Semantically-Informed Regressive Encoder Score

Vasiliy Viskov[1]*, George Kokush[2]*, Daniil Larionov[3]*, Steffen Egger[3], and Alexander Panchenko[1,4]
[1]Skoltech, [2]HSE University, [3]NLLG Group, Bielefeld University, [4]AIRI
{vasiliy.viskov, a.panchenko}@skol.tech {daniil.larionov, steffen.eger}@uni-bielefeld.de

Abstract

Machine translation is a natural language generation (NLG) problem that involves translating source text from one language to another. Like every task in the machine learning domain, it requires an evaluation metric. The most obvious one is human evaluation; however, it is expensive, time-consuming, and not easily reproducible automatically. In recent years, with the introduction of pretrained transformer architectures and large language models (LLMs), state-of-the-art results in automatic machine translation evaluation have significantly improved in terms of correlation with expert assessments. We introduce MRE-Score, which stands for seMantically-informed Regression Encoder Score. It is an approach that constructs an automatic machine translation evaluation system based on a regression encoder and contrastive pretraining for the downstream problem.

1 Introduction

WMT Metrics Shared Task (Freitag et al., 2022) is a machine learning competition where participants have to construct an automatic evaluation system for machine translation for several language pairs. For WMT23 Metrics Shared Task[1], three language pairs are considered: English-German (en-de), Chinese-English (zh-en), and Hebrew-English (he-en). For each source sentence, there is a corresponding target machine-translated text and a reference human translation. The main goals of this competition are:

1. To achieve the strongest correlation with human judgment of translation quality over a diverse set of machine translation systems.

2. To illustrate the suitability of an automatic evaluation metric as a surrogate for human evaluation.

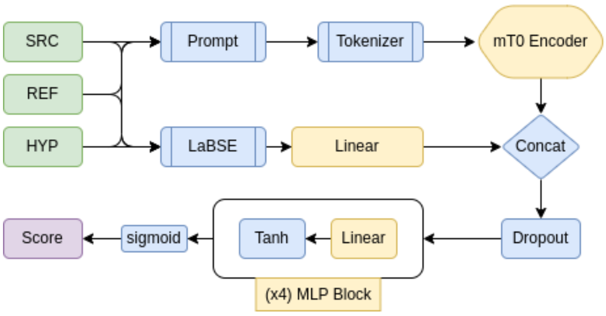

Figure 1: Final model architecture. Blocks in blue represents static components that were not trained. Blocks in yellow represent trained parts of the model.

3. To test the robustness of metrics when evaluating domains other than news data.

4. To create high-quality datasets for developing and evaluating metrics.

Within the WMT23 Metrics competition, our investigation focuses on the approach of constructing evaluation models to solve the regression problem based on expert degrees. Specifically, we construct regression models using a pretrained transformer encoder from the mT0 model family (Muennighoff et al., 2022), both vanilla models and with additional contrastive representation tuning. mT0 is finetuned version of mT5 (Xue et al., 2020), multilingual transformer model, which demonstrated capabilities of crosslingual generalization to unseen tasks and languages. Similar approaches demonstrated the best results in WMT21 (Freitag et al., 2021) and WMT22 (Freitag et al., 2022) Shared Tasks.

We release our code and pre-trained models openly to foster further research.[2]

2 Related Work

Evaluation metrics In NLG evaluation, one can differentiate four types of approaches for model

*Equal contribution
[1] https://wmt-metrics-task.github.io
[2] https://github.com/v-vskv-v/WMT23-MRE-Score

Proceedings of the Eighth Conference on Machine Translation (WMT), pages 815–821
December 6–7, 2023. ©2023 Association for Computational Linguistics

construction: (1) classical lexical overlap models, on the one hand, and LLM-based approaches based on (2) unsupervised matching, (3) regression, and (4) zero-shot prompting, on the other hand. Classical lexical overlap methods measure overlap between source, reference, and target sentence n-grams (Papineni et al., 2002; Lin, 2004; Banerjee and Lavie, 2005). Modern unsupervised matching-based approaches use large language model (LLM)-based encoders such as BERT to compute the semantic similarity between reference and hypothesis texts (Zhang et al., 2019; Zhao et al., 2019) or between source and hypothesis texts (Zhao et al., 2020). In modern regression approaches, models are fine-tuned to predict human evaluation scores. Generally, they consist of a transformer encoder model and a regression head. As input, they can use late binding with source, reference, and target texts or different concatenation combinations (Sellam et al., 2020; Rei et al., 2020). LLM-based zero-shot approaches use prompt engineering for LLMs with the expectation of a score in the generation output (Kocmi and Federmann, 2023). In some research, attempts are made to predict evaluation scores as a weighted sum of digit tokens, where the weights are token probabilities from a Markov chain probability model (Liu et al., 2023).

The previous winner of the WMT Metrics Shared Task competition was the proprietary MetricX(Freitag et al., 2022) model, which used a regression approach. One of the state-of-the-art models in machine translation is GPT-4 with zero-shot scoring. However, due to the time consumption of its inference and the closeness of regression approaches with relatively small backbones (e.g., COMET used the base version of XLM-RoBERTa (Conneau et al., 2019) with 2.5B parameters), task-specific NLG evaluation with sophisticated tricks with vector representation and training datasets may provide better results.

Contrastive Learning Contrastive learning for NLP problems is a popular pretraining approach for improving results in downstream tasks. For example, the recent E5 model (Wang et al., 2022) is pretrained in a contrastive manner using a curated large-scale text pair dataset to solve various tasks that require a single-vector representation of texts, both after finetuning and in a zero- or few-shot manner. Another work that investigated contrastive learning for extrapolating vector representations for different tasks is InstructOR (Su et al., 2022). This model incorporates instructions in contrastive learning and achieved good results for tasks that were unseen during pretraining. The idea of knowledge transfer in the latent space may provide improvements with clean datasets and an appropriate fitting process.

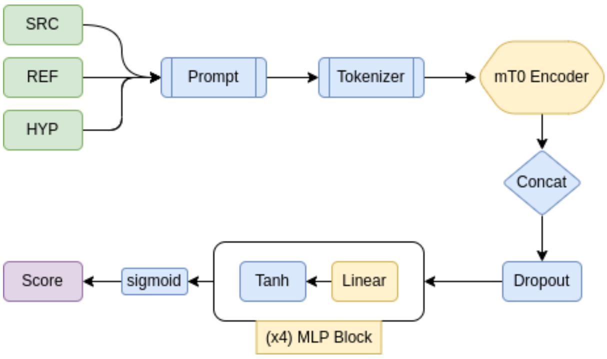

Figure 2: Architecture of the base model.

3 Method

3.1 Architecture

The main essence of our approach is to use vector representations from the encoder of the Big Science mT0 (Muennighoff et al., 2022) model as input for the Feed-Forward layer. This idea has already been proven successful in other approaches, such as COMET (Rei et al., 2020) and BLEURT (Sellam et al., 2020), and we have attempted to further improve upon it in various ways.

All our experiments use the same basic structure, as illustrated in Figure 2. First, the source text is presented as a prompt and is tokenized using the mT0 tokenizer. The resulting tokens are then processed by the mT0 encoder to obtain vector representations. Subsequently, mean pooling is applied to these vectors, and the resulting representations are passed through a Multilayer Perceptron (MLP). Prompting is necessary to present the data (source, reference, hypothesis) in a convenient format. In our approach, the prompt consists of concatenating the source, reference, and hypothesis with a separator token [sep]. It is worth noting that the mT0 tokenizer does not have a specific separator token, so another token can be selected for this purpose. Mean pooling is used to obtain sentence embeddings from the mT0 encoder representations, which are then suitable for further processing by a fully connected layer.

During the process, we tested various configurations of Multilayer Perceptron (MLP). We experimented with the number of linear layers, activation functions between layers and at the end, as well as with dropouts. Insufficient linear layers resulted in a deterioration of metrics as they were unable to extract all the necessary information from the embeddings. On the other hand, adding too many layers did not have any significant impact on the results. Hence, we settled on using 4 layers. We tested Tanh and ReLU as activation functions between layers, and found that Tanh yielded slightly better results. This observation is likely due to the fact that Tanh is commonly used in neural networks for working with text embeddings. For the activation function at the end, we compared two options: sigmoid and a simple threshold approach (rounding to 100 if the result is >100 and to 0 if < 0). The use of sigmoid resulted in significantly better outcomes. In addition, we experimented with the inclusion of dropouts and found that it made sense to add only one dropout before the first linear layer after mean pooling. Otherwise, too much useful information was deleted and the MLP did not have sufficient time to extract it.

We also explored the possibility of incorporating an external «hint» in addition to the embeddings of the mT0 encoder. This approach is illustrated in scheme 1. For obtaining additional representations, we chose LaBSE (Feng et al., 2020), a state-of-the-art model in the bitext mining task, which serves as a proxy task for machine translation. In our architecture, we included an additional component responsible for preprocessing LaBSE embeddings. After obtaining these representations, they were passed through a linear layer, and the resulting vectors were concatenated with the outputs from mT0 passed through Mean Pooling. The inclusion of the linear layer after LaBSE serves as an additional degree of freedom and helps reduce the dimensionality of the vectors.

In the end, we selected a configuration that utilized a pre-trained approach combining contrastive learning mT0 and LaBSE to submit our results. This configuration demonstrated the best metrics on our test data.

3.2 Contrastive pretraining

To enhance the vector representation and address the specific characteristics of the Hebrew language, which is not as widely studied as English or German, we experimented with tuning encoder embeddings using contrastive learning. For each source text, we created two contrastive loss components: one for the reference translation and one for the machine translation. To implement this approach, we needed to specify negative examples that we wanted to be dissimilar to the source text in terms of vector representations. We used the Sentence-T5 model (Ni et al., 2021) to embed each source text and its two translations. Additionally, we constructed two ANN (Approximate Nearest Neighbor) indexes (Johnson et al., 2019): one for human references and another for machine translations. These indexes allowed us to find the K furthest points from the source texts based on the dot product. Note that for normalized vectors:

$$\|x-y\|_2^2 = 2 - 2x^T y \;\longrightarrow$$

$$\min_x \left(x^T y\right) = \max_x \left((-x)^T y\right) = \min_x \left(\|(-x)-y\|_2^2\right) \tag{1}$$

The loss function is defined as the negative log likelihood with an arbitrary similarity function $\mathrm{sim}(x, y)$ (we used $\mathrm{sim}(x, y) =< x, y >$) and a temperature parameter τ. Our goal is to incorporate scaled target values of three types: SQM, DA, and MQM, with different prioritization weights in the loss function. For a given source text s, its reference translation r, and machine translation t, we have an expert degree $a^c_{s,t} \in [0, 1]$ of type c with a prioritization weight γ_c. Each source text s is embedded as e_s, the reference translation r is embedded as e^+_r, and the machine translation t is embedded as e^+_t. In the case of reference translations, we denote the K furthest points from s as $\left\{e^-_{r,k}\right\}_{k=1}^K$. Similarly, in the case of machine translations, we denote the K furthest points from s as $\left\{e^-_{t,k}\right\}_{k=1}^K$.

The component for human reference:

$$\mathbf{L}\left(e_s, e^+_r, \left\{e^-_{r,k}\right\}_{k=1}^K\right) = -\log p_{s,r} \tag{2}$$

$$p_{s,r} = \frac{\exp\left(\frac{\mathrm{sim}(e_s, e^+_r)}{\tau_r}\right)}{\exp\left(\frac{\mathrm{sim}(e_s, e^+_r)}{\tau_r}\right) + \sum_{k=1}^{K} \exp\left(\frac{\mathrm{sim}(e_s, e^-_{r,k})}{\tau_r}\right)} \tag{3}$$

This formula is general negative log-likelihood (NLL) with temperature for self-supervised learning (Wang and Isola, 2022).

The component for machine translation:

$$\mathbf{L}\left(e_s, e_t^+, \left\{e_{t,k}^-\right\}_{k=1}^K\right) = -\log p_{s,t,c} \tag{4}$$

$$p_{s,t,c} = \frac{\exp\left(\alpha_t \frac{\mathrm{sim}(e_s,e_t^+)}{\tau_t}\right)}{\exp\left(\alpha_t \frac{\mathrm{sim}(e_s,e_t^+)}{\tau_t}\right) + \sum_{k=1}^{K} \exp\left(\frac{\mathrm{sim}(e_s,e_{t,k}^-)}{\tau_t}\right)} \tag{5}$$

$$\alpha_t = \gamma_c a_{s,t}^c \tag{6}$$

Here we have a modified version of previous loss where we use target scores and their prior weights as temperature, but only for positive object.

Consider the derivative of the temperatured NLL loss w.r.t. to source text dot product as similarity function:

$$\frac{\left(1 - \frac{\exp\left(e_s^T e^+/\tau\right)}{Z(e_s)}\right)}{e^+\tau} - \sum_{e^-} \frac{\frac{\exp\left(e_s^T e^-/\tau\right)}{Z(e_s)}}{e^-\tau}$$

We have two separate additives with actually independent temperature coefficients. Increasing them removes the gradient changing effect and provides a pipeline for reducing the gradient step for bad and noisy translations. We can model such effect with human degrees with prioritizing ones over others.

Having a batch of quadruplets $\left\{\left(s, r, t, a_{s,t}^c\right)_n\right\}_{n=1}^N$ and using formulas above, the total loss can be written as:

$$\mathbf{L}\left(\left\{\left(s, r, t, a_{s,t}^c\right)_n\right\}_{n=1}^N\right) = \frac{1}{N}\sum_n \mathbf{L}\left(\left(s, r, t, a_{s,t}^c\right)_n\right) \tag{7}$$

$$\mathbf{L}\left(\left(s, r, t, a_{s,t}^c\right)_n\right) = \\ = \mathbf{L}\left(e_{s_n}, e_{r_n}^+, \left\{e_{r_n,k}^-\right\}_{k=1}^K\right) + \mathbf{L}\left(e_{s_n}, e_{t_n}^+, \left\{e_{t_n,k}^-\right\}_{k=1}^K\right) \tag{8}$$

Here we have an empirical risk over the batch, for each point we have two additive components for human reference and machine translation correspondingly.

3.3 Synthetic data

In this year's WMT Metrics Shared Tasks, the organizers presented us with a novel language pair: Hebrew-English. This language pair is not included in any of the available training data for MT evaluation metrics. Consequently, we believe that it was intended to test the ability of novel metrics for zero-shot transfer. To address this challenge, we made the decision to create a synthetic dataset for the Hebrew-English language pair, following the approach proposed by Rei et al. (2022b).

First, we selected a subset of English-Hebrew translations from the publicly available OPUS dataset (Tiedemann, 2012). From a total of approximately 1 million translations, we randomly chose 60,000 translations (Hebrew texts) and translated them back from Hebrew to English. To ensure a diverse range of translation quality, we selected three translation models of different sizes from the NLLB project (Costa-jussà et al., 2022): models with 600M and 1.3B parameters, which were distilled from 54B Mixture-of-Experts teacher models, as well as a 3.3B model that was trained from scratch. Each model was used to generate translations for an equally-sized portion of the dataset. Synthetic quality scores for these translations were computed as the average of scores calculated by the COMET-22 (Rei et al., 2022a) and BLEURT-20 (Sellam et al., 2020) metrics.

4 Experiments

4.1 Data

For our experiments, we utilize datasets from the previous year's WMT Metrics Shared Tasks as both training and evaluation data. These datasets provide three types of scores:

- MQM - Multidimensional Quality Metrics (Burchardt, 2013): This metric encompasses a wide range of issues that occur with translation.
- SQM - Scalar Quality Measure: This metric provides segment-level scalar ratings with document context.
- DA - Direct Assessment: This metric measures the quality of a translation on a scale from 0 to 100, based on the adequacy and fluency of the sentence.

We utilize all the available data and apply min-max scaling to rescale the score values, ensuring they fall within the range of 0 to 1. For DA and SQM scores, we used dataset-level statistics for scaling. However, for MQM scores, we adapted the scaling to accommodate different score ranges. Specifically, the English-German and Chinese-English pairs had a range of -25 to 0, while the English-Russian pair had a range of $-\infty$ to 100.

The resulting composition of the training dataset for our experiments is as follows:

- MQM scores for WMT competitions from the years 2020 and 2021, covering 3 language pairs (en-ru, zh-en, en-de).
- SQM scores for the year 2022, covering 12 language pairs.
- DA scores for the years 2017-2022, covering 41 language pairs.

For the test set, we selected the MQM scores for the year 2022 to ensure comparability with existing metrics.

Furthermore, we included synthetic data for the Hebrew-English language pair, as described in Section 3.3. Out of the total 60,000 examples, we randomly chose 50,000 examples for the training set and the remaining 10,000 examples for the test set. Since the scores for the synthetic data were computed using existing metric models, they naturally fell within the range of 0 to 1, and no additional re-scaling was required. In total, we had 1,527,567 examples in the training set and 77,575 in the test set.

4.2 Experimental settings

All experiments were conducted with a fixed random seed. For the base of the generic model, we chose the encoder part of the mT0-large model introduced in Muennighoff et al. (2022). An MLP on top of the encoder consists of three layers with hidden sizes of 384, 96, and 1, using the hyperbolic tangent activation function. We also apply dropout with a rate of $p = 0.1$. For models that utilize embeddings, we include a resizing dense layer that projects the concatenated embeddings vector into vectors with a size of 512.

For contrastive pretraining, we once again utilize the encoder part of the mT0-large model. Contrastive examples are collected into a total batch size of 128 examples. Furthermore, we accumulate batches across four iterations, resulting in an effective batch size of 512 for each training process.

Pipeline	en-de	zh-en	en-ru	he-en
Comet-22	0.281	0.395	0.330	NA
CometKiwi	0.266	0.343	0.297	NA
Base	0.276	0.179	0.350	0.796
Base + Emb.	0.255	0.173	0.331	0.785
CL Base	0.223	0.101	0.307	0.786
CL Base + Emb.	0.222	0.105	0.315	0.792

Table 1: Experimental results on WMT22 Test Set along with our synthetic test set for He-En. **Base** model represents model that only consits of mT0-large encoder and MLP head. **CL Base** represents model that was pretrained with contrastive loss before fine-tunning.

The first two models, which are based on the original mT0-large encoder, were trained for 3 epochs with an aggressive learning rate of 2×10^{-4}. The other two models, which utilize a contrastively-pretrained encoder, were trained for 1 epoch with a learning rate of 5×10^{-5}. In both cases, the batch size was set to 8 due to the substantially larger sequence sizes.

All our experiments were conducted in a distributed data-parallel setting across 4 GPUs. The learning rate was scaled accordingly based on the number of processes.

4.3 Hardware, Computational Budget and Environmental Impact

For our experiments, we utilized the CITEC computational cluster hosted at Bielefeld University. Each node in the cluster consists of 4xA40 GPUs with 48GB of VRAM, 1xAMD EPYC 7713 64-Core CPU, and 512GB of RAM.

The total computational budget for our experiments is 175 GPU-hours ($\tilde{4}$3.75 hours per node × 4 GPUs). Considering that the A40 GPU has a power draw of 300W under full load, and the current carbon intensity of the German power grid is 510gCO2eq/kWh [3], our estimated total carbon footprint is approximately 26.775 kgCO2eq. It is important to note that this number should be considered a lower bound, as we have not accounted for the power draw of other components of the computing node, such as the CPU and cooling.

5 Results and Discussion

We trained and tested each configuration on our test data using the Kendall-τ correlation metric. The results in Table 1 show that the base configuration has the best performance in most language pairs. However, adding external semantically-informed embeddings improves the quality for the model version with the contrastive loss. We didn't manage to get better results relatively to the base model, even for the rare language pair. We think that it's due to choice of negative sampling strategy, lack of theoretical approach analysis and hyperparameter tuning. Temperature is sensitive parameter, the wrong choice of it could lead to permanent overfitting and noisy results. We need to test more natural approach with adding scaled human degrees as general temperature for all softmax components. Also we should test other approaches with metric learning, e.g. Multi-Class N-pair loss (Sohn, 2016).

6 Conclusion

This paper presents our experiments with semantically informed architectures with a regression head. This led us to conclude that the additional awareness of the encoder and extra pretraining may positively affect the model quality in these conditions. In the future, it would be possible to explore other ways to inform the model and conduct experiments with larger versions of our implemented architectures.

Limitations

While we examine a novel approach to NLG evaluation, it is important to note limitations in our research.

Firstly, due to time and computational resource constraints, we have not conducted hyperparameter search. This opens up a possibility of finding better results for reported model configurations. Additionally, we have only made one experiment with one fixed random seed for each configuration. Increasing the number of runs would improve result stability.

Acknowledgements

The NLLG group is funded by the BMBF project «Metrics4NLG» and the DFG Heisenberg grant EG 375/5-1.

[3] Data obtained from https://app.electricitymaps.com/zone/DE on September 5, 2023

References

Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.

Aljoscha Burchardt. 2013. Multidimensional quality metrics: a flexible system for assessing translation quality. In *Proceedings of Translating and the Computer 35*, London, UK. Aslib.

Alexis Conneau, Kartikay Khandelwal, Naman Goyal, Vishrav Chaudhary, Guillaume Wenzek, Francisco Guzmán, Edouard Grave, Myle Ott, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Unsupervised cross-lingual representation learning at scale. *arXiv preprint arXiv:1911.02116*.

Marta R Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, et al. 2022. No language left behind: Scaling human-centered machine translation. *arXiv preprint arXiv:2207.04672*.

Fangxiaoyu Feng, Yinfei Yang, Daniel Cer, Naveen Arivazhagan, and Wei Wang. 2020. Language-agnostic bert sentence embedding. *arXiv preprint arXiv:2007.01852*.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, Eleftherios Avramidis, Tom Kocmi, George Foster, Alon Lavie, and André F T. Martins. 2022. Results of wmt22 metrics shared task: Stop using bleu – neural metrics are better and more robust. *Association for Computational Linguistics*.

Markus Freitag, Ricardo Rei, Nitika Mathur, Chi-kiu Lo, Craig Stewart, George Foster, Alon Lavie, and Ondřej Bojar. 2021. 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. Association for Computational Linguistics.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. 2019. Billion-scale similarity search with GPUs. *IEEE Transactions on Big Data*, 7(3):535–547.

Tom Kocmi and Christian Federmann. 2023. Large language models are state-of-the-art evaluators of translation quality.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. Gpteval: Nlg evaluation using gpt-4 with better human alignment. *arXiv preprint arXiv:2303.16634*.

Niklas Muennighoff, Thomas Wang, Lintang Sutawika, Adam Roberts, Stella Biderman, Teven Le Scao, M Saiful Bari, Sheng Shen, Zheng-Xin Yong, Hailey Schoelkopf, et al. 2022. Crosslingual generalization through multitask finetuning. *arXiv preprint arXiv:2211.01786*.

Jianmo Ni, Gustavo Hernández Ábrego, Noah Constant, Ji Ma, Keith B Hall, Daniel Cer, and Yinfei Yang. 2021. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. *arXiv preprint arXiv:2108.08877*.

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.

Ricardo Rei, José G. C. de Souza, Duarte Alves, Chrysoula Zerva, Ana C Farinha, Taisiya Glushkova, Alon Lavie, Luisa Coheur, and André F. T. Martins. 2022a. COMET-22: Unbabel-IST 2022 submission for the metrics shared task. In *Proceedings of the Seventh Conference on Machine Translation (WMT)*, pages 578–585, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Ricardo Rei, Ana C Farinha, José G.C. de Souza, Pedro G. Ramos, André F.T. Martins, Luisa Coheur, and Alon Lavie. 2022b. Searching for COMETINHO: The little metric that could. In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, pages 61–70, Ghent, Belgium. European Association for Machine Translation.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. Comet: A neural framework for mt evaluation. *arXiv preprint arXiv:2009.09025*.

Thibault Sellam, Dipanjan Das, and Ankur P Parikh. 2020. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*.

Kihyuk Sohn. 2016. Improved deep metric learning with multi-class n-pair loss objective. *Advances in neural information processing systems*, 29.

Hongjin Su, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, Tao Yu, et al. 2022. One embedder, any task: Instruction-finetuned text embeddings. *arXiv preprint arXiv:2212.09741*.

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

Liang Wang, Nan Yang, Xiaolong Huang, Binxing Jiao, Linjun Yang, Daxin Jiang, Rangan Majumder, and Furu Wei. 2022. Text embeddings by weakly-supervised contrastive pre-training. *arXiv preprint arXiv:2212.03533*.

Tongzhou Wang and Phillip Isola. 2022. Understanding contrastive representation learning through alignment and uniformity on the hypersphere.

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 preprint arXiv:2010.11934*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Wei Zhao, Goran Glavaš, Maxime Peyrard, Yang Gao, Robert West, and Steffen Eger. 2020. On the limitations of cross-lingual encoders as exposed by reference-free machine translation evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1656–1671, Online. Association for Computational Linguistics.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. 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 563–578, Hong Kong, China. Association for Computational Linguistics.