Ranking Machine Translation Systems via Post-Editing

Wilker Aziz[1], Ruslan Mitkov[1] and Lucia Specia[2]

[1] Research Group in Computational Linguistics
University of Wolverhampton, UK
{W.Aziz,R.Mitkov}@wlv.ac.uk
[2] Department of Computer Science
University of Sheffield, UK
l.specia@sheffield.ac.uk

Abstract. In this paper we investigate ways in which information from the post-editing of machine translations can be used to rank translation systems for quality. In addition to the commonly used edit distance between the raw translation and its edited version, we consider post-editing time and keystroke logging, since these can account not only for technical effort, but also cognitive effort. In this system ranking scenario, post-editing poses some important challenges: i) multiple post-editors are required since having the same annotator fixing alternative translations of a given input segment can bias their post-editing; ii) achieving high enough inter-annotator agreement requires extensive training, which is not always feasible; iii) there exists a natural variation among post-editors, particularly w.r.t. editing time and keystrokes, which makes their measurements less directly comparable. Our experiments involve untrained human annotators, but we propose ways to normalise their post-editing effort indicators to make them comparable. We test these methods using a standard dataset from a machine translation evaluation campaign and show that they yield reliable rankings of systems.

Keywords: machine translation evaluation

1 Introduction

Machine Translation (MT) evaluation is the problem of assessing the quality of machine translated text. It is used both to measure the quality of an individual system and to rank a set of systems with respect to the quality of the translations they produce. MT evaluation can be automatic, semi-automatic and manual. In automatic evaluation, MT output is automatically and systematically compared to an independently obtained set of human-produced gold-standard (reference) translations. Popular metrics are based on shallow comparisons such as edit distance (e.g. TER), exact matching of n-grams (e.g. BLEU), or matching of synonyms and short paraphrases (e.g. METEOR, TERp). Overall, these metrics compute a shallow notion of overlap between the MT output and the available reference translation(s). Due to their limited level of linguistic analysis, they only cover a few aspects of quality.

Automatic metrics may easily penalise a valid translation because it has been phrased differently from the available references. Increasing the number of references per input segment is known to be helpful in that it increases the chance of finding a close match to the MT output. However, the space of possible translations is very large [1], and except for very small scale datasets, it is impractical to prepare gold-standard sets to account for such variability. To overcome this problem, in a typical scenario of semi-automatic evaluation, human annotators are asked to post-edit the output of MT systems producing a set of targeted reference translations. Post-editing guidelines may be task dependent, but generally it is expected that meaning and grammar issues (and less often, style issues) in the original MT output are fixed. In settings where the goal is to perform cheap and fast evaluation of MT quality, a common strategy is that of minimally post-editing the MT output, where only major issues are fixed. Once the targeted references are obtained, evaluation can be performed using standard automatic metrics. A popular metric is the Human-mediated Translation Edit Rate([2]), or HTER. HTER estimates the minimum number of edit operations necessary to fix the MT output. In this setting, MT quality is quantified by the amount of post-editing effort necessary to fix it. HTER was used as the official metric to compare MT systems as part of the recent DARPA GALE program [3]. Provided with extensive and comprehensive guidelines and training, professional translators involved in the evaluation were reported to have achieved high inter-annotator agreement, making the HTER assessment reliable. However, most evaluation settings are not able to afford the preparation, training, time and costs required to achieve similar results.

Lastly, manual evaluation is usually done via human scoring/error counts or ranking, that is, annotators are given guidelines on how to score or rank machine translated text. Manual evaluation is known to suffer from low inter-annotator agreement due to the subjectivity of the task and it is also cognitively demanding. It typically involves mentally constructing a reference translation which is then used to penalise/reward the MT at a sub-sentence level. Particularly for MT system ranking, because of the high cost of this type of evaluation, it is often not possible to compare all systems against all other systems for the complete dataset. Instead, samples are considered for partial rankings and global rankings are extrapolated from these using heuristics. For example, the evaluation campaigns organised yearly by the Workshop on Machine Translation (WMT) [4] use human ranking for both MT evaluation and metrics evaluation (i.e., assessment of automatic metrics of MT evaluation). In addition, WMT rely on volunteers or mechanical turkers to produce annotations, who are not professionals and would not be willing to undergo many hours of training activities. Therefore, the reliability of the global rankings provided have been frequently questioned, e.g., [5, 6].

In this paper we focus on post-editing as a more natural and objective way of producing gold-standard annotations for MT evaluation. However, our goal is to move away from standard metrics like HTER. A limitation of this and other edit distance metrics is that they cannot fully capture the effort resulting from post-editing. Certain operations can be more difficult than others, based not only on the type of edit (deletion, insertion, substitution), but also on the words being edited. We use information about the post-editing process to attempt to quantify this effort: post-editing time and keystrokes (counts, groups, etc.). Previous work has proposed post-editing time as a practical way of capturing both technical and cognitive aspects of post-editing effort [7]. However, so far post-editing time has only been used as a measure of translation quality in very controlled environments [8], mostly comparing post-editing against translation from scratch, based on annotations from the same translator performing both tasks. A challenge that arises from the use of post-editing for the ranking of MT systems is the fact that multiple annotators are necessary, since having the same annotator fixing alternative translations for the same source text can bias their post-editing. For example, an annotator may tend to produce a translation that is similar to the one previously post-edited for another system, possibly making fewer edits than if the source sentence has been seen for the first time. This becomes particularly a problem when post-editing time and keystrokes are measured, since different annotators work at different paces and may use different strategies to perform the same post-editing. While on average certain post-editors are simply consistently slower/faster than others, other factors impact post-editing speed and strategies for particular segments (e.g. the length of the segment), making normalisations across annotators far from trivial.

We investigate the use of post-editing to rank MT systems using very simple guidelines and multiple un-trained, non-professional post-editors, followed by strategies to normalise their post-editing effort indicators so that they become comparable across editors. Using a subset of the WMT12's dataset of English-Spanish translations, we show that metrics based on post-editing time and edit distance are subject to large variability across editors and should therefore be normalised before used for comparison.

This paper is organized as follows: Sect. 2 describes the normalization techniques, Sect. 3 contains experimental results and Sect. 4 summarises our findings.

2 Method

The difficulty in using post-editing effort indicators to rank MT systems arises from the "human in the loop" aspect of post-editing. People work at different paces, they have different skills, and they react differently to different type of errors. This variation becomes more evident when non-professional annotation is used.

Consider a set of S input sentences and their alternative translations produced by M systems. In this dataset a task is a pair made of an input sentence and one possible translation produced by a specific system. Each of the $S \times M$ tasks is identified by $t = (i, m)$ where i refers to the input sentence, and m refers to the system that produced the translation. Tasks are assigned to a set of A annotators respecting the constraint that annotators should never be presented with a task that contains an input sentence that they have already seen (let us call this constraint **one-only**). This means that to gather annotation for the whole dataset it is necessary to have as many annotators as there are systems in the comparison (i.e. $A \geq M$). In general terms, to have k annotations for each task, it is necessary to have $k \times M$ annotators. However if it is not possible to have as many annotators, one can randomly skip tasks (this relaxation is similar to observing partial ordering in human ranking).

Assume that annotators post-edit their tasks following simple guidelines producing annotations that include post-editing effort indicators such as the targeted reference, post-editing time, keystrokes count and edit distance. The challenge at hand consists in ranking translations of the same input sentence, therefore post-edited by different annotators, using the effort indicators available without assuming they are directly comparable. For instance, one could be interested in using post-editing time (or the HTER score) as observed from different annotators to rank alternative translations of the same input sentence according to how time consuming they are (or how many edit operations they require), but we know these are not comparable across annotators.

2.1 Mean Normalisation

Consider as variables all measurements from the post-editing process, e.g., post-editing time. Mean normalisation is a fairly standard technique broadly used in machine learning to make a variable assume mean value close to 0 and variance close to 1. This technique adjusts the values of a variable to $(x - \mu)/\sigma$, where x is the original value of the variable and μ and σ are the variable's mean and standard deviation, as observed in a large dataset. This normalisation is done independently for each annotator as well as independently for each variable. We then proceed by sorting the tasks in terms of the normalised feature.

2.2 Regression

Mean normalisation is simply a mathematical trick to adjust a variable as to make it have standard parameters. It oversimplifies the complex problem of comparing indicators from different annotators. Furthermore, it can only be used with one variable at a time, preventing the combination of different types of effort indicators.

To cope with these limitations, we propose to learn how to compare effort indicators across annotators based on a training set in which we can observe their variance in performing the same tasks. In this training set, each input sentence is arbitrarily assigned one - and only one - translation, and all annotators post-edit it. Figure 1 exemplifies how tasks are assigned differently in the two cases. Treating these indicators as features in machine learning scenario is quite appealing, but there is no gold-standard annotation for post-editing effort.[3]

This problem can be seen as the problem of learning a latent scale of post-editing effort onto which all the annotators' effort indicators can be mapped. If we had at our disposal an infinite number of annotators performing the same tasks, we would have access to each task's expected post-editing time (the same for HTER and keystrokes). It would not be absurd to claim that the expected post-editing time can be seen as a gold-standard annotation onto which the features of an individual annotator could be mapped. In practice, because of annotation costs, no more than a few annotators can be used. Therefore we use the sample mean in the training set instead of the unknown expectation. Section 3 presents empirical results to support that such approximation is reasonable.

In a nutshell, we use regression techniques to learn from the training set how to map input features in the form of post-editing effort indicators onto the approximated gold-standard. We learn one such a function for each annotator. At test time, we use them to convert effort indicators of each annotator individually into the gold-standard

[3] Coming up with gold-standard annotation for post-editing effort is a non-trivial task. In Sect. 3 we show that an attempt using human scoring on post-editing effort resulted in a dataset with extremely low inter-annotator agreement.

Input	System	A1	A2	A3	Effort	Approximation
			Training			
50	4	$\mathbf{F}^1_{(50,4)}$	$\mathbf{F}^2_{(50,4)}$	$\mathbf{F}^3_{(50,4)}$	?	$\mu^f_{(50,4)}$
51	5	$\mathbf{F}^1_{(51,5)}$	$\mathbf{F}^2_{(51,5)}$	$\mathbf{F}^3_{(51,5)}$	?	$\mu^f_{(51,5)}$
52	6	$\mathbf{F}^1_{(52,6)}$	$\mathbf{F}^2_{(52,6)}$	$\mathbf{F}^3_{(52,6)}$	?	$\mu^f_{(52,6)}$
			Test			
100	4	$\mathbf{F}^1_{(100,4)}$			?	$y^1_{(100,4)}$
	5		$\mathbf{F}^2_{(100,5)}$		?	$y^2_{(100,5)}$
	6			$\mathbf{F}^3_{(100,6)}$	?	$y^3_{(100,6)}$

Fig. 1: Example of how tasks are assigned differently for training and test. $\mathbf{F}^a_{(i,m)}$ denotes the vector of post-editing indicators from annotator a for task (i, m). At training time annotators perform the same tasks, the gold-standard effort label is unknown and approximated as $\mu^f_{(i,m)}$, i.e., the mean value of feature f observed for task (i, m). At test time each annotator is assigned an alternative translation of the same input. We use $y^a_{(i,m)}$, the predicted value of feature f given annotator's a effort indicators for task (i, m), to rank the alternative translations.

scale where they can be compared (column "Approximation" in Fig. 1). We then rank tasks w.r.t. the predicted mean.

3 Experiments

Our task is to rank alternative translations of an input sentence on the basis of post-editing effort indicators. This section presents the collection of the dataset and experiments with different normalisation and regression techniques.

3.1 Data Collection

We selected a subset of the WMT12's English-Spanish translation evaluation test set. For the training set we selected 200 input sentences. Each of these sentences was then assigned one translation by a randomly selected system (resulting 200 tasks). The test set, on the other hand, is made of 100 input sentences along with their alternative translations by 10 systems[4] (resulting 1,000 tasks). We had 10 volunteer English-Spanish bilingual speakers (native Spanish speakers) performing the post-editing of these machine translations. In summary, there were i) 200 tasks in the training set and all annotators performed each one of them, and ii) 1,000 tasks in the test set shared between 10 annotators observing the **one-only** constraint.

Post-editing was performed using the post-editing tool PET [9], which collects information such as post-editing time and keystrokes at the segment-level in the background, while post-editing is done. The participants were provided with simple guidelines (perform minimum post-editing) and a 10-minute video demonstrating how the

[4] Note that there were 11 systems participating in this task, we left **UK** out.

post-editing tool should be used. After they post-edited each translation, they were asked to assign a score from 1 to 4 to quantify the effort spent on post-editing.[5] To illustrate how post-editing information can be more reliable than human scoring, we computed Coehn's κ inter-annotator agreement [10] for these scores on post-editing effort. We observed very low values of κ, ranging from 0.12 to 0.44 with average 0.269 ± 0.082.

3.2 Ranking with Post-Editing

The most obvious baseline strategy for ranking translations on the basis of post-editing annotation is to chose an effort indicator, assume it is comparable across annotators, and rank translations according to its observed values. We refer to this baseline as **AI** (as in "as is"). The second strategy is mean normalisation, referred to as **MN** (Sect. 2.1), where we extracted the parameters μ and σ from the training set for each annotator and for each of the target features. Finally, we tested regression algorithms (**R** - regularised linear regression, **B** - Bayesian regression, and **S** - SVR with *rbf* kernel)[6] using different feature sets to predict the expected post-editing time.

WMT assigns an overall quality score to each system by drawing pairwise comparisons from the human rankings and computing the proportion $\mathrm{score}(s) = \frac{\mathrm{win}(s)}{\mathrm{win}(s)+\mathrm{loss}(s)}$ [5], that is, how many times a system wins a pairwise comparison out of the comparisons it won or lost. We use the same equation, however rather than using human rankings to decide who wins or loses a pairwise comparison, we use the value of the target feature: post-editing time in **AI**, normalised post-edited time in **MN**, or the expected post-editing time as predicted from effort indicators in **R**, **B** and **S**.

For each alternative translation of an input segment, we observe post-editing features from different annotators. Using one of the aforementioned methods we predict their expected values for the target feature, making them comparable. In this space of comparable feature values we perform ranking of alternative translations. We do that for three different target features (time, keystrokes and HTER) and report the results in terms of rank correlation to WMT's official ranking. Table 1 shows the system-level rank correlation in terms of Spearman's ρ coefficient. We also investigate the agreement between our strategies and WMT's human rankings w.r.t the pairwise comparisons themselves. Table 2 shows the segment-level rank correlation in term of Kendall's τ coefficient.

In *Regression 1*, the annotator's time is used alone to predict the task's expected time (same for keystrokes and HTER). In *Regression 4*, the length of the input (in number of tokens), the annotator's time, keystrokes and HTER score are used as features to predict the task's expected value for time, keystrokes or HTER. We boldface the most successful target in each column and star the best strategy in each row. A double star shows the best among all.

[5] 1 - complete re-translation needed; 2 - a lot of post-editing needed, but quicker than re-translation; 3 - a little post-editing needed; and 4 - no modification needed.

[6] As available in the scikit learn toolkit (`http://scikit-learn.org/`), with hyper-parameters tuned using the toolkit's randomised search in 5-fold cross-validation.

Table 1: System-level rank correlation

Target	AI	MN	Regression 1	Regression 4
time	0.3696	**0.7333****	**0.6969** (R)	**0.5757** (B)
keystrokes	0.4787	0.6121*	0.5878 (R)	0.5636 (R)
HTER	**0.5393**	0.3939	0.4181 (S)	0.5636* (R)

Table 2: Segment-level rank correlation

Target	AI	MN	Regression 1	Regression 4
time	0.0975	0.1555	0.2054 (R)	0.2451* (S)
keystrokes	**0.1941**	0.2189	**0.3065*** (S)	0.2870 (S)
HTER	0.1794	**0.2637**	0.2693 (R)	**0.3559**** (S)

First of all, both tables show that using the post-editing indicators "as is" is the worst strategy. Note that even HTER is improved by the proposed normalisations: segment level rank correlation jumps from 0.17 (AI) to 0.26 without additional features (*Regression 1*) and to 0.35 with additional features (*Regression 4*). Moreover the segment-level correlation for post-editing time improves from 0.09 to 0.245. Table 2 shows that post-editing time is less efficient than HTER, possibly because HTER ranges over a smaller interval than post-editing time, making the regression problem easier (more predictable scores).

It is interesting to notice the mismatch between segment- and system-level correlation w.r.t. the best performing strategies. The best performing system-level metric is the mean normalised time, while HTER using SVR performs the best at the segment-level. Mean normalisation relies on accumulated statistics over the entire training set, perhaps this explains why it is better at reflecting the overall tendency of a system to win comparisons against others. On the other hand, for the harder and finer-grained problem of ranking alternative translations of an input segment, the regression techniques are superior. We note that regression task attempts to predict the expected value of the target feature at the segment-level.

The last row in Tab. 2 shows how the performance of HTER improves with the addition of the other effort indicators as features. We further investigated this direction by removing features from the starred system and noticed that the correlation drops to 0.3458 (no time) and to 0.3213 (neither time, nor keystrokes). These observations show another interesting outcome of using machine learning to normalise the post-editing annotation. We can benefit from effort indicators of different nature despite their low inter-annotator agreement. To further investigate this observation we used the human score on post-editing effort (from 1-4) as one more feature in regression. Segment-level time and HTER with SVR improved to 0.2754 and 0.3633, respectively, in the presence of that feature, even though in isolation the feature had shown low agreement (Sect. 3.1).

Table 3 shows the overall ranking of systems (and their rank correlation ρ) according to the WMT (based on human rankings), and the best performing strategy for each target feature (starred systems in Tab. 1). There are some interesting differences between the overall rankings obtained via post-editing and the one obtained via human ranking.

Table 3: Overall ranking using different annotation

Human ranking	Time	HTER	Keystrokes
0.65: Online-B	0.66: Online-B	0.67: Online-B	0.63: Online-B
0.58: RBMT-3	0.54: Online-A	0.56: UEDIN	0.55: Online-A
0.56: Online-A	0.53: UEDIN	0.55: Online-A	0.53: UEDIN
0.55: PROMT	0.51: PROMT	0.53: UPC	0.52: RBMT-1
0.52: UPC	0.50: UPC	0.49: PROMT	0.51: UPC
0.52: UEDIN	0.48: RBMT-3	0.48: RBMT-1	0.49: PROMT
0.46: RBMT-4	0.46: Online-C	0.46: JHU	0.45: RBMT-3
0.45: RBMT-1	0.45: JHU	0.42: RBMT-3	0.45: JHU
0.43: ONLINE-C	0.43: RBMT-1	0.41: RBMT-4	0.43: RBMT-4
0.36: JHU	0.43: RBMT-4	0.41: Online-C	0.42: Online-C

Note how RBMT-3 is consistently ranked worse by all methods based on post-editing. On the other hand, JHU (officially at the bottom of the ranking) and UEDIN (officially in the middle) both gained a few positions. These results indicate that using post-editing to gather annotation for ranking MT systems could be a promising direction. The mismatches between our ranking and the official one should not be seen as flaws, but rather as the result of a different, perhaps more objective way of addressing the problem as compared to the one used currently, where humans explicatively perform the ranking.

4 Conclusions

In this paper we have argued for using post-editing as a more natural and objective way to produce gold-standard annotation for ranking MT systems for quality. We have proposed normalisation techniques to cope with important challenges from using post-editing for this task, such as low inter-annotator agreement. Our technique learns how to compare effort indicators produced by different annotators. By making post-editing effort indicators comparable, we were able to rank translations from different MT systems without having to directly collect a dataset of explicit human rankings. Extrapolating system rankings from a more natural annotation relieves people from the burden of having to do it themselves – which seems to be very cognitively demanding. Furthermore it allows for different notions of effort to be used altogether, alleviating low inter-annotator agreement issues and strengthening our normalisation techniques. Finally, the data required for our approach, i.e. post-edited translations with implicitly collected effort indicators, is a by-product of a process that is becoming increasingly popular, particularly in the translation industry. Therefore, this can also be considered a more cost-effective way of collecting data.

Future directions for this research include using latent variable models to address a task's expected effort. Related work on post-editing as a way to capture cognitive effort suggests that linguistic features of the input text play an important role at classifying edit operations according to their complexity. Adding such features to our models should increase their prediction power, leading to better normalisation.

References

1. Dreyer, M., Marcu, D.: HyTER: Meaning-Equivalent Semantics for Translation Evaluation. In: Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Montréal, Canada, Association for Computational Linguistics (June 2012) 162–171
2. Snover, M., Dorr, B., Schwartz, R., Micciulla, L., Makhoul, J.: A Study of Translation Edit Rate with Targeted Human Annotation. In: Proceedings of the 7th Conference of the Association for MT in the Americas, Cambridge, Massachusetts (2006) 223–231
3. Olive, J., Christianson, C., McCary, J.: Handbook of Natural Language Processing and Machine Translation: DARPA Global Autonomous Language Exploitation. Springer (2011)
4. Callison-Burch, C., Koehn, P., Monz, C., Post, M., Soricut, R., Specia, L.: Findings of the 2012 Workshop on Statistical Machine Translation. In: Proceedings of the 7th WMT, Montréal (2012) 10–51
5. Bojar, O., Ercegovčević, M., Popel, M., Zaidan, O.: A Grain of Salt for the WMT Manual Evaluation. In: Proceedings of the 6th WMT, Edinburgh (2011) 1–11
6. Lopez, A.: Putting human assessments of machine translation systems in order. In: Proceedings of the 7th WMT, Montréal (2012) 1–9
7. Koponen, M., Aziz, W., Ramos, L., Specia, L.: Post-editing time as a measure of cognitive effort. In: Proceedings of the AMTA 2012 Workshop on Post-editing Technology and Practice, San Diego (2012)
8. Plitt, M., Masselot, F.: A Productivity Test of Statistical Machine Translation Post-Editing in a Typical Localisation Context. The Prague Bulletin of Mathematical Linguistics **93** (2010) 7–16
9. Aziz, W., de Sousa, S.C.M., Specia, L.: PET: a tool for post-editing and assessing machine translation. In: Proceedings of the 8th Conference on Language Resources and Evaluation, Istanbul (2012)
10. Cohen, J.: A coefficient of agreement for nominal scales. Educational and Psychological Measurement **20**(1) (April 1960) 37–46