Evaluation Metrics for Knowledge-Based Machine Translation

Eric H. Nyberg, 3rd
Teruko Mitamura
Jaime G. Carbonell
Center for Machine Translation
Carnegie Mellon University
Pittsburgh, PA 15213

Topical Paper: machine translation

Abstract

A methodology is presented for component-based machine translation (MT) evaluation through causal error analysis to complement existing global evaluation methods. This methodology is particularly appropriate for knowledge-based machine translation (KBMT) systems. After a discussion of MT evaluation criteria and the particular evaluation metrics proposed for KBMT, we apply this methodology to a large-scale application of the KANT machine translation system, and present some sample results.

1 Introduction

Machine Translation (MT) is considered the paradigm task of Natural Language Processing (NLP) by some researchers because it combines almost all NLP research areas: syntactic parsing, semantic disambiguation, knowledge representation, language generation, lexical acquisition, and morphological analysis and synthesis. However, the evaluation methodologies for MT systems have heretofore centered on *black box* approaches, where global properties of the system are evaluated, such as semantic fidelity of the translation or comprehensibility of the target language output. There is a long tradition of such black-box MT evaluations (Van Slype, 1979; Nagao, 1985; JEIDA, 1989; Wilks, 1991), to the point that Yorick Wilks has stated: "MT Evaluation is better understood than MT" (Carbonell&Wilks, 1991). While these evaluations are extremely important, they should be augmented with detailed error analyses and with component evaluations in order to produce *causal* analyses pinpointing errors and therefore leading to system improvement. In essence, we advocate both *causal* component analyses as well as global *behavioral* analyses, preferably when the latter is consistent with the former via composition of the component analyses.

The advent of Knowledge Based Machine Translation (KBMT) facilitates component evaluation and error attribution because of its modular nature, though this observation by no means excludes transfer-based systems from similar analyses. After reviewing the reasons and criteria for MT evaluation, this paper describes a specific evaluation methodology and its application to the KANT system, developed at CMU's Center for Machine Translation (Mitamura, et al. 1991). The KANT KBMT architecture is particularly well-suited for detailed evaluation because of its relative simplicity compared to other KBMT systems, and because it has been scaled up to industrial-sized applications.

2 Reasons for Evaluation

Machine Translation is evaluated for a number of different reasons, and when possible these should be kept clear and separate, as different types of evaluation are best suited to measure different aspects of an MT system. Let us review the reasons why MT systems may be evaluated:

- *Comparison with Humans.* It is useful to establish a global comparison with human-quality translation as a function of task. For general-purpose accurate translation, most MT systems have a long way to go. A behavioral black-box evaluation is appropriate here.
- *Decision to use or buy a particular MT system.* This evaluation is task dependent, and must take both quality of translation as well as economics into account (e.g. cost of purchase and of adapting the MT system to the task, vs. human translator cost). Behavioral black-box evaluations are appropriate here too.
- *Comparison of multiple MT systems.* The comparison may be to evaluate research progress as in the ARPA MT evaluations, or to determine which system should be considered for purchase and use. If the systems employ radically different MT paradigms, such as EBMT and KBMT, only black-box evaluations are meaningful, but if they employ similar methods, then both forms of evaluation are appropriate. It can be very informative to determine which system has the better parser, or which is able to perform certain difficult disambiguations better, and so on, with an eye towards future synthesis of the best ideas from different systems. The speech-recognition community has benefited from such comparisons.
- *Tracking technological progress.* In order to determine how a system evolves over time it is very useful to know which components are improving and which are not, as well as their contribution to overall MT performance. Moreover, a phenomena-based evaluation is useful here: Which previously problematic linguistic phenomena are being handled better and by having improved which module or knowledge source? This is exactly the kind of information that other MT researchers would find extremely valuable to improve their own systems – much more so than a relatively empty global statement such as: "KANT is doing 5% better this month."
- *Improvement of a particular system.* Here is where component analysis and error attribution are most valuable. System engineers and linguistic knowledge source maintainers (such as lexicographers) perform best when

given a causal analysis of each error. Hence module-by-module performance metrics are key, as well as an analysis of how each potentially problematic linguistic phenomenon is handled by each module.

Different communities will benefit from different evaluations. For instance, the MT user community (actual or potential) will benefit most from global black-box evaluations, as their reasons are most clearly aligned with the first three items above. The funding community (e.g., EEC, ARPA, MITI), wants to improve the technological infrastructure and determine which approaches work best. Thus, their interests are most clearly aligned with the third and fourth reasons above, and consequently with both global and component evaluations. The system developers and researchers need to know where to focus their efforts in order to improve system performance, and thus are most interested in the last two items: the causal error analysis and component evaluation both for their own systems and for those of their colleagues. In the latter case, researchers learn both from blame-assignment in error analysis of their own systems, as well as from successes of specific mechanisms tested by their colleagues, leading to importation and extension of specific ideas and methods that have worked well elsewhere.

3 MT Evaluation Criteria

There are three major criteria that we use to evaluate the performance of a KBMT system: Completeness, Correctness, and Stylistics.

3.1 Completeness

A system is *complete* if it assigns some output string to every input string it is given to translate. There are three types of completeness which must be considered:

- *Lexical Completeness.* A system is lexically complete if it has source and target language lexicon entries for every word or phrase in the translation domain.
- *Grammatical Completeness.* A system is grammatically complete if it can analyze of the grammatical structures encountered in the source language, and it can generate all of the grammatical structures necessary in the target language translation. Note that the notion of "grammatical structure" may be extended to include constructions like SGML tagging conventions, etc. found in technical documentation.
- *Mapping Rule Completeness.* A system is complete with respect to mapping rules if it assigns an output structure to every input structure in the translation domain, regardless of whether this mapping is direct or via an interlingua. This implies completeness of either transfer rules in transfer systems or the semantic interpretation rules and structure selection rules in interlingua systems.

3.2 Correctness

A system is *correct* if it assigns a correct output string to every input string it is given to translate. There are three types of correctness to consider:

- *Lexical Correctness.* Each of the words selected in the target sentence is correctly chosen for the concept that it is intended to realize.
- *Syntactic Correctness.* The grammatical structure of each target sentence should be completely correct (no grammatical errors);
- *Semantic Correctness.* Semantic correctness presupposes lexical correctness, but also requires that the compositional meaning of each target sentence should be equivalent to the meaning of the source sentence.

3.3 Stylistics

A correct output text must be meaning invariant and understandable. System evaluation may go beyond correctness and test additional, interrelated *stylistic* factors:

- *Syntactic Style.* An output sentence may contain a grammatical structure which is correct, but less appropriate for the context than another structure which was not chosen.
- *Lexical Appropriateness.* Each of the words chosen is not only a correct choice but the most appropriate choice for the context.
- *Usage Appropriateness.* The most conventional or natural expression should be chosen, whether technical nomenclature or common figures of speech.
- *Other.* Formality, level of difficulty of the text, and other such parameters should be preserved in the translation or appropriately selected when absent from the source.

4 KBMT Evaluation Criteria and Correctness Metrics

In order to evaluate an interlingual KBMT system, we define the following KBMT evaluation criteria, which are based on the general criteria discussed in the previous section:

- *Analysis Coverage* (**AC**). The percentage of test sentences for which the analysis module produces an interlingua expression.
- *Analysis Correctness* (**AA**). The percentage of the interlinguas produced which are complete and correct representations of the meaning of the input sentence.
- *Generation Coverage* (**GC**). The percentage of complete and correct interlingua expressions for which the generation module produces a target language sentence.
- *Generation Correctness* (**GA**). The percentage of target language sentences which are complete and correct realizations of the given complete and correct interlingua expression.

More precise definitions of these four quantities, as well as weighted versions thereof, are presented in Figure 1[1].

Given these four basic quantities, we can define translation correctness as follows:

- *Translation Correctness* (**TA**). This is the percentage of the input sentences for which the system produces a complete and correct output sentence, and can be calculated by multiplying together Analysis Coverage, Analysis Correctness, Generation Coverage, and Generation Correctness:

$$TA = AC \times AA \times GC \times GA \tag{1}$$

For example, consider a test scenario where 100 sentences are given as input; 90 sentences produce interlinguas; 85 of the interlinguas are correct; for 82 of these

[1] An additional quantity shown in Figure 1 is the fluency of the target language generation (**FA**), which will not be discussed further in this paper.

Criterion	*Formula*
No. Sentences	S
No. Sent. w/IL	S_{IL}
No. Comp./Corr. IL	S_{IL-CC}
Analysis Coverage	$AC = S_{IL}/S$
Analysis Accuracy	$AA = S_{IL-CC}/S_{IL}$
IL Error	IL_i
Weighted AA	$WAA = 1 - \Sigma W_i(S_{IL_i})/S_{IL}$
No. TL Produced	S_{TL}
No. Correct TL	S_{TLC}
No. Fluent TL	S_{TLF}
Generation Coverage	$GC = S_{TL}/S_{IL-CC}$
Generation Accuracy	$GA = S_{TLC}/S_{TL}$
TL Corr. Error	TL_i
TL Fluency Error	TLC_i
Weighted GA	$WGA = 1 - \Sigma W_i(S_{TL_i})/S_{TL}$
Generation Fluency	S_{TLF}/S_{TLC}
Weighted FA	$WFA = 1 - \Sigma W_i(S_{TLC_i})/S_{TLC}$

Figure 1: **Definitions and Formulas for Calculating Strict and Error-Weighted Evaluation Measures in Analysis and Generation Components**

interlinguas the system produces French output; and 80 of those output sentences are correct. Then

$$\begin{aligned} TA &= \frac{90}{100} \times \frac{85}{90} \times \frac{82}{85} \times \frac{80}{82} \\ &= .90 \times .94 \times .96 \times .98 = .80 \end{aligned} \quad (2)$$

Of course, we can easily calculate **TA** overall if we know the number of input sentences and the number of correct output sentences for a given test suite, but often modules are tested separately and it is useful to combine the analysis and generation figures in this way. It is also important to note that even if each module in the system introduces only a small error, the cumulative effect can be very substantial.

All interlingua-based systems contain separate analysis and generation modules, and therefore all can be subjected to the style of evaluation presented in this paper. Some systems, however, further modularize the translation process. KANT, for example, has two sequential analysis modules (source text to syntactic f-structures; f-structures to interlingua) (Mitamura, et al., 1991). Hence the evaluation could be conducted at a finer-grained level. Of course, for transfer-based systems the modular decomposition is analysis, transfer and generation modules, and for example-based MT (Nagao, 1984) modules are the matcher and the modifier. Appropriate metrics for completeness and correctness can be defined for each MT paradigm based on its modular decomposition.

5 Preliminary Evaluation of KANT

In order to test a particular application of the KANT system, we identify a set of test suites which meet certain criteria:

- *Grammar Test Suite*. This test suite contains sentences which exemplify all of the grammatical constructions allowed in the controlled input text, and is intended to test whether the system can translate all of them.
- *Domain Lexicon Test Suite*. This test suite contains texts which exemplify all the ways in which general domain terms (especially verbs) are used in different contexts. It is intended to test whether the system can translate all of the usage variants for general domain terms.
- *Preselected Input Texts*. These test suites contain texts from different parts of the domain (e.g., different types of manuals for different products), selected in advance. These are intended to demonstrate that the system can translate well in all parts of the customer domain.
- *Randomly Selected Input Texts*. These test suites are comprised of texts that are selected randomly by the evaluator, and which have not been used to test the system before. These are intended to illustrate how well the system will do on text it has not seen before, which gives the best completeness-in-context measure.

The first three types of test suite are employed for regression testing as the system evolves, whereas the latter type is generated anew for each major evaluation. During development, each successive version of the system is tested on the available test data to produce aggregate figures for **AC**, **AA**, **GC**, and **GA**.

5.1 Coverage Testing

The coverage results (**AC** and **GC**) are calculated automatically by a program which counts output structures during analysis and generation. During evaluation, the translation system is split into two halves: Source-to-Interlingua and Interlingua-to-Target. For a given text, this allows us to automatically count how many sentences produced interlinguas, thus deriving **AC**. This also allows us to automatically count how many interlinguas produced output sentences, thus deriving **GC**.

5.2 Correctness Testing

The correctness results (**AA** and **GA**) are calculated for a given text by a process of human evaluation. This requires the effort of a human evaluator who is skilled in the source language, target language, and translation domain. We have developed a method for calculating the correctness of the output which involves the following steps:

1. The text to be evaluated is translated, and the input and output sentences are aligned in a separate file for evaluation.
2. A scoring program presents each translation to the evaluator. Each translation is assigned a score from the following set of possibilities:
   - **C** (Correct). The output sentence is completely correct; it preserves the meaning of the input sentence completely, is understandable without difficulty, and does not violate any rules of grammar.
   - **I** (Incorrect). The output sentence is incomplete (or empty), or not easily understandable.
   - **A** (Acceptable). The sentence is complete and easily understandable, but is not completely grammatical or violates some SGML tagging convention.
3. The score for the whole text is calculated by tallying the different scores. The overall correctness of the translation is stated in terms of a range between the strictly correct (**C**) and the acceptable (**C + A**) (cf. Figure 2)[2].

[2] In the general case, one may assign a specific error coefficient to each error type, and multiply that coefficient by the number of sentences exhibiting the error. The summation of these products across all the errorful sentences is then used to produce a weighted error rate. This level of detail has not yet proven to be necessary in current KANT evaluation. See Figure 1 for examples of formulas weighted by error.

5.3 Causal Component Analysis

The scoring program used to present translations for evaluation also displays intermediate data structures (syntactic parse, interlingua, etc.) if the evaluator wishes to perform component analysis in tandem with correctness evaluation.

In this case, the evaluator may assign different machine-readable error codes to each sentence, indicating the location of the error and its type, along with any comments that are appropriate. The machine-readable error codes allow all of the scored output to be sorted and forwarded to maintainers of different modules, while the unrestricted comments capture more detailed information.

For example, in figure 2, Sentence 2 is marked with the error codes `(:MAP :LEX)`, indicating that the error is the selection of an incorrect target lexeme (*ouvrez*), occurring in the Target Language Mapper[3]. It is interesting to note that our evaluation method will assign a correctness score of 0% (strictly correct) 25% (acceptable) to this small text, since no sentences are marked with "C" and only one sentences is marked with "A". However, if we use the metric of "counting the percentage of words translated correctly" this text would score much higher (37/44, or 84%). A sample set of error codes used for KANT evaluation is shown in Figure 3.

1. "Do not heat above the following temperature:"
   "Ne réchauffez pas la température suivante au-dessus:"
   Score: I ; Error: :GEN :ORD

2. "Cut the bolt to a length of 203.2 mm."
   "Ouvrez le boulon à une longueur de 203,2 mm."
   Score: I ; Error: :MAP :LEX

3. "Typical location of the 3F9025 Bolts, which must be used on the 826C Compactors:"
   "Position typique des boulons 3F9025 sur les compacteurs:"
   Score: I ; Error: :INT :IR; :MAP :SNM

4. "Use spacers (2) evenly on both sides to eliminate side movement of the frame assembly."
   "Employez les entretoises (2) sur les deux côtés pour éliminer jeu latéral de l'ensemble de bôti uniformément."
   Score: A ; Error: :MAP :ORD

Figure 2: **Sample Excerpt from Scoring Sheet**

5.4 Current Results

The process described above is performed for each of the test suites used to evaluate the system. Then, an aggregate table is produced which derives **AC**, **AA**, **GC**, and **GA** for the system over all the test suites.

At the time of this writing, we are in the process of completing a large-scale English-to-French application of KANT in the domain of heavy equipment documentation. We have used the process detailed in this section to evaluate the system on a bi-weekly basis during development, using a randomly-selected set of texts each time. An example containing aggregate results for a set of 17 randomly-selected texts is shown in Figure 4.

In the strict case, a correct sentence receives a value of 1 and a sentence containing any error receives a value of zero.

[3] For brevity, the sample excerpt does not show the intermediate data structures that the evaluator would have examined to make this decision.

Module	Code	Comment
:PAR	:LEX	Source lexicon, word missing/incorrect
	:GRA	Ungrammatical sentence accepted, Grammatical sentence not accepted
:INT	:SNI	F-structure slot not interpreted
	:FNI	F-structure feature not interpreted
	:IR	Incorrect interlingua representation
:MAP	:LEX	Target lexicon, word missing/incorrect
	:SNM	semantic role not mapped
	:FNM	semantic feature not mapped
:GEN	:GRA	Ungrammatical sentence produced
	:ORD	Incorrect constituent ordering

:PAR	Syntactic Parser
:INT	Semantic Interpreter
:MAP	Target Language Mapper
:GEN	Target Language Generator

Figure 3: **Sample Error Codes Used in KANT Evaluation**

NAME	S	S_{TL}	S_{TLC}	GA	TA
Result 1	608	546	467-491	86-90%	77-81%
Result 2	608	546	467-519.46	86-95%	77-85%

Figure 4: **KANT Evaluation Results, 17 Randomly-Selected Texts, 4/21/94**

In the weighted case, a sentence containing an error receives a partial score which is equal to the percentage of correctly-translated words. When the weighted method is used, the percentages are considerably higher. For both Result 1 and Result 2, the number of correct target language sentences (given as S_{TLC}) is shown as ranging between completely correct (**C**) and acceptable (**C + A**).

We are still working to improve both coverage and accuracy of the heavy-equipment KANT application. These numbers should not be taken as the upper bound for KANT accuracy, since we are still in the process of improving the system. Nevertheless, our ongoing evaluation results are useful, both to illustrate the evaluation methodology and also to focus the effort of the system developers in increasing accuracy.

6 Discussion

Our ongoing evaluation of the first large-scale KANT application has benefitted from the detailed error analysis presented here. Following the tabulation of error codes produced during causal component analysis, we can attribute the majority of the completeness problems to identifiable gaps in lexical coverage, and the majority of the accuracy problems to areas of the domain model which are known to be incomplete or insufficiently general. On the other hand, the grammars of both source and target language, as well as the software modules, are relatively solid, as very few errors can be attributed thereto. As lexical coverage and domain model generalization reach completion, the component and global evaluation of the KANT system will become a more accurate reflection of the potential of the underlying technology in large-scale applications.

As illustrated in Figure 5, traditional transfer-based MT systems start with general coverage, and gradually seek to improve accuracy and later fluency. In contrast, the KBMT philosophy has been to start with high accuracy and gradually improve coverage and fluency. In the KANT system, we combine both approaches by starting with coverage of a large specific domain and achieving high accuracy and fluency

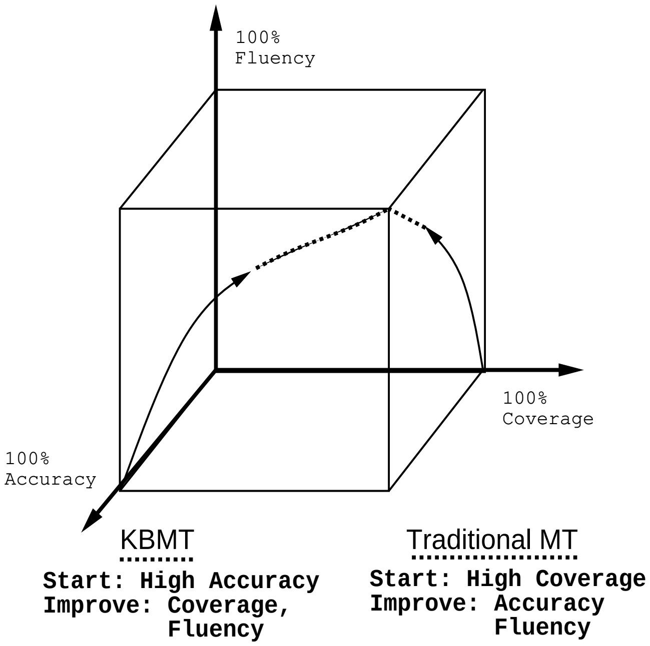

Figure 5: **Longitudinal Improvement in Coverage, Accuracy and Fluency**

within that domain.

The evaluation methodology developed here is meant to be used in conjunction with global black-box evaluation methods, independent of the course of development. The component evaluations are meant to provide insight for the system developers, and to identify problematic phenomena prior to system completion and delivery. In particular, the method presented here can combine component evaluation and global evaluation to support efficient system testing and maintenance beyond development.

7 Acknowledgements

We would like to thank Radha Rao, Todd Kaufmann, and all of our colleagues on the KANT project, including James Altucher, Kathy Baker, Alex Franz, Mildred Galarza, Sue Holm, Kathi Iannamico, Pam Jordan, Kevin Keck, Marion Kee, Sarah Law, John Leavitt, Daniela Lonsdale, Deryle Lonsdale, Jeanne Mier, Venkatesh Narayan, Amalio Nieto, and Will Walker. We would also like to thank our sponsors at Caterpillar, Inc. and our colleagues at Carnegie Group, Inc.

References

[1] Carbonell, J., Mitamura, T., and E. Nyberg (1993). "Evaluating KBMT in the Large," *Japan-US Workshop on Machine-Aided Translation*, Nov. 22-24, Washington, D.C.

[2] Carbonell, J. and Y. Wilks (1991). "Machine Translation: An In-Depth Tutorial," 29th Annual Meeting of the Association for Computational Linguistics, University of California, Berkeley, CA, June 18-21.

[3] Goodman and Nirenburg, eds. (1991). *A Case Study in Knowledge-Based Machine Translation*, San Mateo, CA: Morgan Kaufmann.

[4] Isahara, Sin-nou, Yamabana, Moriguchi and Nomura, (1993). "JEIDA's Proposed Method for Evaluating Machine Translation (Translation Quality)," *Proceedings of SIGNLP 93-NL-96*, July.

[5] Japan Electronic Industry Development Association, *A Japanese View of Machine Translation in Light of the Considerations and Recommendations Reported by ALPAC, U.S.A.*, JEIDA Machine Translation System Research Committee, Tokyo.

[6] King, M. (1993). "Panel on Evaluation: MT Summit IV. Introduction." *Proceedings of MT Summit IV*, July 20-22, Kobe, Japan.

[7] Mitamura, T., E. Nyberg and J. Carbonell (1991). "An Efficient Interlingua Translation System for Multilingual Document Production," *Proceedings of Machine Translation Summit III*, Washington, DC, July 2-4.

[8] Nagao, M. (1984) . "A Framework of a Mechanical Translation between Japanese and English by Analogy Principle," *Artificial and Human Intelligence*, Elithorn, A. and Banerji, R. (eds.), Elsevier Science Publishers, B. V. 1984.

[9] Nagao, M. (1985) . "Evaluation of the Quality of Machine-Translated Sentences and the Control of Language," *Journal of Information Processing Society of Japan*, **26**(10):1197-1202.

[10] Nakaiwa, Morimoto, Matsudaira, Narita and Nomura, (1993). "JEIDA's Proposed Method for Evaluating Machine Translation (Developer's Guidelines)," *Proceedings of SIGNLP 93-NL-96*, July.

[11] Nomura, H. (1993). "Evaluation Method of Machine Translation: From the Viewpoint of Natural Language Processing," *Proceedings of MT Summit IV*, July 20-22, Kobe, Japan.

[12] Nyberg, E. and T. Mitamura (1992). "The KANT System: Fast, Accurate, High-Quality Translation in Practical Domains," *Proceedings of COLING 1992*, Nantes, France, July.

[13] Rinsche, Adriane (1993). "Towards a MT Evaluation Methodology," *Proceedings of the Fifth International Conference on Theoretical and Methodological Issues in Machine Translation*, July 14-16, Kyoto, Japan.

[14] Rolling, L.. (1993). "Panel Contribution on MT Evaluation," *Proceedings of MT Summit IV*, July 20-22, Kobe, Japan.

[15] Takayama, Itoh, Yagisawa, Mogi and Nomura (1993). "JEIDA's Proposed Method for Evaluating Machine Translation (End User System Selection)," *Proceedings of SIGNLP 93-NL-96*, July.

[16] Van Slype, G. (1979). "Evaluation of the 1978 Version of the SYSTRAN English-French Automatic System of the Commission of the European COmmunities," *The Incorporated Linguist* 18:86-89.

[17] Vasconcellos, M. (1993). "Panel Discussion: Evaluation Method of Machine Translation," *Proceedings of MT Summit IV*, July 20-22, Kobe, Japan.

[18] Wilks, Y. (1991). "SYSTRAN: It Obviously Works, but How Much Can it be Improved?," Technical Report MCCS-91-215, Computing Research Laboratory, New Mexico State University, Las Cruces.