The Prague Bulletin of Mathematical Linguistics
NUMBER 108 JUNE 2017 183–195

Towards Optimizing MT for Post-Editing Effort: Can BLEU Still Be Useful?

Mikel L. Forcada,[a] Felipe Sánchez-Martínez,[a] Miquel Esplà-Gomis,[a] Lucia Specia[b]

[a] Departament de Llenguatges i Sistemes Informàtics
Universitat d'Alacant, E-03690 Sant Vicent del Raspeig, Spain
[b] Department of Computer Science
University of Sheffield, Regent Court, 211 Portobello, Sheffield, UK

Abstract

We propose a simple, linear-combination automatic evaluation measure (AEM) to approximate post-editing (PE) effort. Effort is measured both as PE time and as the number of PE operations performed. The ultimate goal is to define an AEM that can be used to optimize machine translation (MT) systems to minimize PE effort, but without having to perform unfeasible repeated PE during optimization. As PE effort is expected to be an *extensive* magnitude (i.e., one growing linearly with the sentence length and which may be simply added to represent the effort for a set of sentences), we use a linear combination of extensive and *pseudo-extensive* features. One such pseudo-extensive feature, 1–BLEU times the length of the reference, proves to be almost as good a predictor of PE effort as the best combination of extensive features. Surprisingly, effort predictors computed using independently obtained reference translations perform reasonably close to those using actual post-edited references. In the early stage of this research and given the inherent complexity of carrying out experiments with professional post-editors, we decided to carry out an automatic evaluation of the AEMs proposed rather than a manual evaluation to measure the effort needed to post-edit the output of an MT system tuned on these AEMs. The results obtained seem to support current tuning practice using BLEU, yet pointing at some limitations. Apart from this intrinsic evaluation, an extrinsic evaluation was also carried out in which the AEMs proposed were used to build synthetic training corpora for MT quality estimation, with results comparable to those obtained when training with measured PE efforts.

Corresponding author: mlf@dlsi.ua.es
Cite as: Mikel L. Forcada, Felipe Sánchez-Martínez, Miquel Esplà-Gomis, Lucia Specia. Towards Optimizing MT for Post-Editing Effort: Can BLEU Still Be Useful?. The Prague Bulletin of Mathematical Linguistics No. 108, 2017, pp. 183–195. doi: 10.1515/pralin-2017-0019.

1. Introduction

Machine translation (MT) applications fall in two main groups: *assimilation* or *gisting*, and *dissemination*. Assimilation takes place when the raw MT output is used to make sense of documents written in a foreign language. Dissemination refers to the use of the MT output as a draft translation that is *post-edited* (corrected) by a professional to generate a publishable translation (Krings and Koby, 2001; O'Brien and Simard, 2014). The requirements of both groups of applications are quite different;[1] however state-of-the-art MT systems are usually optimized to produce translations that resemble existing references in a training or development set, regardless of their application. In statistical MT, this is done by using *automatic evaluation measures* (AEM) such as BLEU (Papineni et al., 2002), the most popular one. In neural MT —usually trained to maximize logarithmic likelihood— AEMs may still be used as a stopping criterion, or even as part of a loss function (Shen et al., 2016).

For dissemination, rather than optimizing the MT system to imitate existing, independently created, reference translations,[2] it would make more sense to optimize it to reduce post-editing (PE) *effort*. PE effort is an *extensive* magnitude, that is, one that grows linearly[3] with the sentence length and which may be simply added to represent the effort for a set of sentences. One straightforward measure of PE effort is PE *time*, since it is directly related to productivity. Additionally, the time devoted to PE is a key metric to budget a translation task.

In addition to PE time, one of the most used metrics for PE effort is human-targeted translation edit rate (HTER) (Snover et al., 2006, 2009; Specia and Farzindar, 2010). This metric computes the translation edit rate (TER) between the raw translation $\mathrm{MT}(s_i)$ produced by an MT system and a given (*human*, hence the *H*) PE of this translation $t_i^{(p)}$, that is, the minimum number of insertions, deletions and substitutions of one word or shifts of blocks of one or more words, divided by the length of the post-edited translation.

One of the main advantages of this metric over time is that it can be computed on any already post-edited translations. However, to use it as an extensive indicator of effort, rather than normalizing it by the length of the reference translation, we need to use the actual number of translation edits (NTE) instead of translation edit rates. The main disadvantage of NTE over PE time is that it disregards the cognitive effort of PE, that is, it does not take into account the time invested by post-editors reading

[1] For instance, a Russian–English translation with no articles (*some*, *a*, *the*), may be just about right for assimilation, but would need significant post-editing for dissemination.

[2] Reference translations that have been produced based on the source text only, and not by post-editing the output of the MT system being evaluated.

[3] The linear growth assumption should be evaluated empirically. For instance the performance of state-of-the-art systems (neural MT systems) seem to degrade with length (Toral and Sánchez-Cartagena, 2017) and could lead to non-linear PE times. However the linear approach seems to be a good starting point, given that in commercial scenarios, the cost of translation is measured based on the length of the text.

the translation and identifying the parts that need to be fixed, the time invested in checking external resources, such as dictionaries or bilingual concordancers, and the time spent revising the final translation. In contrast, PE time can only be measured in a controlled environment, which makes it less practical.

In dissemination applications of MT, it would therefore make sense to use PE effort metrics for model optimization. However, repeatedly collecting PE time or NTE during system optimization is unfeasible. Hundreds of thousands of candidate translations would have to be edited by professionals, a prohibitively expensive and time-consuming process. Datasets with reference translations are, on the other hand, abundant. Therefore, ideally one could optimize MT by using an AEM that, given the MT output and an independent reference translation, predicts the required PE effort.

A number of publicly available corpora provide PE times or raw and post-edited machine translations (see Section 3); however, to the best of our knowledge, while there has been extensive work in predicting PE time or PE rates as a MT quality estimation (QE) task (Specia and Soricut, 2013) (that is, without a reference translation) as part of shared tasks (Bojar et al., 2013, 2014, 2016), no AEM that could be used to optimize MT systems with respect to PE effort has yet been proposed. The only exception is the work of Denkowski (2015), who shows that when an AEM "tuned to post-editing effort is used as an objective function for system optimization, the resulting translations require less effort to edit than those from a BLEU-optimized system".

Denkowski (2015) used unpublished PE data and METEOR, a rather complex AEM relying on resources such as stemmers and paraphrase tables. This paper sets out to define very simple AEMs based on a linear combination of MT system-independent features which aim at predicting PE effort (either time or NTE) as an *extensive* magnitude. It also studies whether sentence-level BLEU computed on independent reference translations could actually be repurposed as a reasonable predictor of PE effort. This work is part of an ongoing research aimed at defining AEMs to be used to optimize MT systems to minimize PE effort.[4]

2. Predicting post-editing effort as an extensive quantity

Since PE effort is expected to be an extensive quantity, we propose using a linear combination of extensive and *pseudo-extensive* features. We will consider time and the number of edits as specific cases of effort (Forcada and Sánchez-Martínez, 2015). The effort of post editing the MT output for segment i in a translation job may be denoted by $T(s_i, MT(s_i))$, which will be approximated by a tunable AEM of the form

$$\hat{T}(s_i, MT(s_i), t_i; \vec{\mu}) = \sum_{j=1}^{n_F} \mu_j f_j(s_i, MT(s_i), t_i), \quad (1)$$

[4]One could imagine this as a linear per-word cost model with a discount proportional to various indicators of closeness to the reference.

where a single reference t_i is assumed, $f_j(s_i, MT(s_i), t_i)$ are the extensive and pseudo-extensive features, and $\vec{\mu}$ is the set of tunable parameters of the AEM. The coefficients μ_j may be obtained by linear regression on a training set.

2.1. Extensive features

The following list of simple extensive features has been preliminarily studied:

- Word-level length of raw MT output $MT(s_i)$ and reference segments t_i and their corresponding character-level counterparts.
- Word- and character-level Levenshtein-edit distances between $MT(s_i)$ and t_i.
- Word- and character-level components of the TER-style distance (Snover et al., 2006) between $MT(s_i)$ and t_i: number of insertions, deletions, substitutions, and block shifts for words and characters.
- $MT(s_i)$ word n-gram mismatches, i.e. number of sub-segments of length n in $MT(s_i)$ that do not appear in t_i, and vice versa, i.e. number of sub-segments of length n in t_i not appearing in $MT(s_i)$.

2.2. Pseudo-extensive features

Pseudo-extensive features may be easily derived from non-extensive AEM by combining them with the length of the reference segment, $len_W(t_i)$. In this paper we have studied the use of $sBLEU_n$, a sentence-level implementation of the well-known AEM $BLEU_n$ where n is the maximum n-gram size used; usually 4. The $sBLEU_n$ indicator takes values in $[0, 1]$ and is expected to be a *reverse* predictor of PE effort —the larger the $sBLEU_n$, the smaller the effort. Consequently, we use a *reversed* version of it so that the feature value is computed as

$$len_W(t_i) \times (1 - sBLEU_4(MT(s_i), t_i)) \quad (2)$$

where $sBLEU_4(\cdot, \cdot)$ is 4-gram sentence-smoothed implementation of BLEU ("Smoothing 3" by Chen and Cherry (2014), implemented in package MultEval as *JBLEU*).[5]

3. Experimental settings

3.1. Data sets

Several experiments were carried out with data sets based on those published for the shared task on MT quality estimation (QE) at the 2013, 2014 and 2016 editions of the Workshop on Statistical Machine Translation (WMT). Each data set consists

[5] As BLEU is unlikely to decrease linearly with effort, we tried a family of suitably transformed versions of Eq. (2), $len_W(t_i) \times (1 - (sBLEU_4(MT(s_i), t_i))^q)^p$, with $p, q > 0$. We found no significant improvement over $p = 1, q = 1$ by doing this in the range $[\frac{1}{3}, 3]$. Eq. (2) has intuitive interpretation: effort (cost) grows linearly with length, but effort is saved (discount) as BLEU gets higher.

	Translation direction	Num. of instances Training	Test
WMT'13	en→es	803	284
WMT'14	en→es	650	208
WMT'16	en→de	13,000	2,000

Table 1. Statistics about the corpora used in the experiments: translation direction, and number of training and test instances.

of: (a) a set of source language segments $\{s_i\}$; (b) the corresponding raw translation produced by an unknown MT system, which may not be the same system in some data sets; (c) an independent reference translation t_i for every source segment s_i, unrelated to the MT system being studied; and (d) the post-edited version $t_i^{(p)}$ of the MT output, together with the corresponding PE time in seconds, $T(s_i, MT(s_i))$. Corpus statistics are provided in Table 1.

Two of the data sets are for translation from English into Spanish (en→es) and were obtained from the data sets distributed as part of WMT'13 (Bojar et al., 2013)[6] and WMT'14 (Bojar et al., 2014),[7] respectively. Independent references were collected from the parallel data distributed for the shared MT task at the 2012 edition of WMT.[8] PE references were provided by the shared-task organizers.[9] The third data set is for English–German (en→de) translation and corresponds to WMT'16 MT QE shared task (Bojar et al., 2016).[10]

In all the experiments, the training–test division is the same performed for the corresponding WMT shared tasks. WMT'16 also provides development data, which was added to the training corpus.

3.2. Training and evaluation

The limited-memory, bound-constrained Broyden–Fletcher–Goldfarb–Shanno (L-BFGS-B) optimization algorithm (Byrd et al., 1995) implemented in the SciPy package (Walt et al., 2011) was used to learn the parameters $\vec{\mu}$ in Eq. (1) by directly mini-

[6] http://www.quest.dcs.shef.ac.uk/wmt13_files/

[7] http://www.quest.dcs.shef.ac.uk/wmt14_files/

[8] Independent references can be downloaded here: https://v.gd/indepref

[9] Post-edited references can be downloaded here: https://v.gd/peref

[10] Training and development data sets are available at: https://lindat.mff.cuni.cz/repository/xmlui/handle/11372/LRT-1646. Test data is available at http://www.quest.dcs.shef.ac.uk/wmt16_files_qe/task1_en-de_test.tar.gz

mizing the mean absolute error (MAE) over the training set

$$\text{MAE} = \frac{1}{n_{\text{train}}} \sum_{i=0}^{n_{\text{train}}} \left| \hat{T}(s_i, \text{MT}(s_i), t_i; \vec{\mu}) - T(s_i, \text{MT}(s_i)) \right|,$$

where n_{train} is the number of training examples. The models trained were then evaluated by computing the Pearson's correlation r between the predicted effort and the actual PE effort, as well as the MAE, this time over the n_{test} examples in the test set. The correlation and the MAE were computed in two situations, namely, using independent references and using the actual post-edited translations, and for: (a) the best combination of extensive features; (b) the pseudo-extensive version of sBLEU proposed in Section 2.2; (c) a combination of (a) and (b); and (d) an example-based baseline (see below).

3.3. A simple baseline

A simple example-based baseline computes, for a test instance $(\text{MT}(s_i), t_i)$, the character-level edit distance $d = \text{ED}_C(\text{MT}(s_i), t_i)$ and then estimates $\hat{T}(s_i, \text{MT}(s_i), t_i)$ as the average PE time of all training-set segments showing a distance d' which is the closest possible to d:

$$\tilde{T}(d) = \frac{1}{|\{s_j : \text{ED}_C(\text{MT}(s_j), t_j) = d'\}|} \sum_{s_j : \text{ED}_C(\text{MT}(s_j), t_j) = d'} T(s_j, \text{MT}(s_j)).$$

4. Results

4.1. Predicting time

Table 2 reports the PE time prediction results obtained with three groups of AEMs: the best-performing combinations of extensive features[11] in Section 2.1, the AEM using the pseudo-extensive feature based on $\text{sBLEU}_4(\text{MT}(s_i), t_i)$ (which will be called the *pseudo-extensive AEM* from now on), an AEM combining both, and the baseline defined in Section 3.3.

As can be seen, both the extensive and pseudo-extensive AEMs significantly outperform our example-based baseline. Note that the pseudo-extensive AEM shows an excellent performance, comparable to the AEM using the best combination of extensive features. This suggests that a simple AEM, $\mu|t_i|(1 - \text{sBLEU}_4(\text{MT}(s_i), t_i))$, with just one coefficient μ, would already be a reasonable estimator of time. The best independently performing extensive features are the number of mismatched n-grams between $\text{MT}(s_i)$ and t_i,[12] where n-gram matching is at the basis of BLEU.

[11] Other combinations were tried but could not be included given the space constraints.

[12] The best results correspond to $n = 2$ and $n = 3$.

		AEM predicting post-editing time							
		Best ext.		Pseudo-ext.		Combined		Baseline	
Corpus	**references**	r	MAE	r	MAE	r	MAE	r	MAE
WMT'13	independent	0.61	49.0 s	0.62	49.1 s	0.62	49.1 s	0.36	64.7 s
	postedited	0.67	45.2 s	0.68	46.0 s	0.68	44.8 s	0.33	72.5 s
WMT'14	independent	0.70	15.9 s	0.69	16.2 s	0.70	15.9 s	0.51	22.4 s
	postedited	0.85	11.8 s	0.81	13.7 s	0.85	11.6 s	0.63	18.3 s
WMT'16	independent	0.46	25.4 s	0.44	26.7 s	0.46	25.4 s	0.24	34.7 s
	postedited	0.55	21.7 s	0.50	24.7 s	0.55	21.7 s	0.36	30.3 s

Table 2. Pearson's correlation r and mean absolute error (MAE) in seconds for four time-predicting AEMs (best extensive, pseudo-extensive (modified BLEU), combination, and example-based baseline) and three different corpora, computed on independent and postedited references.

As expected, all the results included in Table 2 are substantially better for PE references than for independent ones. However, it is worth noting that they are not too distant. These results are encouraging, since they suggest that even when no PE references are available, for instance when optimizing statistical MT systems, the proposed AEMs can be useful.

How good are these results? As mentioned in Section 3, the data sets used in these experiments had previously been used for MT QE. For data set WMT'13, the Pearson correlation r and the MAE are available for the original task (Bojar et al., 2013, Table 18). The results obtained with our (rather simple) linear AEM (having access to a single reference) are around $r = 0.62$ and MAE $= 49$ s while those reported for MT QE (without access to a reference translation) range between $r = 0.42$ and $r = 0.68$ and between MAE $= 48$ s and MAE $= 71$ s. For data set WMT'14, only the MAE is available (Bojar et al., 2014, Table 16); our MAE are around 16 s while the results reported for MT QE range between 16.7 s and 21.5 s. As a contrast, ignoring the quality of $\mathrm{MT}(s)$, and using just the length of $\mathrm{MT}(s)$ as a single feature, without accessing the reference t_i, the results are slightly worse than our best predictors, but far better than the example-based baseline: $r = 0.57$ and MAE $= 52.0$ s for WMT'13, and MAE $= 18.7$ s for WMT'14. These results would suggest that more elaborate AEMs should be explored to give a better estimate of time; improvements are expected to happen through the introduction of both additional extensive features and additional pseudo-extensive versions of features based on non-extensive indicators.

4.2. Predicting the number of edits

Table 3 is analogous to the experiments in Table 2, but here the reference AEM is the number of translation edits (NTE) instead of PE time. Table 3 contains an addi-

	AEM predicting the number of translation edits									
	Best ext.		Pseudo-ext.		Combined		Baseline		Indep. NTE	
Corpus	r	MAE	r	MAE	r	MAE	r	MAE	r	MAE
WMT'13	0.82	3.4	0.81	3.5	0.82	3.4	0.65	4.7	0.81	3.9
WMT'14	0.69	2.8	0.69	2.9	0.70	2.8	0.41	4.3	0.70	5.0
WMT'16	0.75	2.3	0.58	2.5	0.75	2.3	0.42	3.1	0.73	3.7

Table 3. Pearson's correlation r and mean absolute error (MAE) in number of edit operations for four NTE-predicting AEMs (best extensive, pseudo-extensive (modified BLEU), combination, example-based baseline, and using simply the independent-reference NTE as a predictor) and three different corpora, computed on independent references.

tional column that contains the results obtained by using the NTE needed to convert the MT output into an independent reference to predict the actual NTE performed to convert the MT output into its post-edited version (HNTE or *human* NTE). This is used as a second baseline that allows to measure the difficulty of the task of predicting the actual number of edits done when post-editing. As can be seen, the independent value of NTE strongly correlates with the HNTE. It clearly outperforms the example-based baseline used in the previous experiment. However, as regards MAE, the best-extensive MAE and the pseudo-extensive MAE obtain clearly better results, especially for the case of the WMT'14 data. As in the previous experiments, the impact of combining extensive and pseudo-extensive features is almost negligible.

In general, one can see that the approaches in Table 3 correlate much better with HNTE than those in Table 2 with PE time. To explain this, note that HNTE cannot take cognitive (thinking, documentation) effort into account, while PE time naturally includes it. Since none of the features used in this work is capable of directly representing cognitive effort, it would seem logical that using them leads to AEMs showing a better correlation with HNTE than with PE time. It is worth mentioning that when predicting HNTE, the results are also more stable across corpora, ranging between 2 and 5 edit operations.

4.3. Extrinsic evaluation in a quality estimation task

AEMs were also evaluated extrinsically by using them to build synthetic corpora for training MT QE systems that predict time. Synthetic corpora were built as follows: 25% of the training data in each data set in Table 1 was used to train simple pseudo-extensive time-predicting AEMs of the form given in Eq. (2), in view of their performance. Each AEM was then used to predict from independent references the PE time of the remaining 75% of the corresponding corpus, which was then used as the training corpus for a linear regressor built on the baseline features used for Task 1.3 at WMT'13 (Bojar et al., 2013) and WMT'14 (Bojar et al., 2014). The MAE and

Corpus	Training set	PE time		NTE	
		r	MAE	r	MAE
WMT'13	original	0.61	52.8 s	0.75	4.0
	synthetic	0.60	52.1 s	0.71	4.1
WMT'14	original	0.61	18.8 s	0.61	3.3
	synthetic	0.58	17.9 s	0.51	3.4
WMT'16	original	0.40	29.4 s	0.66	2.6
	synthetic	0.39	28.1 s	0.54	2.8

Table 4. Pearson's correlation (r) and mean absolute error (MAE) in seconds for MT QE (PE time estimation) when using both the original training set and a synthetic training set obtained using pseudo-extensive time-predicting AEMs (modified BLEU) with three different corpora.

Pearson's correlation between the estimated PE time using both the original and the synthetic corpora to train the regressor were then compared.

The results of this evaluation, carried out with pseudo-extensive time-predicting and NTE-predicting AEMs for MT QE are shown in Table 4. As can be observed, the MAE obtained with the synthetic corpora are comparable to those obtained with the original training corpora, even though the synthetic corpora used were automatically annotated and are 25% smaller than the original corpora. In the case of the PE time, the Pearson's correlation is comparable, while it is significantly lower in the case of NTE. These results show the usefulness of extensive and pseudo-extensive AEMs for predicting PE time in applications other than optimizing MT.

4.4. Tuning MT with the new AEMs: a sanity check

The most objective way to test the usefulness of the new AEMs would be to tune two different SMT systems with the same development set, one following general practice (i.e., using document-level BLEU) and the other one using the sum of one of the new sentence-level AEMs over the whole development set, and then having professional translators edit the output of both. This would allow us to search for possible savings in PE time and number of edits, much in the same way as reported by Denkowski (2015). While straightforward, this is an expensive experiment that should only be carried out when one has good indications that it will lead to a conclusive result. In addition, the unpredictability of the behaviour of different professional translators may make it more difficult to extract conclusions from such experiment, especially in this very early stage of the research. Therefore, in order to obtain preliminary and more reliable initial results, we will resort to a quick "casting out nines" sanity check, as follows.

We repeatedly and randomly extract simulated development sets of $n_{dev} = 100$ sentences each from the test sets described in Section 3.1 without replacement.

The repeat rate is 0.4 times the size of the test set, to get stable statistics. Over each one of these sets $\{(MT(s_j), t_j)\}_{j=1}^{n_{dev}}$, we will compute three *budgeting* features:

- The total length of references $L = \sum_{k=1}^{n_{dev}} |t_i|$, which will be used as a baseline predictor of total effort for that development set which does not take quality into account.
- A measure based on document-level BLEU over the whole development set, $D = (1 - BLEU((MT(s_j), t_j)\}_{j=1}^{n_{dev}}) \times L$, which takes quality into account by establishing a document-level discount based on BLEU. Minimizing D is equivalent to maximizing $BLEU((MT(s_j), t_j)\}_{j=1}^{n_{dev}})$, which is common practice.
- A measure, $S = \sum_{k=1}^{n_{dev}} \hat{T}(s_k, MT(s_k), t_k; \mu)$, based on the sentence-level AEMs proposed in this paper.

We then study the correlation among them and with actual total effort for that development set $E = \sum_{k=1}^{n_{dev}} T(s_k MT(s_k))$. If the AEM designed is indeed an improvement, the correlation of S with E should be better than that of D (current practice) and much better than that of L (dummy baseline). The results are shown in Table 5. The main findings are as follows:

- The correlation of the pseudo extensive (S-pseudo) and best extensive (S-ext.) AEMs between them and with current BLEU optimization practice (D) is excellent (0.91 or higher). This would mean that current BLEU optimization practice should roughly lead to equivalent results compared to using the new AEMs proposed here.
- The correlation of S-ext., S-pseudo, and current practice (D) with E (time) is reasonable for WMT'13 and WMT'14 while it is only moderate for WMT'16. Correlation with E (edits) is reasonably good for the three corpora.
- The correlation of total length L (which does not take quality into account) with E (time and edits) is surprisingly high and not too far from that obtained with actual AEMs. This could point at limitations of BLEU,[13] as well as of the simple AEMs proposed in this paper, but could also be due to the fact that the underlying MT systems had already been optimized using BLEU and that under those conditions, length is a good enough prediction of PE effort.

5. Concluding remarks

This paper introduces new automatic evaluation measures (AEM) for MT aimed at approximating post-editing (PE) effort. Such metrics would allow optimizing MT systems with respect to PE effort, therefore potentially reducing the cost of translation for dissemination purposes.

We have analyzed the performance of simple AEMs based on extensive and pseudo-extensive features for predicting PE time and the number of translation edits performed during PE (HNTE). The results allow us to conclude that: (a) the AEMs pro-

[13] For instance, Denkowski and Lavie (2012) showed that BLEU did not significantly change after post-editing.

Correlation	Dataset	E (time)	E (edits)	L	D	S-pseudo	S-ext.
E (time)	WMT'13	1.000	0.611	0.588	0.648	0.646	0.609
	WMT'14		0.837	0.649	0.740	0.739	0.728
	WMT'16		0.411	0.356	0.433	0.434	0.455
E (edits)	WMT'13	0.611	1.000	0.688	0.783	0.790	0.810
	WMT'14	0.837		0.605	0.690	0.690	0.645
	WMT'16	0.411		0.417	0.598	0.576	0.770
L	WMT'13	0.588	0.688	1.000	0.878	0.876	0.924
	WMT'14	0.649	0.605		0.906	0.914	0.942
	WMT'16	0.356	0.417		0.684	0.726	0.760
D	WMT'13	0.648	0.783	0.878	1.000	0.999	0.941
	WMT'14	0.740	0.690	0.906		0.999	0.953
	WMT'16	0.433	0.598	0.684		0.990	0.922
S-pseudo	WMT'13	0.646	0.790	0.876	0.999	1.000	0.965
	WMT'14	0.739	0.690	0.914	0.999		0.937
	WMT'16	0.434	0.576	0.726	0.990		0.914
S-ext.	WMT'13	0.609	0.810	0.924	0.941	0.965	1.000
	WMT'14	0.728	0.645	0.942	0.953	0.937	
	WMT'16	0.455	0.770	0.760	0.922	0.914	

Table 5. Pearson correlation observed between randomly-sampled development tests among PE effort E *(time and edits), total length* L*, total length times one minus BLEU (*D*), and effort predictions* S *using pseudo-extensive and extensive AEMs.*

posed perform similarly both on post-edited and independent references, which makes them easier to use to optimize MT systems; (b) the proposed AEMs would not seem to be able to improve current optimization practice (based on BLEU); (c) BLEU is still quite far from actually being able to reliably predict effort (supporting the findings by Denkowski (2015)); and (d) time prediction is however good enough to become useful in other related tasks, such as creating training corpora for MT QE.

Future work will evaluate more elaborate AEMs for predicting PE effort based on the features described and other MT-system-independent features, and will also analyse the impact of using such predictions when actually optimizing MT systems with respect to PE effort in real translation tasks (as was done by (Denkowski, 2015), but using features with simpler interpretation than the METEOR metric).

Acknowledgements: Work supported by the Spanish government through project EFFORTUNE (TIN2015-69632-R) and through grant PRX16/00043 for Mikel L. Forcada, and by the European Commission through QT21 project (H2020 No. 645452).

Bibliography

Bojar, Ondřej, Christian Buck, Chris Callison-Burch, Christian Federmann, Barry Haddow, Philipp Koehn, Christof Monz, Matt Post, Radu Soricut, and Lucia Specia. Findings of the 2013 Workshop on Statistical Machine Translation. In *Proceedings of the Eighth Workshop on Statistical Machine Translation*, pages 1–44, Sofia, Bulgaria, August 2013.

Bojar, Ondrej, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. Findings of the 2014 Workshop on Statistical Machine Translation. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, MD, USA, 2014.

Bojar, Ondřej, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Aurelie Neveol, Mariana Neves, Martin Popel, Matt Post, Raphael Rubino, Carolina Scarton, Lucia Specia, Marco Turchi, Karin Verspoor, and Marcos Zampieri. Findings of the 2016 Conference on Machine Translation. In *Proceedings of the First Conference on Machine Translation*, pages 131–198, Berlin, Germany, August 2016. URL `http://www.aclweb.org/anthology/W/W16/W16-2301`.

Byrd, Richard H, Peihuang Lu, Jorge Nocedal, and Ciyou Zhu. A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, 16(5):1190–1208, 1995.

Chen, Boxing and Colin Cherry. A Systematic Comparison of Smoothing Techniques for Sentence-Level BLEU. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367, Baltimore, Maryland, USA, June 2014. URL `http://www.aclweb.org/anthology/W/W14/W14-3346`.

Denkowski, Michael. *Machine Translation for Human Translators*. PhD thesis, Carnegie Mellon University, May 2015.

Denkowski, Michael and Alon Lavie. Challenges in Predicting Machine Translation Utility for Human Post-Editors. In *Proceedings of AMTA 2012*, 2012.

Forcada, Mikel L. and Felipe Sánchez-Martínez. A general framework for minimizing translation effort: towards a principled combination of translation technologies in computer-aided translation. In *Proceedings of the 18th Annual Conference of the European Association for Machine Translation*, pages 27–34, Antalya, Turkey, 2015.

Krings, Hans P and Geoffrey S Koby. *Repairing texts: empirical investigations of machine translation post-editing processes*, volume 5. Kent State University Press, 2001.

O'Brien, Sharon and Michel Simard. Introduction to special issue on post-editing. *Machine Translation*, 28(3-4):159–164, 2014.

Papineni, Kishore, Salim Roukos, Todd Ward, and Wei-Jing Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

Shen, Shiqi, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. Minimum Risk Training for Neural Machine Translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692, Berlin, Germany, August 2016. URL `http://www.aclweb.org/anthology/P16-1159`.

Snover, Matthew, Bonnie Dorr, Richard Schwartz, Linnea Micciulla, and John Makhoul. A study of translation edit rate with targeted human annotation. In *Proceedings of the Meeting of the Association for Machine Translation in the Americas*, volume 200, pages 223–231, 2006.

Snover, Matthew, Nitin Madnani, Bonnie J. Dorr, and Richard Schwartz. Fluency, adequacy, or HTER?: exploring different human judgments with a tunable MT metric. In *Proceedings of the Fourth Workshop on Statistical Machine Translation*, pages 259–268, Athens, Greece, 2009. Association for Computational Linguistics.

Specia, Lucia and Atefeh Farzindar. Estimating machine translation post-editing effort with HTER. In *Proceedings of the Second Joint EM+/CNGL Workshop Bringing MT to the User: Research on Integrating MT in the Translation Industry*, pages 33–41, Denver, USA, 2010.

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

Toral, Antonio and M. Víctor Sánchez-Cartagena. A Multifaceted Evaluation of Neural versus Phrase-Based Machine Translation for 9 Language Directions. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*, pages 1063–1073. Association for Computational Linguistics, 2017. URL `http://aclweb.org/anthology/E17-1100`.

Walt, Stéfan van der, S Chris Colbert, and Gael Varoquaux. The NumPy array: a structure for efficient numerical computation. *Computing in Science & Engineering*, 13(2):22–30, 2011.

Address for correspondence:
Mikel L. Forcada
`mlf@dlsi.ua.es`
Departament de Llenguatges i Sistemes Informàtics
Universitat d'Alacant, E-03690 Sant Vicent del Raspeig, Spain