Using on-line available sources of bilingual information for word-level machine translation quality estimation

Miquel Esplà-Gomis Felipe Sánchez-Martínez Mikel L. Forcada
Departament de Llenguatges i Sistemes Informàtics
Universitat d'Alacant, E-03071 Alacant, Spain
{mespla,fsanchez,mlf}@dlsi.ua.es

Abstract

This paper explores the use of external sources of bilingual information available on-line for word-level machine translation quality estimation (MTQE). These sources of bilingual information are used as a *black box* to spot sub-segment correspondences between a source-language (SL) sentence S to be translated and a given translation hypothesis T in the target-language (TL). This is done by segmenting both S and T into overlapping sub-segments of variable length and translating them into the TL and the SL, respectively, using the available bilingual sources of information *on the fly*. A collection of features is then obtained from the resulting sub-segment translations, which is used by a binary classifier to determine which target words in T need to be post-edited.

Experiments are conducted based on the data sets published for the word-level MTQE task in the 2014 edition of the Workshop on Statistical Machine Translation (WMT 2014). The sources of bilingual information used are: machine translation (Apertium and Google Translate) and the bilingual concordancer Reverso Context. The results obtained confirm that, using less information and fewer features, our approach obtains results comparable to those of state-of-the-art approaches, and even outperform them in some data sets.

1 Introduction

Recent advances in the field of machine translation (MT) have led to the adoption of this technology by many companies and institutions all around the world in order to bypass the linguistic barriers and reach out to broader audiences. Unfortunately, we are still far from the point of having MT systems able to produce translations with the level of quality required for dissemination in formal scenarios, where human supervision and MT post-editing are unavoidable. It therefore becomes critical to minimise the cost of this human post-editing. This has motivated a growing interest in the field of MT quality estimation (Blatz et al., 2004; Specia et al., 2010; Specia and Soricut, 2013), which is the field that focuses on developing techniques that allow to estimate the quality of the translation hypotheses produced by an MT system.

Most efforts in MT quality estimation (MTQE) are aimed at evaluating the quality of whole translated segments, in terms of post-editing time, number of editions needed, and other related metrics (Blatz et al., 2004). Our work is focused on the sub-field of *word-level MTQE*. The main advantage of word-level MTQE is that it allows not only to estimate the effort needed to post-edit the output of an MT system, but also to guide post-editors on which words need to be post-edited.

In this paper we describe a novel method which uses black-box bilingual resources from the Internet for word-level MTQE. Namely, we combine two on-line MT systems, Apertium[1] and Google Translate,[2] and the bilingual concordancer Reverso Context[3] to spot sub-segment correspondences between a sentence S in the source language (SL) and

[1] http://www.apertium.org
[2] http://translate.google.com
[3] http://context.reverso.net/translation/

a given translation hypothesis T in the target language (TL). To do so, both S and T are segmented into overlapping sub-segments of variable length and they are translated into the TL and the SL, respectively, by means of the bilingual sources of information mentioned above. These sub-segment correspondences are used to extract a collection of features that is then used by a binary classifier to determine the words to be post-edited. Our experiments confirm that our method provides results comparable to the state of the art using considerably fewer features. In addition, given that our method uses (on-line) resources which are publicly available on the Internet, once the binary classifier is trained it can be used for word-level MTQE on the fly for new translations.

The rest of the paper is organised as follows. Section 2 briefly reviews the state of the art in word-level MTQE. Section 3 describes our binary-classification approach, the sources of information, and the collection of features used. Section 4 describes the experimental setting used for our experiments, whereas Section 5 reports and discusses the results obtained. The paper ends with some concluding remarks and the description of ongoing and possible future work.

2 Related work

Some of the early work on word-level MTQE can be found in the context of interactive MT (Gandrabur and Foster, 2003; Ueffing and Ney, 2005). Gandrabur and Foster (2003) obtain confidence scores for each word t in a given translation hypothesis T of the SL sentence S to help the interactive MT system to choose the translation suggestions to be made to the user. Ueffing and Ney (2005) extend this application to word-level MTQE also to automatically reject those target words t with low confidence scores from the translation proposals. This second approach incorporates the use of probabilistic lexicons as a source of translation information.

Blatz et al. (2003) introduce a more complex collection of features for word-level MTQE, using semantic features based on WordNet (Miller, 1995), translation probabilities from IBM model 1 (Brown et al., 1993), word posterior probabilities (Blatz et al., 2003), and alignment templates from statistical MT (SMT) models. All the features they use are combined to train a binary classifier which is used to determine the confidence scores.

Ueffing and Ney (2007) divide the features used by their approach in two types: those which are independent of the MT system used for translation (system-independent), and those which are extracted from internal data of the SMT system they use for translation (system-dependent). These features are obtained by comparing the output of an SMT system T_1 to a collection of alternative translations $\{T_i\}_{i=2}^{N_T}$ obtained by using the N-best list from the same SMT system. Several distance metrics are then used to check how often word t_j, the word in position j of T, is found in each translation alternative T_i, and how far from position j. These features rely on the assumption that a high occurrence frequency in a similar position is an evidence that t_j does not need to be post-edited. Biçici (2013) proposes a strategy for extending this kind of system-dependent features to what could be called a system-independent scenario. His approach consists in choosing parallel data from an additional parallel corpus which are close to the segment S to be translated by means of feature-decay algorithms (Biçici and Yuret, 2011). Once this parallel data are extracted, a new SMT system is trained and its internal data is used to extract these features.

The MULTILIZER approach to (sentence-level) MTQE (Bojar et al., 2014) also uses other MT systems to translate S into the TL and T into the SL. These translations are then used as a pseudo-reference and the similarity between them and the original SL and TL sentences is computed and taken as an indication of quality. This approach, as well as the one by Biçici and Yuret's (2011) are the most similar ones to our approach. One of the main differences is that they translate whole segments, whereas we translate sub-segments. As a result, we can obtain useful information about specific words in the translation. As the approach in this paper, MULTILIZER also combines several sources of bilingual information, while Biçici and Yuret (2011) only uses one MT system.[4]

Among the recent works on MTQE, it is worth mentioning the QuEst project (Specia et al., 2013), which sets a framework for MTQE, both at the sentence level and at the word level. This framework defines a large collection of features which can be divided in three groups: those measuring the complexity of the SL segment S, those measuring the confidence on the MT system, and those measuring both fluency and adequacy directly on the

[4]To the best of our knowledge, there is not any public description of the internal workings of MULTILIZIER.

translation hypothesis T. In fact, some of the most successful approaches in the word-level MTQE task in the Workshop on Statistical Machine Translation in 2014 (WMT 2014) (Bojar et al., 2014) are based on some of the features defined in that framework (Camargo de Souza et al., 2014).

The work described in this paper is aimed at being a system-independent approach that uses available on-line bilingual resources for word-level MTQE. This work is inspired by the work by Esplà-Gomis et al. (2011), in which several on-line MT systems are used for word-level quality estimation in translation-memory-based computer aided translation tasks. In the work by Esplà-Gomis et al. (2011), given a translation unit (S, T) suggested to the translator for the SL segment to be translated S', MT is used to translate sub-segments from S into the TL, and TL sub-segments from T into the SL. Sub-segment pairs obtained through MT that are found both in S and T are an evidence that they are related. The alignment between S and S', together with the sub-segment translations between S and T help to decide which words in T should be modified to get T', the desired translation of S'. Based on the same idea, we built a brand-new collection of word-level features to extend this approach to MTQE. One of the main advantages of this approach as compared to other approaches described in this section is that it uses light bilingual information extracted from any available source. Obtaining this information directly from the Internet allows us to obtain on the fly confidence estimates for the words in T without having to rely on more complex sources, such as probabilistic lexicons, part-of-speech information or word nets.

3 Word-level quality estimation using bilingual sources of information from the Internet

The approach proposed in this work for word-level MTQE uses binary classification based on features obtained through sources of bilingual information available on-line. We use these sources of bilingual information to detect connections between the original SL segment S and a given translation hypothesis T in the TL following the same method proposed by Esplà-Gomis et al. (2011): all the overlapping sub-segments of S and T, up to a given length L, are obtained and translated into the TL and the SL, respectively, using the sources of bilingual information available. The resulting collections of sub-segment translations $M_{S\to T}$ and $M_{T\to S}$ can be then used to spot sub-segment correspondences between T and S. In this section we describe a collection of features designed to identify these relations for their exploitation for word-level MTQE.

Positive features. Given a collection of sub-segment translations M (either $M_{S\to T}$ or $M_{T\to S}$), one of the most obvious features consists in computing the amount of sub-segment translations $(\sigma, \tau) \in M$ that confirm that word t_j in T should be kept in the translation of S. We consider that a sub-segment translation (σ, τ) confirms t_j if σ is a sub-segment of S, and τ is a sub-segment of T that covers position j. Based on this idea, we propose the collection of positive features Pos_n:

$$\mathrm{Pos}_n(j, S, T, M) = \frac{|\{\tau : \tau \in \mathrm{conf}_n(j, S, T, M)\}|}{|\{\tau : \tau \in \mathrm{seg}_n(T) \wedge j \in \mathrm{span}(\tau, T)\}|}$$

where $\mathrm{seg}_n(X)$ represents the set of all possible n-word sub-segments of segment X and function $\mathrm{span}(\tau, T)$ returns the set of word positions spanned by the sub-segment τ in the segment T.[5] Function $\mathrm{conf}_n(j, S, T, M)$ returns the collection of sub-segment pairs (σ, τ) that confirm a given word t_j, and is defined as:

$$\mathrm{conf}_n(j, S, T, M) = \{(\sigma, \tau) \in M : \tau \in \mathrm{seg}_n(T) \wedge \sigma \in \mathrm{seg}_*(S) \wedge j \in \mathrm{span}(\tau, T)\}$$

where $\mathrm{seg}_*(X)$ is similar to $\mathrm{seg}_n(X)$ but without length constraints.[6]

Additionally, we propose a second collection of features, which use the information about the translation frequency between the pairs of sub-segments in M. This information is not available for MT, although it is for the bilingual concordancer we have used (see Section 4). This frequency determines how often σ is translated as τ and, therefore, how reliable this translation is. We define $\mathrm{Pos}_n^{\mathrm{freq}}$ to obtain these features as:

$$\mathrm{Pos}_n^{\mathrm{freq}}(j, S, T, M) = \sum_{\forall(\sigma,\tau)\in \mathrm{conf}_n(j,S,T,M)} \frac{\mathrm{occ}(\sigma, \tau, M)}{\sum_{\forall(\sigma,\tau')\in M} \mathrm{occ}(\sigma, \tau', M)}$$

where function $\mathrm{occ}(\sigma, \tau, M)$ returns the number of occurrences in M of the sub-segment pair (σ, τ).

[5] Note that a sub-segment τ may be found more than once in segment T: function $\mathrm{span}(\tau, T)$ returns all the possible positions spanned.

[6] Two variants of function conf_n were tried: one applying also length constraints when segmenting S (with the consequent increment in the number of features), and one not applying length constraints at all. Preliminary results confirmed that constraining only the length of τ was the best choice.

Both positive features, $\mathrm{Pos}(\cdot)$ and $\mathrm{Pos}^{\mathrm{freq}}(\cdot)$, are computed for t_j for all the values of sub-segment length n up to L. In addition, they can be computed for both $M_{S\to T}$ and $M_{T\to S}$, producing $4L$ positive features in total for each word t_j.

Negative features. Our negative features, i.e. those features that help to identify words that should be post-edited in the translation hypothesis T, are also based on sub-segment translations $(\sigma, \tau) \in M$, but they are used in a different way. Negative features use those sub-segments τ that fit two criteria: (a) they are the translation of a sub-segment σ from S but cannot be matched in T; and (b) when they are aligned to T using the Levenshtein edit distance algorithm (Levenshtein, 1966), both their first word θ_1 and last word $\theta_{|\tau|}$ can be aligned, therefore delimiting a sub-segment τ' of T. Our hypothesis is that those words t_j in τ' which cannot be aligned to τ are likely to need to be post-edited. We define our negative feature collection $\mathrm{Neg}_{mn'}$ as:

$$\mathrm{Neg}_{mn'}(j,S,T,M) = \sum_{\forall \tau \in \mathrm{NegEvidence}_{mn'}(j,S,T,M)} \frac{1}{\mathrm{alignmentsize}(\tau, T)}$$

where $\mathrm{alignmentsize}(\tau, T)$ returns the length of the sub-segment τ' delimited by τ in T. Function $\mathrm{NegEvidence}_{mn'}(\cdot)$ returns the set of τ sub-segments that are considered negative evidence and is defined as:

$$\mathrm{NegEvidence}_{mn'}(j,S,T,M) = \{\tau : (\sigma,\tau) \in M \wedge \sigma \in \mathrm{seg}_m(S) \wedge |\tau'| = n' \wedge \tau \notin \mathrm{seg}_*(T) \wedge \mathrm{IsNeg}(j,\tau,T)\}$$

In this function length constraints are set so that sub-segments σ take lengths $m \in [1, L]$.[7] However, the case of the sub-segments τ is slightly different: n' does not stand for the length of the sub-segments, but the number of words in τ which are aligned to T.[8] Function $\mathrm{IsNeg}(\cdot)$ defines the set of conditions required to consider a sub-segment τ a negative evidence for word t_j:

$$\mathrm{IsNeg}(j,\tau,T) = \exists j', j'' \in [1, |T|] : j' < j < j'' \wedge \mathrm{aligned}(t_{j'}, \theta_1) \wedge \mathrm{aligned}(t_{j''}, \theta_{|\tau|}) \wedge \nexists \theta_k \in \mathrm{seg}_1(\tau) : \mathrm{aligned}(t_j, \theta_k)$$

where $\mathrm{aligned}(X, Y)$ is a binary function that checks whether words X and Y are aligned or not.

[7] In contrast to the positive features, preliminary results showed an improvement in the performance of the classifier when constraining the length of the σ sub-segments used for each feature in the set.

[8] That is, the length of longest common sub-segment of τ and T.

Negative features $\mathrm{Neg}_{mn'}(\cdot)$ are computed for t_j for all the values of SL sub-segment lengths $m \in [1, L]$ and the number of TL words $n' \in [2, L]$ which are aligned to words θ_k in sub-segment τ. Note that the number of aligned words between T and τ cannot be lower than 2 given the constraints set by function $\mathrm{IsNeg}(j, \tau, T)$. This results in a collection of $L \times (L-1)$ negative features. Obviously, for these features only $M_{S\to T}$ is used, since in $M_{T\to S}$ all the sub-segments τ can be found in T.

4 Experimental setting

The experiments described in this section compare the results of our approach to those in the word-level MTQE task in WMT 2014 (Bojar et al., 2014), which are considered the state of the art in the task. In this section we describe the sources of bilingual information used for our experiments, as well as the binary classifier and the data sets used for evaluation.

4.1 Evaluation data sets

Four data sets for different language pairs were published for the word-level MTQE task in WMT 2014: English–Spanish (EN–ES), Spanish–English (ES–EN), English–German (EN–DE), and German–English (DE–EN). The data sets contain the original SL segments, and their corresponding translation hypotheses tokenised at the level of words. Each word is tagged by hand using three levels of granularity:

- **binary**: words are classified only taking into account if they need to be post-edited (class *BAD*) or not (class *OK*);
- **level 1**: extension of the binary classification which differentiates between *accuracy* errors and *fluency* errors;
- **multi-class**: fine-grained classification of errors divided in 20 categories.

In this work we focus on the binary classification, which is the base for the other classification granularities.

Four evaluation metrics were defined for this task:

- The F_1 score weighted by the rate ρ_c of instances of a given class c in the data set:

$$F_1^w = \sum_{\forall c \in C} \rho_c \frac{2 p_c r_c}{p_c + r_c}$$

where C is the collection of classes defined for a given level of granularity (OK and BAD for the binary classification) and p_c and r_c are the precision and recall for a class $c \in C$, respectively;

- The F_1 score of the less frequent class in the data set (class BAD, in the case of binary classification):

$$F_1^{\text{BAD}} = \frac{2 \times p_{\text{BAD}} \times r_{\text{BAD}}}{p_{\text{BAD}} + r_{\text{BAD}}};$$

- The Matthews correlation coefficient (MCC), which takes values in $[-1, 1]$ and is more reliable than the F_1 score for unbalanced data sets (Powers, 2011):

$$\text{MCC} = \frac{T_{\text{OK}} \times T_{\text{BAD}} - F_{\text{OK}} \times F_{\text{BAD}}}{\sqrt[2]{A_{\text{OK}} \times A_{\text{BAD}} \times P_{\text{OK}} \times P_{\text{BAD}}}}$$

  where T_{OK} and T_{BAD} stand for the number of instances correctly classified for each class, F_{OK} and F_{BAD} stand for the number of instances wrongly classified for each class, P_{OK} and P_{BAD} stand for the number of instances classified either as OK or BAD, and A_{OK} and A_{BAD} stand for the actual number of each class; and

- Total accuracy (ACC):

$$\text{ACC} = \frac{T_{\text{OK}} + T_{\text{BAD}}}{P_{\text{OK}} + P_{\text{BAD}}}$$

The comparison between the approach presented in this work and those described by Bojar et al. (2014) is based on the F_1^{BAD} score because this was the main metric used to compare the different approaches participating in WMT 2014. However, all the metrics are reported for a better analysis of the results obtained.

4.2 Sources of Bilingual Information

As already mentioned, two different sources of information were used in this work, MT and a bilingual concordancer. For our experiments we used two MT systems which are freely available on the Internet: Apertium and Google Translate. These MT systems were exploited by translating the sub-segments, for each data set, in both directions (from SL to TL and vice versa). It is worth noting that language pairs EN–DE and DE–EN are not available for Apertium. For these data sets only Google Translate was used.

The bilingual concordancer *Reverso Context* was also used for translating sub-segments. Namely, the sub-sentential translation memory of this system was used, which is a much richer source of bilingual information and provides, for a given SL sub-segment, the collection of TL translation alternatives, together with the number of occurrences of the sub-segments pair in the translation memory. Furthermore, the sub-segment translations obtained from this source of information are more reliable, since they are extracted from manually translated texts. On the other hand, its main weakness is the coverage: although Reverso Context uses a large translation memory, no translation can be obtained for those SL sub-segments which cannot be found in it. In addition, the sub-sentential translation memory contains only those sub-segment translations with a minimum number of occurrences. On the contrary, MT systems will always produce a translation, even though it may be wrong or contain untranslated out-of-vocabulary words. Our hypothesis is that combining both sources of bilingual information can lead to reasonable results for word-level MTQE.

For our experiments, we computed the features described in Section 3 separately for both sources of information. The value of the maximum sub-segment length L used was set to 5, which resulted in a collection of 40 features from the bilingual concordancer, and 30 from MT.[9]

4.3 Binary classifier

Esplà-Gomis et al. (2011) use a simple perceptron classifier for word-level quality estimation in translation-memory-based computer-aided translation. In this work, a more complex *multilayer perceptron* (Duda et al., 2000, Section 6) is used, as implemented in Weka 3.6 (Hall et al., 2009). Multilayer perceptrons (also known as *feedforward neural networks*) have a complex structure which incorporates one or more *hidden layers*, consisting of a collection of perceptrons, placed between the input of the classifier (the features) and the output perceptron. This hidden layer makes multilayer perceptrons suitable for non-linear classification problems (Duda et al., 2000, Section 6). In fact, Hornik et al. (1989) proved that neural networks with a single hidden layer containing a finite number of neurons are universal approximators and may therefore be able to perform better than a simple per-

[9] As already mentioned, the features based on translation frequency cannot be obtained for MT.

ceptron for complex problems. In our experiments, we have used a batch training strategy, which iteratively updates the weights of each perceptron in order to minimise a total error function. A subset of 10% of the training examples was extracted from the training set before starting the training process and used as a validation set. The weights were iteratively updated on the basis of the error computed in the other 90%, but the decision to stop the training (usually referred as the convergence condition) was based on this validation set. This is a usual practice whose objective is to minimise the risk of overfitting. The training process stops when the total error obtained in an iteration is worse than that obtained in the previous 20 iterations.[10]

Hyperparameter optimisation was carried out using a grid search (Bergstra et al., 2011) in a 10-fold cross-validation fashion in order to choose the hyperparameters optimising the results for the metric to be used for comparison, F_1 for class *BAD*:

- *Number of nodes in the hidden layer*: Weka (Hall et al., 2009) makes it possible to choose from among a collection of predefined network designs; the design performing best in most cases happened to have the same number of nodes in the hidden layer as the number of features.
- *Learning rate*: this parameter allows the dimension of the weight updates to be regulated by applying a factor to the error function after each iteration; the value that best performed for most of our training data sets was 0.9.
- *Momentum*: when updating the weights at the end of a training iteration, momentum smooths the training process for faster convergence by making it dependent on the previous weight value; in the case of our experiments, it was set to 0.07.

5 Results and discussion

Table 1 shows the results obtained by the baseline consisting on marking all the words as BAD, whereas Table 2 shows the reference results obtained by the best performing system according to the results published by Bojar et al. (2014). These tables are used as a reference for the results obtained with the approach described in this work.

language pair	weighted F_1	BAD F_1	MCC	accuracy
EN–ES	18.71	52.53	0.00	35.62
ES–EN	5.28	29.98	0.00	17.63
EN–ES	12.78	44.57	0.00	28.67
DE–EN	8.20	36.60	0.00	22.40

Table 1: Results of the "always BAD" baseline for the different data sets.

language pair	weighted F_1	BAD F_1	MCC	accuracy
EN–ES	62.00	48.73	18.23	61.62
ES–EN	79.54	29.14	25.47	82.98
EN–DE	71.51	45.30	28.61	72.97
DE–EN	72.41	26.13	16.08	76.14

Table 2: Results of the best performing systems for the different data sets according to the results published by Bojar et al. (2014).

Table 3 shows the results obtained when using Reverso Context as the only source of information. Using only Reverso Context leads to reasonably good results for language pairs EN–ES and EN–DE, while for the other two language pairs results are much worse, basically because no word was classified as needing to be post-edited. This situation is caused by the fact that, in both cases, the amount of examples of words to be post-edited in the training set is very small (lower than 21%). In this case, if the features are not informative enough, the strong bias leads to a classifier that always recommends to keep all words untouched. However, it is worth noting that with a small amount of features (40 features) state-of-the-art results were obtained for two data sets.[11] Namely, in the case of the EN–ES data set, the one with the largest amount of training instances, the results for the main metric (F_1 score for the less frequent class, in this case *BAD*) were better than those of the state of the art. In the case of the EN–DE data set the results are noticeably lower than the state of the art, but they are still comparable to them.

Table 4 shows the results obtained when combining the information from Reverso Context and the MT systems Apertium and Google Translate. Again, one of the best results is obtained for the EN–ES data set, which would again beat the state of the art for the F_1 score for the *BAD* class, and

[10] It is usual to set a number of additional iterations after the error stops improving, in case the function is in a local minimum, and the error starts decreasing again after a few more iterations. If the error continues to worsen after these 20 iterations, the weights used are those obtained after the iteration with the lowest error.

[11] We focus our comparison on the F_1 score for the *BAD* class because this was the metric on which the classifiers were optimised.

language pair	weighted F_1	BAD F_1	MCC	accuracy
EN–ES	60.18	49.09	16.28	59.46
ES–EN	74.41	0.00	0.00	82.37
EN–DE	65.88	41.24	17.05	65.71
DE–EN	67.82	0.00	0.00	77.60

Table 3: Results of the approach proposed in this paper for the same data sets used to obtain Table 2 using Reverso Context as the only source of bilingual information.

language pair	weighted F_1	BAD F_1	MCC	accuracy
EN–ES	61.43	49.03	17.71	60.91
ES–EN	75.87	10.44	9.61	81.82
EN–DE	66.75	43.07	19.38	78.71
DE–EN	75.00	40.33	25.85	76.03

Table 4: Results of the approach proposed in this work for the same data sets used to obtain Table 2 using both Reverso Context and both Google Translate and Apertium as the sources of bilingual information.

which obtained results still closer to those of the state of the art for the rest of metrics. In addition, the biased classification problem for data sets DE–EN and ES–EN is alleviated. Actually, the results for the DE–EN language pair are particularly good, and outperform the state of the art for all the metrics. The low F_1 score obtained for the ES–EN data set may be explained by the unbalanced amount of positive and negative instances. Actually, the ratio of negative instances is somewhat related to the results obtained: 35% for EN–ES, 17% for ES–EN, 30% for EN–DE and 21% for DE–EN. A closer analysis of the results shows that our approach is better when detecting errors in the *Terminology*, *Mistranslation*, and *Unintelligible* subclasses. The ratio of this kind of errors over the total amount of negative instances for each data set is again related to the results obtained: 73% for EN–ES, 27% for ES–EN, 47% for EN–DE and 35% for DE–EN. This information may explain the differences in the results obtained for each data set.

Again, it is worth noting that this light method using a reduced set of 70 features can obtain, for most of the data sets, results comparable to those obtained by approaches using much more features. For example, the best system for the data set EN–ES (Camargo de Souza et al., 2014) used 163 features, while the winner system for the rest of data sets (Biçici and Way, 2014; Biçici, 2013) used 511,000 features. The sources of bilingual information used in this work are rather rich; however, given that any source of bilingual information could be used on the fly, simpler sources of bilingual information could also be used. It would therefore be interesting to carry out a deeper evaluation of the impact of the type and quality of the resources used with this approach.

6 Concluding remarks

In this paper we describe a novel approach for word-level MTQE based on the use of on-line available bilingual resources. This approach is aimed at being system-independent, since it does not make any assumptions about the MT system used for producing the translation hypotheses to be evaluated. Furthermore, given that this approach can use any source of bilingual information as a black box, it can be easily used with few resources. In addition, adding new sources of information is straightforward, providing considerable room for improvement. The results described in Section 5 confirm that our approach can reach results comparable to those in the state of the art using a smaller collection of features than those used by most of the other approaches.

Although the results described in this paper are encouraging, it is worth noting that it is difficult to extract strong conclusions from the small data sets used. A wider evaluation should be done, involving larger data sets and more language pairs. As future work, we plan to extend this method by using other on-line resources to improve the on-line coverage when spotting sub-segment translations; namely, different bilingual concordancers and on-line dictionaries. Monolingual target-language information could also be obtained from the Internet to deal with fluency issues, for example, getting the frequency of a given n-gram from search engines. We will also study the combination of these features with features used in previous state-of-the-art systems (see Section 2) Finally, it would be interesting to try the new features defined here in word-level quality estimation for computer-aided translation tools, as in Esplà-Gomis et al. (2011).

Acknowledgements

Work partially funded by the Spanish Ministerio de Ciencia e Innovación through projects TIN2009-14009-C02-01 and TIN2012-32615 and by the European Commission through project PIAP-GA-2012-324414 (Abu-MaTran). We specially thank Reverso-Softissimo and Prompsit Language Engineering for providing the access to Reverso Context, and to the University Research Program for Google Translate that granted us access to the Google Translate service.

References

Bergstra, James S., Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. 2011. Algorithms for hyper-parameter optimization. In Shawe-Taylor, J., R.S. Zemel, P.L. Bartlett, F. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2546–2554. Curran Associates, Inc.

Biçici, Ergun and Andy Way. 2014. Referential translation machines for predicting translation quality. In *Proceedings of the 9th Workshop on Statistical Machine Translation*, pages 313–321, Baltimore, USA.

Biçici, Ergun. 2013. Referential translation machines for quality estimation. In *Proceedings of the 8th Workshop on Statistical Machine Translation*, pages 343–351, Sofia, Bulgaria.

Biçici, Ergun and Deniz Yuret. 2011. Instance selection for machine translation using feature decay algorithms. In *Proceedings of the 6th Workshop on Statistical Machine Translation*, pages 272–283.

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2003. Confidence estimation for machine translation. Technical Report Final Report of the Summer Workshop, Center for Language and Speech Processing, Johns Hopkins University, Baltimore, USA.

Blatz, John, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *Proceedings of the 20th International Conference on Computational Linguistics*, COLING '04.

Bojar, Ondrej, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, Radu Soricut, Lucia Specia, and Aleš Tamchyna. 2014. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the 9th Workshop on Statistical Machine Translation*, pages 12–58.

Brown, Peter F., Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Camargo de Souza, José Guilherme, Jesús González-Rubio, Christian Buck, Marco Turchi, and Matteo Negri. 2014. FBK-UPV-UEdin participation in the wmt14 quality estimation shared-task. In *Proceedings of the 9th Workshop on Statistical Machine Translation*, pages 322–328, Baltimore, USA, June. Association for Computational Linguistics.

Duda, R. O., P. E. Hart, and D. G. Stork. 2000. *Pattern Classification*. John Wiley and Sons Inc., second edition.

Esplà-Gomis, Miquel, Felipe Sánchez-Martínez, and Mikel L. Forcada. 2011. Using machine translation in computer-aided translation to suggest the target-side words to change. In *Proceedings of the Machine Translation Summit XIII*, pages 172–179, Xiamen, China.

Gandrabur, Simona and George Foster. 2003. Confidence estimation for translation prediction. In *Proceedings of the 7th Conference on Natural Language Learning at HLT-NAACL 2003 - Volume 4*, CONLL '03, pages 95–102.

Hall, Mark, Eibe Frank, Geoffrey Holmes, Bernhard Pfahringer, Peter Reutemann, and Ian H. Witten. 2009. The WEKA Data Mining Software: an Update. *SIGKDD Explorations*, 11(1):10–18.

Hornik, K., M. Stinchcombe, and H. White. 1989. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2(5):359–366, July.

Levenshtein, V.I. 1966. Binary codes capable of correcting deletions, insertions and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Miller, George A. 1995. Wordnet: A lexical database for English. *Communications of the ACM*, 38(11):39–41.

Powers, David M. W. 2011. Evaluation: From precision, recall and F-measure to ROC, informedness, markedness & correlation. *Journal of Machine Learning Technologies*, 2.

Specia, Lucia and Radu Soricut. 2013. Quality estimation for machine translation: preface. *Machine Translation*, 27(3-4):167–170.

Specia, Lucia, Dhwaj Raj, and Marco Turchi. 2010. Machine translation evaluation versus quality estimation. *Machine Translation*, 24(1):39–50.

Specia, Lucia, Kashif Shah, José GC De Souza, and Trevor Cohn. 2013. QuEst-a translation quality estimation framework. In *ACL (Conference System Demonstrations)*, pages 79–84.

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

Ueffing, Nicola and Hermann Ney. 2007. Word-level confidence estimation for machine translation. *Computational Linguistics*, 33(1):9–40, March.