Automatic Extrapolation of Human Assessment of Translation Quality

Stephan Vogel, Sonja Nießen, Hermann Ney
Lehrstuhl für Informatik VI, RWTH Aachen
D-52056 Aachen, Germany
vogel@informatik.rwth-aachen.de

1 Introduction

The evaluation of machine translation systems is a difficult and time consuming task. To be meaningful and reliable, translation quality has to be evaluated manually by human experts. However, during the development of a translation system and especially in a research environment, a high number of evaluation experiments is necessary to allow for a comparison of different translation approaches, test if modifications are beneficial, monitor improvements over time, and help in fine tuning of the system. For this, an automatic evaluation method seems to be the only practical approach. Several researchers have therefore used the well-known word error rate, acknowledging that this is but a weak indicator of translation quality.

In (Nießen et al., 2000) an evaluation tool has been proposed, in which manually evaluated translations are stored in a database and used to extrapolate the quality of new translations. The score for a new translation is set equal to the score of the best matching translation from the database or to the average of the scores in case there are several equally good matching translations. The standard edit distance is calculated to find the best match. A drawback of this distance measure is that it takes all errors to affect the quality of the translation to the same amount. A missing comma, for example, is treated on a par with a missing content word.

To overcome this shortcoming a method is proposed to define a more meaningful distance measure. The idea is to use the weighted edit distance, i.e. insertion, deletion and substitution errors are associated with individual costs. The main point is, however, to use an adaptation scheme to minimize the overall extrapolation error. More generally speaking, we classify sentences into (subjective) quality classes according to some (objective) criteria which can be calculated automatically. This classifier is trained to improve performance.

In the next section we will give a short overview of quality measures for machine translations. This will be followed by the presentation of a method to extrapolate translation quality and how this extrapolation can be made more reliable through a training procedure. Finally, we will report results for a number of experiments.

2 Quality measures

The quality of a set of translations can be characterized by either objective criteria which can be calculated automatically or by subjective criteria which require inspection by human evaluators. The most common objective criterion is the edit distance $d(t, t^r)$ which gives the minimal number of insertions, deletions and substitutions which transform the given translation t into the reference translation t_r. From this the word error rate for the translation of n sentences can be calculated:

$$\mathrm{WER}[\%] = \frac{\sum_{i=1}^{n} d(t_i, t_i^r)}{\sum_{i=1}^{n} |t_i^r|} * 100 \quad . \tag{1}$$

As a refinement a number of reference translations can be used (Alshawi et al., 1998; Nießen et al., 2000) which we call multi-reference word error rate mWER. Variations along this line include position independent error rate (Ney et al., 2000) which treats two sentences as sets T and T_r and is then defined as $PER = (\max\{|T|, |T_r|\} - |T \cap T_r|)/|T_r|$, or an extension of the edit distance which treats moves as one error only (Alshawi et al., 1998).

These measures have the great advantage that they can be calculated automatically, and that they are reproducible. However, whereas authors frequently contest that word error rate is only a poor substitute for a more thorough manual evaluation, the actual correlation between these objective and a subjective quality assessment has not been investigated.

Subjective evaluation can range from simply classifying translated sentences into one of a small number of quality classes up to assigning figures of merit along different dimensions (syntactic, semantic, stylistic, etc.). This may become very time consuming but gives more reliable information about the quality of the translations. Of course, their is still the problem of inter-evaluator agreement which can be rather serious and the problem that even the judgments of one evaluator can be changing over time.

In the end, we would like to have one or at most a small number of figures as indicators of the translation quality to be able to make a ranking between different translations.

In our research we use as subjective quality criterion the following: Each translation t is assigned a quality index $v(s,t) \epsilon \{0,\ldots,K\}$ indicating translation quality ranging from perfect translation to garbage. The subjective sentence error rate SSER (in percent) of a set of translations $t_1^n = t_1 \ldots t_n$ for a test corpus $s_1^n = s_1 \ldots s_n$ is then defined as

$$\mathrm{SSER}(s_1^n, t_1^n)[\%] = \frac{100}{K \cdot n} \sum_{i=1}^{n} v(s_i, t_i) \quad . \qquad (2)$$

3 Extrapolation of translation quality

In (Nießen et al., 2000) an evaluation tool is proposed, which allows to store evaluated translations in a database and to use them to extrapolate the quality of new translations. This is done in the following way: Let $\mathcal{T}(s)$ denote the set of all stored translations of source sentence s. Find in $\mathcal{T}(s)$ $\{t\}$ the nearest neighbors to t with respect to the edit distance. There may be several t' having the same distance $d(t,t')$ but be assigned to different quality classes $v(s,t')$. Let us denote this set as $\mathcal{T}_{min}(t)$. The extrapolated index $\hat{v}(s,t)$ is then defined as the average of the indices of the nearest neighbors.

$$\hat{v}(s,t) = \frac{1}{|\mathcal{T}_{min}(t)|} \sum_{t' \in \mathcal{T}_{min}(t)} v(s,t') \ . \qquad (3)$$

We define the extrapolated score as follows:

$$\tilde{v}(s,t) = \begin{cases} v(s,t) & \text{if } (s,t) \in \mathcal{DB} \ , \\ \hat{v}(s,t) & \text{otherwise} \ . \end{cases} \qquad (4)$$

h and define the extrapolated subjective sentence error rate eSSER by replacing $v(s,t)$ by $\tilde{v}(s,t)$ in definition (2).

If manual evaluation for this set of translations is actually performed we can calculate the extrapolation error:

$$\mathrm{EE}(s_1^n, t_1^n) = \mathrm{eSSER}(s_1^n, t_1^n) - \mathrm{SSER}(s_1^n, t_1^n) \quad . \ (5)$$

If this extrapolation error is small for a number of translation hypothesis files, then we will feel confident in using the estimate as a reliable indicator. Notice that $\mathrm{EE}(s_1^n, t_1^n)$ can be near zero even if $|\hat{v}(s,t) - v(s,t)|$ is large for a number of translations as these individual extrapolation errors can balance out.

4 Improving the extrapolation

The extrapolation of translation quality of a new translation for a given source sentence can be seen as a classification problem. We have a number of quality classes $V_k, k = 1...K$, a number of translations $t_i, i = 1...I$ for which we know to which quality class they belong, and a distance measure $d(t,t')$ inducing a partial ordering on the set of all translations. Classification of the new translation can then be performed using nearest neighbor search. To minimize the extrapolation error we have to modify the distance measure as the quality classes of the translations are fixed.

Not all translation errors are equally serious. A missing comma will do no harm in most cases whereas a missing 'not' will alter the meaning of the sentence completely. Edit distance does not take into account these differences. A better solution would be to use weighted edit distance, where each insertion, deletion or substitution can be associated with an individual score. Actually, several levels of refinement are possible:

1. There is one insertion score I, one deletion score D, and one substitution score S.
2. There are individual scores $I(w)$, $D(w)$, and $S(w_1, w_2)$.
3. There are for each source sentence s and thereby for each set $\mathcal{T}(s)$ of translations individual scores $I_s(w)$, $D_s(w)$, and $S_s(w_1, w_2)$.

We implemented the second and third of these alternatives.

To minimize the extrapolation error in equation 5 we would need a large number of hypothesis files, their eSSER according to the database before manual evaluation and the correct SSER obtained by manual evaluation. As the database – although under revision control – does not store the information necessary to recall already evaluated translation files, we take the translations stored in the database as representative for new translations to be extrapolated. Thus, if we are able to improve the estimates for the translations in the database, this will improve the estimates for new translation files. That is to say, we want to minimize the sum of all extrapolation errors over the complete database:

$$\mathrm{EE}(\mathcal{DB}) = \frac{100}{K \cdot T} \sum_{s \epsilon \mathcal{DB}} \sum_{t \epsilon \mathcal{T}(s)} v(s,t) - \hat{v}(s,t) \quad , \ (6)$$

where T is the number of target sentences in the database. Actually, we use an even stronger criterion. We want to minimize the absolute extrapolation errors aEE($\mathcal{DB}$), by summing of the absolute difference $|v(s,t) - \hat{v}(s,t)|$ in equation 6.

For each translation t_i in the database, extrapolate the translation quality $\hat{v}(s,t)$ using the remaining translations t' for the same source sentence s. If this extrapolated translation class score is correct, i.e. it is equal to the manually assigned quality class, we are done. However, if there is a mismatch, we have to modify the error scores. Obviously, the distance between t and the best match t_b from all t_j is too small. It should be increased in such a way, that it is no longer the best match. On the other side, if there are translations t_c with $v(s,t_c) = v(s,t)$, their distance should be lowered.

Modification of edit distance means modification of the scores of those insertions, deletions, and substitutions which where necessary to match t and t_b or t and t_c. Notice that there may be conflicting requirements how these scores should be modified. For example, for some word w to increase $d(t,t_b)$ may require to increase deletion score $D(w)$, whereas to decrease $d(t,t_c)$ may require to decrease this score. Therefore we adopt the following heuristic: for each edit distance score count how often a larger value was required and how often a smaller value was required. We experimented with updating the edit distance scores proportionally to the difference of these two counts and with updating them simply by a small constant which decreases from iteration to iteration. The second alternative gave better convergence and was therefore used in the experiments report in the next section.

5 Experiments and Results

5.1 Databases

In our experiments we used two databases. In both cases translation is from German to English. The two translation tasks differ in the size of the underlying vocabularies. For task 1 the vocabulary size is about 7 000 words, whereas for task 2 the vocabulary size is nearly 60 000 words. The Table 1 shows the number of source sentences N, the total number of translations T and the number of reference translations R, i.e. those which have been scored as perfect translation. In addition, the average number of translations and reference translations per source sentence are given. For these databases the quality indices run from 0 (= bad) to 10 (= perfect).

Table 1: Database characteristics.

	N	T	R	T/N	R/N
$\mathcal{DB}$-1	144	6458	922	44.6	6.4
$\mathcal{DB}$-2	120	4530	178	27.7	1.5

5.2 Adaptation of error scores

The complete database at a given time-point was used to train the error scores. Adaptation was run for 500 iterations in the case of database $\mathcal{DB}$-1 whereas for database $\mathcal{DB}$-2, which seems more homogeneous, 200 iterations were sufficient. The extrapolation with unweighted edit distance was used as a baseline. In those cases, where several translations with the same distance exist, the average of the subjective quality score is used as the extrapolated quality score.

In a first experiment only one set of insertion, deletion, and substitution scores for the complete database was used. This gave only a small improvement over the baseline approach. So all experiments reported here used one table of insertion, deletion, and substitution scores for each set of translations $\mathcal{T}(s)$. The results are given in Table 2.

Table 2: Leaving one out extrapolation. B = baseline, W = weighted edit distance.

	$\mathcal{DB}$-1		$\mathcal{DB}$-2	
	B	W	B	W
Correct [%]	45.9	68.8	51.4	80.6
aEE [%]	11.4	7.1	10.3	5.0
EE [%]	3.5	0.1	-1.2	0.3

As can be seen, a clear improvement was achieved. The number of correctly estimated translations goes up by 23% resp. 29% absolute. The absolute extrapolation error aEE shows a clear sharpening of the estimation. That is to say, the estimate of the quality index for each translation in the database is nearer to the correct quality class. For $\mathcal{DB}$-2 this error is cut down to half its former size. It should also be noted, that the estimation became nearly symmetric, whereas the baseline approach shows a systematic overestimation for $\mathcal{DB}$-1 and a systematic underestimation for $\mathcal{DB}$-2.

Figure 1 shows how the extrapolation error decreased with the number of adaptation iterations for $\mathcal{DB}$-1. The extrapolation error of the baseline approach is given as reference line. As is to be expected, the curve is not strictly monotonously decreasing. The largest improvement is in the first two iterations. This comes essentially from the modification of the insertion, deletion and substitution scores for sentence marks. A similar improvement could be achieved by setting these scores manually. After about 250 iterations the curves flatten out. In Figure 2 the number of correct extrapolations as a function of the training iterations is given.

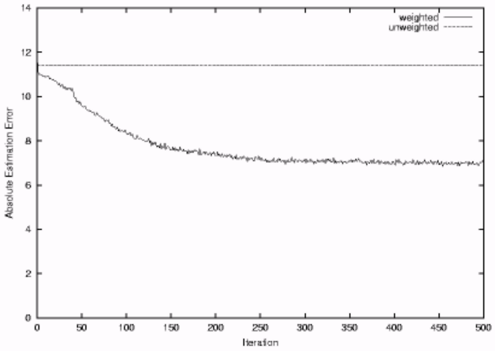

Figure 1: Absolute extrapolation error during training for $\mathcal{DB}$-1.

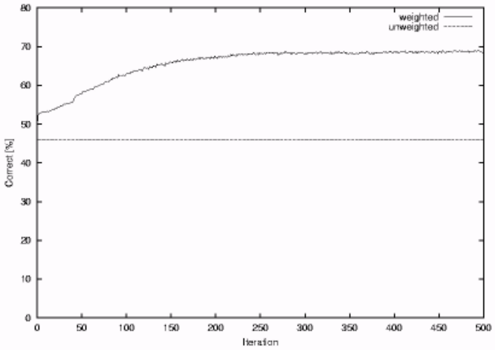

Figure 2: Correct extrapolation during training for $\mathcal{DB}$-1.

5.3 Correlation WER - SSER

In our opinion word error rate is but a poor indicator of translation quality, perhaps with the exception of very simple translation tasks, where word error rate approaches zero. In Figure 3 a scatter plot for unweighted edit distance (mWER) versus translation quality (10 - v) for $\mathcal{DB}$-1 is given. The reference translations ($v = 0$) have been left out. We see the tendency that smaller edit distance gives better quality judgements. However, for a good correlation we would expect a sharp ridge running from the rear edge of the plot to the front.

For database $\mathcal{DB}$-1 we calculated the correlation coefficient. The correlation is defined in terms of the covariance $Cov(d, v)$ between d and v and the standard deviation of d and v:

$$\rho = \frac{Cov(d,v)}{\sigma(d)\sigma(v)} = \frac{\sum(d_i - \bar{d})(v_i - \bar{v})}{\sqrt{\sum(d_i - \bar{d})^2}\sqrt{\sum(v_i - \bar{v})^2}} \quad (7)$$

d_i stands here as shorthand for the (average) distance between t_i and the best matching reference translation(s), v_i is its translation quality index. $\bar{d}$ and $\bar{v}$ are the mean values.

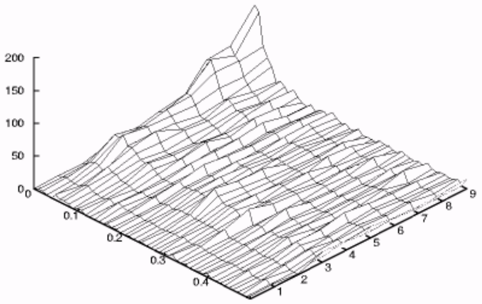

Figure 3: Scatter plot: mWER (0 – 0.5) vs. 10 - v.

The correlation coefficient has been calculated for unweighted and weighted edit distance. Multireference word error rate is used in the case of unweighted edit distance. With weighted edit distance it hardly ever happens that $d(t, t_1) = d(t, t_2)$ for two reference translations. So, only first best match is used.

Table 3: Correlation between word error rate and subjective quality score

	unweighted	weighted
$\mathcal{DB}$-1	-0.44	-0.47
$\mathcal{DB}$-2	-0.48	-0.37

What we see is that the correlation is actually rather week. There is no improvement when using weighted edit distance. The reason for this is the following: The weights were adjusted to improve a local ordering of the stored translations. Translations with the same quality class are brought into closer neighborhood. To make word error rate a stronger indicator for translation quality would call for a global ordering of the translations. A similar adaptation procedure to the one described in this paper could be set up to this end.

5.4 Database size and extrapolation quality

With every test file evaluated the database grows. Therefore, it is interesting to see how extrapolation quality changes with the size of the database. As the database is under revision control it was possible to retrieve older versions and perform the leaving-one-out extrapolation for those versions. This was done for $\mathcal{DB}$-1 using unweighted edit distance as baseline and weighted edit distance with error scores trained on those version. In Figures 4 and 5 the resulting curves are plotted. What is most remarkable is the fast improvement gained for small database sizes.

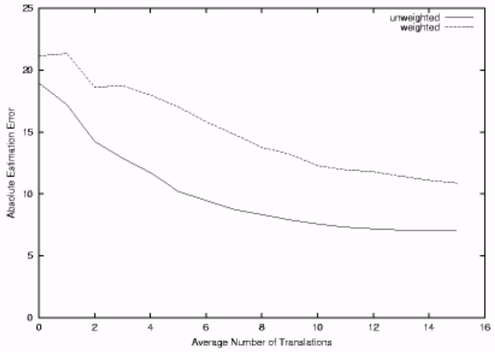

Figure 4: Database size versus extrapolation error.

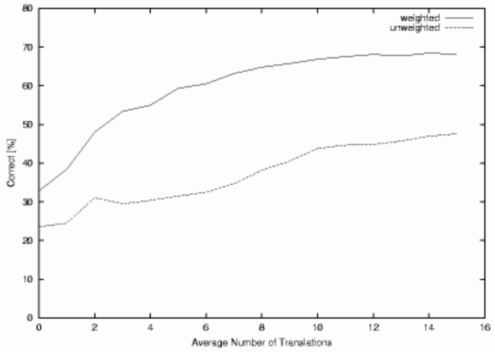

Figure 5: Database size versus number of correct extrapolation.

5.5 Extrapolation of new translations

A number of translation files by different translation methods was used to test the performance of the modified extrapolation method. First, the hypothesis file was extrapolated using both the unweighted and the weighted edit distance. Then, a manual evaluation was performed and the extrapolation errors for the two methods calculated.

In Table 4 the results are given. For each set the number of extrapolated sentences is given. Normally, a number of sentences is given, for which the quality had to be extrapolated, because the sentences are not already in the database. The second column gives the correct SSER after manual evaluation. The last two columns show the extrapolation error when using the unweighted and the weighted edit distance. The last two lines give the sums and averages.

Again, using weighted edit distance gives – on average – superior estimates for the translation quality. This does not mean that manual evaluation is no longer needed. But it helps to sort out significant tests for which manual evaluation should be done.

Table 4: Estimation of Translation Results

Extrapolated	SSER	|EE|	
Sentences		baseline	weighted
47	17.8	0.20	0.54
51	16.7	0.68	0.07
46	19.9	1.36	0.54
54	21.2	2.59	0.27
43	17.1	0.68	1.43
3	16.9	0.14	0.27
54	27.6	1.43	0.88
72	28.0	3.81	2.99
80	27.4	4.22	1.97
45	35.8	1.97	1.70
50	43.0	0.48	0.68
63	39.3	0.75	0.61
63	39.1	0.82	0.34
Average		1.46	0.95

6 Summary

Meaningful assessment of machine translation systems requires manual evaluation. In this paper we addressed the question if objective criteria which can be calculated automatically can be used to improve this subjective evaluation. A method has been proposed to store evaluation results and to use them for extrapolating the quality of new translation results. This extrapolation relies on finding nearest matches between known and new translations. A standard distance measure is edit distance. We showed that extrapolation of translation quality can be improved significantly if weighted edit distance is used where the individual error scores for insertions, deletions and substitutions are adapted to the database using a leaving one out trainings scheme.

References

Hiyan Alshawi, Srinivas Bangalore, and Shona Douglas. 1998. Automatic Acquisition of Hierarchical Transduction Models for Machine Translation. In *Proc. 36th Annual Conference of the Association for Computational Linguistics and the 17th International Conference on Computational Linguistics*, pages 41–47, Montréal, P.Q., Canada, August.

Hermann Ney, Sonja Nießen, Franz Josef Och, Hassan Sawaf, Christoph Tillmann, and Stephan Vogel. 2000. Algorithms for Statistical Translation of Spoken Language. In *IEEE Transactions on Speech and Adio Processing*, pages 24–36.

Sonja Nießen, Franz Josef Och, Gregor Leusch, Hermann Ney. 2000. An Evaluation Tool for Machine Translation: Fast Evaluation for MT Research. In *Proceedings of LREC*, Athens, Greece, May .