Learning Compact Metrics for MT

Amy Pu* **Hyung Won Chung †** **Ankur P. Parikh** **Sebastian Gehrmann** **Thibault Sellam**
Google Research
New York, NY
amy_pu@alumni.brown.edu
{hwchung, aparikh, gehrmann, tsellam }@google.com

Abstract

Recent developments in machine translation and multilingual text generation have led researchers to adopt trained metrics such as COMET or BLEURT, which treat evaluation as a regression problem and use representations from multilingual pre-trained models such as XLM-RoBERTa or mBERT. Yet studies on related tasks suggest that these models are most efficient when they are large, which is costly and impractical for evaluation. We investigate the trade-off between multilinguality and model capacity with RemBERT, a state-of-the-art multilingual language model, using data from the WMT Metrics Shared Task. We present a series of experiments which show that model size is indeed a bottleneck for cross-lingual transfer, then demonstrate how distillation can help addressing this bottleneck, by leveraging synthetic data generation and transferring knowledge from one teacher to multiple students trained on related languages. Our method yields up to 10.5% improvement over vanilla fine-tuning and reaches 92.6% of RemBERT's performance using only a third of its parameters.

1 Introduction

Recent improvements in Machine Translation (MT) and multilingual Natural Language Generation (NLG) have led researchers to question the use of n-gram overlap metrics such as BLEU and ROUGE (Papineni et al., 2002; Lin, 2004). Since these metrics focus solely on surface-level aspects of the generated text, they correlate poorly with human evaluation, especially when models are producing high-quality text (Belz and Reiter, 2006; Callison-Burch et al., 2006; Ma et al., 2019; Mathur et al., 2020a). This has led to a surge of interest in *learned* metrics that cast evaluation as a regression problem and leverage *pre-trained multilingual models* to capture the semantic similarity between references and generated text (Celikyilmaz et al., 2020). Popular examples of those metrics include COMET (Rei et al., 2020a) and BLEURT-EXTENDED (Sellam et al., 2020a), based on XLM-RoBERTa (Conneau and Lample, 2019; Conneau et al., 2020a) and mBERT (Devlin et al., 2019) respectively. These metrics deliver superior performance over those based on lexical overlap, outperforming even crowd-sourced annotations (Freitag et al., 2021; Mathur et al., 2020b).

Large pre-trained models benefit learned metrics in at least two ways. First, they allow for *cross-task transfer*: the contextual embeddings they produce allow researchers to address the relative scarcity of training data that exist for the task, especially with large models such as BERT or XLNet (Zhang* et al., 2020; Devlin et al., 2019; Yang et al., 2019). Second, they allow for *cross-lingual transfer*: MT evaluation is often multilingual, yet few, if any, popular datasets cover more than 20 languages. Evidence suggests that training on many languages improves performance on languages for which there is little training data, including the *zero-shot* setup, in which no fine-tuning data is available (Conneau and Lample, 2019; Sellam et al., 2020b; Conneau et al., 2018; Pires et al., 2019).

However, the accuracy gains only appear if the model is large enough. In the case of cross-lingual transfer, this phenomenon is known as *the curse of multilinguality*: to allow for positive transfer, the model must be scaled up with the number of languages (Conneau and Lample, 2019). Scaling up metric models is particularly problematic, since they must often run alongside an already large MT or NLG model and, therefore, must share hardware resources (see Shu et al. (2021) for a recent use case). This contention may lead to impractical delays, it increases the cost of running experiments, and it prevents researchers with limited resources from engaging in shared tasks.

*Work done during an internship at Google.
†Work done as a Google AI resident.

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 751–762
November 7–11, 2021.

We first present a series of experiments that validate that previous findings on cross-lingual transfer and the curse of multilinguality apply to the metrics domain, using RemBERT (Rebalanced mBERT), a multilingual extension of BERT (Chung et al., 2021). We then investigate how a combination of multilingual data generation and distillation can help us reap the benefits of multiple languages while keeping the models compact. Distillation has been shown to successfully transfer knowledge from large models to smaller ones (Hinton et al., 2015), but it requires access to a large corpus of unlabelled data (Sanh et al., 2019; Turc et al., 2019), which does not exist for our task. Inspired by Sellam et al. (2020a), we introduce a data generation method based on random perturbations that allows us to synthesize arbitrary amounts of multilingual training data. We generate an 80M-sentence distillation corpus in 13 languages from Wikipedia, and show that we can improve a vanilla pre-trained distillation setup (Turc et al., 2019) by up to 12%. A second, less explored benefit of distillation is that it lets us partially bypass the curse of multilinguality. Once the teacher (i.e., larger) model has been trained, we can generate training data for any language, including the zero-shot ones. Thus, we are less reliant on cross-lingual transfer. We can lift the restriction that one model must carry all the languages, and train smaller models, targeted towards specific language families. Doing so increases performance further by up to 4%. Combining these two methods, we match 92.6% of the the largest RemBERT model's performance using only a third of its parameters.

A selection of code and models is available online at `https://github.com/google-research/bleurt`.

2 Multilinguality and Model Size

To motivate our work, we quantify the trade-off between multilinguality and model capacity using data from the WMT Shared Task 2020, the most recent benchmark for MT evaluation metrics. The phenomenon has been well-studied for tasks such as translation (Aharoni et al., 2019) and language inference (Conneau et al., 2020b), but it is less well understood in the context of evaluation metrics.

Task and Data In the WMT Metrics task, participants evaluate the quality of MT systems with automatic metrics for 18 language pairs—10 to-English, 8 from-English. The success criterion is correlation with human ratings.[1] Following established approaches (Ma et al., 2018, 2019), we utilize the human ratings from the previous years' shared tasks for training. Our training set contains 479k triplets `(Reference translation, MT output, Rating)` in 12 languages, and it is heavily skewed towards English. It covers the target languages of the benchmark except Polish, Tamil, Japanese and Inuktitut.[2] We evaluate the first three in a *zero-shot* fashion and do no report results on Inuktitut because its alphabet is not covered by RemBERT.

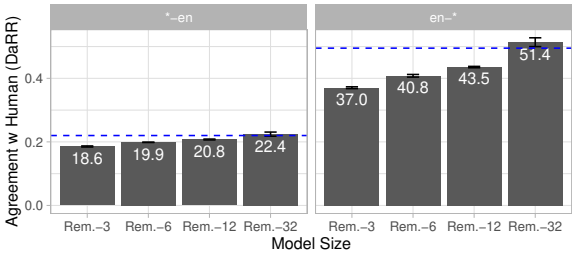

Figure 1: Performance of our models. The dashed line represents the performance of BLEURT-extended (Sellam et al., 2020b). The metric is WMT Metrics DaRR (Mathur et al., 2020b), a robust variant of Kendall Tau, higher is better. We run each experiment with 5 random seeds, report the mean result and Normal-based 95% confidence intervals.

Models Like COMET (Rei et al., 2020a) and BLEURT (Sellam et al., 2020a), we treat evaluation as a regression problem where, given a reference translation $\mathbf{x}$ (typically produced by a human) and predicted translation $\tilde{\mathbf{x}}$ (produced by an MT system), the goal is to predict a real-valued human rating y. As is typical, we leverage pretrained representations (Peters et al., 2018) to achieve strong performance. Specifically, we first embed sentence pairs into a fixed-width vector $\mathbf{v} = \mathbf{F}(\mathbf{x}, \tilde{\mathbf{x}})$ using a pretrained model $\mathbf{F}$ and use this vector as input to a linear layer: $\hat{y} = \mathbf{W}\mathbf{v} + \mathbf{b}$, where $\mathbf{W}$ and $\mathbf{b}$ are the weight matrix and bias vector respectively.

For the pretrained model $\mathbf{F}$, we use RemBERT (Chung et al., 2021), a recently published extension of mBERT (Devlin et al., 2019) pretrained on 104 languages using a combination of Wikipedia and mC4 (Raffel et al., 2020). Be-

[1] The study focuses on segment-level correlation but we also report systems-level results in the appendix.

[2] The target languages are: English, Czech, German, Japanese, Polish, Russian, Tamil, Chinese, Inuktitut. We train on English, German, Chinese, Czech, Russian, Finnish, Estonian, Kazakh, Lithuanian, Gujarati, French, and Turkish.

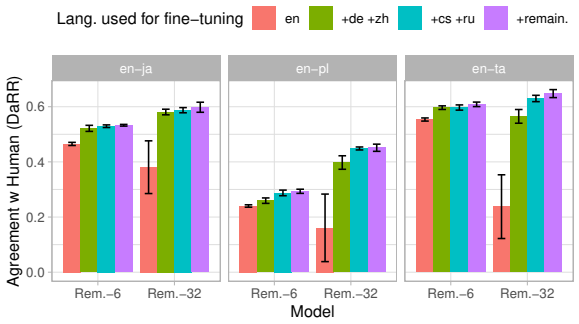

Figure 2: Impact of the number of fine-tuning languages on zero-shot performance, using RemBERT-6 and RemBERT-32 on `en-ja`, `en-pl`, and `en-ta`.

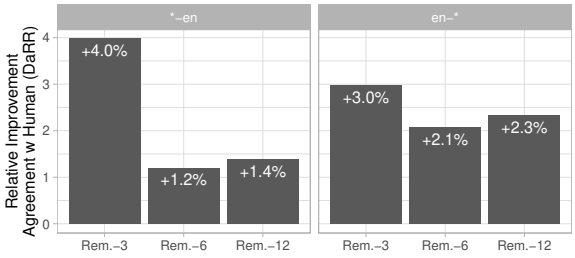

Figure 3: Improvement after removing 86 languages from from pre-training. y-axis: relative performance improvement over a RemBERT of equal size pretrained on 104 languages. Additional details in the appendix.

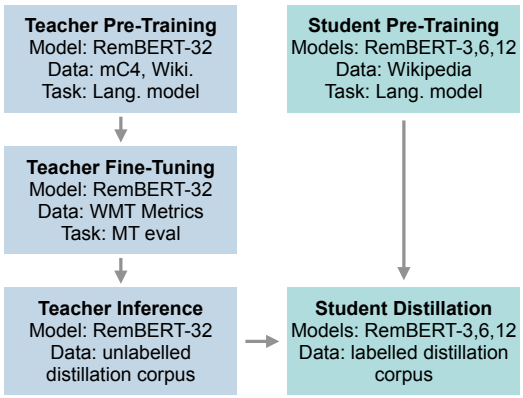

Figure 4: Overview of the default distillation pipeline.

cause RemBERT is massive (32 layers, 579M parameters during fine-tuning) we pre-trained three smaller variants, RemBERT-3, RemBERT-6, and RemBERT-12, using Wikipedia data in 104 languages. The models are respectively 95%, 92%, and 71% smaller, with only 3, 6, and 12 layers. We refer to RemBERT as RemBERT-32 for consistency. The details of architecture, pre-training and fine-tuning are in the appendix.

Figure 1 presents the performance of the models. RemBERT-32 is on par with BLEURT-EXTENDED, a metric based on a similar model which performed well at WMT Metrics 2020.[3] It also corroborates that for a fixed number of languages, larger models perform better.

Cross-lingual transfer during fine-tuning Figure 2 displays the performance of RemBERT-6 and RemBERT-32 on the zero-shot languages as we increase the number of languages used for fine-tuning. We start with English, then add the languages cumulatively, in decreasing order of frequency (without adding data for any of the target languages). Cross-lingual transfer works: in all cases, adding languages improves performance. The effect is milder on RemBERT-6, which consistently starts higher but finishes lower. The appendix presents additional details and results.

Capacity bottleneck in pre-training To further understand the effect of multilinguality, we pre-trained the smaller models from scratch using 18 languages of WMT instead of 104, and fine-tuned on the whole dataset. Figure 3 presents the results: performance increases in all cases, especially for RemBERT-3. This suggests that the models are at capacity and that the 100+ languages of the pre-training corpus compete with one another.

Takeaways Learned metrics are subject to conflicting requirements. On one hand, the opportunities offered by pre-training and cross-lingual transfer encourage researchers to use large, multilingual models. On the other hand, the limited hardware resources inherent to evaluation call for smaller models, which cannot easily keep up with massively multilingual pre-training. We address this conflict with distillation.

3 Addressing the Capacity Bottleneck

The main idea behind distillation is to train a small model (the *student*) on the output of larger one (the *teacher*) (Hinton et al., 2015). This technique is believed to yield better results than training the smaller model directly on the end task because the teacher can provide pseudo-labels for an arbitrary large collection of training examples. Additionally, Turc et al. (2019) have shown that pre-training the student on a language model task before distillation improves its accuracy (in the monolingual setting),

[3]Model provided by the authors. The results diverge from Mathur et al. (2020b) on `en-zh`, for which they submitted a separate metric but the conclusions are similar.

Model		*-en	en-*	en-cs	en-de	en-ja	en-pl	en-ru	en-ta	en-zh
COMET (Rei et al., 2020a)† - *550M params*		-	52.4	66.8	46.8	62.4	46.2	34.4	67.1	43.2
PRISM (Thompson and Post, 2020)† - *745M pa.*		-	45.5	61.9	44.7	57.9	41.4	28.3	44.8	39.7
BLEURT-Ext. (Sellam et al., 2020b) - *425M pa.*		22.0	49.8	68.8	44.7	53.3	43.0	30.6	64.3	44.2
Teacher: RemBERT-32 - *579M params*		22.5	52.3	69.3	45.9	61.7	45.4	31.0	66.6	45.9
RemBERT-3	Fine-tuning	18.5	36.9	42.8	33.0	49.7	26.2	16.0	57.4	33.1
30M params	Distill WMT	16.3	34.8	43.3	29.0	46.8	22.0	15.4	56.1	31.3
	Distill WMT+Wiki	19.1	39.1	42.3	**34.4**	53.6	26.9	18.9	**60.3**	37.6
	1-to-N distill	**19.9**	**40.1**	**47.3**	32.9	**54.4**	**27.3**	**19.3**	60.0	**39.6**
RemBERT-6	Fine-tuning	19.6	40.3	51.4	35.0	53.6	28.5	19.0	60.2	34.8
45M params	Distill WMT	19.9	40.4	53.1	34.8	52.1	28.4	17.9	60.1	36.3
	Distill WMT+Wiki	20.7	42.6	51.6	36.7	55.6	30.2	20.3	**63.1**	40.9
	1-to-N Distill	**21.0**	**44.4**	**56.1**	**38.3**	**57.1**	**34.6**	**22.2**	59.9	**42.9**
RemBERT-12	Fine-tuning	20.6	43.8	57.4	36.7	56.1	33.0	23.4	62.2	37.5
167M params	Distill WMT	21.4	44.8	59.3	39.3	56.0	34.7	22.9	63.6	38.1
	Distill WMT+Wiki	**21.9**	47.3	59.2	**40.8**	**57.9**	37.4	26.4	**65.3**	**44.2**
	1-to-N Distill	21.7	**48.4**	**64.2**	40.2	57.6	**41.3**	**28.4**	63.7	43.5

Table 1: Segment-level agreement with human ratings of our distillation setups. The metric is WMT Metrics DaRR (Mathur et al., 2020b), a robust variant of Kendall Tau, higher is better. The dagger† indicates that the results were obtained from the WMT report. We omit `*-en` for these because of inconsistencies between the benchmark implementations. The number of parameters for COMET corresponds to XLM-RoBERTa-large (Conneau et al., 2020a), mentioned in Rei et al. (2020b). The appendix presents additional details, baselines and systems-level results.

a technique known as *pre-trained distillation*.

Since pre-trained distillation was shown to be simple and efficient, we use it for our base setup. Figure 4 summarizes the steps: we fine-tune RemBERT-32 on human ratings, run it on an unlabelled distillation corpus, and use the predictions to supervise RemBERT-3, 6, or 12. By default, we use the WMT corpus for distillation, i.e., we use the same sentence pairs for teacher fine-tuning and student distillation (but with different labels).

Improvement 1: data generation Distillation requires access to a large multilingual dataset of sentence pairs `(reference, MT output)` to be annotated by the teacher. Yet the WMT Metrics corpus is relatively small, and no larger corpus exists in the public domain. To address this challenge we generate pseudo-translations by perturbing sentences from Wikipedia. We experiment with three types of perturbations: back-translation, word substitutions with mBERT, and random deletions. The motivation is to generate surface-level noise and paraphrases, to expose the student to the different types of perturbations that an MT system could introduce. In total, we generate 80 million sentence pairs in 13 languages. The approach is similar to Sellam et al. (2020a), who use perturbations to generate pre-training data in English. We present the details of the approach in the appendix.

Improvement 2: 1-to-N distillation Another benefit of distillation is that it allows us to lift the constraint that one model must carry all the languages. In a regular fine-tuning setup, it is necessary to pack as many languages as possible in the same model because training data is sparse or non-existent in most languages. In our distillation setup, we can generate vast amounts of data for any language of Wikipedia. It is thus possible to bypass the capacity constraint by training N specialized students, focused on a smaller number of languages. For our experiments, we pre-train five versions of each RemBERT, which cover between 3 and 18 languages each. We tried to form clusters of languages that are geographically close or linguistically related (e.g., Germanic or Romance languages), such that each cluster would cover at least one language of WMT. We list all the languages in the appendix.

Results Table 1 presents performance results on WMT Metrics 2020. For each student model, we present the performance of a naive fine-tuning baseline, followed by vanilla pre-trained distillation on WMT data. We then introduce our synthetic data and 1-to-N distillation. We compare to COMET, PRISM, and BLEURT-EXTENDED, three SOTA metrics from WMT Metrics '20 (Mathur et al., 2020b).

On *en-**, the improvements are cumulative: *Distill WMT+Wiki* outperforms *Distill WMT* (be-

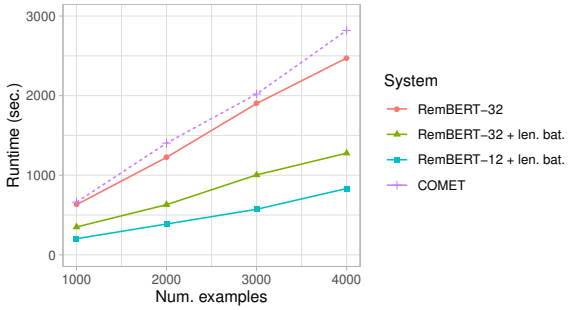

Figure 5: Runtime varying number of examples. We ran each experiment 3 times and averaged the results, using a Google Cloud VM with 4 Intel Haswell vCPUS, 15GB main memory, and an nVidia Tesla T4 GPU. COMET uses the model `wmt-large-da-estimator-1719`.

tween 5 and 12% improvement), and it is itself outperformed by *1-to-N* (up to 4%). Combining techniques improves the baselines in all cases, up to 10.5% improvement compared to fine-tuning. RemBERT-12 matches 92.6% of the teacher model's performance using only a third of its parameters, and it is competitive with current state-of-the-art models.

Runtime To validate the usefulness of our approach, we illustrate how to speed up RemBERT in Figure 5. We first obtain a first 2X speedup by applying length-based batching, a simple optimization which consists in batching examples that have similar a length and cropping the resulting tensor, as done in BERT-Score (Zhang* et al., 2020). Doing so allows us to remove the padding tokens, which cause wasteful computations. We then obtain a 35% improvement by using the distilled version of the model, RemBERT-12. The final model processes 4.8 tuples per second, an approximately 3X improvement over the baselines COMET and RemBERT-32.

4 Conclusion

We experimented with cross-lingual transfer in learned metrics, exposed the trade-off between multilinguality and model capacity, and addressed the problem with distillation on synthetic data. Further work includes generalizing the approach other tasks and experimenting with complementary compression methods such as pruning and quantization (Kim et al., 2021; Sanh et al., 2020), as well as increasing linguistic coverage (Joshi et al., 2020).

Acknowledgments

We thank Vitaly Nikolaev, who provided guidance on language families and created groups for the multiple-students setup. We also thank Iulia Turc, Shashi Narayan, George Foster, Markus Freitag, and Sebastian Ruder for the proof-reading, feedback, and suggestions.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3874–3884, Minneapolis, Minnesota. Association for Computational Linguistics.

Anja Belz and Ehud Reiter. 2006. Comparing automatic and human evaluation of NLG systems. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, Trento, Italy. Association for Computational Linguistics.

Chris Callison-Burch, Miles Osborne, and Philipp Koehn. 2006. Re-evaluating the role of Bleu in machine translation research. In *11th Conference of the European Chapter of the Association for Computational Linguistics*, Trento, Italy. Association for Computational Linguistics.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *CoRR*, abs/2006.14799.

Hyung Won Chung, Thibault Fevry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2021. Rethinking embedding coupling in pre-trained language models. In *International Conference on Learning Representations*.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020a. 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.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2020b. 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.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, volume 32.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Markus Freitag, George F. Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *CoRR*, abs/2104.14478.

Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*.

Pratik Joshi, Sebastin Santy, Amar Budhiraja, Kalika Bali, and Monojit Choudhury. 2020. The state and fate of linguistic diversity and inclusion in the NLP world. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6282–6293, Online. Association for Computational Linguistics.

Sehoon Kim, Amir Gholami, Zhewei Yao, Michael W Mahoney, and Kurt Keutzer. 2021. I-bert: Integer-only bert quantization. *arXiv preprint arXiv:2101.01321*.

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*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Qingsong Ma, Ondřej Bojar, and Yvette Graham. 2018. Results of the WMT18 metrics shared task: Both characters and embeddings achieve good performance. In *Proceedings of the Third Conference on Machine Translation: Shared Task Papers*, pages 671–688, Belgium, Brussels. Association for Computational Linguistics.

Qingsong Ma, Johnny Wei, Ondřej Bojar, and Yvette Graham. 2019. Results of the WMT19 metrics shared task: Segment-level and strong MT systems pose big challenges. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 62–90, Florence, Italy. Association for Computational Linguistics.

Nitika Mathur, Timothy Baldwin, and Trevor Cohn. 2020a. Tangled up in BLEU: Reevaluating the evaluation of automatic machine translation evaluation metrics. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4984–4997, Online. Association for Computational Linguistics.

Nitika Mathur, Johnny Wei, Markus Freitag, Qingsong Ma, and Ondřej Bojar. 2020b. Results of the WMT20 metrics shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 688–725, Online. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Telmo Pires, Eva Schlinger, and Dan Garrette. 2019. How multilingual is multilingual BERT? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4996–5001, Florence, Italy. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020a. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020b. Unbabel's participation in the WMT20 metrics shared task. In *Proceedings of*

the Fifth Conference on Machine Translation, pages 911–920, Online. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS 5th Workshop on Energy Efficient Machine Learning and Cognitive Computing*.

Victor Sanh, Thomas Wolf, and Alexander M. Rush. 2020. Movement pruning: Adaptive sparsity by fine-tuning.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020a. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Thibault Sellam, Amy Pu, Hyung Won Chung, Sebastian Gehrmann, Qijun Tan, Markus Freitag, Dipanjan Das, and Ankur Parikh. 2020b. Learning to evaluate translation beyond English: BLEURT submissions to the WMT metrics 2020 shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 921–927, Online. Association for Computational Linguistics.

Raphael Shu, Kang Min Yoo, and Jung-Woo Ha. 2021. Reward optimization for neural machine translation with learned metrics. *CoRR*, abs/2104.07541.

Brian Thompson and Matt Post. 2020. Automatic machine translation evaluation in many languages via zero-shot paraphrasing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 90–121, Online. Association for Computational Linguistics.

Iulia Turc, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Well-read students learn better: The impact of student initialization on knowledge distillation. *arXiv*.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mT5: A massively multilingual pre-trained text-to-text transformer. *arXiv e-prints*, page arXiv:2010.11934.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

A Training RemBERT for MT Evaluation

A.1 RemBERT Pre-Training

RemBERT is an encoder-only architecture, similar to BERT but with an optimized parameter allocation (Chung et al., 2021). It has reduced input embedding dimension and the saved parameters are reinvested in the form of wider and deeper Transformer layers, keeping the model size constant. In addition, the input and the output embeddings (the weights associated with the softmax layer) are decoupled during pre-training.

Table 2 describes the architecture of the four RemBERT models, along with the number of parameters (note that we remove the output embedding layer during fine-tuning, which reduces the model size). We obtained the original RemBERT model from its authors, and we trained the smaller models for the purpose of this study with a modified version of the public BERT codebase.[4] By default, all models are pre-trained on 104 languages using a masked language modelling objective (Devlin et al., 2019). The setup for the smaller models is similar to Chung et al. (2021), except that RemBERT uses on mC4 (Xue et al., 2020) and Wikipedia while we use Wikipedia only. We train the custom RemBERT models for 2^{17} steps using the Adam optimizer (Kingma and Ba, 2015), using learning rate 0.0002 (with 10,000 linear warm-up followed by inverse square root decay schedule) and batch size 512 on 16 TPU v3 chips. To reduce the size of the models further, we use a smaller SentencePiece model with 120K tokens instead of 250k. Large RemBERT was fine-tuned with sequence size 128, while the student models were fine-tuned with sequence size 512.

A.2 Fine-Tuning for the WMT Metrics Shared Task

We fine-tune RemBERT on the WMT Metrics shared task following the methology of Sellam et al. (2020b). We combine all the sentence pairs of WMT 2015 to 2019, and set aside 5% of the data for continuous evaluation. The data can be downloaded from the WMT Website.[5] The distribution of examples per language is shown in Figure. 6. We sample the sentences randomly, then re-adjust the

[4] https://github.com/google-research/bert

[5] http://www.statmt.org/wmt20/metrics-task.html

Hyperparameter	RemBERT	RemBERT-3	RemBERT-6	RemBERT-12
Number of layers	32	3	6	12
Hidden size	1152	640	640	1024
Vocabulary size	250,000	120,000	120,000	120,000
Input embedding dimension	256	128	128	128
Output embedding dimension	1536	2048	2048	2048
Number of heads	18	8	8	16
Head dimension	64	80	80	64
Num. params. during pre-training	995M	276M	291M	412M
Num. params. during fine-tuning	579M	30M	45M	167M

Table 2: RemBERT architecture.

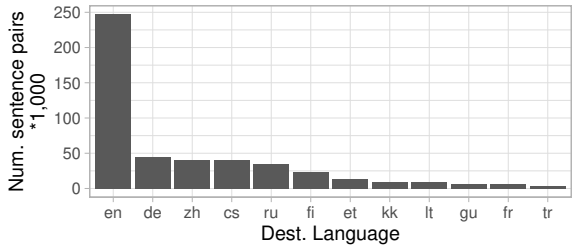

Figure 6: Distribution of the languages in the training set, built from WMT Metrics 2015 to 2019.

sample such that there are not reference translations leaking between the datasets. We train the model with Adam for 5,000 steps and a batch size of 128 while evaluating it on the eval set every 250 steps. We keep the checkpoint that leads to the best performance. To determine the learning rate, we ran a parameter sweep on a previous year of the benchmark (using 2015 to 18 for train and 2019 for test) using the values `[1e-6, 2e-6, 5e-6, 7e-6, 8e-6, 9e-6, 1e-5, 2e-5]`, and kept the learning rate that led the best results (`1e-6`). We also experimented with language rebalancing, batch sizes, dropout, and training duration during preliminary sets of experiments. The setup we used for RemBERT-3, 6 and 12 is similar, except that we used learning rate 1e-5 (obtained with a parameter sweep on a randomly held-out sample), 20,000 training steps, batch size 32, and we evaluate the model every 1,000 steps. We train each model with 4 TPU v2 chips, and evaluate with a Nvidia Tesla V100 GPU.

B Additional Ablation Experiments on WMT Metrics Shared task 2020

We present the detail of our ablation experiments, which expose the trade-off between model capacity and multi-linguality in learned metrics.

In Figure 7, we iteratively expand the number of fine-tuning languages, starting with only English and adding languages in decreasing order of frequency. We add the languages by bucket, such that each bucket contains about the same number of examples (Figure 6 shows the size of the training set for each language).

We start with the five languages for which we have training data. In all cases introducing fine-tuning data for a particular language pair improves the metric's performance on this language. The effect of subsequent additions (that is, cross-lingual transfer) is mixed. For instance, the effect is mild to negative on `*-en`, while it is mostly positive `en-cs`.

Adding data has a different effect on zero-shot languages: in almost all cases, it brings improvements. The effect appears milder on the smaller models, especially RemBERT-3 for which we observe slight performance drops (`en-ta` and `en-ja`), which is consistent the "curse of multilinguality" (Conneau and Lample, 2019).

Figure 8 shows the limit of our smaller models: in 21 cases out of 24 (regardless of whether the language is zero-shot or not), the performance of the model improves when we remove 86 languages from pre-training. This is further evidence that the models are saturated.

C Details of the Distillation Pipeline

C.1 Distillation Data Generation Method

We generate synthetic `(Reference Translation, MT outputs)` pairs by perturbing sentences from Wikipedia. A similar method has been shown to be useful when generating pre-training data in a monolingual context (Sellam et al., 2020a). We apply it to 13 languages: English, Japanese, Lithuanian, Czech, Tamil, Chinese, Russian, Kazakh, Gujarati, Finnish, French, Polish, and German. We chose

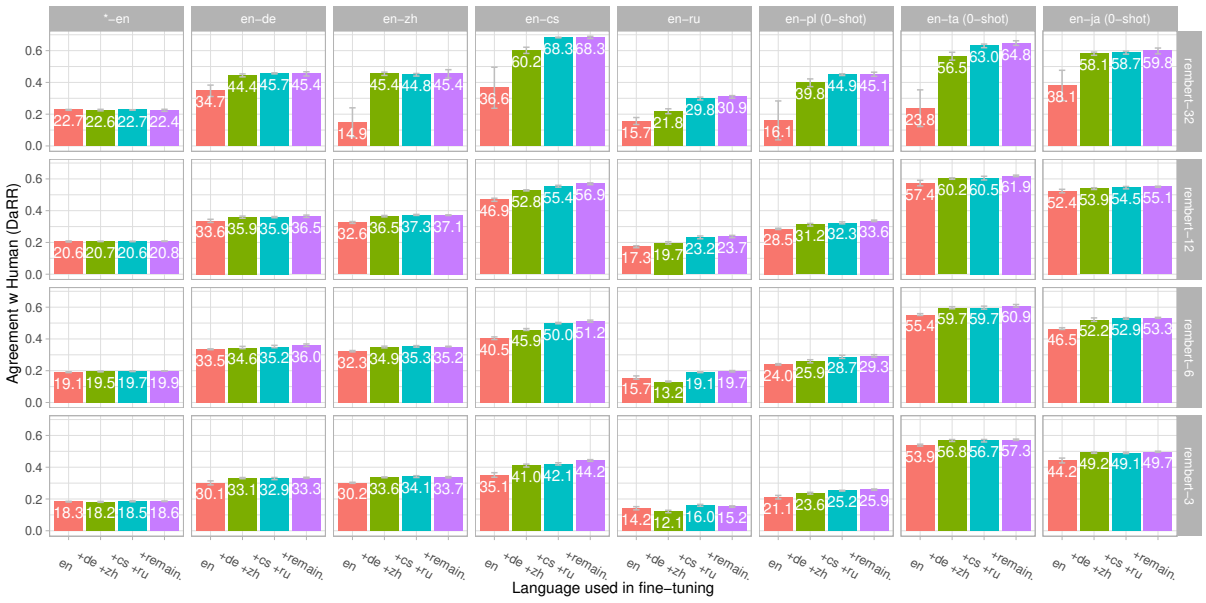

Figure 7: Impact of the number of fine-tuning languages on BLEURT's performance, using all models and all test language pairs of WMT'20 except `en-iu`. The metric is WMT Metrics DaRR (Mathur et al., 2020b), a robust variant of Kendall Tau, higher is better. We run each experiment 5 times, report the mean result and Normal-based 95% confidence intervals.

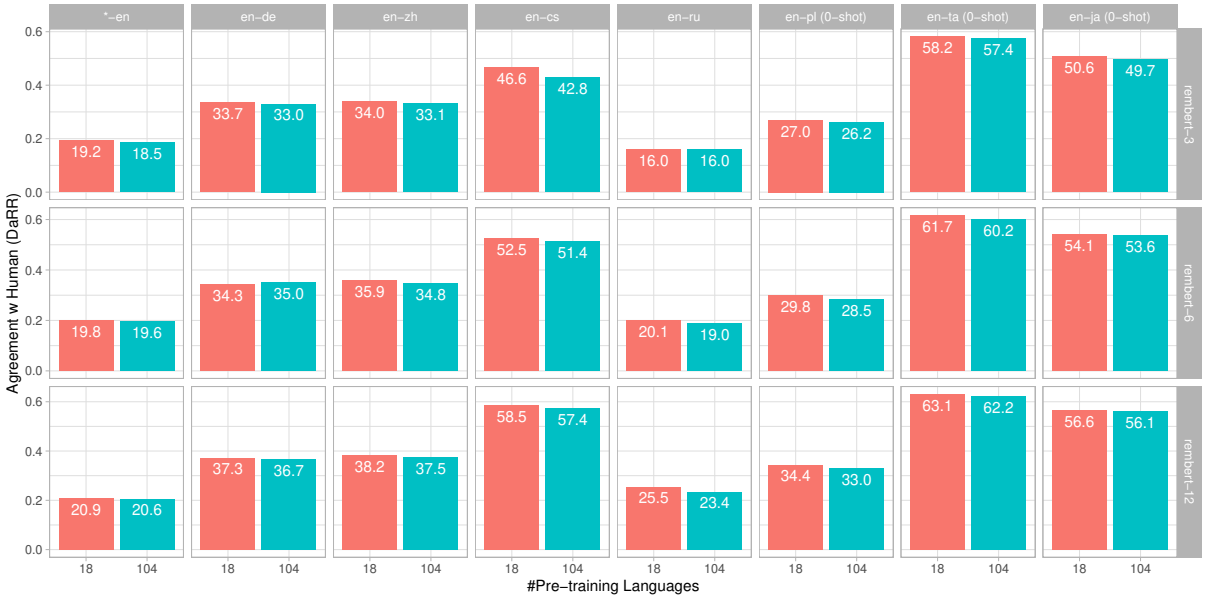

Figure 8: Performance improvement after removing 86 languages from from pre-training (out of 104), using all models and all test language pairs of WMT'20 except `en-iu`.

these languages because they are covered by the WMT Metrics setup during training (e.g., Kazakh, Gujarati, Finnish), testing (Japanese, Polish, Tamil), or both. We emulate the noise introduced by MT systems with three types of perturbations:

- Word substitution: we randomly mask up to 15 WordPiece tokens, and replace the masks by a multilingual model. We sample the number of tokens to be masked uniformly, and we run beam search with mBERT, using beam size 8. We used the official mBERT model.[6]

- Back-translation: we translate the Wikipedia from the source to English, then back in the source language with translation models. We

[6] https://github.com/google-research/bert

Student 1	Student 2	Student 3	Student 4	Student 5
Afrikaans	Catalan	Bengali	Belarusian	Burmese
Danish	French	Gujarati	Bulgarian	Chinese
Dutch	Galician	Hindi	Bulgarian (Latin)	Chinese (Latin)
English	Haitian Creole	Hindi (Latin)	Czech	Japanese
German	Italian	Marathi	Macedonian	
Icelandic	Latin	Nepali	Polish	
Luxembourgish	Portuguese	Persian	Russian	
Norwegian	Romanian	Punjabi	Russian (Latin)	
Swedish	Spanish	Tajik	Serbian	
West Frisian		Urdu	Slovak	
		Tamil	Slovenian	
			Ukranian	
			Finnish	
			Estonian	
			Kazakh	
			Lithuanian	
			Latvian	
			Turkish	

Table 3: Languages per student used for 1-to-N distillation.

used the Tensor2Tensor framework,[7] using models trained on the corresponding WMT datasets.

- Word dropping: we duplicate 30% of the dataset and randomly drop words from the perturbations.

We generate between 1.8M and 7.3M sentence pairs for each language, for a total of 84M unlabelled examples.

C.2 Languages Used in 1-to-N Distillation

Table 3 shows the five language clusters used for the 1-to-N distillation experiments. The groups were created by first joining languages based on their linguistic proximity (e.g., Romance or Germanic languages). Since that left multiple languages in their own cluster, we then combined them based on geographic distance (e.g., Tamil is part of the otherwise Indo-Iranian cluster and Japanese part of a cluster of Sino-Tibetan languages).

C.3 Setup and Hyper-parameters

The hyper-parameters we use for distillation are similar to those of fine-tuning, except that we train the models for 500,000 batches of 128 examples, and thus we learn from 64M sentences instead of 640K. Doing so takes about 1.5 days for RemBERT-3 and 6, and 3.5 days for RemBERT-12. We train the models to completion (i.e., no early-stopping).

D Additional Details of Metrics Performance

We report system-level and segment-level performance of the compact metrics on the MWT Metrics shared task 2020, extending the performance analysis of the distilled models.

We re-implemented the WMT Metrics benchmark using data provided by the organizers. The results are consistent with the published version (Mathur et al., 2020b) except for segment-level to-English pairs, marked with a dagger† in the tables. We ran BLEURT, BLEURT-Tiny, BLEURT-English WMT'20, BLEURT-EXTENDED ourselves. The first two are available online,[8] the latter two were submitted to the WMT Metrics shared task 2020 and were obtained from the authors. We also report results for three state-of-the-art metrics: COMET (Rei et al., 2020a), PRISM (Thompson and Post, 2020), and YISI-1 (Thompson and Post, 2020), using the WMT Metrics report. We only report results for from-English pairs because the benchmark implementations are consistent for these. We also add the baseline `N Fine-tuning`, which describes the performance of fine-tuning the N models presented in Section C.2 directly on WMT data.

As observed in the past (Mathur et al., 2020a,b) system- and segment-level correlations present very different outcomes: the teacher RemBERT-32 is outperformed by several other metrics on both `en-*` and `*-en`, and the impact of the distillation

[7] https://github.com/tensorflow/tensor2tensor

[8] https://github.com/google-research/bleurt

		*-en	cs-en	de-en	iu-en†	ja-en	km-en†	pl-en	ps-en†	ru-en	ta-en	zh-en
BLEURT-Tiny (Sellam et al., 2020a)		17.0	5.2	42.1	22.7	18.4	26.5	2.1	16.8	3.8	21.7	10.7
BLEURT (Sellam et al., 2020a)		21.2	10.7	45.4	27.4	25.8	31.7	4.1	21.1	8.0	24.0	13.9
BLEURT English WMT'20 (Sellam et al., 2020b)		22.1	12.6	45.3	27.6	26.5	33.3	5.7	23.5	9.3	23.1	13.7
BLEURT-Ext (Sellam et al., 2020b)		22.0	12.7	44.6	27.9	27.1	33.8	4.4	20.8	10.1	24.7	13.7
RemBERT-32		22.5	13.9	46.2	29.1	28.0	31.1	4.6	22.6	9.9	25.1	14.5
rembert-3	Fine-tuning	18.5	8.0	42.7	25.0	23.3	26.9	2.6	19.1	4.9	19.7	12.8
	N Fine-tuning	18.0	8.1	42.8	24.1	23.0	26.5	2.8	17.4	4.6	19.1	11.8
	Distill. WMT	16.3	7.0	41.2	23.6	21.5	24.1	0.1	12.6	2.8	19.3	11.2
	Distill. Wiki + WMT	19.1	7.5	43.6	24.7	25.1	25.7	2.8	19.2	7.9	21.5	13.3
	1-to-N Distill	19.9	9.1	44.8	27.0	24.4	27.9	1.5	20.4	7.4	22.7	13.7
rembert-6	Fine-tuning	19.6	10.1	43.9	24.6	23.4	29.5	4.0	20.2	6.5	20.4	13.4
	N Fine-tuning	18.9	9.5	44.2	22.9	21.6	28.9	1.9	20.4	6.4	21.3	12.3
	Distill. WMT	19.9	9.4	43.8	26.9	24.3	30.4	4.3	19.5	5.9	21.1	13.5
	Distill. Wiki + WMT	20.7	10.4	45.2	26.9	25.0	30.3	2.3	21.2	8.4	23.4	14.1
	1-to-N Distill	21.0	10.8	45.7	26.1	25.4	29.5	4.0	21.5	9.2	23.8	14.0
rembert-12	Fine-tuning	20.6	10.5	45.2	25.4	25.2	30.6	4.4	20.4	8.5	21.8	14.0
	N Fine-tuning	19.5	12.1	44.8	23.4	24.7	28.6	2.1	19.1	7.4	20.5	12.7
	Distill. WMT	21.4	11.1	45.6	25.4	26.7	31.1	6.0	21.7	8.6	23.6	14.5
	Distill. Wiki + WMT	21.9	11.9	45.8	28.8	26.0	31.6	4.6	22.6	10.1	23.5	14.1
	1-to-N Distill	21.7	12.3	46.5	29.5	25.7	31.4	3.2	21.7	8.6	23.4	14.9

Table 4: Segment-level agreement with human ratings on to-English language pairs. The metric is WMT Metrics DaRR (Mathur et al., 2020b), a robust variant of Kendall Tau, higher is better. The dagger† indicates that our results may differ from at least 0.5 percentage point from the published WMT results on the language pair (we used the BLEURT submissions for comparison).

		en-*	en-cs	en-de	en-ja	en-pl	en-ru	en-ta	en-zh
BLEURT-Ext. (Sellam et al., 2020b)		49.8	68.8	44.7	53.3	43.0	30.6	64.3	44.2
COMET (Rei et al., 2020a)†		52.4	66.8	46.8	62.4	46.2	34.4	67.1	43.2
PRISM (Thompson and Post, 2020)†		45.5	61.9	44.7	57.9	41.4	28.3	44.8	39.7
YiSi-1 (Thompson and Post, 2020)†		46.9	55.0	42.7	56.8	34.9	25.6	66.9	46.3
RemBERT-32		52.3	69.3	45.9	61.7	45.4	31.0	66.6	45.9
rembert-3	Fine-tuning	36.9	42.8	33.0	49.7	26.2	16.0	57.4	33.1
	N Fine-tuning	32.7	42.7	30.8	43.3	21.8	13.7	44.6	31.7
	Distill. WMT	34.8	43.3	29.0	46.8	22.0	15.4	56.1	31.3
	Distill. Wiki + WMT	39.1	42.3	34.4	53.6	26.9	18.9	60.3	37.6
	1-to-N Distill	40.1	47.3	32.9	54.4	27.3	19.3	60.0	39.6
rembert-6	Fine-tuning	40.3	51.4	35.0	53.6	28.5	19.0	60.2	34.8
	N Fine-tuning	36.6	49.3	31.5	46.5	26.3	18.2	48.8	35.5
	Distill. WMT	40.4	53.1	34.8	52.1	28.4	17.9	60.1	36.3
	Distill. Wiki + WMT	42.6	51.6	36.7	55.6	30.2	20.3	63.1	40.9
	1-to-N Distill	44.4	56.1	38.3	57.1	34.6	22.2	59.9	42.9
rembert-12	Fine-tuning	43.8	57.4	36.7	56.1	33.0	23.4	62.2	37.5
	N Fine-tuning	40.6	58.1	34.1	50.6	35.0	22.7	48.4	35.4
	Distill. WMT	44.8	59.3	39.3	56.0	34.7	22.9	63.6	38.1
	Distill. Wiki + WMT	47.3	59.2	40.8	57.9	37.4	26.4	65.3	44.2
	1-to-N Distill	48.4	64.2	40.2	57.6	41.3	28.4	63.7	43.5

Table 5: Segment-level agreement with human ratings on from-English language pairs. The metric is WMT Metrics DaRR (Mathur et al., 2020b), a robust variant of Kendall Tau, higher is better. We average all the language pairs to English. The dagger† indicates that the results were obtained from the WMT report.

improvements is mixed on to-English. A possible explanation is that system-level involves small sample sizes and that the data is very noisy (Freitag et al., 2021). Another explanation is that systems-level quality assessment is simply another task, which requires its own set of optimizations. In spite of these divergences, Table 7 shows that our contributions bring solid improvements on `en-*` (up to 20.2%), which validates our approach.

		*-en	cs-en	de-en	iu-en	ja-en	km-en	pl-en	ps-en	ru-en	ta-en	zh-en
BLEURT-Tiny (Sellam et al., 2020a)		76.1	81.8	65.8	49.7	86.0	96.2	32.9	95.5	89.5	79.8	84.0
BLEURT (Sellam et al., 2020a)		76.2	73.2	81.3	53.4	81.4	96.6	30.7	94.4	82.6	76.4	92.1
BLEURT-English WMT'20 (Sellam et al., 2020b)		74.9	72.5	77.0	32.0	82.0	98.4	37.1	95.5	84.4	76.8	93.1
BLEURT-Ext. (Sellam et al., 2020b)		73.1	66.8	81.8	35.9	77.2	98.5	29.8	94.2	79.7	74.3	93.1
RemBERT-32		75.7	67.1	79.0	51.2	79.6	99.6	33.4	95.9	80.7	77.4	93.5
Rembert-3	Fine-tuning	77.4	78.4	74.2	53.9	90.3	96.9	27.0	92.3	86.1	80.8	94.0
	N Fine-tuning	77.9	79.8	79.4	52.6	93.5	94.5	27.0	92.6	87.3	78.6	94.2
	Distill. WMT	77.5	76.5	78.8	54.1	91.4	96.2	24.4	92.0	85.6	81.5	94.3
	Distill. Wiki + WMT	77.3	80.8	76.3	54.8	91.5	97.6	23.0	89.0	84.5	80.3	95.3
	1-to-N Distill	78.6	78.9	77.8	59.1	92.3	98.4	25.1	92.2	85.9	80.9	95.1
Rembert-6	Fine-tuning	77.8	80.0	76.9	65.4	88.3	95.2	20.5	91.1	85.0	81.2	94.2
	N Fine-tuning	75.9	77.0	76.5	56.6	89.1	95.7	20.8	91.3	80.5	78.2	93.2
	Distill. WMT	77.0	75.5	78.1	60.0	86.4	96.7	24.3	93.0	82.1	80.5	93.5
	Distill. Wiki + WMT	78.2	80.1	76.7	60.8	88.9	98.9	21.6	93.6	85.5	81.1	94.9
	1-to-N Distill	77.0	76.9	76.7	58.0	86.9	99.1	22.2	94.3	82.6	78.7	94.5
Rembert-12	Fine-tuning	76.7	77.4	77.7	59.8	85.5	95.3	22.9	91.7	82.9	80.2	93.4
	N Fine-tuning	74.4	72.5	76.2	50.4	81.3	96.6	27.4	91.5	80.2	74.7	92.6
	Distill. WMT	76.9	75.2	77.2	63.2	84.0	97.5	23.2	93.1	82.7	79.4	93.4
	Distill. Wiki + WMT	77.4	76.3	75.7	62.1	87.3	99.3	22.1	94.2	84.0	78.4	94.1
	1-to-N Distill	76.7	74.6	77.0	54.9	82.9	99.6	25.8	95.4	83.0	79.3	94.0

Table 6: System-level agreement with human ratings on to-English language pairs excluding outliers where they are available. The metric is Pearson correlation (Mathur et al., 2020b), higher is better.

		en-*	en-cs	en-de	en-ja	en-pl	en-ru	en-ta	en-zh
BLEURT-Ext.(Sellam et al., 2020b)		90.3	96.0	87.0	95.3	82.8	98.0	81.4	91.5
COMET (Rei et al., 2020a)†		75.5	92.6	86.3	96.9	80.0	92.5	79.8	0.7
PRISM (Thompson and Post, 2020)†		67.4	80.5	85.1	92.1	74.2	72.4	45.2	22.1
YiSi-1 (Thompson and Post, 2020)†		86.1	66.4	88.7	96.7	71.4	92.6	90.9	95.9
RemBERT-32		83.6	96.1	86.2	97.1	85.4	91.4	80.8	48.6
Rembert-3	Fine-tuning	65.8	56.7	82.3	95.2	73.2	53.1	90.6	9.7
	N Fine-tuning	73.6	67.8	80.9	92.7	67.1	76.5	88.6	41.7
	Distill. WMT	69.2	69.0	79.0	94.1	75.6	62.4	88.3	16.2
	Distill. Wiki + WMT	76.2	63.9	85.8	94.4	73.4	91.3	90.7	34.1
	1-to-N Distill	78.9	67.9	86.6	93.8	73.7	94.7	89.5	46.3
Rembert-6	Fine-tuning	68.2	64.3	82.9	94.3	74.4	74.1	86.6	0.8
	N Fine-tuning	81.8	75.3	84.3	91.4	79.4	91.9	87.8	62.8
	Distill. WMT	70.3	68.8	84.4	93.7	75.5	69.7	87.7	12.7
	Distill. Wiki + WMT	81.5	70.6	87.6	95.8	75.1	94.8	89.5	57.1
	1-to-N Distill	82.0	75.8	86.1	93.1	76.2	96.0	83.6	63.4
Rembert-12	Fine-tuning	74.2	75.2	83.3	92.4	80.2	89.2	85.6	13.2
	N Fine-tuning	77.1	85.4	86.2	88.6	85.7	85.4	85.1	23.1
	Distill. WMT	72.3	75.9	84.7	93.3	79.3	72.4	85.3	14.9
	Distill. Wiki + WMT	85.8	79.8	88.1	96.2	81.4	97.1	84.8	72.9
	1-to-N Distill	86.3	87.9	87.8	94.0	82.0	96.9	82.2	73.5

Table 7: System-level agreement with human ratings on from-English language pairs, excluding outliers where they are available. The metric is Pearson correlation (Mathur et al., 2020b), higher is better. The dagger† indicates that the results were obtained from the WMT report.