Estimating the Sentence-Level Quality of Machine Translation Systems

Lucia Specia*, Nicola Cancedda and Marc Dymetman
Xerox Research Centre Europe
Meylan, 38240, France
lucia.specia@xrce.xerox.com
nicola.cancedda@xerox.com
marc.dymetman@xerox.com

Marco Turchi* and Nello Cristianini
Department of Engineering Mathematics
University of Bristol
Bristol, BS8 1TR, UK
Marco.Turchi@bristol.ac.uk
nello@support-vector.net

Abstract

We investigate the problem of predicting the quality of sentences produced by machine translation systems when reference translations are not available. The problem is addressed as a regression task and a method that takes into account the contribution of different features is proposed. We experiment with this method for translations produced by various MT systems and different language pairs, annotated with quality scores both automatically and manually. Results show that our method allows obtaining good estimates and that identifying a reduced set of relevant features plays an important role. The experiments also highlight a number of outstanding features that were consistently selected as the most relevant and could be used in different ways to improve MT performance or to enhance MT evaluation.

1 Introduction

The notion of "quality" in Machine Translation (MT) can have different interpretations depending on the intended use of the translations (e.g., fluency and adequacy, post-editing time, etc.). Nonetheless, the assessment of the quality of a translation is in general done by the user, who needs to read the translation and the source text to be able to judge whether it is a good translation or not. This is a very time consuming task and may not even be possible, if the user does not have knowledge about the source language. Therefore, automatically assessing the quality of translations produced by MT systems is a crucial problem, either to filter out the low quality ones, e.g. to avoid professional translators spending time reading / post-editing bad translations, or to present them in such a way as to make end-users aware of the quality. This task, referred to as Confidence Estimation (CE), is concerned about predicting the quality of a system's output for a given input, without any information about the expected output.

CE for MT has been viewed as a binary classification problem (Blatz et al., 2003) to distinguish between "good" and "bad" translations. However, it may be difficult to find a clear boundary between "good" and "bad" translations and this information may not be useful in certain applications (e.g, the time necessary to post-edit translations).

We distinguish the task of CE from that of MT evaluation by the need, in the latter, of reference translations. The general goal of MT evaluation is to compare a machine translation to reference translation(s) and provide a quality score which reflects how close the two translations are. In CE, the task consists in estimating the quality of the translation given only information about the input and output texts and the translation process.

In this paper we consider CE for MT as a wider problem, in which a continuous quality score is estimated for each sentence. This could be seen as a proxy for MT evaluation, but without any form or reference information. This problem is addressed as a regression task, where we train algorithms to predict different types of sentence-level scores. The contribution of a large number of features is epxloited by using a feature selection strategy. We also distinguish between features that depend on the translation process of a given MT system and those that can be extracted given only the input sentences and corresponding output translations,

*L. Specia and M. Turchi contributed equally to this work.

Proceedings of the 13th Annual Conference of the EAMT, pages 28–35,
Barcelona, May 2009

and are therefore independent on MT systems.

In the remaining of this paper we first discuss the previous work on CE for MT (Section 2), to then describe our experimental setting (Section 3) and method (Section 4) and present and discuss the results obtained (Sections 5 and 6).

2 Related work

Early work on CE for MT aimed at estimating the quality at the word level (Gandrabur and Foster, 2003; Ueffing and Ney, 2005; Kadri and Nie, 2006). Sentence-level CE appears to be a more natural set-up for practical applications of MT. One should consider as real-world scenario for CE an MT system in use, which would provide to the user, together with each sentence translation, an estimate of its quality. If this estimate is in the form a numerical score, it sould also be viewed as a proxy to some automatic or manual metric, like NIST (Doddington, 2002) or 1-5 adequacy. Other estimates include the time that would be necessary to post-edit such translation, or simply a "good" / "bad" indicator.

Differently from MT evaluation, in CE reference translations are not available to compute the quality estimates. Therefore, CE approaches cannot be directly compared to the several recently proposed metrics for sentence-level MT evaluation that also use machine learning algorithms and sometimes similar features to those used in CE. For example, (Kulesza and Shieber, 2004) use Support Vector Machines (SVM) with n-gram precision and other reference-based features to predict if a sentence is produced by a human translator (presumably good) or by a MT system (presumably bad) (*human-likeness classification*). (Albrecht and Hwa, 2007a) rely on regression-based algorithms and features, like string and syntax matching of the translation over the corresponding references, to measure the quality of sentences as a continuous score. In (Albrecht and Hwa, 2007b), *pseudo-references* (produced by other MT systems) are used instead of human references, but this scenario with multiple MT systems is different from that of CE.

The most comprehensive study on CE at the sentence level to date is that of (Blatz et al., 2004). Multi-layer perceptrons and Naive Bayes are trained on 91 features extracted for translations tagged according to NIST and word error rate. Scores are thresholded to label the 5th or 30th percentile of the examples as "correct" and the remainder as "incorrect". Regression is also performed, but the estimated scores are mapped into the same classes to make results binary. The contribution of features is investigated by producing classifiers for each feature individually and for combinations of all features except one at a time. In both cases, none of the features is found to be significantly more relevant than the others. This seems to point out that many of the features are redundant, but this aspect is not investigated.

(Quirk, 2004) uses linear regression with features similar to those used in (Blatz et al., 2004) to estimate sentence translation quality considering also a small set of translations manually labeled as correct / incorrect. Models trained on this small dataset (350 sentences) outperform those trained on a larger set of automatically labeled data. Given the small amount of manually annotated data and the fact that translations come from a single MT system and language-pair, it is not clear how results can be generalized. The contribution of different features is not investigated.

(Gamon et al., 2005) train an SVM classifier using a number of linguistic features (grammar productions, semantic relationships, etc.) extracted from machine and human translations to distinguish between human and machine translations (*human-likeness classification*). The predictions of SVM, when combined to a 4-gram language model score, only slightly increase the correlation with human judgements and such correlation is still lower than that achieved by BLEU (Papineni et al., 2002). Moreover, as shown in (Albrecht and Hwa, 2007a), high human-likeness does not necessarily imply good MT quality. Besides estimating the quality of *machine* translations directly, we use a larger set of features, which are meant to cover many more aspects of the translations. These features are all resource-independent, allowing to generalize this method across translations produced by several MT systems and for different language-pairs.

Although our goal is very similar to that of (Blatz et al., 2004; Quirk, 2004), it is not possible to compare our results to these previous works, since we estimate continuous scores, instead of binary ones. We consider the following aspects as main improvements wrt such previous works: (a) evidence that is is possible to accurately estimate continuous scores, besides binary indicators,

which can be more appropriate for certain applications (e.g. post-edition time); (b) the use of learning techniques that are appropriate for the type of features used in CE (Partial Least Squares, which can deal efficiently with multicollinearity of input features); (c) the addition of new features that were found to be very relevant; (d) the proposal of an explicit feature selection method to identify relevant features in a systematic way; and (e) the exploitation of multiple datasets of translations from different MT systems and language pairs, with different types of human and automatic quality annotations, through the use of resource-independent features and the definition of system-independent features.

3 Experimental setting

3.1 Features

We extract all the features identified in previous work for sentence-level CE (see (Blatz et al., 2003) for a list), except those depending on linguistic resources like parsers or WordNet. We also add new features to cover aspects that have not been directly addressed in previous work, including the mismatch of many superficial constructions between the input and output sentences (percentages of punctuation symbols, numbers, etc.), similarity between the source sentence and sentences in a monolingual corpus, word alignment between input and output sentences, length of phrases, etc. This results in a total of 84 features.

Many of these features depend on some aspect of the translation process, and therefore are MT system-dependent and could not be extracted from all translation data used in this paper. We thus divide the features in two subsets: (a) *black-box features*, which can be extracted given only the input sentence and the translation produced by the MT system, i.e., the source and target sentences, and possibly monolingual or parallel corpora, and (b) *glass-box features*, which may also depend on some aspect of the translation process.

The black-box group includes simple features like source and target sentence lengths and their ratios, source and target sentence n-gram frequency statistics in the corpus, etc. This constitutes an interesting scenario and can be particularly useful when it is not possible to have access to internal features of the MT systems (in commercial systems, e.g.). It also provides a way to perform the task of CE across different MT systems, which may use different frameworks. An interesting research question is whether it is possible to produce accurate CE models taking into account only these very basic features. To our knowledge, this issue has not been investigated before.

The glass-box group includes internal features of the MT system, like the SMT model score, phrase and word probabilities, and alternative translations per source word. They also include features based on the n-best list of translation candidates, some of which apply globally to the set of all candidates for a given source sentence (e.g. degree to which phrases are translated in the same way throughout the n-best list), and some to specific candidates (e.g. ratio between scores of the candidate and top candidate). We extract a total of 54 glass-box features.

3.2 Data

We use two types of translation data: (a) translations automatically annotated with NIST scores, and (b) translations produced by different MT systems and for multiple language-pairs, manually annotated with different types of scores.

The automatically annotated dataset, henceforth *NIST dataset*, is produced from the French-English Europarl parallel corpus, as provided by the WMT-2008 shared translation task (Callison-Burch et al., 2008). We translate the three development-test sets available ($\sim$6k sentences) using a phrase-based MT system [omitted for blind review]. These translations and their 1,000 n-best lists are scored according to sentence-level NIST and the 84 features are extracted from them.

The dataset is first sampled into 1,000 subsamples, where each subsample contains all feature vectors for a certain position in all the n-best lists and is randomly split in training (50%), validation (30%) and test (20%) using a uniform distribution.

The first type of manually annotated datasets (*WMT datasets*) is derived from several corpora of the WMT-2006 translation shared task (Koehn and Monz, 2006). These are subsets of sentences from the test data used in the shared task, annotated by humans according to adequacy, with scores from 1 (worst) to 5 (best). Each corpus contains $\sim$100-400 sentences and refers to a given language pair and MT system. Since this number is very small, we put together all sets of translations from a given MT system. We select four among the resulting datasets: the three phrase-based SMT systems (*S1, S2, S3*) with the high-

est numbers of examples and the only rule-based system (*RB*). Each new dataset contains ∼1,300-2,000 sentences, and 4-6 language-pairs. The feature vectors of these datasets contain only black-box features. To account for mixing language-pairs, we add the source and target language indicators as features. The task becomes predicting the quality of a given MT system which translates between different language pairs.

The manually annotated datasets of the second type (*1-4 datasets*) contain 4K sentences of the Europarl domain (English-Spanish), translated by four SMT systems developed by different partners in the project *P* [omitted for blind review]: *P-ES-1, P-ES-2, P-ES-3 and P-ES-4*. The sentences are annotated by professional translators according to 1-4 quality scores, which are commonly used by them to indicate the quality of translations with respect to the need of post-edition: 1 = requires complete retranslation, ..., 4 = fit for purpose.

Datasets of the final type (*post-edition datasets*) contain 3K sentences of the automotive industry domain (English-Russian), translated by three MT systems from the same project *P*: *P-ER-1, P-ER-2 and P-ER-3*. The sentences are annotated according to post-edition time, that is, given a source sentence in English and its translation into Russian, a professional translator post-edited such translation to make it into a good quality sentence, while the time was recorded.

Black-box features are extracted from all datasets in the last two groups (*1-4* and *post-edition*). Additionally, glass-box features are extracted from one of the datasets (*P-ES-1*), since we had access to the SMT system in this case. We call this *P-ES-1gb*. In the *post-edition* datasets, the post-edition time is first normalized by the source sentence length, so that the score refers to the time necessary per source word.

For each manually annotated dataset, the feature vectors are randomly subsampled 100 times in training (50%), validation (30%) and test (20% using a uniform distribution.

In both automatically and manually annotated datasets, we represent each subsample as a matrix of variable predictors (X) times variable response (Y) and normalize feature values using the *z* score.

Datasets covering different language pairs and MT systems and particularly data annotated according to post-edition time for CE have not been investigated before.

3.3 Learning algorithm

We estimate the quality of the translations by predicting the sentence-level NIST, 1-5 / 1-4 scores or post-edition time using Partial Least Squares (PLS) (Wold et al., 1984). Given a matrix X (input variables) and a vector Y (response variable), the goal of PLS regression is to predict Y from X and to describe their common structure. In order to do that, PLS projects the original data onto a different space of latent variables (or "components") and is also able to provide information on the importance of individual features in X. PLS is particularly indicated when the features in X are strongly correlated (multicollinearity). This is the case in our datasets. For example, we consider each of the SMT system features individually, as well as the sum of the all these features (the actual SMT model score). With such datasets, standard regression techniques usually fail (Rosipal and Trejo, 2001). PLS has been widely used to extract qualitative information from different types of data (Frenich et al., 1995), but to our knowledge, it has not been used in NLP applications. More formally, PLS can be defined as an ordinary multiple regression problem, i.e.,

$$Y = XB_w + F$$

where B_w is the regression matrix, F is the residual matrix, but B_w is computed directly using an optimal number of components. For more details see (Jong, 1993). When X is standardized, an element of B_w with large absolute value indicates an important X-variable.

It is well known that feature selection can be helpful to many tasks in NLP, and that even learning methods that implicitly perform some form of feature selection, such as SVMs, can benefit from the use of explicit feature selection techniques. We take advantage of a property of PLS, which is the ordering of the features of X in B_w according to their relevance, to define a method to select subsets of discriminative features (Section 4).

To evaluate the performance of the approach, we compute the average error in the estimation of NIST or manual scores by means of the Root Mean Squared Prediction Error (RMSPE) metric: $\sqrt{\frac{1}{N}\sum_{j=1}^{N}(y_j - \widehat{y}_j)^2}$, where N is the number of points, $\widehat{y}$ is the prediction obtained by the regressor and y is the actual value of the test case. RMSPE quantifies the amount by which the estimator differs from the expected score: the lower the value,

the better the performance.

4 Method

Our method to perform regression supported by an embedded feature selection procedure consists of the following steps: (1) sort all features according to their relevance in the training data; (2) select only the top features according to their relevance in the validation set; (3) apply the selected features to the test data and evaluate the performance. In more details:

1. Given each pre-defined number of components, for each *i-th* subsample of the training data, we run PLS to compute the $B_w(i)$ matrix, generating a list $L_b(i)$ of feature ranked in decreasing order of importance. After generating L_b for all subsamples, we obtain a matrix where each row i contains an $L_b(i)$, e.g.:

66	7	56	...	10
44	56	3	...	10
...	...	...	...	...
66	56	3	...	10

   A list L containing the global feature ordering for all subsamples is obtained by selecting the feature appearing most frequently in each column (i.e., taking the *mode*, without repeating features). In the case shown, $L = \{66, 56, 3, \ldots, 10\}$.
2. Given the list L produced for a certain number of components, for each *i-th* subsample of the validation data, we train the regression algorithms on 80% of the data, adding features from L one by one. We test the models on the remaining validation data and plot the learning curves with the mean error scores over all the subsamples. By analyzing the learning curves, we select the first n features that maximize the performance of the models.
3. Given the selected n features and the number of components that optmized the performance in the validation data, for each *i-th* subsample of the test data, we train (80%) and test (20%) the performance of the regressor using these features, and compute their corresponding metrics over all subsamples.

5 Results

5.1 NIST dataset

Figure 1 illustrates the performance for different numbers of PLS components used to generate ordered lists of features. The maximum performance

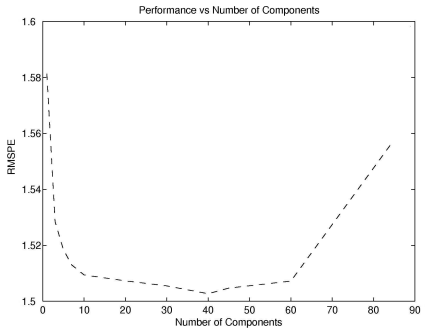

Figure 1: Performance for lists generated with different numbers of components - *NIST dataset*

is obtained from the ordered list generated with 40 components. This resulted in 32 features being selected, and an RMSPE in the test set of **1.503** $\pm$ **0.045**. The RMSPE for all features, without applying the feature selection method, is **1.670** $\pm$ **0.669**. Therefore, the models produced for the selected subset of features perform better than using all features. Moreover, results for the subsets of features are more stable, given the large variance observed in the RMSPE score with all features. To provide a more intuitive measure, we can say that the system deviates on average ~ 1.5 points when predicting the sentence-level NIST score. We believe this is an acceptable deviation, given that the scores vary from 0 to 18.44.

Although the subsets of features selected vary for different numbers of components, some appear in all the top lists:

- average number of alternative translations for words in the source sentence;
- ratio of source and target lengths;
- proportion of aborted nodes in the decoder's search graph.

The first feature reflects the ambiguity and therefore the difficulty of translating the source sentence. The second favors source and target sentences which are similar in size, which is expected for close language-pairs like English-French. The last gives an idea about the uncertainty in the search: nodes are aborted if the decoder is certain that they will not yield good translations.

Other features appear as relevant for most choices in the number of components:

- source sentence length;
- number of different words in the n-best list divided by average sentence length;

MT	RMSPE	RMSPE all features
RB	1.058 ± 0.087	1.171 ± 0.098
S1	1.159 ± 0.064	1.197 ± 0.059
S2	1.116 ± 0.073	1.190 ± 0.073
S3	1.160 ± 0.059	1.201 ± 0.062

Table 1: RMSPE - *WMT datasets*

- 1/3-gram source frequency statistics in the whole corpus or its most frequent quartile;
- 3-gram source language model probability;
- 3-gram target language model probability considering n-best list as corpus;
- phrase probabilities;
- average size of phrases in the target sentence;
- proportion of pruned and remaining nodes in decoder's final search graph.

These features in general point out the difficulty of translating the source sentence, the uniformity of the candidates in the n-best list, how well the source sentence is covered in the training corpus, and how commonplace the target sentence is. They include some SMT model features, but notably not the actual SMT score. Surprisingly, half of these very discriminative features are black-box.

5.2 Manually annotated datasets

Results for the *WMT datasets* are less straightforward to interpret, since the problem has more variables, particularly multiple language pairs, in- / out-of-domain sentences in a single dataset, and reduced dataset sizes. The best numbers of components vary from 1 to 25 and feature selection results in different subsets of features (from 2 to 10 features) for different MT systems. Nevertheless, in all the datasets, feature selection yields better results, as shown in Table 1.

The models deviate on average ~ 1.1 points when predicting 1-5 scores. This means, e.g., that some sentences atually scoring 4 would be given to the user as scoring 5.

Table 2 shows the performance obtained for the *1-4* and *post-edition datasets*. The figures for the subsets of features consistently outperform those for using all features and are also more stable.

The models produced for different MT systems (P-ES-1 to P-ES-4) deviate ~ 0.6-0.7 points when predicting the sentence-level 1-4 scores, which we believe is a satisfactory deviation. For example, one sentence that should be considered as "fit for purpose" (score 4) would never be predicted as "requires complete retranslation" (score 1) and discarded as a consequence.

MT	RMSPE	RMSPE all features
P-ES-1gb	0.690 ± 0.052	0.780 ± 0.385
P-ES-1	0.706 ± 0.059	0.793 ± 0.643
P-ES-2	0.653 ± 0.114	0.750 ± 0.541
P-ES-3	0.718 ± 0.144	0.745 ± 0.287
P-ES-4	0.603 ± 0.262	1.550 ± 3.551
P-ER-1	1.951 ± 0.174	2.083 ± 0.561
P-ER-2	2.883 ± 0.301	3.483 ± 1.489
P-ER-3	3.879 ± 0.339	4.893 ± 2.342

Table 2: RMSPE - *1-4* and *post-edition datasets*

An interesting result is the comparison between the scores for the two variations of the first dataset, i.e., *P-ES-1gb* (glass-box features) and *P-ES-1* (black-box features). The gain in using glass-box features is very little in this case. This shows that although glass-box features may be very informative, it is possible to represent the same information using simpler features. From a practical point of view, this is very important, since black-box features are usually faster to extract and can be used with any MT system.

In order to investigate whether any single feature would be able to predict the quality scores as well as the combination of selected good features, we compare the Pearson's correlation coefficient of each feature and the predicted CE score with the expected human score. The correlation of the best features with the human score is ~ 0.5 (glass-box features) or up to ~ 0.4 (black-box features) across the different *1-4 datasets*. The CE score correlates ~ 0.6 with the human score.

In Table 3 we compare the correlation of the CE and human scores against that of well-known MT evaluation metrics (at the sentence level) and human scores on a test set for *P-ES-1gb* (values are similar for other datasets). The quality estimate predicted by our method correlates better with human scores than reference-based MT evaluation metrics. We apply bootstrapping re-sampling on the data and then use paired t-test to determine the statistical significance of the correlation differences (Koehn, 2004). The differences between all metrics and CE are statistically significant with 99.8% confidence. Different from these metrics, our method requires some training data for a given language-pair and text domain, but once ths training is done, it can be used to estimate the quality of any number of new sentences.

Results for the *post-edition* datasets vary considerably from system to system. This may indicate that different MT systems require more post-

BLEU-2	NIST	TER	Meteor	CE score
0.342	0.298	-0.263	0.376	0.602

Table 3: Correlation of MT evaluation metrics and our score with human annotation - *P-ES-1gb*

edition due to their translation quality. For example, taking the error for *P-ER-1*, of ~ 1.95, we can say that the CE system is able to predict, for a given source sentence, a post-edition time by source word that will deviate up to 1.95 seconds from the real post-edition time needed. The average errors found may seem a very large on a word-basis, but more investigation on the use of this type of CE score to aid translators in their post-edition work is necessary in this direction.

By analyzing the top features in all tasks with the manually annotated datasets we can highlight the following ones:

- source language and in/out-of-domain indicators (*WMT datasets*);
- source & target sentence 3-gram language model probability;
- source & target sentence lengths;
- percentages of types of word alignments;
- percentage and mismatch in the numbers and punctuation symbols in the source and target.

The first two features convey corpus information. Their impact in the performance is expected, given that it may be easier to translate between certain pairs of languages and in-domain sentences. The size of the source and target points out the difficulty of the translation (longer sentences are more difficult). Like the remaining features, it also expresses some form between source and target.

6 Discussion and conclusions

We have presented a series of experiments on a method for confidence estimation to MT that allows taking into account the contribution of different features and have also identified very informative and non-redundant features that improve the performance of the produced CE models. Although it is not directly possible to compare our results to previous work, because of the unavailability of the datasets used before, we consider our results to be satisfactory. Particularly in the case of the regression task, it is possible to have some intuition on what the impact of the error would be. For example, it would indicate crossing on average one category in the quality ranking of the tasks predicting adequacy scores (1 = worst, 5 = best), and only result in uncertainty in the boundaries between two adjacent categories in the *1-4 datasets*.

The sets of relevant features identified includes many features that have not been used before, including the average size of the phrases in the target, several types of mismatchings in the source and target, etc. Some of the others features have been used in previous work, but their exact definition is different here. For example, we use the *proportion* of aborted search nodes, instead of absolute values, and we compute the average number of alternative translations by using probabilistic dictionaries produced from word-alignment.

Besides directly using the estimated scores as quality indicators to professional translators or end-users, we plan to further investigate uses for the features selected across MT systems and language pairs from different MT points of view. In the experiments with the *NIST dataset*, the features found to be the most relevant are not those usually considered in SMT models. Simple features like the ratio of lengths of source and target sentences, the ambiguity of the source words, the coverage of the source sentence in the corpus are clearly good indicators of translation quality. A future direction will be to investigate whether these features could also be useful to improve the translations produced by SMT systems, e.g., in the following ways:

- Complement existing features in SMT models.
- Rerank n-best lists produced by SMT systems, which could make use of the features that are not local to single hipotheses.

As discussed in (Gamon et al., 2005), the readability of the sentence, expressed by features like 3-gram language models, is a good proxy to predict translation quality, even in terms of adequacy. Ultimately, automatic metrics such as NIST aim at simulating how humans evaluate translations. In that sense, the findings of our experiments with the manually annotated datasets could also be exploited from an MT evaluation point of view, for example, in the following ways:

- Provide additional features to a reference-based metric like that proposed by (Albrecht and Hwa, 2007a).
- Provide a score to be combined with other MT evaluation metrics using frameworks like

those proposed by (Paul et al., 2007) and (Giménez and Màrquez, 2008).

Our findings could also be used to provide a new evaluation metric on itself, with some function to optimize the correlation with human annotations, without the need of reference translations.

References

Albrecht, J. and R. Hwa. 2007a. A re-examination of machine learning approaches for sentence-level mt evaluation. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 880–887, Prague.

Albrecht, J. and R. Hwa. 2007b. Regression for sentence-level mt evaluation with pseudo references. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 296–303, Prague.

Blatz, J., E. Fitzgerald, G. Foster, S. Gandrabur, C. Goutte, A. Kulesza, A. Sanchis, and N. Ueffing. 2003. Confidence estimation for machine translation. Technical report, Johns Hopkins University, Baltimore.

Blatz, J., E. Fitzgerald, G. Foster, S. Gandrabur, C. Goutte, A. Kulesza, A. Sanchis, and N. Ueffing. 2004. Confidence estimation for machine translation. In *Proceedings of the 20th Conference on Computational Linguistics*, pages 315–321, Geneva.

Callison-Burch, C., C. Fordyce, P. Koehn, C. Monz, and J. Schroeder. 2008. Further meta-evaluation of machine translation. In *Proceedings of the 3rd Workshop on Statistical Machine Translation*, pages 70–106, Columbus.

Doddington, G. 2002. Automatic evaluation of machine translation quality using n-gram co-occurrence statistics. In *Proceedings of the 2nd Conference on Human Language Technology Research*, pages 138–145, San Diego.

Frenich, A. G., A. G. Jouan-Rimbaud, D. Massart, D. L. Kuttatharmmakul, S. Martinez Galera, and J. L. M. Martinez Vidal. 1995. Wavelength selection method for multicomponent spectrophotometric determinations using partial least squares. *Analist*, 120(12):2787–2792.

Gamon, M., A. Aue, and M. Smets. 2005. Sentence-level mt evaluation without reference translations: beyond language modeling. In *Proceedings of the European Association for Machine Translation Conference*, Budapest.

Gandrabur, S. and G. Foster. 2003. Confidence estimation for translation prediction. In *Proceedings of the 7th Conference on Natural Language Learning*, pages 95–102, Edmonton.

Giménez, J. and L. Màrquez. 2008. Heterogeneous automatic mt evaluation through non-parametric metric combinations. In *Proceedings of the 3rd International Joint Conference on Natural Language Processing*, pages 319–326, Hyderabad.

Jong, S De. 1993. Simpls: An alternative approach to partial least squares regression. *Chemometrics and Intelligent Laboratory Systems*, 18:251–263.

Kadri, Y. and J. Y. Nie. 2006. Improving query translation with confidence estimation for cross language information retrieval. In *Proceedings of the 15th Conference on Information and Knowledge Management*, pages 818–819, Arlington.

Koehn, P. and C. Monz. 2006. Manual and automatic evaluation of machine translation between european languages. In *Proceedings of the Workshop on Statistical Machine Translation*, pages 102–121, New York.

Koehn, P. 2004. Statistical significance tests for machine translation evaluation. In *Conference on Empirical Methods in Natural Language Processing*, Barcelona.

Kulesza, A. and A. Shieber. 2004. A learning approach to improving sentence-level mt evaluation. In *Proceedings of the 10th International Conference on Theoretical and Methodological Issues in Machine Translation*, Baltimore.

Papineni, K., S. Roukos, T. Ward, and W. Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting on Association for Computational Linguistics*, page 311318, Morristown.

Paul, M., A. Finch, and E. Sumita. 2007. Reducing human assessment of machine translation quality to binary classifiers. In *Proceedings of the 11th Conference on Theoretical and Methodological Issues in Machine Translation*, pages 154–162, Skovde.

Quirk, C. B. 2004. Training a sentence-level machine translation confidence measure. In *Proceedings of the 4th Conference on Language Resources and Evaluation*, pages 825–828, Lisbon.

Rosipal, R. and L. J. Trejo. 2001. Kernel partial least squares regression in reproducing kernel hilbert space. *Machine Learning Research*, 2:97–123.

Ueffing, N. and H. Ney. 2005. Application of word-level confidence measures in interactive statistical machine translation. In *Proceedings of the 10th Conference of the European Association for Machine Translation*, pages 262–270, Budapest.

Wold, S., A. Ruhe, H. Wold, and W. J. Dunn. 1984. The covariance problem in linear regression. the partial least squares (pls) approach to generalized inverses. *SIAM Journal on Scientific Computing*, 5:735–743.