

# Findings of the WMT 2019 Shared Task on Quality Estimation

**Erick Fonseca**

Instituto de Telecomunicações, Portugal  
erick.fonseca@lx.it.pt

**Lisa Yankovskaya**

University of Tartu, Estonia  
lisa.yankovskaya@ut.ee

**André F. T. Martins**

Instituto de Telecomunicações  
& Unbabel, Portugal  
andre.martins@unbabel.com

**Mark Fishel**

University of Tartu, Estonia  
fishel@ut.ee

**Christian Federmann**

Microsoft, USA  
chrife@microsoft.com

## Abstract

We report the results of the WMT19 shared task on Quality Estimation, i.e. the task of predicting the quality of the output of machine translation systems given just the source text and the hypothesis translations. The task includes estimation at three granularity levels: word, sentence and document. A novel addition is evaluating sentence-level QE against human judgments: in other words, designing MT metrics that do not need a reference translation. This year we include three language pairs, produced solely by neural machine translation systems. Participating teams from eight institutions submitted a variety of systems to different task variants and language pairs.

## 1 Introduction

This shared task builds on its previous seven editions to further examine automatic methods for estimating the quality of machine translation (MT) output at run-time, without the use of reference translations. It includes the (sub)tasks of word-level, sentence-level and document-level estimation. In addition to advancing the state of the art at all prediction levels, our more specific goals include to investigate the following:

- The predictability of missing words in the MT output. As in last year, our data include this annotation.
- The predictability of source words that lead to errors in the MT output, also as in last year.
- Quality prediction for documents based on errors annotated at word-level with added severity judgments. This is also like in last year.
- The predictability of individual errors within documents, which may depend on a larger

context. This is a novel task, building upon the existing document-level quality estimation.

- The reliability of quality estimation models as a proxy for metrics that depend on a reference translation.
- The generalization ability of quality estimation models to different MT systems instead of a single ones

We present a simpler setup in comparison to last edition, which featured more language pairs, statistical MT outputs alongside neural ones, and an additional task for phrase-based QE. This simplification reflects a more realistic scenario, in which NMT systems have mostly replaced SMT ones, making phrase-level predictions less interesting.

We used both existing data from the previous edition as well as new data. For word and sentence level, we reused the English-German dataset from last year, but also added a new English-Russian one. For document level, we reused last year's English-French data for training and validation, but introduce a new test set from the same corpus. For QE as a metric we ran the evaluation jointly with the WMT19 metrics task, which meant applying the QE systems to news translation submissions and evaluating them against the human judgments collected this year.

## 2 Tasks

This year we present three tasks: Task 1 for word-level and sentence-level quality estimation, Task 2 for document-level, and Task 3 for quality estimation as a metric. In contrast to previous editions, in which there were data from statistical translation systems, all datasets come from neural machine translation systems.

## 2.1 Task 1

The aim of Task 1 is to estimate the amount of human post-editing work required in a given sentence. It is comprised of word-level and sentence-level subtasks, both of which annotated as in last year.

### 2.1.1 Word Level

At the word level, participants are required to produce a sequence of tags for both the source and the translated sentences. For the source, tokens correctly translated should be tagged as OK, and the ones mistranslated or ignored as BAD. For the translated sentence, there should be tags both for words and *gaps* – we consider gaps between each two words, plus one in the beginning and another in the end of the sentence. Words correctly aligned with the source are tagged as OK, and BAD otherwise. If one or more words are missing in the translation, the gap where they should have been is tagged as BAD, and OK otherwise.

As in previous years, in order to obtain word level labels, first both the machine translated sentence and the source sentence are aligned with the post-edited version. Machine translation and post-edited pairs are aligned using the TERCOM tool (<https://github.com/jhclark/tercom>);<sup>1</sup> source and post-edited use the IBM Model 2 alignments from `fast_align` (Dyer et al., 2013).

**Target word and gap labels** Target tokens originating from insertion or substitution errors were labeled as BAD (i.e., tokens absent in the post-edit sentence), and all other tokens were labeled as OK. Similarly to last year, we interleave these target word labels with *gap* labels: gaps were labeled as BAD in the presence of one or more deletion errors (i.e., a word from the source missing in the translation) and OK otherwise.

**Source word labels** For each token in the post-edited sentence deleted or substituted in the machine translated text, the corresponding aligned source tokens were labeled as BAD. In this way, deletion errors also result in BAD tokens in the source, related to the missing words. All other words were labeled as OK.

<sup>1</sup>For back-compatibility with last year’s datasets, when computing word-level labels, we disabled shifts in TERCOM; but for sentence-level, shifts were allowed.

**Evaluation** As in last year, systems are evaluated primarily by  $F_1$ -Mult, the product of the  $F_1$  scores for OK and BAD tags. There are separate scores for source sentences and translated sentences, with the latter having word and gap tags interleaved. Systems are ranked according to their performance on the source side.

Additionally, we compute the Matthews correlation coefficient (MCC, Matthews 1975), a metric for binary classification problems particularly useful when classes are unbalanced. This is the case in QE, in which OK tags are much more common than BAD tags (see Table 2 for the statistics on this year’s data). It is computed as follows:

$$\begin{aligned} S &= \frac{TP + FN}{N} \\ P &= \frac{TP + FP}{N} \\ MCC &= \frac{\frac{TP}{N} - SP}{\sqrt{SP(1-S)(1-P)}}, \end{aligned} \quad (1)$$

where  $TP$ ,  $TN$ ,  $FP$  and  $FN$  stand for, respectively, true positives, true negatives, false positives and false negatives; and  $N$  is the total number of instances to be classified.

### 2.1.2 Sentence Level

At the sentence level, systems are expected to produce the Human Translation Error Rate (HTER), which is the minimum ratio of edit operations (word insertions, deletions and replacements) needed to fix the translation to the number of its tokens, capped at maximum 1.

In order to obtain the number of necessary operations, we run TERCOM on the machine translated and post-edit sentences, with a slightly different parametrization (see footnote 1).

**Evaluation** Also as in last year, systems are primarily evaluated by the Pearson correlation score with the gold annotations. Mean absolute error (MAE), rooted mean squared error (RMSE) and Spearman correlation are also computed.

## 2.2 Task 2

The goal of Task 2 is to predict document-level quality scores as well as fine-grained annotations, identifying which words and passages are incorrect in the translation.

Each document contains zero or more errors,

Gold annotation	
Coup de	sifflet Fox 40 CMG classique doigt
officiel	Grip
System output	
Coup	de sifflet Fox 40 CMG classique doigt
officiel	Grip

Figure 1: Example of fine-grained document annotation. Spans in the same color belong to the same annotation. Error severity and type are not shown for brevity.

annotated according to the MQM taxonomy<sup>2</sup>, and may span one or more tokens, not necessarily contiguous. Errors have a label specifying their type, such as wrong word order, missing words, agreement, etc. They provide additional information, but do not need to be predicted by the systems. Additionally, there are three severity levels for errors: *minor* (if it is not misleading nor changes meaning), *major* (if it changes meaning), and *critical* (if it changes meaning and carries any kind of implication, possibly offensive).

Figure 1 shows an example of fine-grained error annotations for a sentence, with the ground truth and a possible system prediction. Note that there is an annotation composed by two discontinuous spans: a whitespace and the token *Grip* — in this case, the annotation indicates wrong word order, and *Grip* should have been at the whitespace position.

The document-level scores, called MQM scores, are determined from the error annotations and their severity:

$$\text{MQM} = 1 - \frac{n_{\text{minor}} + 5n_{\text{major}} + 10n_{\text{crit}}}{n}. \quad (2)$$

Notice that the MQM score can be negative depending on the number and severity of errors; we truncate it to 0 in that case. Also notice that, while the MQM score can be obtained deterministically from the fine-grained annotations, participants are allowed to produce answers for both subtasks inconsistent with each other, if they believe their systems to work better estimating a single score for the whole document.

<sup>2</sup>Multidimensional Quality Metrics; see <http://www.qt21.eu/mqm-definition/definition-2015-12-30.html> for details.

Gold	R	System	P
Coup de	0.57	Coup	1
classique	1	CMG classique	0.69
Grip	0	officiel	0
<b>Mean Recall</b>			0.52
<b>Mean Precision</b>			0.56
<b>F<sub>1</sub></b>			0.54

Table 1: Scores for the example system output shown in Figure 1. **R** stands for recall and **P** for precision, and are computed based on character overlap.

**MQM Evaluation** MQM scores are evaluated in the same way as the document-level HTER scores: primarily with Pearson correlation with the gold values, and also with MAE, RMSE and Spearman’s  $\rho$ .

**Fine-grained Evaluation** Fine-grained annotations are evaluated as follows. For each error annotation  $a_i^s$  in the system output, we look for the gold annotation  $a_j^g$  with the highest overlap in number of characters. The precision of  $a_i^s$  is defined by the ratio of the overlap size to the annotation length; or 0 if there was no overlapping gold annotation. Conversely, we compute the recall of each gold annotation  $a_j^g$  considering the best matching annotation  $a_k^s$  in the system output<sup>3</sup>, or 0 if there was no overlapping annotation. The document precision and recall are computed as the average of all annotation precisions in the corresponding system output and recalls in the gold output; and therewith we compute the document F<sub>1</sub>. The final score is the unweighted average of the F<sub>1</sub> for all documents. Table 1 shows the precision and recall for each annotation in the example from Figure 1.

### 2.3 Task 3

Task 3 on applying QE as a metric had several purposes:

- To find out how well QE results correlate with general human judgments of MT quality. This mainly means shifting the application focus of quality estimation from professional translators (whose primary interest is the expected number of post-edits to perform,

<sup>3</sup>Notice that if a gold annotation  $a_j^g$  has the highest overlap with a system annotation  $a_i^s$ , it does not necessarily mean that  $a_i^s$  has the highest overlap with  $a_j^g$ .

as estimated by the HTER score) to MT developers and general users.

- To test the generalization ability of QE approaches in a massive multi-system scenario, instead of learning to estimate the quality of just a single MT system
- To directly compare QE models to MT metrics and see how far one can get without a reference translation, or in other words, how much does one gain from having a reference translation in terms of scoring MT outputs

As part of this task sentence-level QE systems were applied to pairs of source segments and translation hypotheses submitted to the WMT19 news translation shared task. System-level results were also computed via averaging the sentence score over the whole test set.

Submission was handled jointly with the WMT19 metrics task. Two language pairs were highlighted as the focus of this task: English-Russian and English-German; however, the task was not restricted to these, and other news translation task languages were also allowed.

Results of this task were evaluated in the same way as MT metrics, using Kendall rank correlation for sentence-level and Spearman rank correlation for system-level evaluations. The overall motivation was to measure how often QE results agree or disagree with human judgments of MT quality.

### 3 Datasets

#### 3.1 Task 1

Two datasets were used in this task: an English-German, the same as in last year with texts from the IT domain; and a novel English-Russian dataset with interface messages present in Microsoft applications. The same data are used for both word-level and sentence-level evaluations.

Table 2 shows statistics for the data. Both language pairs have nearly the same number of sentences, but EN-DE has substantially longer ones. The ratio of BAD tokens in the word-level annotation is also similar in both datasets, as well as the mean HTER, with a increased standard deviation for EN-RU.

#### 3.2 Task 2

There is only one dataset for this task. It is the same one used in last year’s evaluation, but with a

new unseen test set and some minor changes in the annotations; last year’s test set was made available as an additional development set. The documents are derived from the Amazon Product Reviews English-French dataset, a selection of Sports and Outdoors product titles and descriptions. The most popular products (those with more reviews) were chosen. This data poses interesting challenges for machine translation: titles and descriptions are often short and not always a complete sentence. The data was annotated for translation errors by the Unbabel community of crowd-sourced annotators.

Table 3 shows some statistics of the dataset. We see that the new test set has a mean MQM value higher than last year, but actually closer to the training data. On the other hand, the average number of annotations per document is smaller.

#### 3.3 Task 3

Task 3 did not use a specially prepared dataset, as evaluations were done via the human judgments collected in the manual evaluation phase of the news translation shared task.

Suggested training data included last years’ WMT translation system submissions and their collected human judgments, as well as any other additional resources including HTER-annotated QE data, monolingual and parallel corpora.

### 4 Baselines

These are the baseline systems we used for each subtask.

#### 4.1 Word Level

For word-level quality estimation, we used the NuQE (Martins et al., 2017) implementation provided in OpenKiwi (Kepler et al., 2019), which achieved competitive results on the datasets of previous QE shared tasks. It reads sentence pairs with lexical alignments, and takes as input the embeddings of words in the target sentence concatenated with both their aligned counterparts in the source and neighboring words. It then applies linear layers and an RNN to the embedded vectors, outputting a softmax over OK and BAD tags.

#### 4.2 Sentence Level

The sentence-level baseline is a linear regressor trained on four features computed from word-level tags. At training time, it computes the features from the gold training data; at test time, it uses the

Split	Pair	Sentences	Words	BAD source	BAD target	HTER
Train	EN-DE	13,442	234,725	28,549 (12.16%)	37,040 (7.06%)	0.15 ( $\pm 0.19$ )
	EN-RU	15,089	148,551	15,599 (10.50%)	18,380 (6.15%)	0.13 ( $\pm 0.24$ )
Dev	EN-DE	1,000	17,669	2,113 (11.96%)	2,654 (6.73%)	0.15 ( $\pm 0.19$ )
	EN-RU	1,000	9,710	1,055 (10.87%)	1,209 (6.17%)	0.13 ( $\pm 0.23$ )
Test	EN-DE	1,023	17,649	2,415 (13.68%)	3,136 (8.04%)	0.17 ( $\pm 0.19$ )
	EN-RU	1,023	7,778	1,049 (13.49%)	1,165 (7.46%)	0.17 ( $\pm 0.28$ )

Table 2: Statistics of the datasets used in Task 1. Number of sentences is always the same in source and target; number of words refer to the source. Values shown for HTER are mean and standard deviation in parentheses.

Split	Documents	Sentences	Words	MQM	Annotations
Train	1,000	6,003	158,393	29.47 ( $\pm 24.42$ )	23.17 ( $\pm 29.46$ )
Dev	200	1,301	33,959	19.29 ( $\pm 23.28$ )	28.11 ( $\pm 42.94$ )
Test 2018	268	1,640	46,564	18.11 ( $\pm 23.52$ )	27.74 ( $\pm 35.04$ )
Test 2019	180	949	26,279	26.60 ( $\pm 26.80$ )	19.24 ( $\pm 23.94$ )

Table 3: Statistics of the datasets used in Task 2. The column Annotations shows the average number of annotations per document in the dataset. The values for MQM and Annotations are the mean with standard deviation in parentheses

output produced by the word-level baseline. We found this setup to work better than training the regressor with the automatically generated output. The features used are:

1. Number of BAD tags in the source;
2. number of BAD tags corresponding to words in the translation;
3. number of BAD tags corresponding to gaps in the translation;
4. number of tokens in the translation.

During training, we discarded all sentences with an HTER of 0, and during testing, we always answer 0 when there are no BAD tags in the input. This avoids a bias towards lower scores in the case of a high number of sentences with HTER 0, which is the case in the EN-RU data.<sup>4</sup>

### 4.3 Document Level

For the document-level task, we first cast the problem as word-level QE: tokens and gaps inside an

<sup>4</sup>While in principle sentences with no BAD tags should have an HTER of 0, this is not always the case. When pre-processing the shared task data, word-level tags were determined in a case-sensitive fashion, while sentence-level scores were not. The same issue also happened last year, but unfortunately we only noticed it after releasing the training data for this edition.

error annotation are given BAD tags, and all others are given OK. Then, we train the same word-level estimator as in the baseline for Task 1. At test time, for the fine-grained subtask, we group consecutive BAD tags produced by the word-level baseline in a single error annotation and always give it severity *major* (the most common in the training data). As such, the baseline only produces error annotations with a single error span.

For the MQM score, we consider the ratio of bad tags to the document size:

$$\text{MQM} = 1 - \frac{n_{\text{bad}}}{n} \quad (3)$$

This simple baseline contrasts with last year, which used QuEst++ (Specia et al., 2015), a QE tool based on training an SVR on features extracted from the data. We found that the new baseline performed better than QuEst++ on the development data, and thus adopted it as the official baseline.

### 4.4 QE as a Metric

The QE as a metric task included two baselines, both unsupervised. One relied on pre-trained vector representations; the metric consisted of computing LASER (Artetxe and Schwenk, 2018) sentence embeddings for the source segment and the

hypothesis translation and using their cosine similarity.

The second baseline consisted of using bilinearly trained neural machine translation systems to calculate the score of the hypothesis translation, when presented with the source segment as input. Thus, instead of decoding and looking for the best translation with the MT models, we computed the probability of each subword in the hypothesis translation and based on these computed the overall log-probability of the hypothesis under the respective MT model.

## 5 Participants

In total, there were eight participants for Tasks 1 and 2, but not all participated in all of them. Here we briefly describe their strategies and which subtasks they participated in.

There were four participants for Task 3, each team provided two different systems. The description of their methods are presented together with the metrics task<sup>5</sup>.

### 5.1 MIPT

MIPT only participated in the word-level EN-DE task. They used a BiLSTM, BERT and a baseline hand designed-feature extractor to generate word representations, followed by Conditional Random Fields (CRF) to output token labels. Their BiLSTM did not have any pre-training, unlike BERT, and combined the source and target vectors using a global attention mechanism. Their submitted runs combining the baseline features with the BiLSTM and with BERT.

### 5.2 ETRI

ETRI participated in Task 1 only. They pretrained bilingual BERT (Devlin et al., 2019) models (one for EN-RU and another for EN-DE), and then finetuned them to predict all the outputs for each language pair, using different output weight matrices for each subtask (predicting source tags, target word tags, target gap tags, and the HTER score). Training the same model for both subtasks effectively enhanced the amount of training data.

### 5.3 CMU

CMU participated only in the sentence-level task. Their setup is similar to ETRI's, but they pretrain

<sup>5</sup><http://statmt.org/wmt19/metrics-task.html>

a BiLSTM encoder to predict words in the target conditioned on the source. Then, a regressor is fed the concatenation of each encoded word vector in the target with the embeddings of its neighbours and a mismatch feature indicating the difference between the prediction score of the target word and the highest one in the vocabulary.

### 5.4 Unbabel

Unbabel participated in Tasks 1 and 2 for all language pairs. Their submissions were built upon the OpenKiwi framework: they combined linear, neural, and predictor-estimator systems (Chollampatt and Ng, 2018) with new transfer learning approaches using BERT (Devlin et al., 2019) and XLM (Lample and Conneau, 2019) pre-trained models. They proposed new ensemble techniques for word and sentence-level predictions. For Task 2, they combined a predictor-estimator for word-level predictions with a simple technique for converting word labels into document-level predictions.

### 5.5 UTartu

UTartu participated only in the sentence-level task. They combined BERT (Devlin et al., 2019) and LASER (Artetxe and Schwenk, 2018) embeddings to train a regression neural network model. In the second proposed method, they used not only pre-trained embeddings as input features, but also a log-probability score obtained from a neural MT system.

### 5.6 NJUNLP

NJUNLP participated only in the sentence-level EN-DE task. In order to generate word representation vectors in the QE context, they trained transformer models to predict source words conditioned on the target and target words conditioned on the source. Then, they run a recurrent neural network over these representations and a regressor on their averaged output vectors.

### 5.7 BOUN

BOUN turned in a late submission. For word-level predictions, they used referential machine translation models (RTM), which search the training set for instances close to test set examples, and try to determine labels according to them. For sentence level, they used different regressors trained on features generated by their word-level model.

For document level, they treat the whole document as a single sentence and apply the same setup.

## 5.8 USAAR-DFKI

USAAR-DFKI participated only in the sentence-level EN-DE task, and used a CNN implementation of the predictor-estimator based quality estimation model (Chollampatt and Ng, 2018). To train the predictor, they used WMT 2016 IT domain translation task data, and to train the estimator, the WMT 2019 sentence level QE task data.

## 6 Results

The results for Task 1 are shown in Tables 4, 5, 6 and 7. Systems are ranked according to their  $F_1$  on the target side. The evaluation scripts are available at <https://github.com/deep-spin/qe-evaluation>.

We computed the statistical significance of the results, and considered as winning systems the ones which had significantly better scores than all the rest with  $p < 0.05$ . For the word-level task, we used randomization tests (Yeh, 2000) with Bonferroni correction<sup>6</sup> (Abdi, 2007); for Pearson correlation scores used in the sentence-level and MQM scoring tasks, we used William’s test<sup>7</sup>.

In the word-level task, there is a big gap between Unbabel’s winning submission and ETRI’s, which in turn also had significantly better results than MIPT and BOUN. Unfortunately, we cannot do a direct comparison with last year’s results, since i) we now evaluate a single score for target words and gaps, which were evaluated separately before, and ii) only two systems submitted results for source words last year.

The newly proposed metric, MCC, is very well correlated with the  $F_1$ -Mult. If we ranked systems based on their (target) MCC, the only difference would be in the EN-RU task, in which BOUN would be above the baseline. Since this metric was conceived especially for unbalanced binary classification problems, it seems reasonable to use it as the primary metric for the next editions of this shared task.

In the sentence-level task, Unbabel achieved again the best scores, but with a tighter gap to the

<sup>6</sup>We adapted the implementation from <https://gist.github.com/varvara-1/d66450db8da44b8584c02f4b6c79745c>

<sup>7</sup>We used the implementation from <https://github.com/ygraham/nlp-williams>

other participants. For EN-RU, their second submission is statistically tied to ETRI’s first. Comparing to last year’s results in EN-DE, in which the best system had a Pearson correlation of 0.51 and the median was 0.38, we see a great improvement overall. This is likely due to the more powerful pre-trained models, such as BERT and ELMo, that are common now.

In the document-level task, Unbabel achieved the best scores again. Unbabel was also the only participant in the fine-grained annotation subtask, but surpassed the baseline by a large margin. As for the MQM scoring, last year used a different test set, making results not directly comparable, but the best system achieved a Pearson correlation of 0.53. The test set this year is arguably easier because its mean MQM is closer to the training set (see Table 3).

Results for Task 3 are presented together with the metrics task<sup>8</sup>.

## 7 Conclusions

We presented our findings in this year’s shared task on translation quality estimation. This year, the main novelties were a new task that assesses quality estimation as a metric (Task 3), a new subtask related to document-level quality estimation (Task 2) where the goal is to predict error annotations and their severities, and a new dataset for English-Russian used in Task 1.

Following similar trends in other NLP tasks, a common choice from the participants this year was the usage of contextual and pre-trained embedding models such as BERT and XLM along with transfer learning, which includes the systems that obtained the best results. In the future, we plan to implement some strategies to reduce the gap for participants to enter Task 2, as this year we only had two participants. One possibility is to make available pre-processed data or word-level predictions, so that participants can focus more easily on document-level details.

## Acknowledgments

We would like to thank Ramon Astudillo, Frédéric Blain, Carolina Scarton, and Lucia Specia for answering several questions regarding the organization of previous year’s shared task, as well as the Linguistic Services and Platform teams at Unbabel

<sup>8</sup><http://statmt.org/wmt19/metrics-task.html>

Model	Target		Source	
	F <sub>1</sub>	MCC	F <sub>1</sub>	MCC
† UNBABEL Ensemble	0.4752	0.4585	0.4455	0.4094
UNBABEL Stacked	0.4621	0.4387	0.4284	0.3846
ETRI BERT Multitask A	0.4061	0.3778	0.3946	0.3426
ETRI BERT Multitask B	0.4047	0.3774	0.396	0.3446
MIPT Neural CRF Transformer	0.3285	0.2896	0.2662	0.1811
MIPT Neural CRF RNN	0.3025	0.2601	0.26	0.1748
Baseline	0.2974	0.2541	0.2908	0.2126
BOUN RTM GLMd*	0.1846	0.1793	0.0957	0.0372

Table 4: Word-level results for EN-DE. † indicates the winning system. \* indicates late submissions that were not considered in the official ranking.

Model	Target		Source	
	F <sub>1</sub>	MCC	F <sub>1</sub>	MCC
† UNBABEL Ensemble 2	0.478	0.4577	0.4541	0.4212
† UNBABEL Ensemble	0.4629	0.4412	0.4174	0.3729
† ETRI BERT Multitask A	0.4515	0.4294	0.4202	0.3732
ETRI BERT Multitask B	0.43	0.4082	0.4114	0.3644
Baseline	0.2412	0.2145	0.2647	0.1887
BOUN RTM GLMd*	0.1952	0.2271	0.0871	0.0698

Table 5: Word-level results for EN-RU. † indicates the winning systems. \* indicates late submissions that were not considered in the official ranking.

Model	Pearson	Spearman
† UNBABEL Ensemble	0.5718	0.6221
CMULTIMLT	0.5474	0.5947
NJUNLP BiQE BERT Ensemble	0.5433	0.5694
NJUNLP BiQE	0.5412	0.5665
ETRI	0.526	0.5745
Baseline	0.4001	0.4607
UTARTU Labe	-0.319	-0.3768
UTARTU LABEL	0.2487	0.2531
USAAR-DFKI CNNQE	0.2013	0.2806
BOUN RTM1*	0.4734	0.5307
BOUN RTM2*	0.1799	0.2779

Table 6: Sentence-level results for EN-DE. † indicates the winning system. \* indicates late submissions that were not considered in the official ranking.



Model	Pearson	Spearman
† UNBABEL Ensemble 2	0.5923	0.5388
† UNBABEL Ensemble	0.5889	0.5411
ETRI	0.5327	0.5222
CMULTIMLT	0.4575	0.4039
CMULTIMLT 2	0.4292	0.3628
UTARTU LABEL	0.4014	0.3364
Baseline	0.2601	0.2339
UTARTU LACLAS	0.0424	0.1735
BOUN RTM 1*	0.2817	0.2067
BOUN RTM 2*	0.2314	0.1082

Table 7: Sentence-level results for EN-RU. † indicates the winning system. \* indicates late submissions that were not considered in the official ranking.

Model	F <sub>1</sub>
UNBABEL BERT	0.48
Baseline	0.38

Table 8: Document-level fine grained annotation results for EN-FR

Model	Pearson
UNBABEL LINBERT	0.37
UNBABEL BERT	0.37
Baseline	0.35
BOUN RTM 1*	0.22
BOUN RTM 2*	0.05

Table 9: Document-level MQM results for EN-FR. † indicates the winning system. \* indicates late submissions.

for helping creating and annotating the new test set used for Task 2. We also thank Microsoft for the preparation of the new post-edited dataset for Task 1. This work was partly funded by the European Research Council (ERC StG DeepSPIN 758969), the European Union’s Horizon 2020 research and innovation programme under grant agreement No. 825303, the Estonian Research Council grant no. 1226 and by the Fundação para a Ciência e Tecnologia through contract UID/EEA/50008/2019.

## References

H. Abdi. 2007. Bonferroni and Šidák corrections for multiple comparisons. In N. J. Salkind, editor, *Encyclopedia of Measurement and Statistics*, pages 103–107. Sage, Thousand Oaks, CA.

Mikel Artetxe and Holger Schwenk. 2018. [Massively multilingual sentence embeddings for zero-shot cross-lingual transfer and beyond](#). *CoRR*, abs/1812.10464.

Shamil Chollampatt and Hwee Tou Ng. 2018. Neural quality estimation of grammatical error correction. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, Brussels, Belgium.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proceedings of NAACL*.

Chris Dyer, Victor Chahuneau, and Noah A. Smith. 2013. A simple, fast, and effective reparameterization of IBM model 2. In *Proceedings of NAACL*.

Fábio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, and André F. T. Martins. 2019. OpenKiwi: An open source framework for quality estimation. In *Proceedings of ACL 2019 System Demonstrations*.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual Language Model Pretraining. *arXiv preprint arXiv:1901.07291*.

André F. T. Martins, Marcin Junczys-Dowmunt, Fabio N. Kepler, Ramón Astudillo, Chris Hokamp, and Roman Grundkiewicz. 2017. Pushing the Limits of Translation Quality Estimation. *Transactions of the Association for Computational Linguistics*, 5:205–218.

B.W. Matthews. 1975. [Comparison of the predicted and observed secondary structure of t4 phage lysozyme](#). *Biochimica et Biophysica Acta (BBA) - Protein Structure*, 405(2):442 – 451.

Lucia Specia, Gustavo Paetzold, and Carolina Scarton. 2015. [Multi-level translation quality prediction with quest++](#). In *Proceedings of ACL-IJCNLP*

*2015 System Demonstrations*, pages 115–120, Beijing, China. Association for Computational Linguistics and The Asian Federation of Natural Language Processing.

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.