How Much Data is Needed for Reliable MT Evaluation? Using Bootstrapping to Study Human and Automatic Metrics

Paula Estrella
ISSCO/TIM/ETI
University of Geneva
40, bd. du Pont-d'Arve
1211 Geneva, Switzerland
paula.estrella@issco.unige.ch

Olivier Hamon
ELDA
55-57, rue Brillat-Savarin
75013 Paris, France, and
LIPN, University of Paris XIII
99, av. J.-B. Clément
93430 Villetaneuse, France
hamon@elda.org

Andrei Popescu-Belis
ISSCO/TIM/ETI
University of Geneva
40, bd. du Pont-d'Arve
1211 Geneva, Switzerland
andrei.popescu-belis@issco.unige.ch

Abstract

Evaluating the output quality of machine translation system requires test data and quality metrics to be applied. Based on the results of the French MT evaluation campaign CESTA, this paper studies the statistical reliability of the scores depending on the amount of test data used to obtain them. Bootstrapping is used to compute standard deviation of scores assigned by human judges (mainly of adequacy) as well as of five automatic metrics. The reliability of the scores is measured using two formal criteria, and the minimal number of documents or segments needed to reach reliable scores is estimated. This number does not depend on the exact subset of documents that is used.

Introduction

A large number of metrics have been proposed to evaluate machine translation systems, as summarized for instance in the FEMTI framework (Estrella et al. 2005). However, comparatively fewer studies have been devoted to the test data needed by evaluation metrics, and in particular to the amount of data that is required to obtain reliable scores. Indeed, both human and automatic metrics generally assign a score to each translated segment, often comparing it to one or more reference translations of the same segment. While it is commonly acknowledged that a "large" number of segments is needed to obtain statistically significant scores, the goal of this article is to provide empirical estimates of this amount based on observations from a recent MT evaluation campaign.

This article thus analyzes the effect on MT evaluation scores of varying test set sizes, and proposes formal methods to study the robustness of metrics as the numbers of test documents increases. The article first discusses related work, and then the data—systems, test data and scores—used throughout the study. The bootstrapping technique is then introduced, which will be used to compute average values and standard deviations for human metrics (mainly adequacy) as well as for automatic metrics (BLEU, NIST, mWER, mPER and GTM) and its application to the study of reliability is then explained. The scores obtained for various document samples are then discussed along with the effect of document ordering. A method to compute a sufficient number of documents for each metric is further proposed, and its results for both human and automatic metrics are finally discussed.

Studies of the Required Size of Test Data

Studies regarding the influence of the test data on the reliability of scores are not common in MT evaluation. For instance, few guidelines indicate the number and size of documents to be used in an evaluation, or the effect on scores of various sizes of the test set. This is unlike the case of training data for statistical NLP systems, where studies of the influence of size of training data on output quality are more frequent, e.g. for statistical or example-based MT (Germann 2001), as well as for many other domains, e.g. question answering systems (Clarke et al. 2002; Dumais et al. 2002).

Closer to our present goal, Elliott et al. (2003) explicitly attempt to answer the question of how much text to include in a multilingual corpus for MT evaluation, given the general hypothesis that more text would lead to more reliable scores. Their work concerns human metrics—fluency, adequacy and informativeness—and mainly focuses on the ranking of systems based on the results of the FR/EN, SP/EN and JP/EN DARPA 1994 MT evaluation campaign. The scores were compared for an increasing number of texts, starting with 1 and ending with 100 texts, the average length of texts being 350 words. Based on an empirical assessment of score variation, the authors estimate that systems could be reliably ranked with around 40 texts (ca. 14,000 words), and that using ten texts already separate the highest and the lowest ranked systems. These figures can be compared with the amounts used in a number of previous evaluations which generally use several hundred to several thousand sentences (Elliott et al. 2003: Table 1).

Zhang and Vogel (2004) also studied the influence of the amount of test data on the reliability of automatic metrics, focusing on confidence intervals for BLEU and NIST scores. They used the data of the CH/EN track of the TIDES 2002 MT evaluation campaign (100 documents of 7-9 sentences each), with the output of the 7 participating systems and 4 reference translations. Their results show that BLEU and NIST scores become stable when using around 40% of the data (around 40 documents or 300 sentences), though stability is defined here in terms of the distance between scores of different systems.

These two studies suggest that an evaluation can be *reliably* performed with less text than is often used. We reinforce this hypothesis here, and propose a formal method to estimate the necessary amount of test data, which evaluators could use to assess the amount of test data needed by a given metric.

Data and Metrics: CESTA EN/FR Campaign

The experiments presented here were done using the test data, system outputs and evaluation metrics from the French MT evaluation campaign, CESTA (Hamon et al. 2006). The test data comes from the first run of the campaign, on the English to French translation task, in which five systems have participated. The results of the systems are anonymized, and for the present purpose the systems will simply be referred to by the codes S1 to S5 in no particular order. The systems participating to this run were: Comprendium, RALI / University of Montreal, Reverso / Softissimo, SDL, and Systran.

One of the goals of the CESTA campaign was to validate the use of automatic evaluation metrics with French as a target language, by comparing the results of well-known automatic metrics with fluency and adequacy scores assigned by human judges. The following automatic metrics were applied to the translations produced by the five systems participating in the CESTA campaign, with four reference translations: mWER, multiple reference Word Error Rate (Niessen et al. 2000), mPER, position independent Word Error Rate (Tillmann et al. 1997), BLEU (Papineni et al. 2001), NIST version of BLEU (Doddington 2002). We added to this experiment the GTM (General Text Matcher) metric (Turian et al. 2003).

The test data, i.e. the corpus created for the CESTA evaluation campaign, English to French first run, consists of 15 documents from the Official Journal of the European Communities (JOC, 1993), with a total of 790 segments or sentences, with an average of 25 words per segment (Hamon et al. 2006). The data consists of transcribed questions and answers in a parliamentary context, and since no particular domain was targeted when putting together the corpus, the CESTA campaign considered this as *general domain* data.

The goal of the experiments presented here is to observe how the average scores obtained by human judges and automatic metrics evolve, as more documents are incrementally added to the evaluation corpus. More specifically, the experiments attempt to test whether these scores stabilize towards their final value as more documents are added, and to find a method to determine a sufficient amount of test data to reach this value with reasonable precision.

Bootstrapping over MT Evaluation Scores

This section describes the bootstrapping technique used to compute average scores and related statistics, first from a theoretical point of view, then in the setup used here.

Estimating Variables Using Bootstrapping

Bootstrapping is a statistical technique that is used to study the distribution of a variable based on an existing set of values (Efron and Gong 1983). This is done by randomly resampling *with replacement* (i.e. allowing repetition of the values) from the full existing sample and computing the desired parameters of the distribution of the samples. The method has the practical advantage of being easy to implement and the theoretical advantage of not presupposing anything about the underlying distribution of the variable. A simple programming routine can thus calculate the estimators of the mean, variance, etc., of any random variable distribution. Moreover, when the original sample is resampled a large number of times (theoretically close to infinite), the law of large numbers ensures that the observed probability approaches (almost certainly) the actual probability. The bootstrapping algorithm can be summarized as follows:

1. Given a sample $X = (X_1, X_2, \ldots, X_n)$ from a population **P**, generate N random samples of size n by drawing n values from the sample, with replacement (each value having probability $1/N$).
2. The resulting population $\mathbf{P}^*$, noted $X^* = (X_1^*, \ldots, X_N^*)$, with $X_i^* = (X_{i1}^*, X_{i2}^*, \ldots, X_{in}^*)$, $i = 1..N$, constitute the N bootstrapped samples.
3. If the original estimator of a given population parameter was $\theta(X)$, with the bootstrapped samples we can calculate the same estimator as $\theta(X^*)$.

An important parameter for bootstrapping is N, the number of bootstrapped samples, or the number of times the process is repeated. This number should be large enough to build a representative number of samples. It appears that, for instance, $N = 200$ leads to slightly biased estimations (Efron and Gong 1983; Zhang and Vogel 2004), so a larger N is preferred, for example $N = 1{,}000$ (Efron and Gong 1983; Koehn 2004) or even $N = 10{,}000$ (Bisani and Ney 2004). Based on these examples, we decided to use $N = 1{,}500$.

Another source of error in inference statistics is the error induced by using a particular sample to represent a whole (unknown) population. In the present case, this amounts to considering that the scores on the 15 documents (or 790 segments) are fully representative of a system's performance on this type of text.

Application to MT Evaluation Scores

In the MT field, bootstrapping has been mainly used to estimate confidence intervals for automatic metrics and to compute the statistical significance of comparative performance of different MT systems, e.g. using the BLEU (Koehn 2004; Kumar and Byrne 2004; Zhang and Vogel 2004) or WER metric (Bisani and Ney 2004).

Here, bootstrapping will be used to compute reliable estimators for different automatic metrics for MT, namely mean, standard deviation (often expressed as a percentage of the mean) and confidence intervals (based on standard deviations) for the mean of the bootstrapped sample.

For the application of bootstrapping in MT, the original sample X is the set of text segments arranged in documents, each segment being accompanied by a list of scores obtained by each MT system, according to the metrics mentioned in the previous section.

Described in pseudo code, the routine computing the various estimators is particularly simple: M is the number of segments to be considered, `sample[m]` is the m-th element of the *sample* while `sample*` is a pointer to the list of bootstrapped samples:

```
for(n=0; n<N; n++){
    for(m=0; m<M; m++){
       sample[m] = selectRandSeg();
    }
    scoreList[n] = calcMetric(sample*);
}
calcEstimators(scoreList);
```

The test corpus consists of 15 documents, noted d_1 … d_{15}. Despite the slight differences between their lengths, a document is the most reasonable incremental step in our bootstrapping study, since a document offers in theory the highest topical homogeneity across sentences—as the exact topics may change from one document to another.
The following algorithm evaluates the systems and computes parameters related to each of the metrics, for each document subset D, incrementally constructed by adding one document at a time, starting with $D = \{d_1\}$.

1. Select one system and one metric to be applied, say S_k and m_i , where $k = 1, 2, 3, 4, 5$ and $m_i \in$ {adequacy, fluency, BLEU, NIST, GTM, mWER, mPER}.
2. Apply m_i to each translated segment of D output by S_k.
3. Bootstrap N times to compute mean, relative standard deviation and confidence intervals for the mean score of m_i.
4. Add one more d_j to the evaluation set D, following the order $j = 2, .., 15$ (or a random order).
5. Repeat steps 2 – 4.

The process is of course repeated for every metric and every system. At the end of the process, the mean, standard deviation and confidence intervals are available for each system and each metric. The following sections make use of these results to analyze the sufficient size of the subset of documents D based on formal criteria to get reliable scores.

Variation of Average Scores Depending on the Size of the Test Data

The results of bootstrapping with 1, 2, and up to 15 documents are discussed in this section, in terms of standard deviation and comparison with the global scores obtained when the full test data (15 documents) is used. The results are given first for the human metrics, then for automatic ones. The next section will then attempt to determine the minimal number of documents leading to evaluation scores that are not substantially different from those obtained on the full data set.

Human Metrics

Bootstrapping was performed on human metrics, computing the average scores for one document first, then for two, three, etc. The order of the documents was firstly the one used in the CESTA campaign, and secondly, in a different experiment, a random order. Figure 1 shows the evolution of average adequacy scores computed over 1, 2, …, 15 documents, for the five systems evaluated in the CESTA campaign (fluency values show a similar pattern).
The observed trend is that, after some initial variation due to the heterogeneity of documents and to the systems' performance, the scores quickly reach their final values over the entire test set. A notable exception is system S5, having scores particularly low on the first document, which penalizes also its performance on $\{d_1, d_2\}$, $\{d_1, d_2, d_3\}$, etc.
The lower performance of S5 on d_1 may simply be due to the inevitable variation of system performance on different texts (e.g. caused by missing vocabulary) as no other cause could be identified. Similarly, S1 performs better on documents d_4 and d_5; however, in the case of S5 and d_1, the lower performance on the first document of the series is much more perceptible graphically.

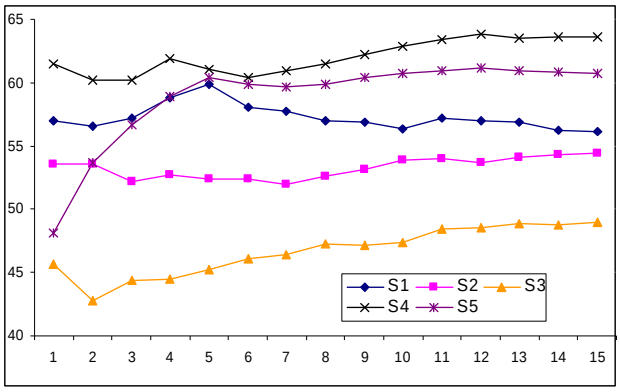

Figure 1: Average adequacy values (on a 0-100% scale) for systems S1 to S5, computed on 1, 2, …, 15 documents

Automatic Metrics

Turning now to average scores of automatic metrics, Figure 2 and Figure 3 display the scores obtained using the GTM and mWER metrics (error rate means that lower scores are better). These figures display a similar pattern: after chaotic variation on the first subsets of documents, the ranking of the systems becomes quickly close to the final one, and the average scores reach their final values quite soon as well.

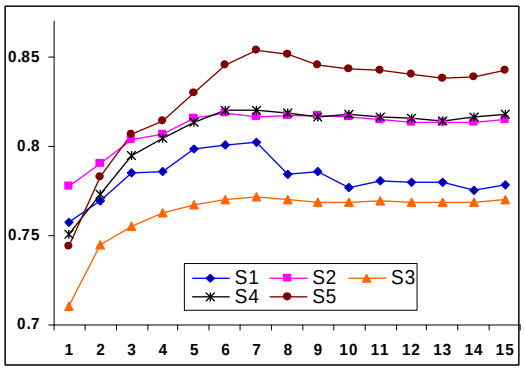

Figure 2: GTM scores (on a 0-1 scale) for systems S1 to S5, computed on 1, 2, …, 15 documents

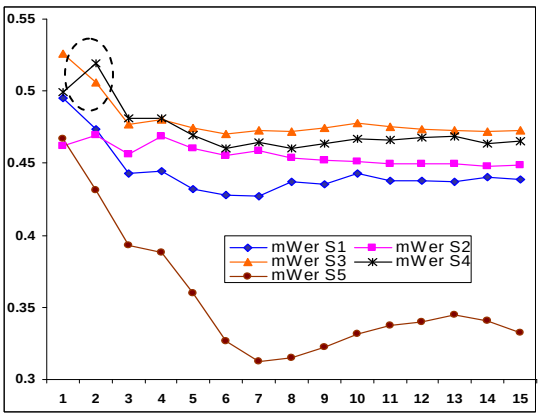

Figure 3: mWER scores for systems S1 to S5, computed on 1, 2, …, 15 documents (error rates: lower is better)

The other metrics that were studied (not shown here) also have similar behaviors. So, despite having very different mechanisms—n-grams *vs.* precision-recall *vs.* edit

distance—metrics behave similarly across systems, i.e. they show coherent rankings for the different subsets of documents. For instance, regarding their final results (for 15 documents), all automatic metrics rank S5 as the best and S3 as the poorest system: these average results obtained using bootstrapping are thus coherent with the official results of the CESTA campaign.

The figures also indicate a significant qualitative agreement between human judgments and automatic metrics (Hamon et al. 2006). Document d_1 appears to be "difficult" for most systems, but especially for the best system S5, while system S1 performs quite well on documents d_1-d_6 but then its scores decrease.

Specific Domain Experiment

To go further, we tested this method on the data used in the second run of the CESTA campaign, consisting of documents from the health domain. The size of the documents and average number of words per segment are similar to the corpus of the general domain and also five systems were evaluated.

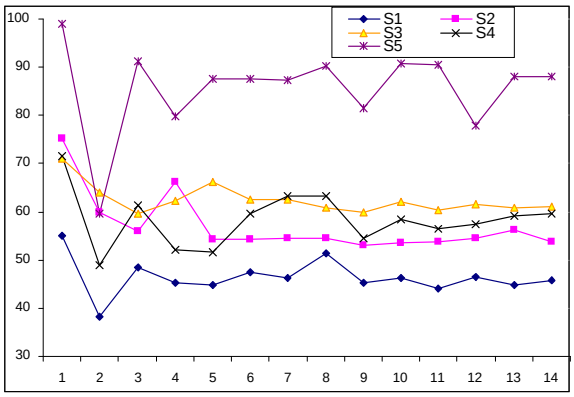

Figure 4: Health domain adequacy results for systems S1 to S5, for 1, 2, …, 14 documents, in random order

As in the general domain case, scores stabilize around 10 documents and the definitive ranking is visible around 9 documents, the scores being quite chaotic before this point, shown in Figure 4. Although the trend is similar to that of the general domain, curves seem to be more chaotic; this could be explained by the specific characteristics of the data but in general the conclusions for human metrics of previous sections are valid for this domain.

The Number of Documents Needed to Compute Reliable Scores: Two Criteria

In this section we propose a method to estimate both the minimum number of documents that are sufficient to obtain reliable scores, and the theoretical, maximum number of documents needed to minimize the standard deviation.

We consider that an average evaluation score is *reliable* when the STDEV computed using bootstrapping is close to the STDEV obtained using the whole test set of 15 documents. In addition, since our limit is 15 documents, we have no evidence of what happens beyond that limit, hence raising the question if there is any other way to predict or estimate how many additional documents would be necessary to obtain more reliable results, i.e. with a lower STDEV. To provide additional guidelines about the use of the CESTA corpus, we propose two methods to assess reliability. The first one estimates the number of documents needed for the STDEV to reach "acceptable" values, and the second one estimates the number of documents needed to reach an SDTEV close to zero, assuming that its tendency is accurately described by the values obtained with 1, 2, …, 15 documents.

The Evolution of Standard Deviation with the Number of Documents

The "convergence" of scores towards their final value when the number of documents is increased can also be studied through the standard deviation (STDEV) of each average score, obtained using bootstrapping. Indeed, the 1500 samples of documents obtain different average scores, therefore the standard deviation across these samples is not zero, but is expected to decrease as the number of documents increases.

For instance, Figure 5 shows the average NIST score and the confidence interval (based on standard deviations) for S1. The STDEV (and the width of the confidence interval) decreases from d_1 to d_7, but as performance varies slightly with d_8, the STDEV increases again at this point, and decreases again afterwards. However, it appears in this case that the width of the confidence interval remains below ±0.3 starting with the 6th document. Overall, it appears that STDEVs do not change much after the 9th document.

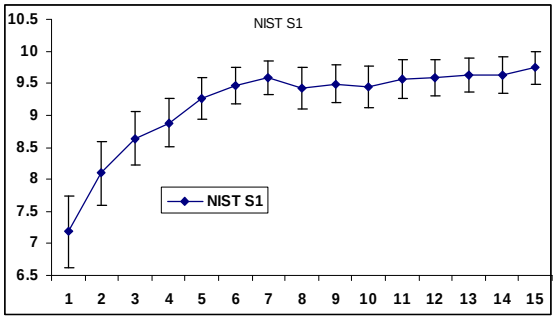

Figure 5: Average scores and confidence intervals (based on standard deviations) for the NIST metric applied to S1, computed on 1, 2, …, 15 documents

The average scores obtained using bootstrapping for each subset of documents also offer insights about the systems' output, not based on the scores themselves but rather on their variation. As an example, in Figure 2 and Figure 3, GTM and mWER disagree for S4 when adding document d_2: both mWER and GTM increase while they are not expected to (shown in dashed circle). To explain this observation, S4's output was inspected, and it appeared that many words are followed by their synonyms in parenthesis, for instance (in French*): "l'avenir (le contrat à terme)", "la Réglementation (le Règlement)", "la fourniture (l'apport, la provision)"*. This might be one reason why the BLEU score also decreases from $\{d_1\}$ to $\{d_1, d_2\}$ due to the brevity penalty. The mWER score suggests that S4 does not use the same vocabulary as the references. Therefore, as NIST and GTM do not penalize too much and reward more for longer matches, the scores increases from $\{d_1\}$ to $\{d_1, d_2\}$, as shown in Table 1. Interestingly, S4 is ranked best according to the human scores, who

might have been positively biased by the words in parenthesis.

NIST	BLEU [%]	GTM [%]	mWER [%]	mPER [%]
7.2612 ±0.54	0.326 ±0.06	0.7509 ±0.04	0.499 ±0.06	0.405 ±0.04
7.8868 ±0.42	0.3192 ±0.04	0.7732 ±0.02	0.518 ±0.05	0.410 ±0.041

Table 1: Average scores and confidence intervals for S4 evaluated with documents d_1 and d_2 using all the automatic metrics

Influence of Document Ordering

As we have seen, the scores obtained using the very first documents in the series above have a great influence on the overall pattern of the results. It is therefore normal to explore the influence of document order by changing the order of documents d_1… d_{15} and repeating the analyses above. To completely discard the effect of document ordering, one could perform the analyses for every possible ordering, i.e. 15 factorial times (15!), or at least a sufficiently representative number of times.

Here, we performed the same task two more times (run 1 and run 2) choosing a random order for the documents to be added. We found that the scores change at the beginning, as expected, but again become stable around document d_9. The average values of adequacy are represented in Figure 6: again, after initial chaotic variation, the scores stabilize towards their final values. Scores reach stable values after about 10 documents, while ranks reach stable values after 4 documents.

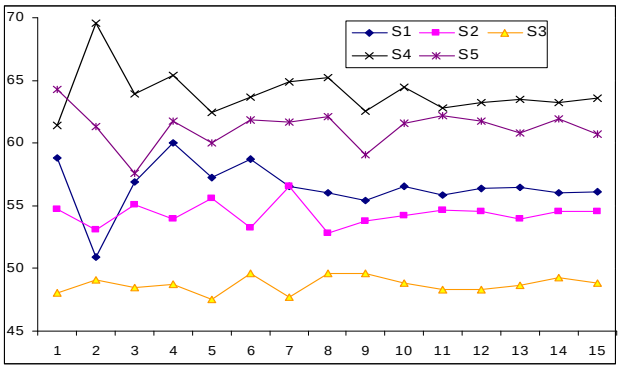

Figure 6: Adequacy results for systems S1 to S5, for 1, 2, …, 15 documents, in random order

A representation of GTM average scores with two random orderings of the documents is shown in Figure 7, in which the two curves have a different pattern, although they converge towards the same score.

The main question arising at this point is: what is the number of documents needed to reach a score which is "close" to the final one, i.e. to the objective measure of the system's output quality? Figure 8 provides an initial hint: it shows the STDEV of average GTM scores with the two random orderings of documents. The curves show quite clearly that the STDEVs evolve similarly in the two cases. The next section will exploit this fact to define a formal criterion for score reliability.

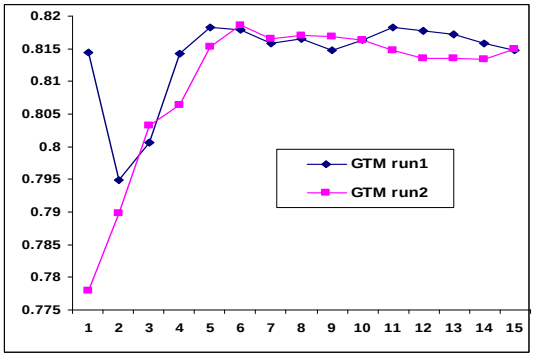

Figure 7: GTM average scores for S2 using two different random orderings for documents

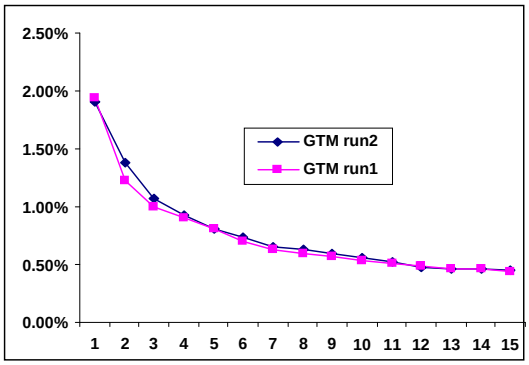

Figure 8: Relative STDEV (%) of GTM scores for S2 using two different random orders for documents

Principles and Computation

The STDEVs of the scores obtained for every metric (human and automatic) and every system exhibit similar behaviors, as shown respectively in Figure 9 for adequacy and in Figure 10 for BLEU which is typical of the other automatic metrics. STDEVs start with relatively high values and decrease considerably, ending with a relatively low value.

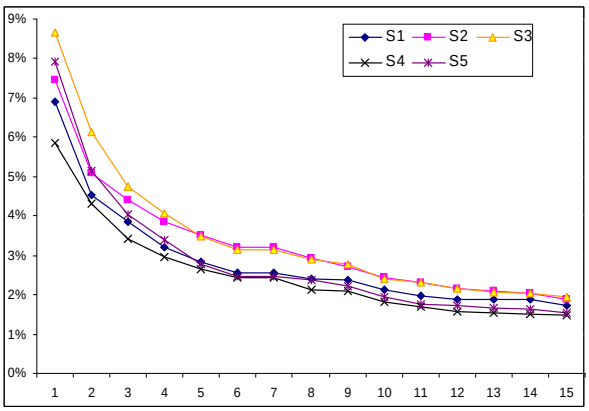

Figure 9: STDEV for adequacy scores, computed using bootstrapping, depending on the number of documents considered (1, 2, …, 15)

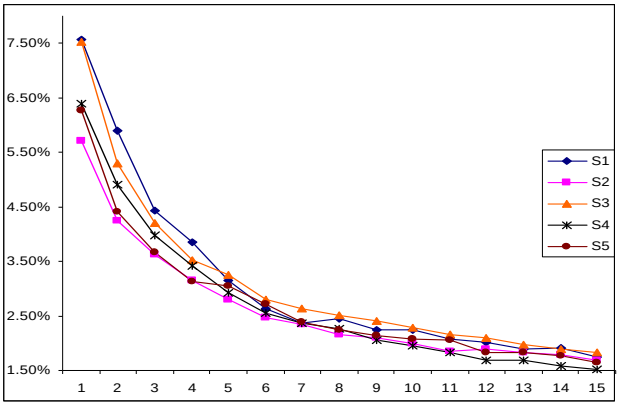

Figure 10: STDEV for BLEU scores, computed using bootstrapping, depending on the number of documents considered (1, 2, …, 15)

Using the values that we computed, it is possible to find a function *f(x)* that approximates the STDEV curves from Figures 8 to 10, performing a regression analysis. It is then possible to study the evolution of the STDEV using the first order derivative of *f(x)*, i.e. the tangent to *f(x)* in any of its points, as follows. We consider two possible approaches that allow us to define two points of interest on the *x* axis, that is two sizes of the test data set (expressed as numbers of documents) that are related to specific values of the STDEV, called here x_{min} and x_{max}.

The first characteristic value, x_{min}, is the point where the tangent line at the beginning of the STDEV curve crosses the x-axis, which suggests a first number of documents beyond which the STDEV does not decrease significantly. The second point, x_{max}, is the point where the slope of the tangent line becomes very close to zero, after which STDEV will stop decreasing drastically. Empirical tests must be carried to find out if this value corresponds to an acceptable number of documents ($x_{max} << \infty$).

The two proposed parameters can be determined using any regression function that properly fits the data. We explored two functions, namely the cubic (3rd degree polynomial) and power functions, because we found the R^2 coefficient of correlation between these curves and the empirical STDEV curves was quite high.

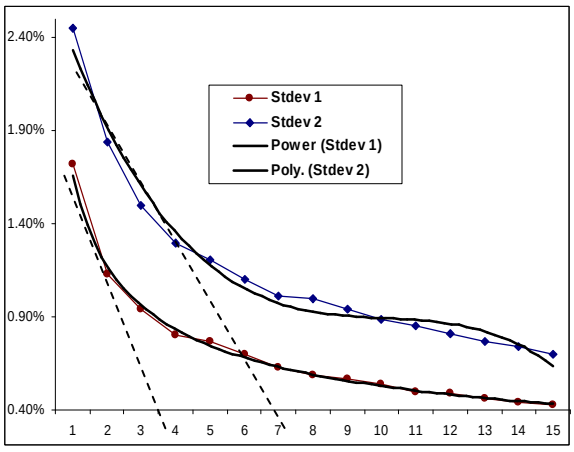

Figure 11: Graphical representation of x_{min} for two systems, S1 and S4—the values are the intersection points of the tangents at the beginning of the curves with the x axis

Depending on the metric, one function was more appropriate than the other. For instance, as shown in Figure 11, *Stdev2* (for S4) is better represented by a cubic function, while the power function is better for *Stdev1* (S1).

In the following subsections, we apply this method to the concrete cases of the power and cubic functions and in the next section we present the results obtained regarding the minimal number of documents.

Regression with power function

The power function that estimates the STDEV curves has negative exponent and is defined in $(0, \infty)$.

$$f(x) = a \cdot x^{-b}$$

This function approximates SDTEV by having high values for small number of documents (x) and low values for larger amount of documents. More specifically:

$$\lim_{x \to \infty} x^{-b} = 0 \quad \text{and} \quad \lim_{x \to 0^+} x^{-b} = +\infty$$

In practice, *f(x)* can be very close but not equal to 0 or equivalently, and the tangent line to *f(x)* will never be parallel to the x-axis. Therefore, we fix a threshold ε under which the value of the derivative is considered equal to zero, for example, of 0.01%. Formally, x_{max} is the first value such that:

$$\frac{dy}{dx} = \varepsilon \cong 0 \quad \text{for some value of } x << \infty$$

Given the derivative of *f(x)*: $f' = -b \cdot a \cdot x^{-(b+1)}$ it is then possible to calculate $x_{\max}$, the point where the function begins to stabilize, using the equation:

$$f' = -b \cdot a \cdot x_{\max}^{-(b+1)} = \varepsilon \quad \Rightarrow \quad \boxed{x_{\max\, p} = \left(-\frac{\varepsilon}{a \cdot b} \right)^{\frac{1}{-(b+1)}}}$$

Regarding now x_{min}, given the tangent line

$$y_1 = a_1 \cdot x + b_1$$

we can calculate the slope a_1 and the y-intercept b_1 with the point $(x_i, f(x_i))$ that they share. Since this point is known, we can make the following operations:

$$f(x_i) = y_1(x_i)$$

$$\Rightarrow \quad a \cdot x_i^{-b} = a_1 \cdot x_i + b_1$$

$$\Rightarrow \quad \boxed{b_1 = a \cdot x_i^{-b} - a_1 \cdot x_i}$$

Recall that the derivate of *f(x)* in x_i is the slope of the line y_1, thus

$$\boxed{f'(x_i) = -b \cdot a \cdot x_i^{-(b+1)} = a_1}$$

Now, we have all of coefficients of the tangent line and we can calculate the $x_{\min\, p}$, e.g. the point where the tangent line intercepts the x-axis (first equation below)

and an approximate minimal number of documents to use in the evaluation.

$$y_1 = a_1 \cdot x_{\min p} + b_1 = 0$$

$$\Rightarrow x_{\min p} = -\frac{b_1}{a_1} \equiv -\frac{a \cdot x_i^{-b} - a_1 \cdot x_i^{-b}}{-b \cdot a \cdot x_i^{-(b+1)}}$$

Resolving the latter equation we get:

$$\boxed{x_{\min p} = \frac{x_i + b \cdot x_i}{b}}$$

Regression with cubic function

Let *f(x)* be the 3rd order polynomial function that best approximates STDEV:

$$f(x) = d \cdot x^3 + c \cdot x^2 + b \cdot x + a$$

Its derivative is:

$$f' = 3 \cdot d \cdot x^2 + 2 \cdot c \cdot x + b = 0$$

Setting *f'=0* and calculating its roots is equivalent to looking for a local minimum of *f(x)*; for our purpose, this will serve to calculate $x_{max3,}$, the number of documents where the function stops decreasing drastically. To find these roots, we use the well-known formula for solving second order equations:

$$r_{1,2} = \frac{-b \pm \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a}$$

We only consider the cases where the roots are positive and discard all other cases (negative or imaginary roots), leading to the following potential cases:

a) Two positive roots, we choose the smallest
b) One positive and other negative root, we choose the positive one
c) If the discriminant equals 0 there are two identical roots and we only consider the case when they are > 0

Replacing *f'* in the formula above, we define x_{max3} as

$$\boxed{x_{\max 3} = -\frac{c}{2 \cdot d} \pm \frac{\sqrt{c^2 - 3 \cdot d \cdot b}}{3 \cdot d}}$$

Following the same procedure, we now calculate $x_{\min 3}$ in a similar way as in the previous case. Having the linear function representing the tangent,

$$y_1 = a_1 \cdot x + b_1$$

we calculate the slope a_1 and the y-intercept b_1 as follows:

$$\boxed{f'(x_i) = 3 \cdot d \cdot x_i^2 + 2 \cdot c \cdot x_i + b = a_1}$$

Using a shared point between *f* and y_1

$$f(x_i) = y_1(x_i) \Rightarrow$$

$$d \cdot x_i^3 + c \cdot x_i^2 + b \cdot x_i + a = a_1 \cdot x_i + b_1$$

$$\boxed{b_1 = d \cdot x_i^3 + c \cdot x_i^2 + b \cdot x_i + a - a_1 \cdot x_i}$$

replacing in $y_1 = a_1 \cdot x + b_1 = 0$ we obtain

$$\boxed{x_{\min 3} = -\frac{d \cdot x_i^3 + c \cdot x_i^2 + (b - a_1) \cdot x_i + a}{3 \cdot d \cdot x_i^2 + 2 \cdot c \cdot x_i + b}}$$

Number of Documents Needed for Reliable Scores: Observed Results

We applied the method explained above, fixing $x_i = 1$ for both functions and $\varepsilon = 0.001$ for the power function. Recall that ε is the acceptable error threshold, thus the choice of its value is arbitrary, depending on the evaluator applying the method. We observed that if we choose a larger value of ε, the resulting x_{max} decreases but if we are stricter about the error threshold and choose a small ε, the value of x_{max} increases. These results are coherent with the practice, since we need more data to obtain less deviated scores.

In general, S2's to S5's standard deviation are better represented with the power function and S1 with the cubic function. In the case of mWER and mPER for S5, neither function seems to be suitable; we can check it graphically and with the R^2 coefficients.

Table 2 shows the values obtained empirically from the CESTA data for x_{min} and x_{max}, using always the power function, which in the majority of cases correlates better (in terms of R^2) with the STDEV function.

The conclusion is that $\boldsymbol{x_{min,}}$ the characteristic number of documents, is almost uniformly equal to four (here, documents have about 65 sentences).

Metric	Var.	S1	S2	S3	S4	S5
GTM	R^2	.674	.996	.995	.987	.996
	$\boldsymbol{x_{min}}$	**5**	**3**	**3**	**4**	**4**
	x_{max}	20	20	20	20	20
NIST	R^2	.905	.995	.997	.990	.972
	$\boldsymbol{x_{min}}$	**4**	**4**	**4**	**4**	**4**
	x_{max}	27	24	26	27	26
BLEU	R^2	.984	.995	.998	.993	.995
	$\boldsymbol{x_{min}}$	**3**	**4**	**3**	**3**	**4**
	x_{max}	50	46	51	47	48
mPER	R^2	.960	.995	.975	.924	.707
	$\boldsymbol{x_{min}}$	**4**	**4**	**4**	**4**	**6**
	x_{max}	38	37	34	41	40
mWER	R^2	.991	.975	.970	.963	.752
	$\boldsymbol{x_{min}}$	**4**	**4**	**4**	**4**	**5**
	x_{max}	39	15	7	41	43
Fluency	R^2	.989	.987	.989	.985	.993
	$\boldsymbol{x_{min}}$	**4**	**3**	**3**	**3**	**3**
	x_{max}	53	64	61	50	50
Adequacy	R^2	.992	.987	.995	.990	.993
	$\boldsymbol{x_{min}}$	**9**	**4**	**3**	**3**	**3**
	x_{max}	49	54	54	45	47

Table 2: Values of here x_{min} and x_{max} using the power regression as an approximation of the STDEV curves

Conclusions and Future work

This study shows that different metrics behave coherently across systems and documents. The study also takes advantage of the particular cases that were found to gain

more insight about systems' output; e.g., we were able to detect documents that are "difficult to translate", pointed out by the disagreement between metrics for the same set of documents and system; inspecting the last document added to the set we discovered information that was useful to understand the variation of metrics. We also reinforce the hypothesis that we can obtain reliable evaluation results with fewer documents than expected, reducing evaluation cost (effort and time). Our results show that for human or automatic evaluation about five documents from the same domain—with ca. 250 segments or 6,000 words—seem sufficient to establish the ranking of the systems and about ten documents are sufficient to obtain reliable scores.

Finally, we propose a method to empirically determine the minimum number of documents needed to obtain acceptably reliable results. The results presented here are also a valuable resource, which could complement the guidelines for users of the CESTA corpus—made public by ELDA—along with reference translations and scores for automatic and human metrics.

At this moment, we use the corpus in a black box evaluation but if it is intended to be used in glass box evaluation, other methods could be used to reduce the amount of text to evaluate. For example, the method proposed in (Eck et al. 2005), which consists in extracting from the corpus only the unduplicated n-grams, i.e. it eliminates redundancy.

We plan to apply our method to other experimental setup, such as different corpora or language pairs. Special cases of study are the human based metrics, since the average of two human judgments was used for the adequacy and fluency metrics in the EN/FR first CESTA campaign. These metrics are limited by the loss of information about the difference between judgments. So far, it was not so easy to find a method of human weighted scores. Indeed, having (at least) three evaluations by segments would allow us to weight the scores according to similar judgments (i.e. if one judgment is strongly different from the two others, it would have lower weight for the scoring).

Acknowledgments

This work has made use of the results of the CESTA MT Evaluation campaign organized in France under the Technolangue program. The authors are grateful to the Swiss National Science Foundation (grants n. 200021-103318 and 200020-113604) and to Professor Margaret King for valuable discussions and advice.

References

Bisani M., and Ney H. 2004. Bootstrap Estimates for Confidence Intervals in ASR Performance Evaluation. In IEEE International Conference on Acoustics, Speech, and Signal Processing, 409-412. Montreal, Canada.

Callison-Burch, C., Osborne, M. and Koehn, P., Re-evaluating the Role of BLEU in Machine Translation Research. In EACL 2006, 249-256, Trento, Italy.

Clarke C., Cormack G., Laszlo M., Lynam T., and T. E. 2002. The impact of corpus size on question answering performance. In 25th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR '02), 369-370. New York, NY, USA: ACM Press.

Doddington G. 2002. Automatic Evaluation of Machine Translation Quality Using N-gram Co-Occurrence Statistics. In HLT 2002 (Human Language Technology Conference). San Diego, CA.

Dumais S., Banko M., Brill E., Lin J., and N. A. 2002. Web Question Answering: Is More Always Better? In 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, 291-298. Tampere, Finland: ACM Press.

Eck M., Vogel S., and Waibel A. 2005. Low Cost Portability for Statistical Machine Translation based on N-gram Frequency and TF-IDF. In International Workshop on Spoken Language Translation (IWSLT). Pittsburgh, PA.

Efron B. and Gong G. 1983. A Leisurely Look at the Bootstrap, the Jackknife, and Cross-Validation. In The American Statistician, 37 (1), 36-48.

Elliott D., Hartley A., and Atwell E. 2003. Rationale for a multilingual aligned corpus for machine translation evaluation. In International Conference on Corpus Linguistics (CL2003), 191-200. Lancaster, UK.

Estrella P., Popescu-Belis A., and U. N. 2005. Finding the System that Suits you Best: Towards the Normalization of MT Evaluation. In ASLIB (27th International Conference on Translating and the Computer). London.

Germann U. 2001. Building a statistical machine translation system from scratch: how much bang for the buck can we expect? In Proceedings of the workshop on Data-driven methods in machine translation, 1-8. Toulouse, France: ACM Press.

Hamon O., Popescu-Belis A., Choukri K., Dabbadie M., Hartley A., Mustafa El Hadi W., Rajman M., and T. I 2006. CESTA First Conclusions of the Technolangue MT Evaluation Campaign. In LREC 2006 (Fourth International Conference on Language Resources and Evaluation). Genoa, Italy.

Koehn P. 2004. Statistical Significance Tests for Machine Translation Evaluation. In EMNLP 2004, Barcelona.

Kumar S., and Byrne W. 2004. Minimum Bayes-risk decoding for statistical machine translation. In HLT-NAACL 2004, 169-176.

Niessen S., Och F., Leusch G., and Ney H. 2000. An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research. In LREC 2000 (2nd International Conference on Language Resources and Evaluation), 39-45. Athens, Greece.

Papineni K, Roukos S, Ward T, and Wei-Jing Zhu 2001. BLEU: a Method for Automatic Evaluation of Machine Translation. IBM Research Division, Thomas J. Watson Research Center.

Tillmann C., Vogel S., Ney S., Zubiaga A., and S. H. 1997. Accelerated DP Based Search for Statistical Translation. In Eurospeech 1997, 2667-2670. Rhodes, Greece.

Turian J., Shen L., and Melamed D. 2003. Evaluation of Machine Translation and its Evaluation. In Machine Translation Summit IX, 386-393. New Orleans, LA.

Zhang Y., and Vogel S. 2004. Measuring Confidence Intervals for the Machine Translation Evaluation Metrics. In International Conference on Theoretical and Methodological Issues in Machine Translation (TMI 2004). Baltimore, MD.