Automatic Ranking of MT Systems

Martin Rajman*, Anthony Hartley[†]

* EPFL Lausanne, CH
martin.rajman@epfl.ch
[†] Centre for Translation Studies, Leeds, UK
a.hartley@leeds.ac.uk

Abstract

In earlier work, we succeeded in automatically predicting the relative rankings of MT systems derived from human judgments on the Fluency, Adequacy or Informativeness of their output. In this paper, we present an experiment – using human evaluators and additional data – designed to test the robustness of our earlier results. These had yielded two promising automatically computable predictors, the D-score based on semantic features of the MT output, and the X-score based on syntactic features. We conclude that the X-score is indeed a robust and reliable predictor, even on new data for which it has not been specifically tuned.

1. Introduction

The manual assessment by human judges of the quality of an MT system's output is a costly exercise. It is, therefore, of central concern to explore whether there are any automatically computable scores that correlate well with such expensive, manually produced evaluations. Indeed, this concern has motivated a number of recent research initiatives (e.g. Papineni et al., 2001; Reeder et al., 2001; Vanni and Miller, 2001; White and Forner, 2001) that have produced promising initial results. These authors recognise, however, that further work needs to be done to test the robustness of their findings.

Our own work in this vein (Rajman and Hartley, 2001) attempted initially to automatically compute scores that would correlate with human judgments on the Fluency, Adequacy or Informativeness of individual documents translated by five different MT systems. This proved to be beyond us; however, we succeeded instead in predicting the relative rankings of the MT systems yielded by the human scores. We identified, in fact, two automatically computable predictors: the X-score based on syntactic features of the MT output, and the D-score based on semantic features.

In this paper, we report on an experiment with human evaluators designed to test the validity of these two scores with additional data.

2. Experimental set-up

2.1. Data

From 1992 through 1994, DARPA conducted a series of MT evaluations as part of the Human Language Technology initiative (White, O'Connell and O'Mara, 1994). The largest of these included 100 newspaper articles in each of three language pairs (Spanish, French, and Japanese into English). Each pair was represented by several MT systems in various states of maturity, and also by two sets of human, professional translations. Each translation, in turn, was subjected to three separate evaluations, for Fluency, Adequacy and Informativeness.

We have focused on the translations derived from French, that is, 100 human translations and 500 outputs generated by five MT systems: Candide (CD), Globalink (GL), MetalSystem (MS), Systran (SY) and XS (XS). Our previous work showed a large discrepancy between the manual and automatic rankings of the human translations, and so we excluded the human translations from consideration in the current experiment. However, we added the 100 outputs generated by a new system, Reverso (RV). Thus, we had 100 series of outputs, each series consisting of six competing target versions derived from the same source text by the six MT systems.

Following (White and Forner, 2001), we sorted the 100 series by their average score for Adequacy. We then selected every 5th series, so retaining a representative sample of 20 series to submit to our human evaluators.

2.2. Evaluators' task

We employed 12 evaluators, all graduate students with English as their mother tongue, ranging in age from mid-twenties to early sixties. We expressly recruited subjects who professed no knowledge of French, so that they would be equally disadvantaged in any attempt to reconstruct a translation by conjecturing about the source text.

Each evaluator read 10 series of outputs, i.e. a total of 60 outputs; each series was read by six evaluators. The order in which each evaluator read each series was randomized, as was, for each series and each evaluator, the order of the six texts in the series. The evaluators were required to take at least a 10-minute break after every two series.

The evaluators were given the following instruction: 'Rank these six texts from best to worst. If you can't distinguish between two or more texts, bracket them together, e.g. 4 [1 2] 6 [3 5]'. They were not told that the texts had been generated by MT systems.

This procedure departs from a long tradition of using anchored scales to assess attributes of MT output, from (Carroll 1966) to (Papineni et al., 2001). It most closely resembles the approach adopted by (Brew and Thompson, 1994), which asked evaluators to assign unanchored, relative scores. The instruction was so designed as to produce responses that would be directly comparable with the rankings predicted from the automatically computed scores. The underlying objective is the simple but useful one of predicting where a new, unknown MT system ranks relative to one or more previously evaluated systems.

The instruction makes no mention of any attribute of text quality, leaving the subjects to determine what they mean by 'better' or 'worse', which we assume to be captured by some combination of Fluency, Adequacy and

Informativeness. In this event, the underspecificity of the instruction is not problematic, since our previous results showed identical rankings according to Adequacy and Informativeness, which varied only slightly from the ranking according to Fluency (Rajman and Hartley, 2001).

It is worth stressing also the fact that our evaluators were ranking texts of some 250-300 hundred words each, thus making summative judgments not only over attributes of quality but also over collections of sentences. This differs from the approach of, for example, (Papineni et al., 2001), who asked their evaluators to judge translations of just 50 sentences selected at random from a corpus of 40 (short) texts

3. Automatically computed scores

In our previous experiments, we defined three types of scores that all can be automatically computed for the translations produced by the MT systems: two scores relying on syntactic features, the X-score and the C-score, and one relying on semantic features, the D-score. As the new results confirmed the already observed low predictive power of the C-score, we will not present this score again and the rest of this section will focus on a brief description of the two most promising candidates: the X-score and the D-score.

3.1. X-score

The X-score is taken to measure the grammaticality of the translations. For any given document, the X-score is obtained as follows. First, the document is analyzed by the Xerox shallow parser XELDA[1] in order to produce the syntactic dependencies for each sentence constituent. For example, for the sentence *The Ministry of Foreign Affairs echoed this view*, the following syntactic dependencies are produced: SUBJ (Ministry, echoed); DOBJ (echoed, view); NN (Foreign, Affairs) and NNPREP (Ministry, of, Affairs).

On the corpus used in our previous experiments, XELDA produced 22 different syntactic dependencies, among which (the figure within brackets indicates the dependence occurrence frequency):

RELSUBJ[2501]: for example, RELSUBJ(hearing, lasted) in "*a hearing that lasted more than two hours*";

RELSUBJPASS[108]: for example, RELSUBJPASS(program, agreed) in "*a public program that has already been agreed on ...*";

PADJ[2358]: for example, PADJ(effects, possible) in *"to examine the effects as possible"*;

ADVADJ[433]: for example, ADVADJ(brightly, colored) in *"brightly colored doors"*.

After each document has been parsed, we compute its dependency profile (i.e. the number of occurrences of each of the 22 dependencies in the document). This profile is then used to derive the X-score using the following formula:

X-score=(#RELSUBJ+#RELSUBJPASS-#PADJ-#ADVADJ)

Note that several formulae would have been possible for computing the X-scores. The above-mentioned one was selected in such a way that, if applied to the average dependency profile, it correctly predicts the average rank ranking (see Section 4.1 below) derived from the F-scores (Fluency evaluation). In this sense, one can say that the computation of the X-score was specifically tuned to the test data and so it was considered quite ad hoc in our previous experiments. However, this is no longer true for the current experiments. We retained exactly the same formula for the X-scores, while completely changing the human evaluations – evaluators directly assigned rankings to series of translations instead of assigning individual scores to each of the translations. Moreover, we added a new MT system, not present at all in the data that was used for the tuning. Thus, in our new experimental setup, there is no reason to believe the X-scores to be ad hoc, which strongly increases their chances of being highly portable to other experimental data.

3.2. D-score

The D-score is held to measure how well the semantic content of a document has been preserved during translation. The underlying idea is to use a vector-based semantic model (similar to those used in domains such as IR) and a large reference corpus of aligned translations[2]. The part of the reference corpus consisting of the documents in the source (or the target) language will be called hereafter the source (or target) reference corpus. In such a setup, the goal is then to measure how far the position of any source document in the vector space defined by the source reference corpus is comparable to the position of its translation in the vector space defined by the target reference corpus.

More precisely, for any document in the source language, we compute its semantic similarity with each of the documents in the source reference corpus and consider the resulting vector of similarities as an indirect characterization of the position of the document in the vector space. The similarity measure used in all our experiments is the cosine similarity between the document lexical profiles (with the SMART ltn weighting scheme (Salton and Buckley, 1988)). We proceed in the same way for the translation of the document in the target language, and thus produce a vector of similarities between the translation and the documents in the target reference corpus.

We then use the following hypothesis for which we provided convincing evidence in (Rajman and Hartley, 2001) and in (Besançon and Rajman, 2002): the structure of the vector space spanned by the documents in the source reference corpus is well preserved by translation in the target language, and thus is very similar to the one of the vector space spanned by the documents in the target reference corpus. This in turn implies the following useful property: if the semantic content of a document is well preserved during translation, then the vector of similarities associated with this document in the source vector space should be very similar to the vector of similarities associated with its translation in the target vector space.

We therefore use the Euclidean distance between the vectors of similarities as an indicator of the quality of the preservation of the semantic content after translation, and,

[1] http://www.xrce.xerox.com/ats/xelda/

[2] As reference corpus, we used the JOC corpus containing 6729 documents comprising questions and answers to the European Community as published in the *Journal Officiel de la Communauté Européenne*.

in order to have a score (hereafter called the D-score) that varies in the same direction as quality (the higher the value, the higher the quality), we actually use an inverse function of the distance:

$$\textbf{D-score}(D_{tgt})=1(1+d(V_{src}(D_{src}),V_{tgt}(D_{tgt})))$$

where $V_{src}(D_{src})$ (or $V_{tgt}(D_{tgt})$) is the vector of similarities for the source document D_{src} (or the translation D_{tgt}) in the source (or target) semantic vector space.

4. Computing the rankings

The very first problem we face when trying to predict overall rankings is the production of the reference overall rankings that should be predicted. For the six systems evaluated in our new experiment, the raw evaluation material consists of the 12x10=120 rank series produced by the evaluators. Each of these series corresponds to a ranking of the six MT systems made by one of the 12 evaluators (on the basis of one of the 10 translation sequences he/she was provided with), and can be therefore considered as one individual preference indication (or vote) over these systems. The overall ranking we are looking for is then the one that optimally globally represents the set of 120 individual preferences.

Once the overall reference rankings are defined, we further have to decide how the automatically computable scores (X-score and D-score) should be taken into account to derive overall rankings that will be used as predictors for the reference rankings. In this case, the raw evaluation material consists of the 100 series of six scores produced for each of the 100 available series of translations. Here again, we consider each of the sequences of scores as one individual preference indication (or vote) over the MT systems. The targeted overall ranking is therefore once again the one that optimally globally represents the set of 100 available individual preferences.

In both cases we are faced with the problem of optimally aggregating individual preferences. In fact, this problem is a very hard mathematical problem well known to economists and political scientists (in the domain of voting theory for example), and for which it has been shown (Arrow, 1963) that there is no indisputable optimal solution. However, various simple aggregation techniques can be used to produce sub-optimal solutions. For our experiments, we have considered three such aggregations techniques: (1) ranking by average scores (average score ranking or ASR); (2) ranking by average ranks (average rank ranking or ARR); and (3) ranking by average binary preferences (average preference ranking or APR). We did not consider other more sophisticated aggregation techniques, such as approval voting or multiple round voting schemes.

4.1. Average score and average rank ranking (ASR and ARR)

ASR has the advantage of great simplicity: for each of the systems, its scores are averaged over all the available score sequences and the resulting average values are used to produce the overall ranking of the systems. If the original evaluation data consists of rank sequences instead of score sequences (as it is the case for our human evaluations), the rank values need to be transformed in order to vary in the same way as scores (the bigger, the better). For our experiments, we have simply multiplied all the rank values by –1.

ARR is another very simple aggregation technique: for each of the systems, its ranks are averaged over all the available rank sequences and the resulting average values are used to produce the overall ranking of the systems. If the original evaluation data consists of score sequences instead of rank sequences (as it is the case for the X-scores and D-scores), each of the score sequences is first transformed in a rank sequence.

Note that in the case of original evaluation data consisting of ranks, ARR and ASR necessarily produce exactly the same results. Computing ASRs is then of no interest and therefore, for the production of the reference overall rankings (i.e. those derived from the human evaluations), we compute only ARRs. This is not the case if the original evaluation data consists of scores, in which case ASRs and ARRs do not necessarily coincide. One potential advantage of ARR over ASR is the fact that the former are less sensitive than the latter to outlying values. However, for well-conditioned scores, we observed that ASRs tend to be more stable than ARRs in the case of bootstrap replication (see section 5). We therefore systematically produced both ASRs and ARRs for the overall rankings derived from the X-scores and D-scores.

4.2. Average preference ranking (APR)

APR represents another way of producing an overall ranking as the synthesis of a set of several individual rankings. The method for producing an average preference ranking is quite simple. The individual rankings are first converted into a set of binary comparisons on translation pairs. For each of the pairs i:j, we then compute how many times i has been ranked higher than j and the resulting average ranking is the one corresponding to simple majority decisions for all the pairs. By convention, for each pair i:j, an associated value 1 (or -1) indicates that element i has a better (or worse) rank than element j. For partial rankings, a value 0 indicates that for the pair i:j, no ranking decision has been made.

The APR rankings are in fact far more complicated than they appear. Indeed, with the procedure described above, it is not guaranteed that the resulting set of binary decisions effectively corresponds to a ranking.

Two types of problems may arise. First, some of the average binary decisions might not be taken on the basis of a simple majority vote because the number of votes for each of the 2 possible decisions (1 and -1) are equal; in such a case, a partial ranking is produced and the corresponding decision value is set to 0 as already mentioned earlier.

Second, the resulting set of average binary decisions might not correspond to a ranking. This is due to the fact that some transitivity relation is not verified .This is the well known Condorcet paradox, stating that the aggregation of rational – i.e. verifying transitivity – preference sets can result in a irrational set of preferences (Saari, 1999). One possibility for dealing with such situations is to relax the binary decisions that violate transitivity to ‘unknown’ (value 0), again turning the set of binary decisions into a partial ranking.

In our experiments, we used APR mainly as a second step, to produce partial rankings when the total rankings produced as ASRs or ARRs appear too unstable.

5. Quality measures used for the rankings

5.1. ASR and ARR stability

Since the average rankings analyzed in the previous section are derived from a limited number of preference sequences (either 120 rank sequences or 100 score sequences), it is essential to have evidence of how sensitive the overall rankings are to the specific raw evaluation data they have been derived from. One standard method for testing the stability of results derived from finite sets of data is bootstrapping. The general idea of the method is very simple: the original data set is used to produce a large number of random samples (called bootstrap replicates) of the same size N as the original data set. The random samples are used to produce the result for which we would like to estimate the stability, which is then measured by a statistic computed on the set of bootstrap duplicates.

In our case, the random samples are simply built by N times randomly selecting among the original N raw evaluation data sets. Note that it often happens that the same raw evaluation data is duplicated several times in the bootstrap replicates. To measure stability, we simply compute how many times the evaluated ranking is produced among all the rankings derived from the bootstrap replicates. Additional insight into stability is provided by checking whether the overall ranking produced is indeed the most frequent one among the replicates, and by computing the frequency of the second most frequent one.

In our experiments, for each of the scores, we produced 10,000 bootstrap replicates of the original evaluation data set.

As we have already observed when we compared the ASR and ARR rankings, an important part of the instability of the ranking produced comes from the fact that the data they derive from simultaneously substantiates not one single overall ranking but in fact several competing ones.

In order to analyse such phenomena, it is important to be able to produce the different rankings that are substantiated by a given evaluation data set. One possibility would be to explore the different rankings frequently produced during bootstrapping. This method however is not optimal, as it does not allow us to make use of the fact that the several competing rankings probably share important common parts (i.e. a subset of identical pairwise orders). A better approach is to focus on the reliability of the average preference rankings, as we see in the next section.

5.2. APR reliability

As it was the case for the ASR and ARR rankings, the issue of reliability is an important concern for the APR rankings. However, the approach for measuring reliability is quite different. As the APR is built by deriving the average binary decision from the counts of the individual binary decisions, a statistical test can quite easily be used instead of the simple majority rule. More precisely, this corresponds to replacing the rule: ‘Select a decision if the proportion of individual decisions it corresponds to is greater than a half’ by the statistical test ‘Select a decision if the proportion of individual decisions it corresponds to is significantly greater than a half’. As we are dealing with proportions, we used a Student test. The level of confidence that can be associated with a produced APR is then the lowest of the levels of confidence that were used to select the average binary decisions it consists of.

5.3. Comparing rankings

Another interesting point is to define a proximity measure on the rankings in order to quantify the quality of the ranking predictions, i.e. how well the ranking derived from an automatically computed score corresponds to the one derived from the human evaluations.

A possible distance on rankings is the Hamming distance, which computes the number of pairwise differences. We extended this definition to partial rankings by adding a value of 0.5 (or 0) for all the pairwise differences that involve exactly one pair (or two pairs) for which no preference decision has been taken. With such a definition, the distance between the partial rankings R1=(2[4 3]1) and R2=([2 4][1 3]) (where the numbers identify the ranked entities) corresponding, respectively, to the two sequences of six pairwise comparisons S1 = (–1 –1 –1 +1 +1 0) and S2 = (–1 0 –1 +1 0 –1), is d(R1,R2) = 1.5.

As defined above, the distance d(R1,R2) between two rankings over N elements is necessarily between 0 and N(N-1)/2, and can therefore easily be transformed into a similarity s(R1,R2) by: s(R1,R2) = 1 – 2d(R1,R2)/N(N-1). With this definition, the distance d(R1,R2) = 1.5 corresponds to a similarity s(R1,R2) = 75%.

Finally, two additional interesting measures of the quality of the predicted ranking are its precision and recall. The precision is the proportion of binary comparisons correctly predicted among all the binary relations predicted. The recall is the proportion of binary comparisons correctly predicted among all the binary relations in the reference ranking. In the above example, if we consider R1 as the reference ranking, we have $P_{R1}(R2)$ = 75% and $R_{R1}(R2)$ = 60%.

6. Description of the results obtained

6.1. Overall rankings derived from the human evaluations

For the human evaluations (hereafter called the H-ranks), both ARR (and of course ASR) and APR produce the same overall ranking H:

H: RV SY CD GL MS XS

The corresponding stability/reliability measures are:

	stability	reliability
ASR/ARR	57%	
APR		41%

Table 1: stability/reliability measures (H-ranks)

The derived ASR/ARR ranking is indeed the most frequent one among the bootstrap replicates, the second most frequent one being more than 3 times less frequent. In addition, the most complete partial ranking that can be produced by APR is:

H_p: RV [SY CD] GL MS XS with a reliability of 63%

which indicates that the least reliable pairwise preference (in the complete ranking H) is SY > CD and that relaxing it (i.e. not taking a decision on the relative position of SY and CD) provides the partial ranking H_p with a reliability of 63%.

6.2. Overall rankings derived from the X-scores

Again, all 3 aggregation methods (ASR, ARR and APR) produce the same ranking X:

X: RV SY CD MS GL XS

The corresponding stability/reliability measures are:

	stability	reliability
ASR	52%	
ARR	34%	
APR		83%

Table 2: stability/reliability measures (X-scores)

The derived ASR and ARR rankings are indeed the most frequent ones among the bootstrap replicates, the second most frequent being more than 3 times less frequent for ASR and 1.5 less frequent for ARR. In addition, the most complete partial ranking that can be produced by APR is:

X_p: RV SY CD [MS GL] XS with a reliability of 89%.

6.3. Overall rankings derived from the D-scores

Again, all three aggregation methods (ASR, ARR and APR) produce the same ranking D:

D: GL SY RV MS CD XS

The corresponding stability/reliability measures are:

	stability	reliability
ASR	75%	
ARR	57%	
APR		45%

Table 3: stability/reliability measures (D-scores)

The derived ASR and ARR rankings are indeed the most frequent ones among the bootstrap replicates, the second most frequent being more than 2 times less frequent for ASR and 2 less frequent for ARR. In addition, the most complete partial ranking that can be produced by APR is:

D_p: (GL SY) RV MS CD XS with a reliability of 99%.

6.4. Quality of the predictions

The following tables summarize all the quality measures for the prediction of H (resp. H_p) by X and D (resp X_p and D_p).

	distance	similarity	precision	recall
X	1	93.3%	93.3%	93.3%
D	5	66.7%	66.7%	66.7%

Table 4: prediction of H

	distance	similarity	precision	recall
X_p	0.5	96.7%	100%	93.3%
D_p	4.5	70%	71.4%	66.7%

Table 5: prediction of H_p

7. Interpreting the results

7.1. Reliability of the ranking methods

All three ways of deriving an overall ranking – ASR, ARR and APR – systematically produce the same ranking. This strong agreement is a good indicator of the robustness of the method and simplifies the interpretation.

7.1.1. ASR and ARR

As far as the robustness of ASR and ARR is concerned (according to bootstrap replication), the scores obtained are reasonably good – almost always above 50% and better than the scores obtained by (Rajman and Hartley, 2001), which were all below 47%. Even the ranking produced by the single low value – ARR(34%) for the X-score (Table 2) – is indeed the most frequent in the 10,000 bootstrap replications performed and the second most frequent ranking (which has a score of 23%) is more than 1.5 less frequent. In short, the rankings produced are without any doubt reliable in that they are representative of the specific data used for the experiment.

7.1.2. APR

As far as the reliability of APR is concerned, the scores obtained are lower than those obtained previously. However, to put this fact in perspective, it is important to remember the interpretation of the reliability of APR: the APR reliability scores essentially measure how well the overall ranking produced represents the underlying individual rankings produced for each of the documents,

For the ranking derived from the human evaluations (H-ranks) in particular, the APR reliability score is therefore to some extent indicative of inter- and intra-annotator variability. Thus, the low score of 41% obtained for the rankings derived from the H-ranks (Table 1) in comparison to the scores of over 80% obtained previously for the A-, F-, and I-scores could probably be taken to reflect the fact that the annotators used in our experiment were less trained than those participating in the DARPA experiments.

The APR reliability for the X- and D-scores is fully comparable with that obtained previously. The X-scores produce an overall ranking which shows a quite good agreement of 83% (Table 2) and the weakest pair-wise comparison (MS > GL) is indeed the one that does not correspond to the ranking derived from the H-scores;

The D-scores produce a ranking with a quite low agreement of 45% (Table 3), but this is due solely to the difficulty of discriminating between the first two systems (GL and SY). The partial ranking that does not decide any relative ordering for these two systems nonetheless corresponds to an excellent overall agreement (99%).

7.2. Predictive power of the automated scores

7.2.1. X-score

The X-scores clearly represent a very good predictor of the ranking derived from the human evaluations (H-ranks). The distance between the H-ranking and the X-ranking is 1, corresponding to a similarity of 93.3%, a precision of 93,3% and a recall of 93.3% (Table 4). If we restrict ourselves to the most complete partial ranking, these values improve to a distance of 0.5, a similarity of 96.7%, a precision of 100% and a recall of 93.3%.

These results confirm those obtained by (Rajman and Hartley, 2001), but put them in a fresh perspective. In the previous experiments, the X-scores were optimised on the data, and so it was hardly surprising that they yielded such good results. We doubted that such ad hoc scores would generalise well to other data.

However, the formula used to compute the X-scores were derived only from the outputs of the five 'old' MT systems, excluding the outputs of Reverso (See Section 3.1). The fact that they still perform extremely well for a new set of data to which they had not been tuned indicates that they might have a generality that was not suspected initially.

7.2.2. D-score.

The D-scores appear to be quite disappointing as predictors for the rankings derived from the human evaluations. The distance between the D-ranking and the H-ranking is 5, corresponding to a similarity of 66.7%, a precision of 66.7% and a recall of 66.7% (Table 5). For the most complete partial ranking, these values are respectively: distance = 4.5, similarity = 70%, precision = 71.4% and recall = 66.7%.

The D-scores therefore clearly do not confirm the promise they offered in (Rajman and Hartley, 2001).

8. Conclusions

Our general conclusion from these results is that we have identified an excellent automated predictor for the ranking of MT systems. This predictor relies on purely syntactic features; a predictor based on more semantic features does not perform as well, despite earlier expectations of its robustness.

Moreover, the method does not require the production of reference translations by human translators, up to four in the case of (Papineni et al., 2001). These authors limit, however, the expense of this part of the process by translating only 500 sentences, leaving themselves with a rather small data set.

We hold to our earlier intuition (Rajman and Hartley, 2001) that humans recognise translations produced by machines on the basis of syntactic features and rank them accordingly. We offered this in order to account for the large discrepancy between the manual rankings of the human translations (relative to the machine translations) and the automatically produced rankings. While the former consistently ranked the human translation in first place, the latter did not.

In other words, we believe that a machine translation is perceived as 'good' on the grounds of its well-formedness, rather than its semantic plausibility. This viewpoint finds some support in the results of (Papineni et al., 2001), who observed that bilingual judges were more lenient and tolerant of a bad translation than monolingual judges because their perceptions of its fidelity moderated their penalisation of its ill-formedness. Recall that the judges in our experiments did not know the source language (See Section 2.2) and so, as effective monolinguals, had no basis for moderation.

We are also basing our intuition on experiments which required judgments on whole texts rather than on single sentences. Clearly, a given MT system may produce a translation of a single sentence that is indistinguishable from a human translation and thus qualify for the highest mark on the rating scale. On that used by (Papineni et al, 2001), that equates to: 'Has writing proficiency equal to that of a well-educated native'. But sustaining such a performance over a number of sentences in a naturally occurring text is highly unlikely. Errors made by MT systems are qualitatively different from those made by a human translator, even by one who does not have the target language as their mother tongue; (Papineni et al., 2001) found a marked gap between the worst human translation and the best machine translation in their experiment.

There are clearly general questions about the how the profile of the evaluators (monolingual or bilingual), the nature of the input (sentences or texts) and the terms of the scale affect the results. For example, how the evaluators employed by (Papineni et al., 2001) were able to award the lowest mark on their scale – 'Writing tends to be a collection of sentences on a given topic and provides little evidence of conscious organisation' – when they were judging single sentences selected at random, is unclear to us. It is also hard to see how the criterion 'Can write simply about a very limited number of current events or daily situations' can be applied to single sentences. Moreover, it relates to the subject matter, which is given by the source text, and so could harshly penalise an accurate translation of simple input.

More work is clearly called for to better understand the implications of using single sentences or whole texts as the input to the evaluation tasks, and of using evaluators with no or considerable knowledge of the source language.

Another open research question is whether we can extend our evaluation framework with a good automated predictor that reliably distinguishes between a machine-generated text and a human-generated text, without relying on several reference translations produced by human translators. Evaluation of the applicability of the method presented in (Papineni et al., 2001) to our data is currently ongoing.

9. References

Arrow, K.J., 1963. *Social Choice and Individual Values*. Wiley: New York.

Brew, C. and H. Thompson, 1994. Automatic Evaluation of Computer Generated Text. *Procs. ARPA/ISTO Workshop on Human Language Technology*, 104-109.

Besançon, R. and Rajman, M., 2002. Evaluation of a Vector Space similarity measure in a multilingual framework. *Procs. 3rd International Conference on Language Resources and Evaluation, Las Palmas, Spain*.

Carroll, J.B., 1966. An experiment in evaluating the quality of translations. In J. Pierce (ed.) *Language and machines*. Report by ALPAC. NASNRC, 67-75.

Papineni, K., S. Roukos, T. Ward and Z. Wei-Jing, 2001. Bleu: a method for automatic evaluation of machine translation. *IBM Research Report RC22176*. Yorktown Heights, NY: IBM.

Rajman, M. and T. Hartley, 2001. Automatically predicting MT systems rankings compatible with Fluency, Adequacy or Informativeness scores. *Procs. 4th ISLE Workshop on MT Evaluation*, *MT Summit VIII*, 29-34.

Reeder, F., K. Miller, J. Doyon, and J. White, J. 2001. The naming of things and the confusion of tongues: an MT metric. *Procs. 4th ISLE Workshop on MT Evaluation, MT Summit VIII*, 55-59.

Saari, D., 1999. Explaining all three-alternative voting outcomes. *Journal of Economic Theory. 87*, 313-355.

Salton,G. and Buckley, Cm 1988. Term Weighting Approaches in Automatic Text Retrieval. *Information Processing and Management, 24*, 513-523.

Vanni, M. and K. Miller 2001. Scaling the ISLE framework: validating tests of machine translation quality for multi-dimensional measurement. *Procs. 4th ISLE Workshop on MT Evaluation, MT Summit VIII*, 21-27.

White, J., T. O'Connell and F. O'Mara, 1994. The ARPA MT evaluation methodologies: evolution, lessons and future approaches. *Procs. 1st Conference of the Association for Machine Translation in the Americas*, 193-205.

White, J. and M. Forner, 2001. Predicting MT fidelity from noun-compound handling. *Procs. 4th ISLE Workshop on MT Evaluation, MT Summit VIII*, 45-48.