Using Test Suites in Evaluation of Machine Translation Systems.

Margaret King and Kirsten Falkedal
ISSCO and ETI
University of Geneva
54 rte des Acacias
CH-1227 Geneva

e-mail: king@divsun.unige.ch
falkedal@divsun.unige.ch

1. Background.

As awareness of the increasing need for translations grows, readiness to consider computerized aids to translation grows with it. Recent years have seen increased funding for research in machine aids to translation, both in the public and the private sector, and potential customers are much in evidence in conferences devoted to work in the area. Activity in the area in its turn stimulates an interest in evaluation techniques: sponsors would like to know if their money has been well spent, system developers would like to know how well they fare compared to their rivals, and potential customers need to be able to estimate the wisdom of their proposed investment. Indeed, interest in evaluation extends beyond translation aids to natural language processing as a whole, as a consequence of attempts to facilitate storage and retrieval of large amounts of information. Concrete manifestations of this interest include a workshop on evaluation in Philadelphia in late 1988, and, in the particular field of machine translation, the publication of two books, the first [4] dedicated to the topic, the second [7] containing much discussion of it.

This paper is concerned with only one sub-topic within the large general area. It is based on work (carried out under mandate for the Suissetra Association) aimed at defining an evaluation strategy for a translation service interested in acquiring a ready-made translation system from a commercial firm, with no possibility of examining the internal workings of the system other than, perhaps, being able to glean clues from the content of dictionary entries. The type of test discussed here is only one of several proposed, all of which assume an evaluation set-up permitting an initial up-dating of the dictionary to cover the vocabulary of the test corpora, plus at least one system up-date with subsequent re-execution of the tests. It is hoped that the particular type of test - the construction and use of test suites - will be of interest to the natural language processing community as a whole.

2. The evaluation strategy.

Although space restrictions prevent any detailed account of the whole proposal for an evaluation strategy, a brief discussion of some basic assumptions will help to set the concerns of this paper in perspective. The most fundamental assumption is that no two potential purchasers of a machine translation system are ever alike in their needs or in their constraints. Thus an essential preliminary to any evaluation is analysis of what is actually needed and what the constraints are. A less obvious consequence is that just as there can be no one "best" translation system, which is the most suitable for all purposes, so there can be no fixed general evaluation methodology, which will unequivocally determine which is the "best" system. However, it is possible to distinguish different factors which may be relevant to making an evaluation judgement, and to suggest ways of collecting data on the basis of which a system's satisfactoriness with respect to each of those factors may be judged. In any particular case, the evaluation can then take account of the relative importance of the individual factors, either in deciding what data should be collected, or when making a final judgement.

In this paper, we concentrate on one method of collecting data relevant to coverage of the source language as evidenced by the acceptability of the translations produced for specific test cases, to the treatment of specific translational problems, and, to some extent, to the ease with which the system can be modified or extended. The full proposal further considers factors like operational behaviour, need for and ease of dictionary up-dating, post-editing and through-put time, user reactions etc., all of them factors whose interaction determines judgements on speed and efficiency.

Once a particular factor has been retained as important in the particular context of evaluation, our proposals require the construction of test materials to be used in obtaining the relevant data. This assumes the availability of someone with at least a knowledge of the languages concerned, of linguistics and preferably some experience of machine translation. Actual practice, as demonstrated by the reports of evaluations already carried out, reveals the vulnerability of this assumption. Nonetheless, we have chosen to retain it, on the grounds that common sense must eventually prevail; it is a critical assumption for the rest of this paper. For the sake of brevity, we shall refer to this person as "the evaluator" in what follows. Also for the sake of brevity, we shall neglect the special questions raised by systems relying on pre-editing or controlled input, and by inter-active systems.

3. Some Standard Evaluation Measures and their Weaknesses.

Three kinds of metrics have prevailed in past evaluations of machine translation systems. First, a very common technique has been to obtain ratings on some scale for aspects of quality such as intelligibility, fidelity or clarity. A second common approach has been to count the number of errors, typically by simply counting the number of corrections made by the post-editor(s). Thirdly, the perception that some errors are more important than others has led to attempts to classify errors according to pre-established classification schemes.

Each of these metrics has its own set of inherent weaknesses, which we shall not go into here. (A detailed discussion can be found in [5] and [1]). Beyond these inherent weaknesses, however, they all suffer from the same two major deficiences: the resulting data, whatever their reliability, do not provide the kind of information necessary for an assessment of the actual acceptability of the translation quality to the users, be they readers or post-editors of the raw machine output; moreover, these metrics do not provide the slightest hint about the ease with which the system can be extended or modified, factors which have acquired increasing importance along with growing insights into the slow but permanently developing nature of machine translation systems.

4. The problem with trying to be more informative.

A natural reaction to the above is to propose setting up a series of systematically organised test inputs to test at least the syntactic coverage of the system, along the lines of the test suite set up at Hewlett Packard for systematic testing of English language interfaces to data bases [2].

The major problem with setting up test suites of this kind is that of interaction between different linguistic phenomena. The standard solution is to reduce the linguistic complexity of all items other than the item of direct interest to an absolute minimum. Thus a sentence like "John gave Mary a book" would be used to test correct treatment of ditransitive verbs, rather than, say, a sentence with complex noun phrases or with a modal included in the verbal group.

Even when a test suite is designed to test only coverage of syntactic structures in a single language, its construction is a lengthy and delicate task, requiring much linguistic insight; adding even rudimentary testing of semantic phenomena would seriously increase the complexity of the task. When the system to be tested is a translation system yet more complications arise, even if testing is limited to testing syntactic coverage.

The most serious of these occurs already during the construction of the test suite, where the fact that the "proof" of a correct treatment of a test input is its intended - or at least an acceptable - translation into the target language imposes non-negligeable constraints on the sub-structures, and especially the vocabulary used: the test inputs should not only be adequate for testing the source language, but also translationally unproblematic, in the sense that they do not introduce difficulties irrelevant to the problem tested. As a concrete example, imagine that a test input is designed to test the treatment of verbs which take an infinitive verbal complement. If the English test input is "The chairman decides to come", the corresponding French sentence is equally simple, "Le président décide de venir", but if the English is "The chairman expects to come", the French equivalent of "expect" imposes a completely different structure ("Le président s'attend à ce qu'il vienne (lui-même)"), and, if the output translation is incorrect, it may be difficult to determine whether the reason is a generally wrong treatment of the class of English verbs, or whether it is something specific to "expect".

Thus, test inputs designed primarily to illuminate the system's treatment of specific source language phenomena should avoid the introduction of "noise" triggered by an injudicious choice of lexical material. However, since translational difficulties between any given pair of languages do exist, a test suite for testing a machine translation system will have to include a substantial component of contrastively based test inputs, an area which has received far less investigation than mono-lingual syntax, and which is specific for each language pair.

Thus, although a translation test suite for a given pair of languages would be of great utility to the community at large, its construction and debugging is a long term project, which no individual responsible for defining test material in a limited time can hope to undertake. Building test suites is perhaps one of the areas, like text-encoding and data collection, best undertaken as collaborative work.

Furthermore, given our basic assumption that no two evaluations are alike as to the needs and constraints which determine whether a system will or will not prove acceptable, the fact that general test suites do not, by their nature, reflect the particular characteristics of the application actually envisaged, may limit their usefulness.

Below, we try to suggest a way of overcoming some of these difficulties, and also of exploiting what seems to us the real value of using test suites, that is, their ability to serve as indicators of a system's potential for improvement.

5. A Productive Compromise.

Despite all the difficulties, there is no way round the fact that data on a system's coverage of the source language, on its ability to produce acceptable translations and on its improvability are crucial input to any judgement of its suitability in a particular environment. Somewhat paradoxically, we therefore suggest that an appropriate way out of the dilemma is to construct not one test suite, but two, each in turn divided into a test suite concerned with syntactic coverage of the source language and one concerned with specific translational problems.

The first of these is based on (computer aided) study of a substantial bi-lingual corpus of the kinds of texts the system will actually be expected to tackle. It will be used to collect data relevant to the system's ability to fulfill the needs currently determined.
The part of this test suite concerned with source language coverage will include structures which are expected to be unproblematic as well as structures known to be likely to cause problems. Particular attention should be paid to structures which are specific to the corpus, for example, orders in French manuals expressed by the infinitive, and to any deviations from standard grammar, for example, "Bonnes connaissances d'Allemand" as a complete sentence in a job advertisement. If possible, the relative importance of any given structure on the evidence of its frequency in the corpus will be indicated. For reasons which will become clear later, the test suite should comprise at least two test inputs for each structure selected, organised in two parallel lists, referred to as the A list and the B list below.

Every test input, in all of the test suites, should be accompanied by an acceptable translation. This does not, of course, imply that the translation constitutes the only acceptable output from the system, but the existence of a pre-defined set of translations will help in maintaining a consistent judgement of acceptability during the data collection exercise.

The second section of the test suite concerns translational problems. The first goal is to include inputs based on mismatches between the two languages. Such mismatches may range from differences in the lexicons of the two languages, different behaviour of otherwise corresponding lexical items (classic examples are "like" vs."plaire"), to much wider ranging differences such as a lack of correspondence in the use of tenses or in the use of articles. Secondly, inputs will be included to test the system's treatment of lexical and structural ambiguity. Once again, though, the choice of what test inputs to include will be informed by study of the corpus of actual texts and their translations: no attempt should be made to think up or import from the literature tricky examples. Here, too, A and B lists of examples should be produced for all phenomena other than those dealing with idiosyncratic behaviour of individual lexical items.

Where the aim of the first test suite is to collect data on the system's ability (present and future) to fulfill the needs currently determined, the aim of the second is, broadly speaking, to collect data on what more or what else the system is capable of, to give some idea of to what degree the system could be used to treat satisfactorily texts other than those seen as its immediate domain of application. This idea will, though, necessarily be more impressionistic than founded on exhaustive evidence.

The test suite will again be divided into a section aimed at looking at source language coverage and a section examining translation problems. Obvious sources of inspiration include those few test suites already elaborated and the literature, including prescriptive grammars and translators' aids. As before, A and B lists of examples should be produced.

6. Using the test suites.

As mentioned previously, the test suites are only one type of test in a full data collection strategy which space constraints prevent us from even outlining here, beyond indicating that, as a function of other parts of the strategy, the dictionaries will already have been up-dated to cover the vocabulary needed for all of the tests below.

First, the test suite based on actual texts is submitted to the system, and the resulting outputs classified by the evaluator as acceptable or unacceptable, with no attempt made at more refined classification. That is, no attempt is made either to count mistakes or to classify them individually.

The results will give a first indication of how well the system can deal with the texts for which it is intended. However, it is totally unrealistic to imagine that any ready-made system will produce only acceptable results without further modification. Given the assumption that the dictionaries contain all the necessary vocabulary, the root cause of unacceptable outputs must lie in inadequacies in the dictionary coding or elsewhere in the system. As a general rule (though not a universal truth) dictionary information is quicker and easier to modify than other parts of the system. Thus, data on which of the unacceptable outputs is the result of dictionary inadequacies, which not, is likely to be of considerable interest.

The system manufacturer is therefore given the A-list inputs and their results, together with the evaluator's classification of their acceptability and asked to sort the unacceptable outputs according to the above criterion. He is then given an agreed lapse of time in which he may attempt to remedy errors said to be due to dictionary coding. The whole test suite is then re-submitted, in order to check improvement/deterioration relative to the first run. The role of the B-lists (not used by the manufacturer for corrections) becomes evident here: the B-list examples serve as a control corpus, since changes in their treatment will be indicative of the carry-over effects of the corrections based on the A-list. For example, they will indicate whether changes have only a local effect or whether the effects extend to whole classes of lexical items. This procedure may be repeated if time allows and if the evaluator considers it useful.

The use of the non-corpus based test suite is essentially similar. The only variation is that, in an attempt to develop intuitions about the capacities of the system, the evaluator will attempt an independent estimate of which, amongst the unacceptable outputs in the A-list, are the consequence of inadequate dictionary coding, checking his estimate against that of the system manufacturer.

Even after modification of the dictionary coding, it is probable that some inputs in the two test suites will still give rise to unacceptable outputs. The next step is to use these remaining "intractable" inputs to acquire data on the system's extensibility. If a technical description of the system's principles is available, an experienced evaluator will probably be able to make informed guesses about the system's extensibility. In many cases though, the manufacturer may be unwilling to give the necessary information, and, in any case, some empirical evidence is required to confirm or disconfirm the evaluator's guesses.

We therefore suggest taking the remaining intractable cases in the A-lists of both test suites, and asking the manufacturer to classify them according to his estimate of how long it would take to modify the system to deal with each case, e.g.

- less than a week
- less than a month
- more than a month
- means of correction not known.

As before, the manufacturer should be asked to demonstrate the reliability of his estimate by actually arranging for the modifications to be carried out for some agreed number of the problem inputs, the actual choice of inputs being left to the evaluator. Both test suites (B-lists included) are then re-executed and re-analyzed for improvements and deteriorations. Once again, the B-lists serve as a control corpus indicating carry-over effects.

7. Conclusion.

In summary, the test suites are used to produce lists of inputs the system can and can not deal with, and evidence on the distribution of problem cases across five classes, the first due to dictionary coding, the remaining four based on how long it would take to improve or extend the system to deal succesfully with the problem. The use of a corpus based test suite adapts the testing to the actual foreseen needs, whilst the use of a non text specific test suite provides information on what more or what else the system might do. Systematic testing of source language structures is separated from contrastively based testing of translational problems, and the use of A and B lists gives an indication of how changes intended to improve the treatment of specific problems may produce unwanted side effects or introduce larger benefits.

The only qualitative (and therefore subjectivity prone) measure involved is the estimate of the acceptability of the translations produced, and even here, some defence against bias is provided by ensuring that each test input is accompanied by a pre-specified acceptable translation. However, as argued earlier, straightforward lists of what a system can or cannot do will not necessarily prove sufficiently informative when forming a judgement about a system's overall acceptability.

The main virtue of the test suite approach is not, then, its relative objectivity, but flexibility in accounting for both the competence and the user-related performance of the system under test. Furthermore, it furnishes a basis for judgements on the potential for and cost effectiveness of systematic improvement of the system. When several systems are to be evaluated for the same application, this procedure can be used to provide reliable comparisons. Finally, the raw data obtained lend themselves readily to a variety of additional quantifications which decision makers may find informative, as well as to more refined linguistic analysis of potential value to the system developers.

Acknowledgement.
The authors would like to record their gratitude to Ulrich Heid and Jonathan Slocum, who read draft reports, offered much constructive criticism, and provided generous opportunities for discussion.

Selected Bibliography.

[1] Falkedal, K. *Evaluation Methods for Machine Translation Systems: An Historical Survey and A Critical Account.* ISSCO: Interim Report to Suissetra. Forthcoming.

[2] Flickinger, D. et al. *HP-NL Test Suite*, (as of June 30, 1987).

[3] King, Margaret. *A Practical Guide to the Evaluation of Machine Translation Systems.* ISSCO. Intermediate Report to Suissetra, Feb., 1989. Revised version in preparation.

[4] Lehrberger, J. and Bourbeau, L. *Machine Translation: Linguistic Characteristics of MT Systems and General Methodology of Evaluation*. John Benjamin. 1988.

[5] Slocum, J. and Bennet, W. *An Evaluation of METAL: the LRC Machine Translation System*. Second Conference of the European Chapter of the ACL, 1985.

[6] Van Slype, G. *Critical Study of Methods for Evaluating the Quality of Machine Translation Systems*. Bureau Marcel Van Dijk, Bruxelles and CCE, 1979.

[7] Vasconcellos, M. (ed.) *Technology as Translation Strategy*. SUNY, 1988.

[8] Wilks, Y. and LATSEC Inc. *Comparative Translation Quality Analysis*, Final Report, Latsec Inc., 1979.