Define you team management needs: How many translators need access to the TM database at any one time? Who validates translations? Who controls translation consistency over the team? How is quality control managed?
- What post-editing needs do you have?
- Terminology needs, tools required: What are your needs for terminology leveraging?
- Are you planning to reuse TM data in other environments?
- Do you (or your team) have time to learn a new tool? If so, how much? Gauge the human factor.

Measuring System Performance in 10 Attributes

Once the evaluation scenario is drawn up and we know what we are looking for in a TM system, the next step is to decide the relative value of each feature on the overall performance of the system.

Consider, for instance, how different team-management needs might be for freelance translators working on their own compared to those of a translation agency. The former would probably think that this feature is not applicable to their context of use, while the latter would assign a high relevance.

We will assume that the ideal TM scores 100 percent, this meaning the optimum performance. In order to find out how the different features contribute to this final 100 percent, we have to go over each of them and assign a weight (percentage) according to our own needs. In the example above, freelance translators might assign a low percentage to team-management features, since they are not applicable to their context of use, while an agency could easily give a 50-percent weighting to these same features. Similarly, handling different file formats would be a must in the case of, say, a localization company, while for the individual translator it might only account for a mere 4 percent on the overall system performance.

The attributes to be considered for measuring system performance are the following:

1. Functionality: this is broken down into:
   - Accuracy: measure system performance in terms of precision (percentage of valid segments from all those retrieved) and recall (percentage of segments retrieved from all those valid in the TM database)
   - Interoperability: check whether the system allows interaction with other translation aids.
   - Compliance with standards: check if the system supports different file formats.
   - Security: check whether the system covers your needs for translation validation and control of consistency over your team. If so, what are the mechanisms for translation validation?
2. Portability: if you need to reuse materials in other environments, does the system comply with the required standards?
3. Usability: how much effort is needed for recognizing the logical concept behind the system tasks and workflow? Usability also measures the effort needed for learning the application. What is the learning curve for effective use of the TM? What level of retraining does the TM impose on translation staff?
4. Efficiency: measure here time behavior in terms of retrieval time.
5. Maintainability: how does the system respond to fault-tolerance and recoverability?
6. Backup and service: does the company selling the product offer good service?
7. Pricing policy: can you afford to buy the system?
8. Investment in technical equipment: what level of investment is required before you can actually have the system running?
9. Customization: does the system allow easy customization?
10. Updates: is there any updating policy? Does it suit your needs?

As explained above, assign a weight to each of the 10 features in the form of a percentage, depending on the needs you stated before. System performance will be then measured against the assigned weights so that when the evaluation is finally completed, the TM system which gets a score close to 100 percent would be the one that really suits your needs.

Finally, we are ready to execute the evaluation. Proceed as follows: check each of the features above and, for those where the system shows an excellent performance, allocate all the stated weight. Conversely, no percentage (0 percent) would be allocated if the system shows a poor performance on this same feature. For any performance in between these two poles (from poor to excellent) give the weight that best mirrors the attribute.

As I mentioned at the beginning, evaluation is never an easy job. The aspects to be taken into account are of such variety that its plan and aim have to be clearly stated well in advance, while avoiding subjective statements. My contribution to this issue is far from solving the matter once and for all, but it offers a practical framework which is readily applicable to different contexts of use.

References

Eagles (1995): Evaluation of Natural Language Processing Systems. Eagles document eag-ewg-pr.2. Available at: http://www.ilc.pi.cnr.it/eagles96/browse.html#wg3. 7, November 2000

Kay M. (1997), "The Proper Place of Men and Machines in Language Translation," Machine Translation: 12: 3-23.