Papago's Submission to the WMT22 Quality Estimation Shared Task

Seunghyun S. Lim[*]
Papago, Naver Corp.
shaun.lim@navercorp.com

Jeonghyeok Park[*]
Papago, Naver Corp.
jeonghyeok.park@navercorp.com

Abstract

This paper describes anonymous submission to the WMT 2022 Quality Estimation shared task. We participate in `Task 1: Quality Prediction` for both sentence and word-level quality prediction tasks. Our system is a multilingual and multi-task model, whereby a single system can infer both sentence and word-level quality on multiple language pairs. Our system's architecture consists of Pretrained Language Model (PLM) and task layers, and is jointly optimized for both sentence and word-level quality prediction tasks using multilingual dataset. We propose novel auxiliary tasks for training and explore diverse sources of additional data to demonstrate further improvements on performance. Through ablation study, we examine the effectiveness of proposed components and find optimal configurations to train our submission systems under each language pair and task settings. Finally, submission systems are trained and inferenced using K-folds ensemble. Our systems greatly outperform task organizer's baseline and achieve comparable performance against other participants' submissions in both sentence and word-level quality prediction tasks.

1 Introduction

Quality Estimation (QE) evaluates the quality of machine translated output without human reference translation (Blatz et al., 2004). Apart from QE models' most obvious usage as being reference-less metrics for MT, it has variety of other applications in Machine Translation (MT) pipeline including but not limited to: parallel corpus filtering (Schwenk et al., 2021), curriculum learning (Ramnath et al., 2021) and decoding (Fernandes et al., 2022).

High performance in both sentence and word-level quality prediction tasks is achieved by incorporating PLM as part of QE model architecture as demonstrated in previous WMT QE findings (Specia et al., 2020, 2021). Previous years' top performers generally incorporate various data augmentation techniques in order to account for limited amount of annotated gold data (Lim et al., 2021; Chen et al., 2021). Multi-task training, ensembling, or incorporating features extracted from external models are few other popular approaches that proved to work well (Lim et al., 2021; Chen et al., 2021; Zerva et al., 2021; Wang et al., 2021a).

Our system is a multilingual and multi-task model, whereby a single system can infer both sentence and word-level quality on multiple language pairs. Our system's architecture (§3.1) consists of PLM and task layers, and is jointly optimized for both sentence and word-level quality prediction tasks (§3.2.1) using multilingual dataset. We propose novel auxiliary tasks (§3.2.2) for training and explore diverse sources of additional data (§3.3) to demonstrate further improvements on performance. Through ablation study (§5), we evaluate each components of our proposed model and use optimal configurations to train our submission systems under each language pair and task settings. Finally, submission systems are trained and inferenced using K-folds ensemble (§3.4.2). Our systems greatly outperform task organizer's baseline and perform very competitively against other participants' submissions in both sentence and word-level quality prediction tasks.

2 Quality Prediction Task and Dataset

In this section we briefly overview two subtasks and their datasets in `Task 1`. Apart from provided `Gold` data as described below, participants are allowed to leverage additional sources of data.

2.1 Sentence Level Quality Prediction

The goal of sentence-level quality prediction is to predict the quality score for each *(source, hypothesis)* sentence pair. Participants are provided with two types of sentence-level quality prediction

[*] These authors contributed equally to this work

Proceedings of the Seventh Conference on Machine Translation (WMT), pages 627–633
Abu Dhabi, December 7–8, 2022. ©2022 Association for Computational Linguistics

data depending on how annotations are created: `Multi-dimensional Quality Metrics (MQM)`[1] and `Direct Assessments (DA)`[2]. All three language pairs in `MQM`, and En-Mr in `DA` are supervised, while remaining four language pairs in `DA` are unsupervised. Submission systems are evaluated on aforementioned eight language pairs and one `surprise language pair`[3]. Note that `MQM` scores are inverted in order to align `MQM` scores with `DA` scores.

2.2 Word Level Quality Prediction

The goal of word-level quality prediction is to predict translations errors, assigning OK/BAD tags to each word in hypothesis, given *(source, hypothesis)* sentence pairs. Word-level tags are provided for language pairs same as in sentence-level task, and tags are derived from either MQM annotations (`MQM`) or post-edited sentences (`DA`).

3 Approach

Below we describe relevant components of our proposed QE model.

3.1 Model Architecture

Our system employs the Predictor-Estimator architecture (Kim et al., 2017). For our **predictor** we use a PLM, and our choice of PLM is XLM-RoBERTa-large (Conneau et al., 2020) due to its impressive performance on crosslingual downstream tasks. Given source sentence src^X in language X and target sentence tgt^Y in language Y, the concatenation of src^X and tgt^Y are fed as input to the PLM and feature vectors relevant to each task are then passed as inputs to the **estimator**. We utilize four independent 2-layer feed-forward networks as estimators, which are 1024 and 200 dimensions, and are stacked in parallel above PLM. The Predictor-Estimator architecture can be described as:

$$\begin{aligned}
&f(src^x, tgt^y)\\
&= H_{sent}, H_{word}, H_{sentaux}, H_{wordaux}\\
&V_{sent} = \phi_{sent}(H_{sent})\\
&V_{word} = \phi_{word}(H_{word})\\
&V_{sentaux} = \phi_{sentaux}(H_{sentaux})\\
&V_{wordaux} = \phi_{wordaux}(H_{wordaux}),
\end{aligned} \tag{1}$$

where f, H, ϕ, and V are predictor, feature extracted from predictor, estimator, and our final prediction, respectively. We describe H, ϕ, and their corresponding training objectives in §3.2.

3.2 Training Objective

The full training objective of our QE model is shown below in equation (2),

$$\begin{aligned}
\mathcal{L} = (&w_{sent} \cdot \mathcal{L}_{sent}\\
&+ (1 - w_{sent}) \cdot \mathcal{L}_{sentaux})+\\
(&w_{word} \cdot \mathcal{L}_{word}\\
&+ (1 - w_{word}) \cdot \mathcal{L}_{wordaux}).
\end{aligned} \tag{2}$$

$\mathcal{L}$ and w denote loss functions and loss weight values. w_{sent} and w_{word} are 0.6 and 0.7 respectively. We describe each loss function components in the following subsections.

3.2.1 Multi-task Training

To build a system that is capable of predicting both sentence and word-level quality, our proposed training objective optimizes for $\mathcal{L}_{sent}$ and $\mathcal{L}_{word}$ jointly as shown in equation (2). We use mean squared error (MSE) and weighted cross entropy loss[4] as loss functions for $\mathcal{L}_{sent}$ and $\mathcal{L}_{word}$ respectively. Therefore, ϕ_{sent} is a classification layer with input H_{sent}, which is PLM's last layer [CLS] representation; ϕ_{word} is a classification layer with input H_{word}, which is created by mean pooling PLM's last layer token hidden states[5]. Since sentence and word-level quality prediction tasks are two closely related tasks, we assume some level of transferability of task knowledge between the two when jointly trained.

3.2.2 Auxiliary-task Training

Auxiliary-tasks are additional objectives that are jointly optimized with losses described in §3.2.1.

[1] English-Russian (En-Ru), English-German En-De), and Chinese-English (Zh-En)

[2] English-Marathi (En-Mr), English-Czech (En-Cs), English-Japanese (En-Ja), Khmer-English (Km-En), and Pashto-English (Ps-En)

[3] English-Yoruba (En-Yo), where no train and development data is provided at all

[4] In order to reduce the problem of label imbalance between `OK` and `BAD`, we use weighted cross entropy with ratio of `OK`:`BAD` = 1:3

[5] If $word_n$ spans $token_{i:j}$, then $H_{word_n} = mean(H_{token_i}, ..., H_{token_j})$

Task	Train Data	Train Method	Train Objective	En-De	En-Ru	Zh-En	En-Mr	Km-En	Ps-En	En-Ja	En-Cs	Multi
Sent	Gold	Vanilla	Single	0.490	0.483	**0.283**	0.551	0.621	0.606	0.315	0.539	0.556
			Multi (3.2.1)	0.493	**0.529**	0.261	0.552	**0.644**	0.614	0.301	0.548	0.562
			Multi+Aux (3.2.1, 3.2.2)	**0.499**	0.516	0.252	**0.555**	0.633	**0.617**	**0.319**	**0.570**	**0.573**
	Gold	Vanilla	Multi + Aux	0.499	0.516	0.252	0.555	0.633	0.617	0.319	0.570	0.573
	Augmented (§3.3)	Vanilla		0.550	0.575	0.274	0.563	0.618	0.611	0.350	0.582	0.609
	Augmented	K-folds ensemble (§3.4.2)		**0.576**	**0.584**	**0.287**	**0.682**	**0.639**	**0.627**	**0.385**	**0.594**	**0.611**
Word	Gold	Vanilla	Single	**0.238**	0.381	**0.208**	0.332	0.404	**0.375**	**0.186**	**0.383**	0.487
			Multi	0.220	0.378	0.201	0.339	0.439	0.367	0.174	0.367	**0.494**
			Multi+Aux	0.229	**0.386**	0.198	**0.358**	**0.455**	0.343	0.169	0.366	0.476
	Gold	Vanilla	Multi + Aux	0.229	**0.386**	0.198	0.358	0.455	0.343	0.169	0.366	0.476
	Augmented	Vanilla		0.285	0.363	0.397	0.443	0.412	0.351	0.153	0.332	0.507
	Augmented	K-folds ensemble (§3.4.2)		**0.301**	0.380	**0.413**	**0.459**	**0.488**	**0.378**	**0.234**	**0.421**	**0.531**

Table 1: Ablation on Train Data, Train Method, and Train Objective. Multi column contains development portion of Gold for all 14 language pairs.

Our intuition is that quality prediction is inherently a complex task even for humans such that human-labels may contain noise. Hence, we appropriately craft original gold labels into secondary labels and use those labels during training as additional learning signals. We expect that training with auxiliary labels can make training more robust and produce a model that is more generalizable.

Sentence-level auxiliary task is a classification task and labels are made as follows: given the n_{th} train set sample's z-standardized score $score_n$, we scale $score_n$ by applying min-max normalization and assign bin (class) labels to each sample. For our experiments, the number of bins is set to 10. Note that min-max scaling is applied to each language pair dataset in order to account for different scales of $score_n$ per dataset. $\phi_{sentaux}$ is a regression layer with input $H_{sentaux}$, which is PLM's last layer [CLS] representation. Likewise, **word-level auxiliary task** is also a classification task and labels are made as follows: given a sample's word-level tags, a sample is assigned to BAD if there exists at least one BAD tags in word-level tags, else OK. $\phi_{wordaux}$ is a classification layer with input $H_{wordaux}$, which is created by mean pooling PLM's last layer token hidden states, excluding special tokens.

3.3 Data Augmentation

We augment training data with additional data, which can be categorized as follows: task-related or pseudo-generated. **Task-related** data are open source data of other downstream tasks, but are similar or can be useful to quality prediction task. We collect data from previous years' WMT Metrics Shared Task[6] and WMT APE Task[7]. Since WMT Metrics Shared Task data contain human DA scores for *(source, hypothesis)* pairs, and WMT APE Task data contain *(source, hypothesis, post-edited)* triplets such that word-level quality annotations can be built using provided word label tagging conventions[8], sentence or word quality labels for this dataset type can be considered high quality.

Pseudo-generated data first assumes bitext[9], *(source, reference)* pairs. We then use NMT models provided by organizers to create *(source, reference, hypothesis)* triplets. Sentence quality labels are generated using COMET[10], which is an open source reference-less QE model. Word quality labels are generated adhering to word label tagging conventions. Labels for pseudo-generated data are considered less accurate compared to task-related data since either labels are pseudo-generated via external model instead being human generated (sentence-level) or do not use actual *(hypothesis, post-edited)* pairs to compute labels (word-level). Refer to Appendix A for detailed list of augmented data.

3.4 Final Model Training

3.4.1 Optimized Configuration

Although we can submit a single model for all language pairs because all our models are multilingual, we submit optimized models for each language pair and task submissions. This is done by choosing and training with optimal configuration for each language pair and task as found in our ablation study (§5, Table 1) or summarized in Appendix B. Our final submissions are optimized for three configurations: train data, train objective, and train method. We further explain each configurations in detail below.

[6] WMT17-21 Metrics Shared Task

[7] WMT16-21 APE Shared Task

[8] https://github.com/deep-spin/qe-corpus-builder#1

[9] Sources of bitext are Europarl, OPUS, Tatoeba and WMT News Translation Task

[10] wmt21-comet-qe-da, https://github.com/Unbabel/COMET

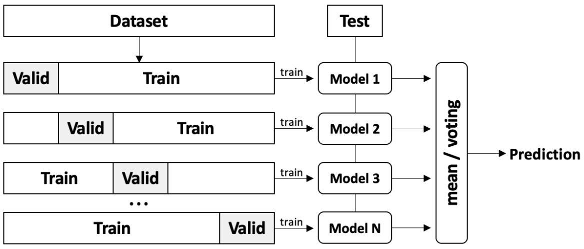

Figure 1: K-folds Ensemble

We have two sources of **train data**: `Gold` which is data provided by task organizers, and `Additional` as described in §3.3. This leads us to experiment on two different compositions of train data: `Gold` and `Augmented`, where the latter is the aggregation of `Gold` and `Additional`.

There are three variants of **train objective**: `Single`, `Multi`, and `Multi+Aux`. `Single` refers to models that are trained with a single-task objective, either being $\mathcal{L}_{sent}$ or $\mathcal{L}_{word}$. `Multi` are multi-task models that are trained jointly on both sentence and word-level quality prediction objectives $\mathcal{L}_{sent} + L_{word}$, as described in §3.2.1. `Multi+Aux` refers to models that are trained with multi-task and auxiliary objectives, as described in equation (2).

For models trained with `Vanilla` **train method**, we can train with variants of train data but always select best checkpoint using the development set portion of `Gold`. We explore advanced training methods such as `K-folds ensemble` (§3.4.2) to further improve model performance.

3.4.2 K-folds Ensemble

As demonstrated in Figure 1, K-folds ensemble (Domingo et al., 2022) distributes the dataset in different training and validation folds such that each individual model uses discrete dataset for training and validation. Compared to vanilla ensembles, where all models are trained using same train data, we expect this method to generate more robust final predictions and become less over-fitted to validation data.

Given a complete set of data[11], we randomly select 1,000 samples for each supervised language pair `Gold` dataset to create validation set, while the rest are used for training. We repeat this process N=5 times with the constraint that N=5 mutually exclusive validation sets are created. We then train and select best checkpoints for each partition using discrete datasets. An ensemble of N=5 best models, one from each partition, are taken to make final predictions. Prediction mean and majority voting is used for sentence-level and word-level quality prediction respectively.

4 Settings

For all training phases and experiments, we train our model in data parallel on multiple NVIDIA Tesla V100 GPUs for maximum 10 epochs with batch size of 16 and is optimized with Adam (Kingma and Ba, 2015) with a learning rate of $7e^{-6}$. Our implementation is based on PyTorch[12] framework.

All models trained within the scope of this paper are multilingual QE models. We concatenate dataset of all individual language pairs to create a multilingual train dataset for both sentence and word-level quality prediction tasks. We apply the same for development set and always perform model selection using a multilingual dataset[13].

5 Ablation

In this section, we present ablation study of individual components to our model described in §3. All evaluations for ablation in Table 2 are conducted on `development` portion of `Gold`.

5.1 Does multilingual training help?

Task	Language	En-De	En-Ru	En-Mr	Ne-En	Si-En
Sent	Single	0.491	0.399	**0.560**	0.798	0.538
	Multi	**0.499**	**0.516**	0.555	**0.805**	**0.550**
Word	Single	0.225	0.306	**0.360**	0.438	0.408
	Multi	**0.229**	**0.386**	0.358	**0.469**	**0.452**

Table 2: Ablation on multilingual training

Training multiple language pairs in a single model through parameter sharing can significantly reduce the cost of model training and maintenance compared with training multiple separate models (Wang et al., 2021b) in Neural Machine Translation. Moreover, we argue that multilingual QE models can collectively learn knowledge from multiple language pairs, which can be particularly be useful in this shared task scenario considering limited training data available per language pair. Table 2 compares the performance of single language pair

[11] We concatenate `train` and `development` portion in the case of `Gold`

[12] https://pytorch.org/

[13] Checkpoints for our final submissions (Table 3, 4) are selected base on performance on multilingual dataset

QE models to multilingual QE models. We see that the performance of `multi` models are higher than or similar to the performance of `single` models. Since the performance of multilingual models are at least on par with separate models, this motivates us to use multilingual training when considering additional training and parameter costs of maintaining multiple separate models.

5.2 Does multi-task or auxiliary-task training help?

Row 1 to 3 and 7 to 9 in Table 1 demonstrates ablation on different training objectives. For sentence-level quality prediction tasks, we see that the performance of `Multi` or `Multi+Aux` is higher than that of `single` in all cases except for Zh-En. We observe that adding auxiliary tasks to multi-tasking can give further improvements in most cases. In general, we argue that multi-tasking or adding auxiliary tasks help improve performance of sentence-level QE models. The results for word-level quality prediction tasks are a bit mixed; `single` achieves highest performance compared to adding any additional training tasks at all for En-De, Zh-En, Ps-En, En-Ja, and En-Cs. We conjecture that multi-tasking or adding auxiliary tasks do not help in word-level as much as they do in sentence-level quality predictions tasks.

5.3 How does train data and train method impact performance?

Row 4 to 6 and 10 to 12 in Table 1 demonstrates ablation on train data and train method. Using additional data (i.e `Augmented`) improves over `Gold` in most cases, confirming the importance of using augmented data for quality prediction tasks which mostly are low-resource condition. `K-folds ensemble`[14] further improves over `Vanilla` in all cases, again confirming the widely accepted fact that ensembling techniques are useful to give additional boost in performance.

6 Results

Table 3 and 4 demonstrate our final submission systems for sentence-level and word-level quality prediction task respectively. Refer to Appendix B for detailed configurations used for each final submission models.

	Our Submission		Organizer's Baseline	
	Spearman	Rank	Spearman	Rank
Multi	0.4490	4th	0.3172	6th
En-De	0.5815	3rd	0.4548	10th
En-Ru	0.4963	3rd	0.3327	11th
Zh-En	0.3254	4th	0.1641	11th
Multi	0.5015	2nd	0.4148	5th
Multi (w/o En-Yo)	0.5710	3rd	0.4974	6th
En-Mr	0.6038	1st	0.4356	9th
En-Cs	0.6362	2nd	0.5598	7th
En-Ja	0.3266	4th	0.2716	9th
Km-En	0.6526	3rd	0.5788	7th
Ps-En	0.6713	3rd	0.6410	6th
# params	560M		564M	
Disk space	2,243MB		2,280MB	

Table 3: Submission results on Sentence-level Quality Prediction Task

	Our Submission		Organizer's Baseline	
	MCC	Rank	MCC	Rank
Multi	0.3167	2nd	0.2345	3rd
Multi (w/o En-Yo)	0.3431	2nd	0.2569	3rd
En-De	0.3186	2nd	0.1824	5th
En-Ru	0.4207	2nd	0.2027	5th
Zh-En	0.3514	2nd	0.1036	5th
En-Mr	0.4178	1st	0.3058	5th
En-Cs	0.3961	3rd	0.3245	4rd
En-Ja	0.2573	2nd	0.1751	4th
Km-En	0.4291	1st	0.4016	4th
Ps-En	0.3735	2nd	0.3593	3rd
# params	560M		564M	
Disk space	2,243MB		2,280MB	

Table 4: Submission results on Word-level Quality Prediction Task

7 Conclusions

In this work, we describe our system submission to the WMT 2022 Quality Estimation shared task. Our system is a multilingual and multi-task model for both sentence and word level quality prediction tasks. We demonstrate through ablation study that additional training objectives and data can further improve quality prediction performance. Our final model is trained and inferenced using K-folds ensemble which show remarkable performance in all language pairs and tasks. However, we find that multi-task or auxiliary-task training do not help in word-level as much as they do in sentence-level quality prediction. Further analysis to understand the dynamics of training with multiple objectives and improvements on word-level quality prediction are challenges that we need to overcome in future work.

[14] We leave out `development` portion of `Gold` for final evaluation within the scope of ablation

References

John Blatz, Erin Fitzgerald, George Foster, Simona Gandrabur, Cyril Goutte, Alex Kulesza, Alberto Sanchis, and Nicola Ueffing. 2004. Confidence estimation for machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 315–321, Geneva, Switzerland. COLING.

Yimeng Chen, Chang Su, Yingtao Zhang, Yuxia Wang, Xiang Geng, Hao Yang, Shimin Tao, Guo Jiaxin, Wang Minghan, Min Zhang, Yujia Liu, and Shujian Huang. 2021. HW-TSC's participation at WMT 2021 quality estimation shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 890–896, Online. Association for Computational Linguistics.

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.

Jaime Duque Domingo, Roberto Medina Aparicio, and Luis Miguel Gonzx00E1;lez Rodrigo. 2022. Cross validation voting for improving cnn classification in grocery products. *IEEE Access*, 10:20913–20925.

Patrick Fernandes, António Farinhas, Ricardo Rei, José G. C. de Souza, Perez Ogayo, Graham Neubig, and André F. T. Martins. 2022. Quality-aware decoding for neural machine translation.

Hyun Kim, Hun-Young Jung, Hongseok Kwon, Jong-Hyeok Lee, and Seung-Hoon Na. 2017. Predictor-estimator: Neural quality estimation based on target word prediction for machine translation. *ACM Trans. Asian Low-Resour. Lang. Inf. Process.*, 17(1).

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Seunghyun Lim, Hantae Kim, and Hyunjoong Kim. 2021. Papago's submission for the WMT21 quality estimation shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 935–940, Online. Association for Computational Linguistics.

Sahana Ramnath, Melvin Johnson, Abhirut Gupta, and Aravindan Raghuveer. 2021. HintedBT: Augmenting Back-Translation with quality and transliteration hints. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1717–1733, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Holger Schwenk, Guillaume Wenzek, Sergey Edunov, Edouard Grave, Armand Joulin, and Angela Fan. 2021. CCMatrix: Mining billions of high-quality parallel sentences on the web. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6490–6500, Online. Association for Computational Linguistics.

Lucia Specia, Frédéric Blain, Marina Fomicheva, Erick Fonseca, Vishrav Chaudhary, Francisco Guzmán, and André F. T. Martins. 2020. Findings of the WMT 2020 shared task on quality estimation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 743–764, Online. Association for Computational Linguistics.

Lucia Specia, Frédéric Blain, Marina Fomicheva, Chrysoula Zerva, Zhenhao Li, Vishrav Chaudhary, and André F. T. Martins. 2021. Findings of the WMT 2021 shared task on quality estimation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 684–725, Online. Association for Computational Linguistics.

Jiayi Wang, Ke Wang, Boxing Chen, Yu Zhao, Weihua Luo, and Yuqi Zhang. 2021a. Qemind: Alibaba's submission to the WMT21 quality estimation shared task. *CoRR*, abs/2112.14890.

Rui Wang, Xu Tan, Renqian Luo, Tao Qin, and Tie-Yan Liu. 2021b. A survey on low-resource neural machine translation.

Chrysoula Zerva, Daan van Stigt, Ricardo Rei, Ana C Farinha, Pedro Ramos, José G. C. de Souza, Taisiya Glushkova, Miguel Vera, Fabio Kepler, and André F. T. Martins. 2021. IST-unbabel 2021 submission for the quality estimation shared task. In *Proceedings of the Sixth Conference on Machine Translation*, pages 961–972, Online. Association for Computational Linguistics.

A Data Augmentation

Augment Data Type	Label Type	Language	# of samples	Original Source
Pseudo-generated	Sent, Word	En-De	500,000	Europarl, OPUS, Tatoeba, and WMT News Translation Task
		En-Ru	500,000	
		Zh-En	500,000	
		En-Mr	500,000	
		Km-En	423,583	
		Ps-En	131,163	
		En-Ja	500,000	
		En-Cs	500,000	
Task-related	Sent	En-De	78,616	WMT Metrics Shared Task Data
		En-Ru	72,024	
		Zh-En	136,938	
		Km-En	4,722	
		Ps-En	4,611	
		En-Ja	24,429	
		En-Cs	70,911	
		En-Zh	94,667	
		De-En	109,907	
		Ru-En	70,276	
		Ja-En	23,399	
	Word	En-De	37,000	WMT APE Shared Task Data
		En-Ru	17,112	
		En-Mr	19,000	
		De-En *	28,000	
		En-Zh *	9,000	

Table 5: Details on augmented data

B Optimal Configuration

Task	Label Type	Language Pair	Train Objective	Train Data
Sent	MQM	Multi	Multi+Aux	Gold, Task-related
		En-De	Multi+Aux	Gold, Task-related
		En-Ru	Multi	Gold, Task-related
		Zh-En	Single	Gold, Task-related
	DA	Multi	Multi+Aux	Gold, Task-related
		Multi (w/o En-Yo)	Multi+Aux	Gold, Task-related
		En-Mr	Multi+Aux	Gold, Task-related
		En-Cs	Multi	Gold, Task-related
		En-Ja	Multi+Aux	Gold, Task-related
		Km-En	Multi+Aux	Gold, Task-related
		Ps-En	Multi+Aux	Gold, Task-related
Word	MQM + DA	Multi	Multi	Gold, Task-related, Pseudo-generated
		Multi (w/o En-Yo)	Multi	Gold, Task-related, Pseudo-generated
	MQM	En-De	Single	Gold, Task-related
		En-Ru	Multi+Aux	Gold, Task-related
		Zh-En	Single	Gold, Task-related
	DA	En-Mr	Multi+Aux	Gold, Task-related
		En-Cs	Single	Gold, Task-related, Pseudo-generated
		En-Ja	Single	Gold, Task-related, Pseudo-generated
		Km-En	Multi+Aux	Gold, Task-related, Pseudo-generated
		Ps-En	Single	Gold, Task-related, Pseudo-generated

Table 6: Details on optimal configuration