Searching for COMETINHO: The Little Metric That Could

Ricardo Rei[*,1,2,3] **Ana C Farinha**[*,1] **José G. C. de Souza**[*,1]
Pedro G. Ramos[2] **André F. T. Martins**[1,3,4] **Luisa Coheur**[2,3] **Alon Lavie**[1]
[1]Unbabel [2]INESC-ID [3]Instituto Superior Técnico [4]Instituto de Telecomunicações
{ricardo.rei, catarina.farinha, jose.souza}@unbabel.com

Abstract

Recently proposed neural-based machine translation evaluation metrics, such as COMET and BLEURT, exhibit much higher correlations with human judgments than traditional lexical overlap metrics. However, they require large models and are computationally very costly, preventing their application in scenarios where one has to score thousands of translation hypotheses (e.g. outputs of multiple systems or different hypotheses of the same system, as in minimum Bayes risk decoding). In this paper, we introduce several techniques, based on pruning and knowledge distillation, to create more compact and faster COMET versions—which we dub COMETINHO. First, we show that just by optimizing the code through the use of *caching* and *length batching* we can reduce inference time between 39% and 65% when scoring multiple systems. Second, we show that pruning COMET can lead to a 21% model reduction without affecting the model's accuracy beyond 0.015 Kendall τ correlation. Finally, we present DISTIL-COMET, a lightweight distilled version that is 80% smaller and 2.128x faster while attaining a performance close to the original model. Our code is available at: https://github.com/Unbabel/COMET

© 2022 The authors. This article is licensed under a Creative Commons 3.0 licence, no derivative works, attribution, CC-BY-ND.
* Corresponding authors.

1 Introduction

Traditional metrics for machine translation (MT) evaluation rely on lexical similarity between a given hypothesis and a reference translation. Metrics such as BLEU (Papineni et al., 2002) and CHRF (Popović, 2015) remain popular due to efficient memory usage and fast computational performance, even though several studies have shown that they correlate poorly with human judgements, specially for high quality MT (Ma et al., 2019; Mathur et al., 2020a).

In contrast, neural fine-tuned metrics on top of pre-trained models such as mBERT (Devlin et al., 2019) and XLM-R (Conneau et al., 2020) (e.g BLEURT (Sellam et al., 2020) and COMET (Rei et al., 2020) have demonstrated significant improvements in comparison to other metrics (Mathur et al., 2020b; Kocmi et al., 2021; Freitag et al., 2021b). The improvements made them good candidates for revisiting promising research directions where the metric plays a more central role in candidate selection during decoding, such as N-best reranking (Ng et al., 2019; Bhattacharyya et al., 2021; Fernandes et al., 2022) and minimum Bayes risk (MBR) decoding (Eikema and Aziz, 2021; Müller and Sennrich, 2021). Nonetheless, the complexity of such strategies using metrics based on large transformer models can become impractical for a large set of MT hypotheses.

In this paper, we describe several experiments that attempt to reduce COMET computational cost and model size to make it more efficient at inference. Our techniques are particularly useful in settings where we have multiple translations from different systems on the same source sentences. Since the models are based on triplet encoders, we will first analyse the impact of *embedding caching*

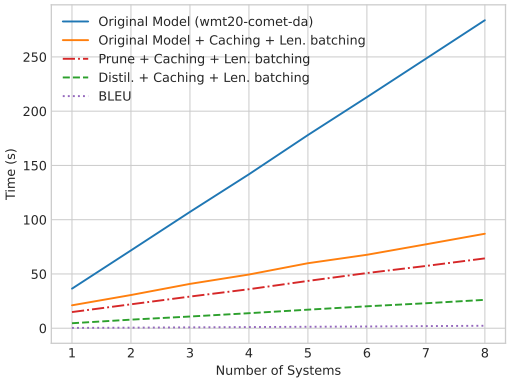


Figure 1: Comparison between the vanilla COMET, COMET with caching and length batching, PRUNE-COMET and DISTIL-COMET. We report the average of 5 runs for each model/metric for a varying number of systems. All experiments were performed using the German→English WMT20 Newstest, with a `NVIDIA GeForce GTX 1080 TI` GPU and a constant batch size of 16. For comparison we also plot the runtime of BLEU in a `Intel(R) Core(TM) i7-6850K CPU @ 3.60GHz`.

and *length batching*. Then, we will try to further reduce the computational cost by using *weight pruning* and *knowledge distillation*. Our results show that embedding caching and length batching alone can boost COMET performance 39.19% when scoring one system and 65.44% when scoring 8 systems over the same test set. Furthermore, with knowledge distillation we are able to create a model that is 80% smaller and 2.128x faster with a performance close to the original model and above strong baselines such as BERTSCORE and PRISM. Figure 1 shows time differences for all proposed methods when evaluating a varying number of systems.

2 Related Work

In the last couple of years, learned metrics such as COMET (Rei et al., 2020) and BLEURT (Sellam et al., 2020) proved to achieve high correlations with human judgments (Mathur et al., 2020b; Freitag et al., 2021a; Kocmi et al., 2021). They are cast as a regression problem and capture the semantic similarity between the translated text and a reference text, going beyond the simple surface/lexical similarities—the base of popular metrics like BLEU (Papineni et al., 2002) and CHRF (Popović, 2015). The fact that COMET and BLEURT metrics leverage large pre-trained multilingual models was a huge turning point. By using contextual embeddings trained on a different task, researchers were able to overcome the scarcity of data in MT evaluation (as well as in other tasks in which data is also limited). With such multilingual models, high-quality MT evaluation is now a possibility, even for language pairs without labeled data available (i.e. zero-shot scenarios). However, this multilingual property usually comes with a trade-off. For example, for cross-lingual transfer task, gains in performance (higher accuracy with human labels) only occur by adding new language pairs until a certain point, after which adding more languages actually decreases the performance, unless the model capacity is also increased (a phenomena called "the curse of multilinguality" (Conneau et al., 2020).

Besides the curse of multilinguality phenomena, the NLP community has been motivated to build larger and larger transformer models because, generally, the bigger the model the better it performs. This was demonstrated in several tasks like the ones in the GLUE benchmark (Goyal et al., 2021) and in multilingual translation tasks (Fan et al., 2020). Hence, models are achieving astonishing sizes like BERT with 340M parameters (Devlin et al., 2019), XLM-R$_{XXL}$ with 10.7B parameters (Goyal et al., 2021), M2M-100 with 12B parameters (Fan et al., 2020), and GPT-3 with 175B parameters (Brown et al., 2020). However, this growth comes with computational, monetary and environmental costs. For example, training a model with 1.5B parameters costs from 80k dollars up to 1.6M dollars[1] when doing hyper-parameter tuning and performing multiple runs per setting (Sharir et al., 2020). Such scale makes running similar experiments impractical to the majority of research groups, and the high energy and high response latency of such models are preventing them from being deployed in production (e.g. (Sun et al., 2020)).

To deal with the above problem, it is necessary to apply techniques for making models more compact, such as pruning, distillation, quantization, among others. In a recent review (Gupta and Agrawal, 2022) summarizes these techniques for increasing inference efficiency, i.e., for making the model faster, consuming fewer computational resources, using less memory, and less disk space. DistilBERT (Sanh et al., 2019) is a successful example: using distillation with BERT as the

[1]Estimates from (Sharir et al., 2020) calculated using internal AI21 Labs data; cloud solutions such as GCP or AWS can differ.

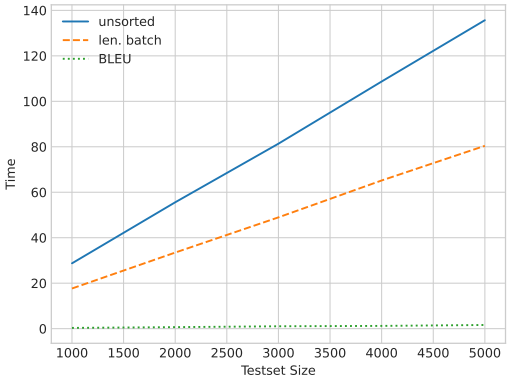


Figure 2: Runtime (in seconds) varying number of examples, with a `NVIDIA GeForce GTX 1080 TI` GPU and a constant batch size of 16. The time is calculated with the average of 10 runs using the default COMET model `wmt20-comet-da`. For comparison we also plot the runtime of BLEU in a `Intel(R) Core(TM) i7-6850K CPU @ 3.60GHz`.

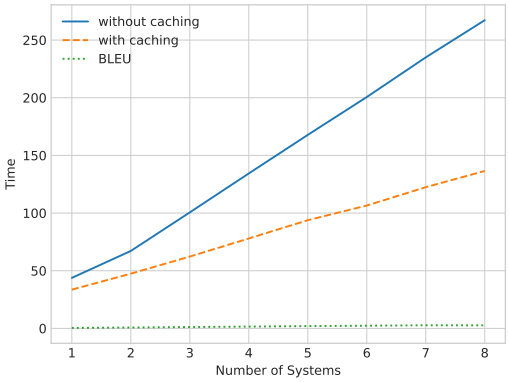


Figure 3: Runtime (in seconds) varying number of systems for the `de-en` WMT20 Newstest, with a `NVIDIA GeForce GTX 1080 TI` GPU and a constant batch size of 16. The time is calculated with the average of 5 runs using the default COMET model `wmt20-comet-da`. For comparison we also plot the runtime of BLEU in a `Intel(R) Core(TM) i7-6850K CPU @ 3.60GHz`.

teacher and reducing the amount of layers from the regular 12 to only 6, the model retains 97% of BERT's performance while reducing the size by 40% and being 60% faster. The authors have also shown that when used for a mobile application (iPhone), the DistilBERT was 71% faster than BERT. Another example, closer to our research, is the metric obtained from using synthetic data and performing distillation using a new variation of BLEURT as the teacher (Pu et al., 2021). The resulting metric obtains up to 10.5% improvement over vanilla fine-tuning and reaches 92.6% of teacher's performance using only a third of its parameters. Nonetheless, the architecture of BLEURT-based models requires that the reference is always encoded together with MT hypothesis which is extremely inefficient in use cases such as MBR, where the metric has a $\mathcal{O}(N^2)$ complexity (with N being the number of hypotheses), and system scoring where for a fixed source and reference we can have several translations being compared.

3 Length Sorting and Caching

Before exploring approaches that reduce the number of model parameters, we experiment with techniques to optimize the inference time computational load. One which is commonly used is to sort the batches according to sentence length to reduce tensor padding (Pu et al., 2021). Since COMET receives three input texts (source, hypothesis and reference), for simplicity, we do length sorting according to the source length. Figure 2 shows the speed difference between an unsorted test set with varying size and length-based sorting.

As previously pointed out, COMET metrics are based on triplet encoders[2] which means that the source and reference encoding does not depend on the provided MT hypothesis as opposed to other recent metrics such as BLEURT (Sellam et al., 2020) which have to repetitively encode the reference for every hypotheses. With that said, using COMET we only need to encode each unique sentence (source, hypothesis translation or reference translation) once. This means that we can cache previously encoded batches and reuse their representations. In Figure 3, we show the speed gains, in seconds, when scoring multiple systems over the same test set. This reflects the typical MT development use case in which we want to select the best among several MT systems.

These two optimizations altogether are responsible for reducing the inference time of COMET from 34.7 seconds to 21.1 seconds while scoring 1 system (39.19% faster) and from 265.9 seconds to 91.9 seconds when scoring 8 systems (65.44% faster). For all experiments performed along the rest of the paper we always use both optimization on all COMET models being compared.

[2]A triplet encoder, is a model architecture where three sentences are encoded independently and in parallel. Architectures such as this have been extensively explored for sentence retrieval applications due to its efficiency (e.g. Sentence-BERT (Reimers and Gurevych, 2019))

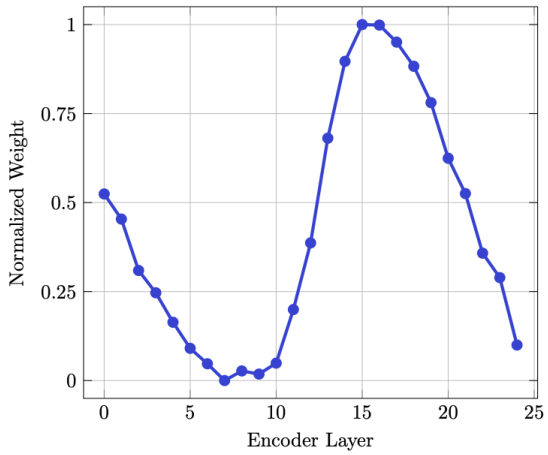


Figure 4: Normalized weights distribution for the COMET default model (`wmt20-comet-da`). As we can observe layers between 15-19 are the most relevant ones with a normalized weight between 0.75 and 1. The representations learnt by layers 15-19 depend on previous layers but we can prune the top layers (20-25) without impacting the layers that the model deemed more relevant.

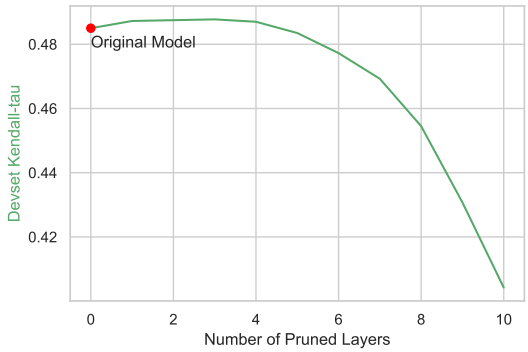


Figure 5: Impacts in performance of Layer Pruning for the WMT 2020 development set. We can observe that removing up to 5 layers does not affect model performance but provides a 10% reduction in model size.

4 Model Pruning

Model pruning has been widely used in natural language processing to remove non-informative connections and thus reducing model size (Zhu and Gupta, 2018). Since most COMET parameters come from the XLM-R model, we attempt to reduce its size. We start with layer pruning by removing the top layers of XLM-R. Then we experiment with making its encoder blocks smaller either by reducing the size of the feed-forward hidden layers or by removing attention heads. The main advantage of these approaches is their simplicity: within minutes we are able to obtain a new model with reduced size and memory footprint with minimal performance impact.

For all the experiments in this section, we used the development set from the Metrics shared task of WMT 2020. This set contains direct assessment annotations (DA; (Graham et al., 2013)) for English→German, English→Czech, English→Polish and English→Russian. We use these language pairs because they were annotated by experts exploring *document context* and in a *bilingual setup* (without access to a reference translation)[3]. Nonetheless, in Section 6 we show the resulting model performance on all language pairs from WMT 2021 for both DA and multi-dimensional quality metric annotations (MQM; (Lommel et al., 2014)).

4.1 Layer Pruning

In large pre-trained language models, different layers learn representations that capture different levels of linguistic abstractions, which can impact a downstream task in different ways (Peters et al., 2018; Tenney et al., 2019). In order to let the model learn the relevance of each layer during training, (Peters et al., 2018) proposed a layer-wise attention mechanism that pools information from all layers. This method has been adopted in COMET.

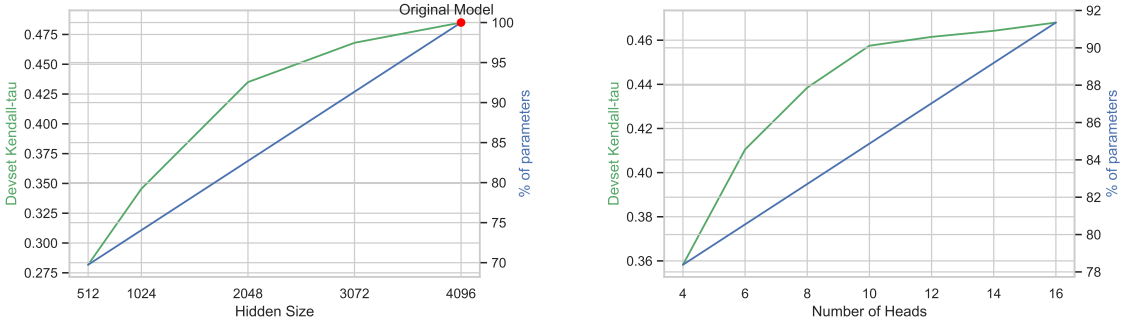
After analyzing the weights learnt by COMET (`wmt20-comet-da`) for each layer of XLM-R (Figure 4), we realized that the topmost layers (20-25) are not the most relevant ones. This means that we can prune those layers without having an impact on the most relevant features.

Each removed layer decreases the number of total parameters by 2.16%. Figure 5 shows the impacts in performance after removing a varying number of layers. As we can observe, performance starts to decrease only after removing 5 layers. Yet, removing 5 layers already produces a 10.8% reduction in model parameters. Surprisingly, removing the last layer (pruning 1 layer) slightly improves the performance in terms of Kendall-tau (Kendall, 1938).

4.2 Transformer Block Pruning

The Transformer architecture is composed of several encoder blocks (layers) stacked on top of the other. In the previous section, we reduce model

[3]In the WMT 2020 findings paper (Mathur et al., 2020b), most metrics showed suspiciously low correlations with human judgements based on crowd-sourcing platforms such as Mechanical Turk. Thus, we decided to focus just on 4 language pairs in which annotations are deemed as trustworthy.



(a) Feed-forward hidden size pruning.

(b) Attention head pruning.

Figure 6: Impact of gradient based pruning techniques on model size (in blue) and performance on the WMT 2020 development set (in green). Note that in Figure (a) we apply pruning just for the feed-forward hidden size. In Figure (b) pruning is applied to several heads while freezing the hidden size to 3072 (3/4 of the original hidden size of XLM-R).

size by removing the topmost blocks (depth pruning). In this section we reduce the size of each block instead (width pruning).

Each transformer block is made of two components: a *self-attention* (composed of several attention heads) and a *feed-forward neural network*. In XLM-R-large, each block is made of 16 self-attention heads followed by a feed-forward of a single hidden layer with 4092 parameters.

Using the `TextPruner` toolkit[4], we can easily prune both the attention heads and the feed-forward hidden sizes. Figure 6a shows the impact of pruning the hidden sizes from 4096→{512, 1024, 2048, 3072} while Figure 6b shows the impact of reducing the attention heads from 16→{4, 6, 8, 10, 12, 14}.

4.3 PRUNED-COMET

After experimenting with these three different pruning techniques, we created a pruned version of COMET in which we keep only 19 XLM-R layers, we reduced the feed-forward hidden size by $3/4$ (3072 hidden size) and we removed 2 heads (out of 16). According to our experiments above, the resulting model's performance drop should be almost the same as the original model but the resulting model is 21.1% smaller.

The resulting model is able to score 1000 samples in just 19.74 seconds, while the original model takes around 31.32 seconds. It is important to notice that most of the XLM-R parameters come from its huge embedding layer. Since the embedding size memory does not affect the inference time, the obtained 20% reduction in parameters translates into speed improvements of around 36.97%.[5]

5 Distillation

Another commonly used way to compress neural networks is through knowledge distilation (Bucilua et al., 2006; Hinton et al., 2015) in which, for large amounts of unlabeled data, a smaller neural network (the student) is trained to mimic a more complex model (the teacher).

As the teacher network, we used an ensemble of 5 COMET models trained with different seeds (Glushkova et al., 2021). The student network follows the same architecture as the original model and the same hyper-parameters. However, instead of using XLM-R-large, it uses a distilled version with only 12 layers, 12 heads, embeddings of 384 features, and intermediate hidden sizes of 1536. This model has only 117M parameters compared to the 560M parameters from the large model.

Regarding the unlabeled data for distillation, we extracted 25M sentence pairs from OPUS (Tiedemann, 2012) ranging a total of 15 language pairs. To guarantee high quality parallel data we used Bicleaner tool (Ramírez-Sánchez et al., 2020) with a threshold of 0.8. Then, using pre-trained MT models available in Hugging Face Transformers, we created 2 different translations for each source: one using a bilingual model (in theory a high quality translation) and another using pivoting (which can be thought as lower quality). Finally, we scored all the data using our teacher ensem-

[4] `https://textpruner.readthedocs.io/en/latest/`

[5] Experiments performed in a `NVIDIA GeForce GTX 1080 TI` GPU and a constant batch size of 16. The resulting time is the average of 5 runs.

Table 1: Kendall's tau correlation on high resource language pairs using the MQM annotations for both News and TED talks domain collected for the WMT 2021 Metrics Task.

		zh-en		en-de		en-ru		
Metric	**# Params**	**News**	**TED**	**News**	**TED**	**News**	**TED**	**avg.**
BLEU	-	0.166	0.056	0.082	0.093	0.115	0.067	0.097
CHRF	-	0.171	0.081	0.101	0.134	0.182	0.255	0.154
BERTSCORE	179M	0.230	0.131	0.154	0.184	0.185	0.275	0.193
PRISM	745M	0.265	0.139	0.182	0.264	0.219	0.292	0.229
BLEURT	579M	**0.345**	**0.166**	**0.253**	**0.332**	**0.296**	**0.347**	**0.290**
COMET	582M	0.336	0.159	0.227	0.290	0.284	0.329	0.271
PRUNE-COMET	460M	0.333	0.157	0.219	0.293	0.274	0.319	0.266
DISTIL-COMET	119M	0.321	0.161	0.202	0.274	0.263	0.326	0.258

Table 2: Kendall's tau-like correlations on low resource language pairs using the DARR data from WMT 2021 Metrics task.

Metric	**# Params**	**zu-xh**	**xh-zu**	**bn-hi**	**hi-bn**	**en-ja**	**en-ha**	**en-is**	**avg.**
BLEU	-	0.381	0.1887	0.070	0.246	0.315	0.124	0.278	0.229
CHRF	-	0.530	0.301	0.071	0.327	0.371	0.186	0.373	0.308
BERTSCORE	179M	0.488	0.267	0.074	0.365	0.413	0.161	0.354	0.303
BLEURT	579M	**0.563**	**0.362**	**0.179**	0.498	0.483	0.186	0.469	0.391
COMET	582M	0.550	0.285	0.156	**0.526**	**0.521**	**0.234**	**0.474**	**0.392**
PRUNE-COMET	460M	0.541	0.264	0.163	0.519	0.513	0.197	0.439	0.377
DISTIL-COMET	119M	0.488	0.254	0.135	0.498	0.471	0.145	0.419	0.344

ble. The resulting corpus contains 45M tuples with (`source`, `translation`, `reference`, `score`).

The resulting model which name DISTIL-COMET, scores 1000 sentences in 14.72 seconds resulting in a 53% speed improvement over the original model[3].

6 Correlation with Human Judgements

In this section, we show results for {PRUNE and DISTIL}-COMET in terms of correlations with MQM annotations from WMT 2021 Metrics task for two different domains: News and TED talks. Since these annotations only cover high-resource language pairs (English→German, English→Russian, Chinese→English), we also evaluate models on low resource language pairs using DA Relative Ranks from WMT 2021, namely we test these models for: Hindi↔Bengali, Zulu↔Xhosa, English→Hausa, English→Icelandic, English→Japanese. For a detailed comparison, we also present results for CHRF (Popović, 2015) and BLEU (Papineni et al., 2002), two computationally efficient lexical metrics, and other neural metrics such as PRISM[6] (Thompson and Post, 2020), BLEURT (Sellam et al., 2020) and BERTSCORE (Zhang et al., 2020).

From Table 1, we can observe that PRUNE-COMET has minimal performance drops compared with vanilla COMET with only 80% of its parameters. DISTIL-COMET performance is on average 0.013 Kendall's bellow vanilla COMET for high resources languages, which is impressive for a model that only has 20% of COMET's parameters. For low-resource languages, we can observe bigger performance differences between COMET, PRUNE-COMET, and DISTIL-COMET which confirm results by (Pu et al., 2021) that shows that smaller MT evaluation models are limited in their ability to generalize to several language pairs. Nonetheless, when comparing with other recently proposed metrics such as PRISM and BERTSCORE, {PRUNE and DISTIL}-COMET have higher correlations with human judgements for both high and low resource language pairs. The only exception is BLEURT which shows stronger correlations than COMET on high-resource language pairs and com-

[6] PRISM does not support the low-resource language pairs used in our experiments, thus we only report PRISM correlations with MQM data

petitive performance in low-resource ones.[7]

7 Use Case: Minimum Bayes Risk Decoding

In minimum Bayes risk (MBR) decoding, a machine translation evaluation metric can be used as the utility function for comparing the translation hypotheses. This kind of approach, also known as "consensus decoding", derived from the idea that the top ranked translation is the one with the highest average score when compared to all other hypotheses. This process requires that each hypothesis translation be compared to every other hypotheses in an hypotheses candidate list. Having faster neural metrics could directly impact research and computational performance of using MBR decoding approaches with such metrics.

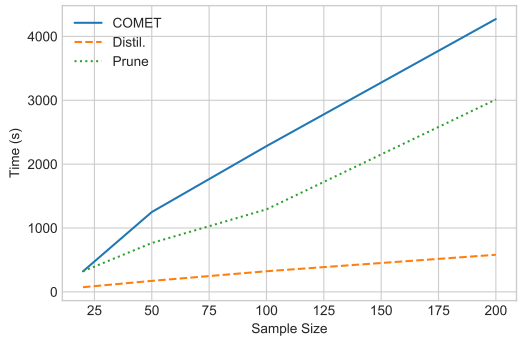


Figure 7: Runtime for performing MBR with a different number of samples using one `NVIDIA GeForce GTX 1080 TI GPU`.

Using COMET models with distillation or pruning can have a considerable effect at the performance of MBR decoding using such models as the utility function. Figure 7 shows that DISTIL-COMET is always substantially faster than the original COMET model especially for larger candidate list sizes such as 200 candidates. Likewise, PRUNE-COMET performs better than the original model but its performance is also considerably higher than DISTIL-COMET.

Regarding the two COMET variants there is a clear trade-off that needs to be taken into consideration, as evidenced by the results in Section 6: while DISTIL-COMET is faster, PRUNE-COMET is more accurate, leaving the choice of each model to use up to the most important aspect for the application. In the case of MBR decoding, this might depend on the hardware available for performing the computations.

[7]For a more detailed comparison between COMET and BLEURT metrics we refer the reader to the WMT 2021 Metrics shared task results paper (Freitag et al., 2021b) where both metrics ended up statistically tied for most language pairs and domains.

8 Conclusion and Future Work

In this paper we presented two simple optimizations that lead to significant performance gains on neural metrics such as COMET and two approaches to reduce its number of parameters. Together these techniques achieve impressive gains in performance (both speed and memory) at a very small cost in performance.

To showcase the effectiveness of our methods, we presented DISTIL-COMET and PRUNE-COMET. These models were obtained using COMET knowledge distillation and pruning respectively. To test the proposed models, we used the data from the WMT 2021 Metrics task which covers low resource languages as well as high resource languages. Overall the results of PRUNE-COMET are stable across the board with only a small degradation compared to the original metric. Knowledge distillation leads to much higher compression rates but seems to confirm previous findings (Pu et al., 2021) which suggest the lack of model capacity when it comes to the multilingual generalization for low resource languages.

A primary avenue for future work is to study how decreasing the model size can further impact on robustness of the metric, inspired by recent studies which identified weaknesses of COMET metrics when dealing with numbers and named entities (Freitag et al., 2021b; Amrhein and Sennrich, 2022). Also, in this work we explored knowledge distillation directly from the teacher output but an interesting avenue for improving the quality of the student model is to explore alternative distillation approaches that learn directly from internal representations of the teacher model such as self-attention distillation (Wang et al., 2020).

Acknowledgments

We would like to thank João Alves and Craig Stewart and the anonymous reviewers for useful feedback. This work was supported by the P2020 Program through project MAIA (contract No 045909) and by the European Union's Horizon 2020 research and innovation program (QUARTZ grant agreement No 951847).

References

Amrhein, Chantal and Rico Sennrich. 2022. Identifying Weaknesses in Machine Translation Metrics Through Minimum Bayes Risk Decoding: A Case Study for COMET. *CoRR*, abs/2010.11125.

Bhattacharyya, Sumanta, Amirmohammad Rooshenas, Subhajit Naskar, Simeng Sun, Mohit Iyyer, and Andrew McCallum. 2021. Energy-based reranking: Improving neural machine translation using energy-based models. 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 4528–4537, Online, August. Association for Computational Linguistics.

Brown, Tom B., Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. *CoRR*, abs/2005.14165.

Bucilua, Cristian, Rich Caruana, and Alexandru Niculescu-Mizil. 2006. Model compression. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, page 535–541, New York, NY, USA. Association for Computing Machinery.

Conneau, Alexis, 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, July. Association for Computational Linguistics.

Devlin, Jacob, 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, June. Association for Computational Linguistics.

Eikema, Bryan and Wilker Aziz. 2021. Sampling-based minimum bayes risk decoding for neural machine translation. *CoRR*, abs/2108.04718.

Fan, Angela, Shruti Bhosale, Holger Schwenk, Zhiyi Ma, Ahmed El-Kishky, Siddharth Goyal, Mandeep Baines, Onur Celebi, Guillaume Wenzek, Vishrav Chaudhary, Naman Goyal, Tom Birch, Vitaliy Liptchinsky, Sergey Edunov, Edouard Grave, Michael Auli, and Armand Joulin. 2020. Beyond english-centric multilingual machine translation. *CoRR*, abs/2010.11125.

Fernandes, Patrick, Antonio Farinhas, Ricardo Rei, José G. C. de Souza, Perez Ogayo, Neubig Graham, and André F. T. Martins. 2022. Quality-Aware Decoding for Neural Machine Translation. In *Accepted at the 2022 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, Seattle, Washington, july. Association for Computational Linguistics.

Freitag, Markus, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021a. Experts, Errors, and Context: A Large-Scale Study of Human Evaluation for Machine Translation. *Transactions of the Association for Computational Linguistics*, 9:1460–1474, 12.

Freitag, Markus, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, George Foster, Alon Lavie, and Ondřej Bojar. 2021b. Results of the WMT21 metrics shared task: Evaluating metrics with expert-based human evaluations on TED and news domain. In *Proceedings of the Sixth Conference on Machine Translation*, pages 733–774, Online, November. Association for Computational Linguistics.

Glushkova, Taisiya, Chrysoula Zerva, Ricardo Rei, and André F. T. Martins. 2021. Uncertainty-aware machine translation evaluation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3920–3938, Punta Cana, Dominican Republic, November. Association for Computational Linguistics.

Goyal, Naman, Jingfei Du, Myle Ott, Giri Anantharaman, and Alexis Conneau. 2021. Larger-scale transformers for multilingual masked language modeling. In *Proceedings of the 6th Workshop on Representation Learning for NLP (RepL4NLP-2021)*, pages 29–33, Online, August. Association for Computational Linguistics.

Graham, Yvette, Timothy Baldwin, Alistair Moffat, and Justin Zobel. 2013. Continuous measurement scales in human evaluation of machine translation. In *Proceedings of the 7th Linguistic Annotation Workshop and Interoperability with Discourse*, pages 33–41, Sofia, Bulgaria, August. Association for Computational Linguistics.

Gupta, Manish and Puneet Agrawal. 2022. Compression of deep learning models for text: A survey. *ACM Trans. Knowl. Discov. Data*, 16(4), jan.

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

Kendall, M. G. 1938. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93.

Kocmi, Tom, Christian Federmann, Roman Grundkiewicz, Marcin Junczys-Dowmunt, Hitokazu Matsushita, and Arul Menezes. 2021. To ship or not to ship: An extensive evaluation of automatic metrics for machine translation. In *Proceedings of the Sixth Conference on Machine Translation*, pages 478–494, Online, November. Association for Computational Linguistics.

Lommel, Arle, Aljoscha Burchardt, and Hans Uszkoreit. 2014. Multidimensional quality metrics (MQM): A framework for declaring and describing translation quality metrics. *Tradumàtica: tecnologies de la traducció*, 0:455–463, 12.

Ma, Qingsong, 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, August. Association for Computational Linguistics.

Mathur, Nitika, 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, July. Association for Computational Linguistics.

Mathur, Nitika, 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, November. Association for Computational Linguistics.

Müller, Mathias and Rico Sennrich. 2021. Understanding the properties of minimum bayes risk decoding in neural machine translation. In *Proceedings of the Joint Conference of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL-IJCNLP 2021)*.

Ng, Nathan, Kyra Yee, Alexei Baevski, Myle Ott, Michael Auli, and Sergey Edunov. 2019. Facebook FAIR's WMT19 news translation task submission. In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 314–319, Florence, Italy, August. Association for Computational Linguistics.

Papineni, Kishore, 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, July. Association for Computational Linguistics.

Peters, Matthew E., 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, June. Association for Computational Linguistics.

Popović, Maja. 2015. chrF: character n-gram F-score for automatic MT evaluation. In *Proceedings of the Tenth Workshop on Statistical Machine Translation*, pages 392–395, Lisbon, Portugal, September. Association for Computational Linguistics.

Pu, Amy, Hyung Won Chung, Ankur Parikh, Sebastian Gehrmann, and Thibault Sellam. 2021. Learning compact metrics for MT. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 751–762, Online and Punta Cana, Dominican Republic, November. Association for Computational Linguistics.

Ramírez-Sánchez, Gema, Jaume Zaragoza-Bernabeu, Marta Bañón, and Sergio Ortiz-Rojas. 2020. Bifixer and Bicleaner: two open-source tools to clean your parallel data. In *Proceedings of the 22nd Annual Conference of the European Association for Machine Translation*, pages 291–298, Lisboa, Portugal, November. European Association for Machine Translation.

Rei, Ricardo, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. 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, November. Association for Computational Linguistics.

Reimers, Nils and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China, November. Association for Computational Linguistics.

Sanh, Victor, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. In *NeurIPS EMC2̂ Workshop*.

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

Sharir, Or, Barak Peleg, and Yoav Shoham. 2020. The cost of training NLP models: A concise overview. *CoRR*, abs/2004.08900.

Sun, Zhiqing, Hongkun Yu, Xiaodan Song, Renjie Liu, Yiming Yang, and Denny Zhou. 2020. MobileBERT: a compact task-agnostic BERT for resource-limited devices. In *Proceedings of the 58th Annual*

Meeting of the Association for Computational Linguistics, pages 2158–2170, Online, July. Association for Computational Linguistics.

Tenney, Ian, Dipanjan Das, and Ellie Pavlick. 2019. BERT rediscovers the classical NLP pipeline. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4593–4601, Florence, Italy, July. Association for Computational Linguistics.

Thompson, Brian 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, November. Association for Computational Linguistics.

Tiedemann, Jörg. 2012. Parallel data, tools and interfaces in OPUS. In *Proceedings of the Eighth International Conference on Language Resources and Evaluation (LREC'12)*, pages 2214–2218, Istanbul, Turkey, May. European Language Resources Association (ELRA).

Wang, Wenhui, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and Ming Zhou. 2020. Minilm: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. In Larochelle, H., M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 5776–5788. Curran Associates, Inc.

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

Zhu, Michael and Suyog Gupta. 2018. To prune, or not to prune: Exploring the efficacy of pruning for model compression. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Workshop Track Proceedings*. OpenReview.net.