Metrics for Evaluation of Word-Level Machine Translation Quality Estimation

Varvara Logacheva, Michal Lukasik and **Lucia Specia**
Department of Computer Science
University of Sheffield, UK
{v.logacheva, m.lukasik, l.specia}@sheffield.ac.uk

Abstract

The aim of this paper is to investigate suitable evaluation strategies for the task of word-level quality estimation of machine translation. We suggest various metrics to replace F_1-score for the "BAD" class, which is currently used as main metric. We compare the metrics' performance on real system outputs and synthetically generated datasets and suggest a reliable alternative to the F_1-BAD score — the multiplication of F_1-scores for different classes. Other metrics have lower discriminative power and are biased by unfair labellings.

1 Introduction

Quality estimation (QE) of machine translation (MT) is a task of determining the quality of an automatically translated text without any oracle (reference) translation. This task has lately been receiving significant attention: from confidence estimation (i.e. estimation of how confident a particular MT system is on a word or a phrase (Gandrabur and Foster, 2003)) it evolved to system-independent QE and is performed at the word level (Luong et al., 2014), sentence level (Shah et al., 2013) and document level (Scarton et al., 2015).

The emergence of a large variety of approaches to QE led to need for reliable ways to compare them. The evaluation metrics that have been used to compare the performance of systems participating in QE shared tasks[1] have received some criticisms. Graham (2015) shows that Pearson correlation better suits for the evaluation of sentence-level QE systems than mean absolute error (MAE), often used for this purpose. Pearson correlation evaluates how well a system captures the regularities in the data, whereas MAE essentially measures the difference between the true and the predicted scores and in many cases can be minimised by always predicting the average score as given by the training set labels.

Word-level QE is commonly framed as a binary task, i.e., the classification of every translated word as "OK" or "BAD". This task has been evaluated in terms of F_1-score for the "BAD" class, a metric that favours 'pessimistic' systems — i.e. systems that tend to assign the "BAD" label to most words. A trivial baseline strategy that assigns the label "BAD" to all words can thus receive a high score while being completely uninformative (Bojar et al., 2014). However, no analysis of the word-level metrics' performance has been done and no alternative metrics have been proposed that are more reliable than the F_1-BAD score.

In this paper we compare existing evaluation metrics for word-level QE, suggest a number of alternatives, and show that one of these alternatives leads to more objective and reliable results.

2 Metrics

One of the reasons word-level QE is a challenging problem is the fact that "OK" and "BAD" labels are not equally important: we are generally more interested in finding incorrect words than in assigning a suitable category to every single word. An ideal metric should be oriented towards the recall for the "BAD" class. However, the case of F_1-BAD score shows that this is not the only requirement: in order to be useful the metric should not favour pessimistic labellings, i.e., all or most words labelled as "BAD". Below we describe possible alternatives to the F_1-BAD score.

2.1 F_1-score variants

Word-level F_1-scores. Since F_1-BAD score is too pessimistic, an obvious solution would be to

[1] http://statmt.org/wmt15/quality-estimation-task.html

Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, pages 585–590,
Berlin, Germany, August 7-12, 2016.

balance it with F_1-score for the "OK" class. However, the widely used weighted average of F_1-scores for the two classes is not suitable as it will be dominated by F_1-OK due to labels imbalance. Any reasonable MT system will nowadays generate texts where most words are correct, so the label distribution is very skewed towards the "OK" class. Therefore, we suggest instead the **multiplication of F_1-scores** for individual classes: it is equal to zero if one of the components is zero, and since both are in the [0,1] range, the overall result will not exceed the value of any of the multipliers.

Phrase-level F_1-scores. One of the features of MT errors is their phrase-level nature. Errors are not independent: one incorrect word can influence the classification of its neighbours. If several adjacent words are tagged as "BAD", they are likely to be part of an error which spans over a phrase.

Therefore, we also evaluate word-level F_1-scores and alternative metrics which are based on correctly identified erroneous or error-free spans of words. The phrase-level F_1-score we suggest is similar to the one used for the evaluation of named entity recognition (NER) systems (Tjong Kim Sang and De Meulder, 2003). There, precision is the percentage of named entities found by a system that are correct, recall is the percentage of named entities present in the corpus that are found by a system. For the QE task, instead of named entities we have spans of erroneous (or correct) words. Precision is the percentage of correctly identified spans among all the spans found by a system, recall is the percentage of correctly identified spans among the spans in the test data.

However, in NER the correct borders of a named entity are of big importance, because failure to identify them results in an incorrect entity. On the other hand, the actual borders of an error span in QE are not as important: the primary goal is to identify the erroneous region in the sentence, the task of finding the exact borders of an error cannot be solved unambiguously even by human annotators (Wisniewski et al., 2013). In order to take into account partially correct phrases (e.g. a 4-word "BAD" phrase where the first word was tagged as "OK" by a system and the remaining words were correctly tagged as "BAD"), we compute the number of true positives as the sum of percentages of words with correctly predicted tags for every "OK" phrase. The number of true negatives is defined analogously.

2.2 Other metrics

Matthews correlation coefficient. MCC (Powers, 2011) was used as a secondary metric in WMT14 word-level QE shared task (Bojar et al., 2014). It is determined as follows:

$$MCC = \frac{TP \times TN + FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

where TP, TN, FP and FN are true positive, true negative, false positive and false negative values, respectively.

This coefficient results in values in the [-1, 1] range. If the reference and hypothesis labellings agree on the majority of the examples, the final figure is dominated by the $TP \times TN$ quantity, which gets close to the value of the denominator. The more false positives and false negatives the predictor produces, the lower the value of the numerator.

Sequence correlation. The sequence correlation score was used as a secondary evaluation metric in the QE shared task at WMT15 (Bojar et al., 2015). Analogously to the phrase-level F_1-score, it is based on the intersection of spans of correct and incorrect words. It also weights the phrases to give them equal importance and penalises the difference in the number of phrases between the reference and the hypothesis.

3 Metrics comparison

One of the most reliable ways of comparing metrics is to measure their correlation with human judgements. However, for the word-level QE task, asking humans to rate a system labelling or to compare the outputs of two or more QE systems is a very expensive process. A practical way of getting the human judgements is the use of quality labels in downstream human tasks — i.e. tasks where quality labels can be used as additional information and where they can influence human accuracy or speed. One such a downstream task can be computer-assisted translation, where the user translates a sentence having automatic translation as a draft, and word-level quality labels can highlight incorrect parts in a sentence. Improvements in productivity could show the degree of usefulness of the quality labels in this case. However, such an experiment is also very expensive to be performed. Therefore, we consider indirect ways of comparing the metrics' reliability based on prelabelled gold-standard test sets.

3.1 Comparison on real systems

One of the purposes of system comparison is to identify the best-performing system. Therefore, we expect a good metric to be able to distinguish between systems as well as possible. One of the quality criteria for a metric will thus be the number of significantly different groups of systems the metric can identify. Another criterion to evaluate metrics is to compare the real systems' performance with synthetic datasets for which we know the desirable behaviour of the metrics. If a metric gives the expected scores to all artificially generated datasets, it detects some properties of the data which are relevant to us, so we can expect it to work adequately also on real datasets.

Here we compare the performance of six metrics:

- F_1**-BAD** — F_1-score for the "BAD" class.
- F_1**-mult** — multiplication of F_1-scores for "BAD" and "OK" classes.
- **phr** F_1**-BAD** — phrase-level F_1-score for the "BAD" class.
- **phr** F_1**-mult** — multiplication of phrase-level F_1-scores.
- **MCC** — Matthews Correlation Coefficient.
- **SeqCor** — Sequence Correlation.

We used these metrics to rank all systems submitted to the WMT15 QE shared task 2 (word-level QE).[2] In addition to that, we test the performance of the metrics on a number of synthetically created labellings that should be **ranked low** in comparison to real system labellings:

- **all-bad** — all words are tagged as "BAD".
- **all-good** — all words are tagged as "OK".
- **optimistic** — 98% words are tagged as "OK", with only a small number of "BAD" labels generated: this system should have high precision (0.9) and low recall (0.1) for the "BAD" label.
- **pessimistic** — 90% words are tagged as "BAD": this system should have high recall (0.9) for the "BAD" label, but low recall (0.1) for the "OK" label.
- **random** — labels are drawn randomly from the label probability distribution.

We rank the systems according to all the metrics and compute the level of significance for every pair of systems with **randomisation tests** (Yeh, 2000) **with Bonferroni correction** (Abdi, 2007). In order to evaluate the metrics' performance we compute the system distinction coefficient *d* — the probability of two systems being significantly different, which is defined as the ratio between the number of significantly different pairs of systems and all pairs of systems. We also compute *d* for the top half and for the bottom half of the ranked systems list separately in order to check how well each metric can discriminate between better performing and worse performing systems.[3]

The results are shown in Table 1. For every synthetic dataset we show the number of real system outputs that were rated lower than this dataset, with the rightmost column showing the sum of this figure across all the synthetic sets.

We can see that three metrics are better at distinguishing synthetic results from real systems: SeqCor and both multiplied F_1-scores. In the case of SeqCor this result is explained by the fact that it favours longer spans of "OK" and "BAD" labels and thus penalises arbitrary labellings. The multiplications of F_1-scores have two components which penalise different labellings and balance each other. This assumption is confirmed by the fact that F_1-BAD scores become too pessimistic without the "OK" component: they both favour synthetic systems with prevailing "BAD" labels. Phrase-F_1-BAD ranks these systems the highest: **all-bad** and **pessimistic** outperform 16 out of 17 systems according to this metric.

MCC is, in contrast, too 'optimistic': the **optimistic** dataset is rated higher than most of system outputs. In addition to that, it is not good at distinguishing different systems: its system distinction coefficient is the lowest among all metric. SeqCor and phrase-F_1-multiplied, despite identifying artificial datasets, cannot discriminate between real systems: SeqCor fails with the top half systems, phrase-F_1-multiplied is bad at finding differences in the bottom half of the list.

Overall, F_1-multiplied is the only metric that performs well both in the task of distinguishing

[2] Systems that took part in the shared task are listed and described in (Bojar et al., 2015).

[3] d_{bottom} is always greater than d_{top} in our experiments because better performing systems tend to have closer scores under all metrics and more often are not significantly different from one another. When comparing two metrics, greater d does not imply greater d_{top} and d_{bottom}: we use Bonferroni correction for which the significance level depends on the number of compared values, so a difference which is significant when comparing eight systems, for example, can become insignificant when comparing 16 systems.

	d	d_{top}	d_{bottom}	all-bad	all-good	optimistic	pessimistic	random	total
F_1-BAD	0.79	**0.61**	**0.81**	4	-	1	4	1	10
F_1-mult	0.81	0.57	0.75	-	-	2	-	2	4
phr F_1-BAD	**0.86**	**0.61**	0.78	16	-	1	16	-	33
phr F_1-mult	0.75	0.54	0.47	-	-	1	-	-	1
MCC	0.63	**0.61**	0.34	-	-	15	-	-	15
SeqCor	0.77	0.39	0.75	-	-	1	1	2	4

Table 1: Results for all metrics. Numbers in synthetic dataset columns denote the number of system submissions that were rated lower than the corresponding synthetic dataset.

synthetic systems from real ones and in the task of discriminating among real systems, despite the fact that its d scores are not the best. However, F_1-BAD is not far behind: it has high values for d scores and can identify synthetic datasets quite often.

3.2 Comparison on synthetic datasets

The experiment described above has a notable drawback: we evaluated metrics on the outputs of systems which had been tuned to maximise the F_1-BAD score. This means that the system rankings produced by other metrics may be unfairly considered inaccurate.

Therefore, we suggest a more objective metric evaluation procedure which uses only synthetic datasets. We generate datasets with different proportion of errors, compute the metrics' values and their statistical significance and then compare the metrics' discriminative power. This procedure is further referred to as **repeated sampling**, because we sample artificial datasets multiple times.

Our goal is for the synthetic datasets to simulate real systems' output. We achieve this by using the following procedure for synthetic data generation:

- Choose the proportion of errors to introduce in the synthetic data.
- Collect all sequences that contain incorrect labels from the outputs of real systems.
- Randomly choose the sequences from this set until the overall number of errors reaches the chosen threshold.
- Take the rest of segments from the gold-standard labelling (so that they contain no errors).

Thus our artificial datasets contain a specific number of errors, and all of them come from real systems. We can generate datasets with very small differences in quality and identify metrics according to which this difference is more significant.

Let us compare the discriminative power of metrics m_1 and m_2. We choose two error thresholds e_1 and e_2. Then we sample a relatively small number (e.g. 100) of random datasets with e_1 errors. Then — 100 random datasets with e_2 errors. We compute the values for both metrics on the two sets of random samples and for each metric we test if the difference between the results for the two sets is significant (we compute the statistic significance using **non-paired t-test with Bonferroni correction**). Since we sampled the synthetic datasets a small number of times it is likely that the metrics will not detect any significant differences between them. In this case we repeat the process with a larger (e.g. 200) number of samples and compare the p-values for two metrics again. By gradually increasing the number of samples at some point we will find that one of the metrics recognises the differences in scores as statistically significant, while another one does not. This means that this metric has higher discriminative power: it needs less samples to determine that the systems they are different. The procedure is outlined in Algorithm 1.

In our experiments in order to make p-values more stable we repeat each sampling round (sampling of a set with e_i errors 100, 200, etc. times) 1,000 times and use the average of p-values. We used fixed sets of sample numbers: [100, 200, 500, 1000, 2000, 5000, 10,000] and error thresholds: [30%, 30.01%, 30.05%, 30.1%, 30.2%]. The significance level α is 0.05.

Since we compare all six metrics on five error thresholds, we have 10 p-values for each metric at every sampling round. We analyse the results in the following way: for every difference in the percentage of errors (e.g. thresholds of 30% and 30.01% give 0.01% difference, thresholds of 30% and 30.2% — 0.2% difference), we define the minimum number of samplings that a metric

	0.01	0.04	0.05	0.1	0.15	0.2
F_1-mult	10000	**2000**	**2000**	**500**	**200**	**100**
MCC	10000	**2000**	**2000**	**500**	**200**	**100**
F_1-BAD	10000	5000	**2000**	1000	500	200
phr F_1-mult	10000	5000	5000	1000	500	200
SeqCor	10000	5000	5000	1000	500	500
phr F_1-BAD	10000	10000	5000	1000	500	500

Table 2: Repeated sampling: the minimum number of samplings required to discriminate between samples with a different proportions of errors.

Result: $m_x \in \{\mathrm{m}_1, m_2\}$, where m_x — metric with the highest discriminative power on error thresholds e_1 and e_2
$N \leftarrow 100$
$\alpha \leftarrow$ significance level
while $p\text{-}val_{m_1} \geqslant \alpha$ *and* $p\text{-}val_{m_2} \geqslant \alpha$ **do**
  $s_1 \leftarrow$ N random samples with e_1 errors
  $s_2 \leftarrow$ N random samples with e_2 errors
  $p\text{-}val_{m_1} \leftarrow t\text{-}test(m_1(s_1), m_1(s_2))$
  $p\text{-}val_{m_2} \leftarrow t\text{-}test(m_2(s_1), m_2(s_2))$
  if $p\text{-}val_{m_1} < \alpha$ *and* $p\text{-}val_{m_2} \geqslant \alpha$ **then**
    return m_1
  else if $p\text{-}val_{m_1} \geqslant \alpha$ *and* $p\text{-}val_{m_2} < \alpha$ **then**
    return m_2
  else
    $N \leftarrow N + 100$
end

Algorithm 1: Repeated sampling for metrics m_1, m_2 and error thresholds e_1, e_2.

needs to observe significant differences between datasets which differ in this number of errors. Table 2 shows the results. Numbers in cells are minimum numbers of samplings. We do not show error differences greater than 0.2 because all metrics identify them well. All metrics are sorted by discriminative power from best to worst, i.e. metrics at the top of the table require less samplings to tell one synthetic dataset from another.

As in the previous experiment, here the discriminative power of the multiplication of F_1-scores is the highest. Surprisingly, MCC performs equally well. Similarly to the experiment with real systems, the F_1-BAD metric performs worse than the F_1-multiply metric, but here their difference is more salient. All phrase-motivated metrics show worse results.

4 Conclusions

The aim of this paper was to compare evaluation metrics for word and phrase-level quality estimation and find an alternative for F_1-BAD score, which has been used as primary metric in recent research but has a number of drawbacks, in particular tendency to overrate labellings with predominantly"BAD" instances.

We found that the multiplication of F_1-BAD and F_1-OK scores is more stable against "pessimistic" labellings and has bigger discriminative power when comparing synthetic datasets. However, other tested metrics, including advanced phrase-based scores, could not outperform F_1-BAD.

This work should be seen as a proxy for real user evaluation of word-level QE metrics, which could be done on downstream tasks (e.g. computer-assisted translation).

References

Hervé Abdi. 2007. The bonferroni and šidák corrections for multiple comparisons. *Encyclopedia of measurement and statistics*, 3:103–107.

Ondrej Bojar, 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 Ninth Workshop on Statistical Machine Translation*, pages 12–58, Baltimore, Maryland, USA, June. Association for Computational Linguistics.

Ondřej Bojar, Rajen Chatterjee, Christian Federmann, Barry Haddow, Matthias Huck, Chris Hokamp, Philipp Koehn, Varvara Logacheva, Christof Monz, Matteo Negri, Matt Post, Carolina Scarton, Lucia Specia, and Marco Turchi. 2015. Findings of the 2015 workshop on statistical machine translation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 1–46, Lisbon, Portugal.

Simona Gandrabur and George Foster. 2003. Confidence estimation for translation prediction. In *HLT-NAACL-2003*, pages 95–102, Edmonton, Canada.

Yvette Graham. 2015. Improving evaluation of machine translation quality estimation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1804–1813.

Ngoc Quang Luong, Laurent Besacier, and Benjamin Lecouteux. 2014. Lig system for word level qe task at wmt14. In *WMT-2014*, pages 335–341, Baltimore, USA, June.

David M.W. Powers. 2011. Evaluation: from precision, recall and F-measure to ROC, informedness, markedness and correlation. *Journal of Machine Learning Technologies*, 2(1):37–63.

Carolina Scarton, Liling Tan, and Lucia Specia. 2015. Ushef and usaar-ushef participation in the wmt15 qe shared task. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 336–341, Lisbon, Portugal, September.

Kashif Shah, Trevor Cohn, and Lucia Specia. 2013. An investigation on the effectiveness of features for translation quality estimation. In *MT Summit XIV*, pages 167–174, Nice, France.

Erik F. Tjong Kim Sang and Fien De Meulder. 2003. Introduction to the conll-2003 shared task: Language-independent named entity recognition. In *Proceedings of CoNLL-2003*, pages 142–147.

Guillaume Wisniewski, Anil Kumar Singh, Natalia Segal, and François Yvon. 2013. Design and Analysis of a Large Corpus of Post-Edited Translations: Quality Estimation, Failure Analysis and the Variability of Post-Edition. In *MT Summit XIV: 14th Machine Translation Summit*, pages 117–124, Nice, France.

Alexander Yeh. 2000. More accurate tests for the statistical significance of result differences. In *Coling-2000: the 18th Conference on Computational Linguistics*, pages 947–953, Saarbrücken, Germany.