Reference-Free Word- and Sentence-Level Translation Evaluation with Token-Matching Metrics

Christoph Wolfgang Leiter
CS Department, TU Darmstadt
christoph.leiter@stud.tu-darmstadt.de

Abstract

Many modern machine translation evaluation metrics like BERTScore, BLEURT, COMET, MonoTransquest or XMoverScore are based on black-box language models. Hence, it is difficult to explain why these metrics return certain scores. This year's Eval4NLP shared task tackles this challenge by searching for methods that can extract feature importance scores that correlate well with human word-level error annotations. In this paper we show that unsupervised metrics that are based on token-matching can intrinsically provide such scores. The submitted system interprets the similarities of the contextualized word-embeddings that are used to compute (X)BERTScore as word-level importance scores. We make our code available[1].

1 Introduction

In recent years, machine translation evaluation metrics constantly improved in their correlation with human judgements (e.g. Mathur et al., 2020; Specia et al., 2020). However, this improvement comes at a loss of understandability. Early metrics such as BLEU (Papineni et al., 2002) and METEOR (Lavie et al., 2004; Banerjee and Lavie, 2005) follow a clearly defined algorithm without learnable weights. Therefore, these metrics are interpretable by design and could even be computed per hand. Newer metrics such BERTScore (Zhang et al., 2020), BLEURT (Sellam et al., 2020), COMET (Rei et al., 2020a), MonoTransquest (Ranasinghe et al., 2020a,b), MoverScore (Zhao et al., 2019) or XMoverScore (Zhao et al., 2020) instead leverage transformer (Vaswani et al., 2017) based language models. As these base their predictions on thousands of learned parameters, they are too complex to understand without employing further techniques. Such techniques that aim to support the understanding of black-box models are the scope of XAI (eXplainable Artificial Intelligence) (e.g. Carvalho et al., 2019; Bodria et al., 2021).

This year's Eval4NLP shared task (Fomicheva et al., 2021a) considers to what extend XAI techniques extract feature importance scores from metrics that correlate with word-level error annotations. Some embedding based metrics, such as MoverScore, XMoverScore and BERTScore can be categorized as *unsupervised matching* (Yuan et al., 2021). These metrics are unsupervised, as they are not fine-tuned on human annotated translation scores. And they perform *matching*, as the sentence-level score is calculated based on how well each token in one sentence matches to tokens in the other sentence.

This work evaluates the usage of the token-level matches of BERTScore and XMoverScore as feature-importance explanation of the sentence-level score. It was conducted as part of a master thesis by Leiter (2021).

2 Related Work

This system paper is related to work in the fields of machine translation evaluation metrics and explainable artificial intelligence.

2.1 Metrics

A large number of metrics has been proposed to grade the quality of machine translations (e.g. Mathur et al., 2020; Specia et al., 2020). Reference-based metrics grade machine translations based on one or more reference translations. Reference-free metrics grade machine translations based on the source sentence. Due to the structure of the shared task this paper considers reference-free metrics, in specific BERTScore (Zhang et al., 2020) with multilingual language embeddings (reference-free usage is proposed by Zhou et al., 2020; Song et al., 2021) and XMoverScore (Zhao et al., 2020). To differentiate, we will refer to the reference-free BERTScore

[1] https://github.com/Gringham/WordAndSentScoresFromTokenMatching

Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, pages 157–164

https://doi.org/10.26615/978-954-452-056-4_016

as *XBERTScore*. Other reference-free metrics are for example MonoTransquest (Ranasinghe et al., 2020a,b) and COMET for quality estimation (Rei et al., 2020b). Many reference-free metrics have been enabled by the pre-training of multilingual language models on large scale datasets. Examples are multilingual BERT (Devlin et al., 2018) and XLM-Roberta (Conneau et al., 2020). The discussed metrics produce a single score per translation. In contrast, word-level metrics such as the metrics by Lee (2020) and Ranasinghe et al. (2021) predict word-level errors. Word-level metrics are closely related to the goal of the Eval4NLP shared task, as the extracted feature importance scores are evaluated with word-level error annotations (Fomicheva et al., 2021a).

2.2 Explainable Artificial Intelligence

As summarized in related surveys (e.g. Carvalho et al., 2019; Lertvittayakumjorn and Toni, 2021; Linardatos et al., 2021), explainability techniques can be categorized along several dimensions. Intrinsic (self-explaining) models explain their output during the original computation, while post-hoc methods are applied afterwards. Model-agnostic techniques can be applied to any model, while model-specific techniques are specific to certain architectures. Also, global methods try to explain a model as a whole, while local methods give insights into single pairs of input/output.

The goal of the Eval4NLP shared task is the extraction of feature importance scores as word-level error indications (Fomicheva et al., 2021a), i.e. each input feature (here tokens) should be assigned a score of how important it is for a predicted output. As these are assigned per input, they can be counted towards the local techniques. Further, the methods proposed in this paper are intrinsic and model specific. Note that even though the model itself produces the explanation, i.e. a token level output, the approaches we present do not explain the internal workings of the underlying language model.

Other model-specific post-hoc feature importance methods are, for example, Integrated Gradients (Sundararajan et al., 2017), DiffMask (De Cao et al., 2020) and (Guan et al., 2019). Model-agnostic post-hoc feature importance methods are for example LIME (Ribeiro et al., 2016), SHAP (Lundberg and Lee, 2017) and Input Marginalization (Kim et al., 2020). Fomicheva et al. (2021b) present the first evaluation of explainabilitiy techniques in the same context as the shared task.

3 Feature Importance from Token-Matching

In this section we describe the extraction of word-level importance scores from XBERTScore and XMoverScore. In specific, we consider that words that are well aligned between source and translation are important for the sentence-level score and are likely to be correct translations. If a word does not align well, it is likely to be an error. Hence, the maximal similarity (or minimal dissimilarity) of each word between source and translation can be interpreted as word-level (importance) score. We choose $x = (x_1, ..., x_n)$ to represent a source sentence and $y = (y_1, ..., y_m)$ to represent a translation where x_i and y_j refer to arbitrary token embeddings in x and y.

3.1 XBERTScore

XBERTScore computes a reference-free sentence score as follows (Zhang et al., 2020; Zhou et al., 2020; Song et al., 2021):

1. A multilingual pre-trained transformer model is chosen and contextualized embeddings are extracted for each word in translation and source. These are obtained by performing a forward pass and extracting the hidden states at a layer of choice.

2. A matrix $S \in \mathbb{R}^{n \times m}$ of cosine similarities between each embedding of source and translation is constructed. In other words, entries in S are computed as $S_{ij} = \frac{x_i^T y_j}{||x_i|| \, ||y_j||}$.

3. Two vectors x_{max} and y_{max} are determined. x_{max} contains the maximum similarity of each token in x to tokens in y:

   $$x_{max} = (\max \ S_{1,*}, ..., \max \ S_{n,*})$$

   Respectively y_{max} contains the maximum similarity to each token in y to tokens in x:

   $$y_{max} = (\max \ S_{*,1}, ..., \max \ S_{*,m})$$

4. Zhang et al. (2020) propose three different scores: R_{BERT}, P_{BERT} and F_{BERT}. R_{BERT} computes the recall $R_{BERT} = mean(x_{max})$. P_{BERT} computes the precision $P_{BERT} = mean(y_{max})$. The F_{BERT}-Score is computed as $2\frac{P_{BERT} * R_{BERT}}{P_{BERT} + R_{BERT}}$.

System	Hypothesis			Source			
	AUC	AP	RtopK	AUC	AP	RtopK	Pearson
XBERTScore(XLMR)	0.741	0.600	0.485	0.734	0.579	0.448	0.520
XBERTScore(XLMR$_{NLI1}$)	0.772	0.640	0.523	0.753	0.606	0.475	0.575
XBERTScore(XLMR$_{NLI2}$)	0.757	0.628	0.518	0.747	0.597	0.464	0.575
XBERTScore(XLMR$_{Ensemble}$)	0.778	0.655	0.540	0.755	0.608	**0.481**	**0.582**
XBERTScore(mBERT)	0.673	0.506	0.396	0.683	0.514	0.377	0.303
XBERTScore(mBART)	0.648	0.504	0.392	0.664	0.503	0.378	0.255
XMoverScore(mBERT)	0.676	0.528	0.425	0.660	0.503	0.372	0.530
XMoverScore(mBERT)-KEEP	0.746	0.608	0.497	0.731	0.571	0.432	0.52
XMoverScore(XLMR$_{Ensemble}$)-KEEP	**0.781**	**0.658**	**0.544**	**0.759**	**0.609**	0.479	0.543
XMoverScore + SHAP (Baseline)	0.593	0.444	0.338	0.513	0.394	0.262	0.415

Table 1: Results on the et-en dev set of the shared task. Metrics for word level outputs are Area Under the Curve, Average Precision and Recall at top K. The sentence-level correlation to human judgements is denoted as Pearson.

System	Hypothesis			Source			
	AUC	AP	RtopK	AUC	AP	RtopK	Pearson
XBERTScore(XLMR)	0.818	0.685	0.507	0.779	0.599	0.466	0.742
XBERTScore(XLMR$_{NLI1}$)	0.837	0.710	0.584	0.798	0.632	0.514	0.765
XBERTScore(XLMR$_{NLI2}$)	0.828	0.705	0.589	0.801	**0.658**	**0.531**	0.763
XBERTScore(XLMR$_{Ensemble}$)	**0.848**	**0.730**	**0.615**	**0.808**	0.652	0.525	**0.770**
XBERTScore(mBERT)	0.777	0.613	0.491	0.749	0.567	0.433	0.645
XBERTScore(mBART)	0.738	0.587	0.473	0.738	0.591	0.474	0.556
XMoverScore(mBERT)	0.719	0.562	0.450	0.705	0.537	0.427	0.634
XMoverScore(mBERT)-KEEP	0.790	0.636	0.505	0.759	0.584	0.461	0.623
XMoverScore(XLMR$_{Ensemble}$)-KEEP	0.842	0.721	0.607	0.794	0.624	0.497	0.725
XMoverScore + SHAP (Baseline)	0.641	0.462	0.341	0.541	0.384	0.265	0.638

Table 2: Results on the ro-en dev set of the shared task. Metrics for word level outputs are Area Under the Curve, Average Precision and Recall at top K. The sentence-level correlation to human judgements is denoted as Pearson.

5. They describe further steps such as idf-weighting and rescaling of scores, which we don't apply in this paper. Idf-weighting over many sentences potentially increases the sentence level scores.

Zhang et al. (2020) compute R_{BERT} and P_{BERT} from embeddings in a single formula. In above's description we describe the construction of the matrix S and the vectors x_{max} and y_{max} as extra steps, as we interpret these vectors as token-level importance scores. To explain, we treat x_{max_i} as the importance score for embedding x_i in x (and the token at the i-th position of x), the same applying for y.

Many language-models use sub-word tokenization (e.g. Sentencepiece (Kudo and Richardson, 2018)), so that the importance-scores are at a sub-word level. To receive word-level scores, we parse the scored tokens to be aligned with the input sentences. Multiple scores that belong to a single word are averaged. If a token did not receive a score, e.g. as punctuation was dropped (see XMoverScore(mBERT) in section 4), we assign the score of the previous token.

To further improve the correlation to word-level error annotations, we ensemble word-level and sentence-level (F_{BERT}) scores by summing them across different models:

$$F_{ensemble} = \sum_{i=1}^{z} F_{BERT_i}$$

$$x_{max_{ensemble}} = \sum_{i=1}^{z} x_{max_i}$$

Here, F_{BERT_i} denotes the XBERTScore returned by using the i-th of z models to extract contextualized embeddings and $x_{max_{ensemble}}$ describes the element-wise sum of respective x_{max} vectors. Again, $x_{max_{ensemble_i}}$ is treated as importance score for embedding x_i in x. $y_{max_{ensemble}}$ is calculated analogous.

In section 4, the F-Score is evaluated in terms of its pearson correlation to sentence-level scores. $x_{max_{ensemble}}$ is evaluated in terms of its correlation to word-level error annotations of the source and $y_{max_{ensemble}}$ is evaluated in terms of its correlation to word-level error annotations of the hypothesis.

3.2 XMoverScore

Zhao et al. (2020) propose XMoverScore (XMS), a metric that matches n-grams of tokens based on the word mover's distance (WMD) (Kusner et al., 2015). In the case of unigrams, they first compute a matrix $C \in \mathbb{R}^{n \times m}$, with $C_{ij} = ||x_i - y_j||_2$. Then, based on C, they minimize the WMD to determine the optimal alignment between the two sentences.

Using the same notation as for XBERTScore, we obtain token-level scores as follows:

$$x_{min} = (\min\ C_{1,*}, ..., \min\ C_{n,*})$$

$$y_{min} = (\min\ C_{*,1}, ..., \min\ C_{*,m})$$

As for XBERTScore, we obtain word-level scores by aligning the token-level scores based on the input sentences. Again, word- and sentence-level scores can be ensembled via summation.

Zhao et al. (2020) further improve the sentence-level score by remapping the token-embeddings and employing a target-side language model. The remapping assumes that tokens in the cross-lingual embedding space are not fully aligned between languages. They propose two techniques for mitigation. Linear cross-lingual projection (CLP) learns a projection matrix that projects tokens of the source language such that the distance to tokens of the target language is minimized. Universal language mismatch-direction (UMD) determines a global direction along which the embeddings of two languages are misaligned. Then the projection along this direction is subtracted form each embedding. Both techniques use embeddings that were aligned using small parallel corpora. Zhao et al. (2020) employ the target-side language model as an additional measure of fluency of translations. In our experiments we do not use this model, as it might lower the degree to which the word-level scores explain the sentence-level scores.

3.3 Inversion

In the Eval4NLP shared task errors are considered as important for the sentence-level score (Fomicheva et al., 2021a), i.e. they should receive a higher feature-importance than correct words. Hence, we invert the word-level scores and use $-x_{max}$ and $-y_{max}$ for XBERTScore (likewise $-x_{min}$ and $-y_{min}$ for XMoverScore).

4 Experiment Setup

We calculate word- and sentence- level scores for the dev sets[2] of the Eval4NLP shared task (Fomicheva et al., 2021a), which are a subset of the MLQE-PE corpus by Fomicheva et al. (2020b,a). The organizers provide 1000 samples for the *ro-en* (Romanian-English) and *et-en* (Estonian-English) language pairs each. For every sample they provide a source sentence, a translation, a sentence-level ground truth score and word-level ground truth labels for source and translation. On the word-level they label a word with 1 if it is erroneous and 0 if it is correct.

Zhao et al. (2019) show that the usage of language models fine-tuned for Natural Language Inference (NLI) improves the results of MoverScore. Therefore, we evaluate models fine-tuned for NLI for XBertScore and XMoverScore. The results of the following configurations are reported:

- **XBERTScore(XLMR):** XBERTScore using the pre-trained XLMR-large model (Conneau et al., 2020).
- **XBERTScore(XLMR$_{NLI1}$):** XBERTScore using an XLMR-large model fine-tuned on XNLI (Conneau et al., 2018) from the Huggingface model hub[3].
- **XBERTScore(XLMR$_{NLI2}$):** XBERTScore using another XLMR-large model fine-tuned on XNLI (Conneau et al., 2018) and ANLI (Nie et al., 2020) from the Huggingface model hub[4].
- **XBERTScore(XLMR$_{Ensemble}$):** An ensemble version of the three models above that uses the ensembling step described in section 3.1.
- **XBERTScore(mBERT):** XBERTScore using multilingual BERT (Devlin et al., 2018) to extract contextualized embeddings.

[2] https://github.com/eval4nlp/SharedTask2021/tree/main/data/dev

[3] https://huggingface.co/joeddav/xlm-roberta-large-xnli

[4] https://huggingface.co/vicgalle/xlm-roberta-large-xnli-anli

- **XBERTScore(mBART):** XBERTScore using mBart-large 50 many-to-many (Tang et al., 2020).
- **XMoverScore(mBERT):** We report the scores for XMS[5] with unigrams and CLP remapping mode. XMS is based on the 12th layer of multilingual BERT.
- **XMoverScore(mBERT)-KEEP:** The original implementation of XMS by Zhao et al. (2020) drops embeddings of sub-words that are not the start of a word as well as punctuation. This configuration keeps them during the computation.
- **XMoverScore(XLMR$_{Ensemble}$)-KEEP:** XMS using the ensemble configuration described for XBERTScore above. Additionally, CLP and UMD mappings were trained on 30k sentences for each ensembled model and respective layer. The scores were summed across CLP and UMD mappings. Embeddings of punctuation and sub-words were kept.
- **XMoverScore+SHAP (Baseline):** A baseline copied from the shared task (Fomicheva et al., 2021a). The output score of XMS is explained with SHAP (Lundberg and Lee, 2017).

The result of (X)BERTScore by Zhang et al. (2020) depends on the choice of the layer to extract embeddings from. For the models already included in their library[6], we use the layers they tested perform best in a reference-based setting. For XLMR-NLI1 we choose layer 16 and for XMLR-NLI2 we choose layer 17, which we determined to perform best on a small subset of *et-en* data from the MLQE-PE corpus. Appendix A lists hashes produced by the BERTScore library that summarize the configurations. For XMoverScore(XLMR$_{Ensemble}$)-KEEP we choose the same layers.

The word-level scores are evaluated with Area Under the Curve (AUC), Recall at top K (RtopK) and Average Precision (AP) using the implementation by the organizers of the Eval4NLP shared task[7].

[5] https://github.com/AIPHES/ACL20-Reference-Free-MT-Evaluation/blob/master/score_utils.py

[6] https://github.com/Tiiiger/bert_score

[7] https://github.com/eval4nlp/SharedTask2021/blob/main/scripts/evaluate.py

5 Results

Table 1 and 2 show the results for the different configurations and language pairs. Metrics based on XLMR-large achieve the highest correlations. This is expected as it uses 24 layers in contrast to mBERT and mBART (encoder) with 12 layers. Also, the models fine-tuned for NLI perform better than the pre-trained XLMR model. Amongst all configurations, the XLMR-Ensembles perform best. Only for the AP and RtopK of the source in ro-en a single NLI model performed better. XMoverScore(mBERT)-KEEP achieves higher word-level scores than XBERTScore(mBERT), which indicates the successfulness of the applied remapping of embeddings. XMoverScore(mBERT) is worse at the word-level, as the scores of the dropped punctuation are inferred from the previous token. Further, XMoverScore(mBERT) being worse than XBERTScore(mBERT) on sentence-level might be caused by XMS using the 12th layer instead of the 9th. XMoverScore(XLMR$_{Ensemble}$)-KEEP, which also uses remappings, achieves slightly higher word-level correlations than XBERTScore(mBERT) for et-en but not for ro-en. This indicates that the applied remapping techniques are less effective for XLMR-large. Another interesting observation is that the sentence-level scores of XBERTScore with mBERT and mBART are much lower than the others for et-en, suggesting a weakness of these embeddings when compared with greedy matching rather than XMS's word mover's distance.

In the test-phase of the shared task we submitted XBERTScore(XLMR$_{Ensemble}$), which achieved its highest rank for the zero-shot language pair *ru-de* (Russian-German) and its lowest rank for *de-zh* (German-Chinese). For the latter one, the sentence scores even had a negative correlation. The cause of this remains to be investigated in the future.

6 Conclusion

In this paper we have evaluated XBERTScore and XMoverScore for word-level error annotations in a reference-free setup. The best reported configurations are based on multiple XLMR models. For future work it might be interesting to apply XLMR models that are remapped with novel cross-lingual alignment techniques. Also, it could be considered to incorporate the token-probabilities of the target-side language model of XMS into the word-level scores.

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, Ann Arbor, Michigan. Association for Computational Linguistics.

Francesco Bodria, Fosca Giannotti, Riccardo Guidotti, Francesca Naretto, Dino Pedreschi, and Salvatore Rinzivillo. 2021. Benchmarking and survey of explanation methods for black box models.

Diogo V. Carvalho, Eduardo M. Pereira, and Jaime S. Cardoso. 2019. Machine learning interpretability: A survey on methods and metrics. *Electronics*, 8(8).

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

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.

Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. 2020. How do decisions emerge across layers in neural models? interpretation with differentiable masking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3243–3255, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

Marina Fomicheva, Piyawat Lertvittayakumjorn, Wei Zhao, Steffen Eger, and Yang Gao. 2021a. The eval4nlp shared task on explainable quality estimation: Overview and results. In *Proceedings of the 2nd Workshop on Evaluation and Comparison of NLP Systems*.

Marina Fomicheva, Lucia Specia, and Nikolaos Aletras. 2021b. Translation error detection as rationale extraction.

Marina Fomicheva, Shuo Sun, Erick Fonseca, Frédéric Blain, Vishrav Chaudhary, Francisco Guzmán, Nina Lopatina, Lucia Specia, and André F. T. Martins. 2020a. MLQE-PE: A multilingual quality estimation and post-editing dataset. *arXiv preprint arXiv:2010.04480*.

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020b. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

Chaoyu Guan, Xiting Wang, Quanshi Zhang, Runjin Chen, Di He, and Xing Xie. 2019. Towards a deep and unified understanding of deep neural models in NLP. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2454–2463. Pmlr.

Siwon Kim, Jihun Yi, Eunji Kim, and Sungroh Yoon. 2020. Interpretation of NLP models through input marginalization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3154–3167, Online. Association for Computational Linguistics.

Taku Kudo and John Richardson. 2018. SentencePiece: A simple and language independent subword tokenizer and detokenizer for neural text processing. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 66–71, Brussels, Belgium. Association for Computational Linguistics.

Matt Kusner, Yu Sun, Nicholas Kolkin, and Kilian Weinberger. 2015. From word embeddings to document distances. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 957–966, Lille, France. Pmlr.

Alon Lavie, Kenji Sagae, and Shyamsundar Jayaraman. 2004. The significance of recall in automatic metrics for mt evaluation. In *Machine Translation: From Real Users to Research*, pages 134–143, Berlin, Heidelberg. Springer Berlin Heidelberg.

Dongjun Lee. 2020. Two-phase cross-lingual language model fine-tuning for machine translation quality estimation. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1024–1028, Online. Association for Computational Linguistics.

Christoph Wolfgang Leiter. 2021. Explaining machine translation metrics - application and assessment of explainability techniques in the domain of machine translation evaluation. Unpublished thesis. TU Darmstadt.

Piyawat Lertvittayakumjorn and Francesca Toni. 2021. Explanation-based human debugging of nlp models: A survey. *arXiv preprint arXiv:2104.15135*.

Pantelis Linardatos, Vasilis Papastefanopoulos, and Sotiris Kotsiantis. 2021. Explainable ai: A review of machine learning interpretability methods. *Entropy*, 23(1).

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

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

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial NLI: A new benchmark for natural language understanding. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4885–4901, 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.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020a. TransQuest at WMT2020: Sentence-level direct assessment. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1049–1055, Online. Association for Computational Linguistics.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2020b. TransQuest: Translation quality estimation with cross-lingual transformers. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5070–5081, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Tharindu Ranasinghe, Constantin Orasan, and Ruslan Mitkov. 2021. An exploratory analysis of multilingual word-level quality estimation with cross-lingual transformers. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 434–440, Online. Association for Computational Linguistics.

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.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Kdd '16, page 1135ÔÇô1144, New York, NY, USA. Association for Computing Machinery.

Thibault Sellam, 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. Association for Computational Linguistics.

Yurun Song, Junchen Zhao, and Lucia Specia. 2021. SentSim: Crosslingual semantic evaluation of machine translation. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3143–3156, Online. Association for Computational Linguistics.

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

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3319–3328. JMLR.org.

Yuqing Tang, Chau Tran, Xian Li, Peng-Jen Chen, Naman Goyal, Vishrav Chaudhary, Jiatao Gu, and Angela Fan. 2020. Multilingual translation with extensible multilingual pretraining and finetuning. *CoRR*, abs/2008.00401.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation.

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*.

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.

Lei Zhou, Liang Ding, and Koichi Takeda. 2020. Zero-shot translation quality estimation with explicit cross-lingual patterns. In *Proceedings of the Fifth Conference on Machine Translation*, pages 1068–1074, Online. Association for Computational Linguistics.

A BERTScore Hashes

The BERTScore library by Zhang et al. (2020) provides a function to generate hashes of the metric's configuration to allow better reproducibility[8]. Here we list the hashes of the configurations we used:

- **XBERTScore(XLMR):**
  xlm-roberta-large_L17_no-idf_version=0.3.10(hug_trans=4.4.0)
- **XBERTScore(XLMR_$NLI1$):**
  joeddav/xlm-roberta-large-xnli_L16_no-idf_version=0.3.10(hug_trans=4.4.0)
- **XBERTScore(XLMR_$NLI2$):**
  vicgalle/xlm-roberta-large-xnli-anli_L17_no-idf_version=0.3.10(hug_trans=4.4.0)
- **XBERTScore(XLMR_$Ensemble$):**
  - xlm-roberta-large_L17_no-idf_version=0.3.10(hug_trans=4.4.0)
  - joeddav/xlm-roberta-large-xnli_L16_no-idf_version=0.3.10(hug_trans=4.4.0)
  - vicgalle/xlm-roberta-large-xnli-anli_L17_no-idf_version=0.3.10(hug_trans=4.4.0)
- **XBERTScore(mBERT):**
  bert-base-multilingual-cased_L9_no-idf_version=0.3.10(hug_trans=4.4.0)
- **XBERTScore(mBART):**
  facebook/mbart-large-50-many-to-many-mmt_L12_no-idf_version=0.3.10(hug_trans=4.4.0)

[8] https://github.com/Tiiiger/bert_score