IST-Unbabel Participation in the WMT20 Quality Estimation Shared Task

João Moura
Instituto Superior Técnico, Lisbon
joaopcmoura@tecnico.ulisboa.pt

Miguel Vera
Unbabel, Lisbon
miguel.vera@unbabel.com

Daan van Stigt
Unbabel, Lisbon
daan.stigt@unbabel.com

Fabio Kepler
Unbabel, Lisbon
kepler@unbabel.com

André F. T. Martins
Instituto de Telecomunicações
Instituto Superior Técnico
Unbabel, Lisbon
andre.t.martins@tecnico.ulisboa.pt

Abstract

We present the joint contribution of IST and Unbabel to the WMT 2020 Shared Task on Quality Estimation. Our team participated on all tracks (Direct Assessment, Post-Editing Effort, Document-Level), encompassing a total of 14 submissions. Our submitted systems were developed by extending the OpenKiwi framework to a transformer-based predictor-estimator architecture, and to cope with glass-box, uncertainty-based features coming from neural machine translation systems.

1 Introduction

Quality estimation (QE) is the task of evaluating a translation system's quality without access to reference translations (Blatz et al., 2004; Specia et al., 2018). This paper describes the joint contribution for Instituto Superior Técnico (IST) and Unbabel to the WMT20 Quality Estimation shared task, where systems were submitted to all three tasks: 1) sentence-level direct assessment; 2) word and sentence-level post-editing effort; and 3) document-level annotation and scoring.

Unbabel's participation in previous editions of the shared task (2016, 2017, 2019) used ensemble of strong individual systems, with varying architectures and hyper-parameters. While this strategy led to very strong results, large system ensembles are not a very practical solution, complicating model deployment and requiring expensive computation and memory usage. This year, in contrast, our focus was on simplicity: only single model systems were submitted and, in a few cases, an additional simple ensemble of the same model. Transfer learning on top of pretrained multilingual models was also used for avoiding manual pretraining for each language pair.

Last year's winning submission (Kepler et al., 2019a) combined strong individual systems built on top of the OpenKiwi framework (Kepler et al., 2019b) and pretrained Transformer models. We consolidated those changes with support for newly released pretrained models and packages and published a new version 2.0 of the OpenKiwi framework.[1] We trained and submitted single model systems in OpenKiwi for all tasks, beating all baselines by a large margin. Additionaly, we also used OpenKiwi with small adaptations to handle specific sources of information in Tasks 1 and 3.

Task 1, in particular, was introduced this year with Direct Assessment scores as targets. Further, it introduced the novelty of providing the trained NMT models that were used for producing the translations. Previously, only black-box QE was considered in the WMT Shared Task, as it is one of the main uses cases. With the availability of the NMT models, new glass-box approaches can be explored. Our best submitted systems drew inspiration from (Fomicheva et al., 2020) to leverage this information, improving in performance and robustness over a black-box approach.

Our main contributions are:

- We release the second version of OpenKiwi along with our submission, with a variety of new features, including the ability to use pretrained Transformer-based Language Models;
- We show that transfer learning techniques still perform well, by fine-tuning *XLM-Roberta* in a Predictor-Estimator architecture;
- We incorporate features extracted from the provided NMT models into our existing architectures and show that glass-box QE improves upon black-box approaches.

[1] The new version will be publicly available at https://github.com/unbabel/openkiwi.

Proceedings of the 5th Conference on Machine Translation (WMT), pages 1029–1036
Online, November 19–20, 2020.

2 Quality Estimation Tasks

This year's shared task edition comprised three tasks: 1) a newly introduced one for sentence-level direct assessment; 2) one for word and sentence-level post-editing effort; and 3) one for document-level. Refer to the Findings paper (Specia et al., 2020) for full descriptions.

Of noteworthy mention is that the NMT models for Tasks 1 and 2 where provided along with the data, which opened up the possibility of using glass-box approaches.

3 Implemented Systems

To avoid the complexity of ensemble of several systems, all our submitted systems consisted of a single model type. In addition to standard OpenKiwi 2.0 systems submitted to Tasks 1 and 2 (§3.1), we implemented two types of extensions on top of OpenKiwi, one for exploring glass-box approaches for Tasks 1 and 2 (§3.2), and one for handling document-level QE for Task 3 (§3.3).

3.1 Base OpenKiwi System

Given the success in doing transfer learning with pretrained Language Models in last year's shared task edition, we published support for them as part of the open source QE framework OpenKiwi in a new 2.0 version. BERT, XLM, and XLM-Roberta are currently supported via the `Transformers`[2] Python package (Wolf et al., 2019), which means different models can be easily used. For this year's shared task, we based all systems on this version of OpenKiwi and used pretrained XLM-Roberta models (Conneau et al., 2020), either `base` or `large` versions. We chose XLM-Roberta (called XLM-R from here on) instead of XLM, used in last year's best individual model, due to its reported state-of-the-art performance on downstream cross-lingual tasks and based on preliminary experiments.

The architecture follows the overall pattern introduced originally in the Predictor-Estimator model (Kim et al., 2017), comprising a "Feature Extractor" module with a "Quality Estimator" module on top. Figure 1 depicts this general architecture.

The Feature Extractor module consists of a pretrained XLM-R model and feature extraction methods on top, such that features for the target sentence, the target tokens, and the source tokens are returned separately. Source and target sentences

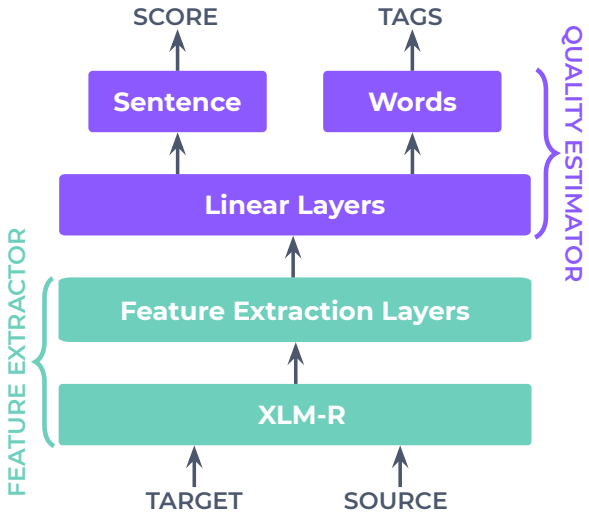

Figure 1: General architecture of the implemented OpenKiwi-based systems.

are passed as inputs in the format `<s> target </s> <s> source </s>`. Output features for tokens in the target sentence are averaged and then concatenated with the classifier token embedding (first `<s>` in the input), and returned as sentence features.[3]

For the Quality Estimator module we used linear layers instead of a bi-LSTM (as used by Kim et al. (2017)), since initial experiments showed similar performance. Additional linear layers were stacked on top for each output type: target words, target gaps, source words, and sentence regression.

For the plain OpenKiwi submissions we used the XLM-R `base` model and a Quality Estimator block with two linear layers. Hyper-parameter search was performed for each language pair and task[4] and submitted as a single model system to Tasks 1 and 2, and used as basis for the submission to Task 3. These systems will be referred to as OPENKIWI-BASE through the rest of the paper.

[2] https://github.com/huggingface/transformers

[3] Even though XLM-R was not trained on the Next Sentence Prediction objective (therefore not using the classification token in its original pretraining), preliminary experiments showed that concatenating inputs, average pooling, and using the classification token resulted in better performance compared to feeding source and target separately and extracting sentence features with other strategies (only pooled target, only the classifier token, classifier token + pooled source, and others).

[4] Hyper-parameters that were searched are: learning rate, dropout, number of warmup steps, and number of freeze steps.

3.2 Glass-Box QE

3.2.1 Glass-Box Features

Recent work on MT confidence estimation (Fomicheva et al., 2020) showed that useful information coming from an MT system, obtained as a by-product of translation, can be competitive with supervised black-box QE models in terms of correlation to human judgements of translation quality, in settings where the labeled data is scarce. The approach described in Fomicheva et al. (2020) requires access to the MT system that produced the translations (unlike the black-box regime). This year's new Task 1, and the fact it shares datasets with Task 2, allowed us to explore this approach on both tasks. In our work, we investigated how to combine the richness of this extra information coming from the provided Neural MT (NMT) system with the strength of state-of-the-art approaches to supervised QE.

To this end, we extract features (referred to as *glass-box features* henceforth) using the output probability distribution obtained from (i) a standard deterministic NMT and (ii) using uncertainty quantification. For (ii) we use Monte Carlo Dropout (Gal and Ghahramani, 2015) as a way of circumventing the miscalibration problem of Deep Neural Networks (Guo et al., 2017) and obtaining measures indicative of the model's uncertainty.

We obtain 7 different features for each sentence of each language-pair, the first 3 via (i) and the last 4 via (ii) (full details are in Fomicheva et al. (2020)):

- `TP` - sentence average of word translation probability
- `Softmax-Ent` - sentence average of softmax output distribution entropy
- `Sent-Std` - sentence standard deviation of word probabilities
- `D-TP` - average `TP` across $N (N = 30)$ stochastic forward-passes
- `D-Var` - variance of `TP` across N stochastic forward-passes
- `D-Combo` - combination of `D-TP` and `D-Var` defined by $1 - \texttt{D-TP}/\texttt{D-Var}$
- `D-Lex-Sim` - lexical similarity - measured by `METEOR` score (Banerjee and Lavie, 2005) - of MT output generated in different stochastic passes.

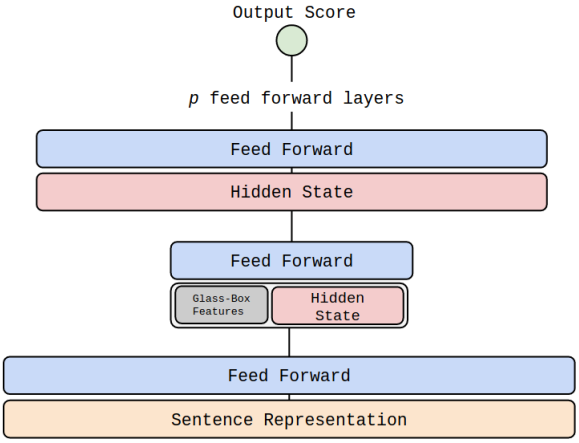

Figure 2: Architecture of the "Quality Estimator" module modified to include *glass-box features*.

Table 1 shows the correlation between each one of these features and human DAs for every language pair in Task 1. As expected, features obtained using uncertainty quantification consistently display higher correlations across all language-pairs, `D-TP` being the most effective for high and medium resource languages, and `D-Lex-Sim` for low resource languages.

3.2.2 Glass-box + Black-box Model

Different configurations were attempted in order to introduce the extracted glass-box features into the OpenKiwi system. The best empirical performance was observed with a simple method: we reduced the dimension of the pooled sentence features output from XLM-R by about five fold (onto `bottleneck_size`), creating a dimensional bottleneck and forcing a more compact sentence representation, and then concatenated the seven extracted glass-box features to this hidden state, followed by an expansion back to a higher dimensional state of `hidden_size`. The result is used as input feature for regression on the sentence score, employing `p` progressively smaller feed-forward layers (halving in size). A visualization of this process can be seen in Figure 2.

The glass-box features were individually normalized a priori, according to their mean and variance in the training dataset, allowing for their integration in the network's training in a scale-independent way.

Systems were trained for all language pairs in Tasks 1 and 2. XLM-R `large` was used instead of `base` version. We ran experiments with and without glass-box features. From here on we will

	Feature	Language Pair						
		En-De	En-Zh	Ro-En	Et-En	Ne-En	Si-En	Ru-En
(i)	TP	0.0993	0.2808	0.5951	0.3992	0.3653	0.3658	0.3658
	Softmax-Ent	0.0858	0.2919	0.5595	0.3546	0.4133	0.4077	0.3790
	Sent-Std	0.0691	**0.3252**	0.5049	0.3985	0.3669	0.3912	0.3510
(ii)	D-TP	**0.1078**	0.3158	**0.6404**	**0.4936**	0.3905	0.3797	**0.4441**
	D-Var	0.0782	0.1943	0.3550	0.2780	0.2336	0.2338	0.2329
	D-Combo	0.0487	0.1259	0.2620	0.1335	0.2938	0.2244	0.2013
	D-Lex-Sim	0.0994	0.2903	0.6210	0.3940	**0.4751**	**0.4318**	0.4092

Table 1: Pearson correlation (r) between the employed *glass-box features* and human DA's for every language pair in Task 1 (validation set) - best results are in bold.

call KIWI-GLASS-BOX the system as described here, which was the one used for the official submissions, but for comparison we will also refer to KIWI-LARGE as the same system but without using the glass-box features.

Hyper-parameter search was performed over `p`, `bottleneck_size`, `hidden_size`, `warmup_steps` (number of warm up steps for optimizer), `freeze_steps` (number of steps for which XLM-R's weights are not updated) and `lr` (learning rate). The exact values can be found in Table 6 in Appendix A.

All submissions of KIWI-GLASS-BOX to Task 1 were created by simple linear ensembles, combining 5 of the models obtained through hyper-parameter search for each language pair. We used the validation set predictions of these 5 models to train a LASSO regression model. However since we do not possess labels for the test set, these ensembles were trained using k-fold cross-validation ($k = 10$) on the validation set.

3.3 Document-level QE

For Task 3 we submitted two systems, both of which are based on the general OpenKiwi architecture described in Section 3.1. The two systems differ only in the type of tags they predict, and the subsequent post-processing that is applied to these tags to obtain annotations and document-level MQM (Multidimensional Quality Metrics) scores. We submitted single systems that predict both tasks of document-level annotation and scoring.

The first system, henceforth referred to as KIWI-DOC, is OPENKIWI-BASE with additional data processing to convert between word- and sentence-level predictions, and document-level predictions. The data approach is the exact same as Kepler et al. (2019a). To obtain training data, annotations are converted to binary word-level tags (`OK` and `BAD` tags) and sentence-level MQM scores are computed from the annotations pertaining to the sentence. After training, document-level annotation predictions are obtained by the following heuristic: contiguous `BAD` tags in the word-level predictions are grouped into a single annotation span and are given the severity label `major`. Predicted document-level MQM scores are obtained by averaging predicted sentence-level MQM weighted by sentence-length (`regression`) or by direct computation from the predicted annotations using the MQM formula (`direct`).

The second system, KIWI-DOC-IOB, is a new contribution in which the task of annotating is approached as Named-entity recognition by using severity tags in IOB (Inside-Outside-Beginning) format.[5] This richer tag scheme addresses two types of information loss that occur in the approach taken for KIWI-DOC: the severity information is kept, and adjacent but disjoint annotations are not collapsed into single annotations during prediction.[6] This approach has the advantage that the predicted tag sequences can be converted to annotations directly by converting the token spans into character spans and using the predicted label as severity. The architecture of KIWI-DOC-IOB is identical to that of KIWI-DOC except that it is trained with a linear chain CRF[7] that enforces correctness of the IOB tag-sequence at prediction time[8].

For both systems we trained a final linear regression model that combines the two types of pre-

[5]The full label set is hence: `B-minor`, `I-minor`, `B-major`, `I-major`, `B-critical`, `I-critical`, and `O`.

[6]The two other types of information loss that were noted by Kepler et al. (2019a) are left unaddressed: tags are still defined at the token-level, and annotations consisting of multiple spans are still split into individual annotations.

[7]Each edge score is a single learned parameter that is independent of the input.

[8]During decoding, the edge scores corresponding to the impossible transitions are set manually to $-\infty$.

dicted MQM scores (`regression` and `direct`) with features derived from the tag-level predictions. We use the following additional features (when available[9]) computed over the document: the fraction of predicted tags corresponding to an error tag;[10] and the mean, variance, minimum, and maximum of the probability of the `BAD`. For simplicity we train the linear regression on the same training data as the systems. For each system, we perform search over all combinations of features, and choose the subset that gives the highest Pearson score on the validation set for that particular system.

4 Experimental Results

4.1 Task 1: Sentence-Level Direct Assessment

The results achieved over the validation set on all language pairs for Task 1 are shown in Table 2. We also include the best correlation achieved by any *glass-box feature* (denoted by BEST GB FEATURE), showing that indeed the proposed method allows for this rich information to complement and enhance the model's training, resulting in a performance increase when compared to model or GB-feature independently.

High resource language pair models (*En-De, En-Zh, Ru-En* benefit the most from the aid of NMT internal information, in particular English-German, where an increase of $\approx 4.5\%$ occurs; this might indicate the usefulness of incorporating nuanced information when sentence scores have less variability.

Scored test set predictions submitted during the development of this approach served as informative feedback, revealing the drop from validation to test performance to be smaller on KIWI-GLASS-BOX models when compared to KIWI-LARGE models, suggesting better generalization capabilities.

4.2 Task 2: Word and Sentence-Level Post-editing Effort

We trained OPENKIWI-BASE and KIWI-GLASS-BOX on all three subtasks at the same time: source tags, target tags, and sentence HTER. The best model was selected by the highest sum of the three metrics on the validation set. We used a single run of each of the two models to simultaneously predict the three outputs. The results can be seen in Table 3. Using the glass-box features provided a significant boost to the Pearson score, showing our strategy for sentence-level DA estimation performed well also when estimating sentence-level HTER.

Even though we only have a single model for all subtasks, our models outperformed the baselines by a large margin and performed very competitively in the test leaderboard (to cite Findings paper).

Pair	System	Pearson VAL	Pearson TEST
En-De	(*)**KIWI-GLASS-BOX-ENSEMBLE**	0.5715	0.5230
	KIWI-GLASS-BOX	0.5263	-
	KIWI-LARGE	0.4794	-
	OPENKIWI-BASE	0.3499	0.2670
	BEST GB FEATURE	0.1078	-
	Openkiwi 1.0	-	0.1455
En-Zh	(*)**KIWI-GLASS-BOX-ENSEMBLE**	0.5711	0.4940
	KIWI-GLASS-BOX	0.5461	-
	KIWI-LARGE	0.5258	-
	OPENKIWI-BASE	0.4199	0.3460
	BEST GB FEATURE	0.3252	-
	OpenKiwi 1.0	-	0.1902
Ro-En	(*)**KIWI-GLASS-BOX-ENSEMBLE**	0.8968	0.8910
	KIWI-GLASS-BOX	0.8841	-
	KIWI-LARGE	0.8790	-
	OPENKIWI-BASE	0.6672	0.7080
	BEST GB FEATURE	0.6404	-
	OpenKiwi 1.0	-	0.6845
Et-En	(*)**KIWI-GLASS-BOX-ENSEMBLE**	0.7697	0.7700
	KIWI-GLASS-BOX	0.7611	-
	KIWI-LARGE	0.7496	-
	OPENKIWI-BASE	0.6728	0.6900
	BEST GB FEATURE	0.4936	-
	OpenKiwi 1.0	-	0.4770
Ne-En	(*)**KIWI-GLASS-BOX-ENSEMBLE**	0.7994	0.7920
	KIWI-GLASS-BOX	0.7804	-
	KIWI-LARGE	0.7711	-
	OPENKIWI-BASE	0.6987	0.6040
	BEST GB FEATURE	0.4751	-
	OpenKiwi 1.0	-	0.3860
Si-En	(*)**KIWI-GLASS-BOX-ENSEMBLE**	0.6896	0.6390
	KIWI-GLASS-BOX	0.6604	-
	KIWI-LARGE	0.6521	-
	OPENKIWI-BASE	0.5727	0.5650
	BEST GB FEATURE	0.4318	-
	OpenKiwi 1.0	-	0.3737
Ru-En	(*)**KIWI-GLASS-BOX-ENSEMBLE**	0.7391	0.7670
	KIWI-GLASS-BOX	0.7137	-
	KIWI-LARGE	0.6938	-
	OPENKIWI-BASE	-	-
	BEST GB FEATURE	0.4441	-
	OpenKiwi 1.0	-	0.5479

Table 2: Task 1 results on the validation and test sets for all language pairs in terms of Pearson's r correlation. Systems in **bold** were officially submitted. (*) Lines with an asterisk use LASSO regression to tune ensemble weights on the validation set, therefore their numbers cannot be directly compared to the other models.

4.3 Task 3: Document-Level QE

The results for the document-level scoring are shown in Table 4. For both systems we observe

[9] Because of the non-binary tags and CRF model the probability based features are not used for the KIWI-DOC-IOB model (posterior marginals could be used for this).

[10] This correspond to the `BAD` tag for KIWI-DOC and all tags different from `O` for KIWI-DOC-IOB.

Pair	System	Target MCC		Source MCC		Pearson	
		Val	Test	Val	Test	Val	Test
En-De	KIWI-GLASS-BOX	**0.460**	**0.465**	**0.357**	**0.349**	**0.618**	**0.633**
	OPENKIWI-BASE	0.445	0.432	0.330	0.324	0.561	0.531
	(*)OpenKiwi 1.0	-	0.358	-	0.266	-	0.392
En-Zh	KIWI-GLASS-BOX	0.567	0.567	**0.348**	0.287	**0.691**	**0.651**
	OPENKIWI-BASE	**0.576**	**0.575**	0.298	0.287	0.615	0.593
	(*)OpenKiwi 1.0	-	0.509	-	0.270	-	0.506

Table 3: Task 2 word and sentence-level results on the validation and test sets. Results for OPENKIWI-BASE and KIWI-GLASS-BOX were obtained from a single model trained by multi-tasking on the 3 different subtasks. (*) Baseline results on the validation set were not made available by the organizers.

System	Validation	Test
KIWI-DOC-`regression`	0.5146	0.4127
KIWI-DOC-`direct`	0.3131	0.3156
KIWI-DOC-`linear`	0.5635	0.4014
KIWI-DOC-IOB-`regression`	0.5731	**0.4746**
KIWI-DOC-IOB-`direct`	0.5483	0.3363
KIWI-DOC-IOB-`linear`	**0.6023**	0.4493

Table 4: Results of document-level (task 3) submissions for MQM scoring (Pearson). The results of KIWI-DOC and KIWI-DOC-IOB are for the same single model. For model selection during training we used the summed validation set Pearson of `direct` and `regression` to obtain a model that performs well in both methods.

System	Validation	Test
KIWI-DOC	**0.4934**	**0.4716**
KIWI-DOC-IOB	0.4016	0.4147

Table 5: Results of document-level (task 3) submissions for annotation (F1). For model selection during training we used validation set MCC for KIWI-DOC and validation set tagging F1 for KIWI-DOC-IOB.

a large drop in Pearson score from validation set to test set, in the range of 0.1-0.2,[11] which suggests that there is a difference in data distribution between the two sets. On the validation set, KIWI-DOC and KIWI-DOC-IOB obtain comparable Pearson correlation, albeit for different MQM methods. While both models perform comparably in the sentence score prediction (`regression`), the KIWI-DOC-IOB system clearly outperforms KIWI-DOC on the MQM scores that are computed directly from the predicted annotations (`direct`). The improvements made by linear regression on the validation set do not consistently translate to the test set. This suggests that our method of search over features for the linear regression is overly optimizing the performance to the validation data. It may also reflect our choice to train the linear model on system predictions on training data.

Table 5 shows the results for the annotation task. The best results are obtained by KIWI-DOC. Surprisingly, the strong scoring results of KIWI-DOC-IOB with `direct` (derived from predicted annotations) do not translate to good results on the annotation F1. The difference between the models is caused by the different trade-off between precision and recall: KIWI-DOC-IOB produces less annotations that are more precise, but KIWI-DOC catches much more errors.[12] The most likely cause for this is the more complex tag-set and constrained decoding of KIWI-DOC-IOB.

5 Conclusions

Our approach to this year's edition of the QE shared task was simplicity. Our submissions consisted of either single models, or simple ensembles of multiple runs of the same model. Moreover, we used multi-task models in Task 2, where a system was trained on all three possible outputs (target and source word level and sentence level). We implemented a new version of OpenKiwi and used it as our baseline. It significantly outperformed the official shared task baseline across the board, which was based on the previous version of OpenKiwi. Finally, we showed that having access to NMT models enables using glass-box approaches to QE, which in turn improves performance when used in

[11] The only exception is KIWI-DOC-IOB-`direct`, which performanced equally poorly on both.

[12] On the validation set KIWI-DOC-IOB predicted 2555 annotations, whereas KIWI-DOC predicted 4028 (the gold set has 5626 annotations). Extending the output message of the annotation evaluation script allowed us to further validate this hypothesis on the validation set: for KIWI-DOC-IOB precision/recall is 0.6287/0.3322; for KIWI-DOC precision/recall is 0.4549/0.6092.

combination with a black-box QE system.

Acknowledgments

This work was supported by the P2020 programs MAIA (contract 045909) and Unbabel4EU (contract 042671), by the European Research Council (ERC StG DeepSPIN 758969), and by the Fundação para a Ciência e Tecnologia through contract UID/50008/2019.

References

Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. 2019. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence Estimation for Machine Translation. In *Proc. of the International Conference on Computational Linguistics*, page 315.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020. Unsupervised cross-lingual representation learning at scale. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8440–8451, Online. Association for Computational Linguistics.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computation Linguistics*.

Yarin Gal and Zoubin Ghahramani. 2015. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. *Proceedings of The 33rd International Conference on Machine Learning*.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. *ArXiv*, abs/1706.04599.

Fabio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, António Góis, M. Amin Farajian, António V. Lopes, and André F. T. Martins. 2019a. Unbabel's participation in the WMT19 translation quality estimation shared task. In *Proceedings of the Fourth Conference on Machine Translation (Volume 3: Shared Task Papers, Day 2)*, pages 78–84, Florence, Italy. Association for Computational Linguistics.

Fabio Kepler, Jonay Trénous, Marcos Treviso, Miguel Vera, and André F. T. Martins. 2019b. OpenKiwi: An open source framework for quality estimation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 117–122, Florence, Italy. Association for Computational Linguistics.

Hyun Kim, Jong-Hyeok Lee, and Seung-Hoon Na. 2017. Predictor-Estimator using Multilevel Task Learning with Stack Propagation for Neural Quality Estimation. In *Conference on Machine Translation (WMT)*.

Lucia Specia, Frederic Blain, Marina Fomicheva, Erick Fonseca, Vishrav Chaudhary, Francisco Guzman, and Andre FT Martins. 2020. Findings of the wmt 2020 shared task on quality estimation. In *Proceedings of the Fifth Conference on Machine Translation: Shared Task Papers*.

Lucia Specia, Carolina Scarton, and Gustavo Henrique Paetzold. 2018. Quality Estimation for Machine Translation. *Synthesis Lectures on Human Language Technologies*, 11(1):1–162.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

A Hyper-parameters

Table 6 shows the hyperparameters used in Task 1.

Language Pair	Hyper-parameters				
	hidden_size	bottleneck_size	lr	warmup_steps	freeze_steps
EN-DE	900	200	1.00E-05	6535	750
EN-ZH	700	300	7.00E-06	3280	4375
RO-EN	900	200	9.00E-06	2625	5687
ET-EN	500	200	7.00E-06	655	3935
NE-EN	900	200	1.20E-05	2625	3060
SI-EN	900	200	7.00E-06	5250	5250
RU-EN	700	200	1.70E-05	3800	6125

Table 6: Hyper-parameters of the best models trained for each language pair in Task 1. 70 trials were performed for each search, using the OPTUNA framework (Akiba et al., 2019), and hyper-parameter values were sampled with the TPE (Tree-structured Parzen Estimator) algorithm. The criterion for trial selection was r Pearson correlation to validation set DA's.