# Layer or Representation Space: What makes BERT-based Evaluation Metrics Robust?

**Doan Nam Long Vu**[1], **Nafise Sadat Moosavi**[2], **Steffen Eger**[3]

[1] Department of Computer Science, Technical University of Darmstadt, Germany
[2] Department of Computer Science, The University of Sheffield, UK
[3] NLLG, Faculty of Technology, Bielefeld University, Germany
doannamlong.vu@stud.tu-darmstadt.de

## Abstract

The evaluation of recent embedding-based evaluation metrics for text generation is primarily based on measuring their correlation with human evaluations on standard benchmarks. However, these benchmarks are mostly from similar domains to those used for pretraining word embeddings. This raises concerns about the (lack of) generalization of embedding-based metrics to new and noisy domains that contain a different vocabulary than the pretraining data. In this paper, we examine the robustness of BERTScore, one of the most popular embedding-based metrics for text generation. We show that (a) an embedding-based metric that has the highest correlation with human evaluations on a standard benchmark can have the lowest correlation if the amount of input noise or unknown tokens increases, (b) taking embeddings from the first layer of pretrained models improves the robustness of all metrics, and (c) the highest robustness is achieved when using character-level embeddings, instead of token-based embeddings, from the first layer of the pretrained model.[1]

## 1 Introduction

Evaluating the quality of generated outputs by Natural Language Generation (NLG) models is a challenging and open problem. Human judgments can directly assess the quality of generated texts (Popović, 2020; Escribe, 2019). However, human evaluation, either with experts or crowdsourcing, is expensive and time-consuming. Therefore, automatic evaluation metrics, which are fast and cheap, are commonly used alternatives for the rapid development of text generation systems (van der Lee et al., 2019). Traditional metrics such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), and ROUGE (Lin, 2004) measure $n$-gram overlap between generated and reference texts. While these metrics are easy to use, they cannot correctly assess generated texts that contain novel words or a rephrasing of the reference text.

Recent metrics like BERTScore (Zhang et al., 2020), MoverScore (Zhao et al., 2019), COMET (Rei et al., 2020), BARTScore (Yuan et al., 2021), and BLEURT (Sellam et al., 2020) adapt pretrained contextualized word embeddings to tackle this issue. These novel metrics have shown higher correlations with human judgments on various tasks and datasets (Ma et al., 2019; Mathur et al., 2020). However, the correlations are measured on standard benchmarks containing text domains similar to those used for pretraining the embeddings themselves. As a result, it is unclear how reliable these metrics are on domains and datasets containing words outside the vocabulary of the pretraining data.

The goal of this paper is to investigate the *robustness* of embedding-based evaluation metrics on new and noisy domains that contain a higher ratio of unknown tokens compared to standard text domains.[2] We examine the robustness of BERTScore, one of the most popular recent metrics for text generation.[3] In order to perform a systematic evaluation on the robustness of BERTScore with regard to the ratio of unknown tokens, we use character-based adversarial attacks (Eger and Benz, 2020) that introduce a controlled ratio of new unknown tokens to the input texts. Our contributions are:

- We investigate whether the use of character-based embeddings instead of token-based embeddings improves the robustness of embedding-

[1] The code of our experiments is available at https://github.com/long21wt/robust-bert-based-metrics

[2] We connect to recent research that investigates the behavior of metrics in adversarial situations (Sai et al., 2021; Kaster et al., 2021; Leiter et al., 2022; Zeidler et al., 2022).

[3] E.g., as of September 2022, BERTScore is cited ∼1200 times while it is ∼200 and 400 for MoverScore and BLEURT, respectively.

based generation metrics. Our results show that the evaluations based on character-level embeddings are more robust.

- We examine the impact of the hidden layer used for computing the embeddings in BERTScore. We show that the choice of hidden layer affects the robustness of the evaluation metric.
- We show that by using **character-level embeddings from the first layer**, we achieve the highest robustness, i.e., similar correlation with human evaluations for different ratios of unknown tokens.

## 2 BERTScore

BERTScore (Zhang et al., 2020) computes the pairwise cosine similarity between the reference and hypothesis using contextual embeddings. It forward-passes sentences through a pretrained model, i.e., BERT (Devlin et al., 2019), and extracts the embedding information from a specific hidden layer. To select the best hidden layer, BERTScore uses average Pearson correlation with human scores on WMT16 (Bojar et al., 2016) over five language pairs. For instance, the best layer is the ninth layer for $\text{BERT}_{\text{base-uncased}}$.

**BERTScore with character-level embeddings.** Existing embedding-based metrics, including BERTScore, use token-based embeddings that are taken from pretrained models like BERT (Devlin et al., 2019). In this paper, we investigate the impact of using character-level embeddings instead of token-level embeddings in BERTScore (Zhang et al., 2020). We use ByT5 (Xue et al., 2021), which encodes the input at the byte level. It tokenizes a word into a set of single characters or converts it directly to UTF-8 characters before forwarding the input sequence into the model. Xue et al. (2021) show that ByT5 is more robust to noise compared to word-level embeddings. For computing BERTScore using character-level embeddings, we use ByT5 instead of BERT in BERTScore computations. We adapt three variants of ByT5 (small, base, large) in BERTScore. Table 1 presents the best layer of ByT5 models for computing BERTScore.

| Model | Best Layer | Score |
|---|---|---|
| ByT5-small | 1 | 0.510 |
| ByT5-base | 17 | 0.581 |
| ByT5-large | 30 | 0.615 |

Table 1: Best layers with different ByT5 variants and their average Pearson correlation score on WMT16.

| Language Pairs | No. Segment Sample (DARR) |
|---|---|
| de-en (German-English) | 85365 |
| fi-en (Finnish-English) | 38307 |
| gu-en (Gujarati-English) | 31139 |
| kk-en (Kazakh-English) | 27094 |
| lt-en (Lithuanian-English) | 21862 |
| ru-en (Russian-English) | 46172 |
| zh-en (Chinese-English) | 31070 |

Table 2: To-English language pairs of WMT19. DARR denotes *Direct Assessment Relative Ranks*, in which all available sentence pairs of DA (Direct Assessment) scores are taken into account.

## 3 Experimental settings

### 3.1 Evaluation on a standard benchmark

We report the results on the WMT19 dataset (Ma et al., 2019) that contains seven to-English language pairs. Each language pair has 2800 sentences, each corresponding to one reference, plus the systems' output sentences. Totally, the human evaluation in WMT19 has 281k segment sample scores for each of the output translation in to-English language pairs. Table 2 shows the language pairs considered, as well as the number of segments per language pair.

### 3.2 Evaluating Robustness

**Evaluation on different ratios of unknown tokens.** To evaluate the robustness of evaluation metrics on new domains, we use character-level attacks to introduce a controlled ratio of unknown tokens in the corresponding reference texts of the evaluation sets.[4] We examine five different attacks from Eger and Benz (2020): (a) **intruders:** inserting a character—e.g., '.', '/', ':'—in between characters of a word, (b) **disemvoweling**: removing vowels—e.g., 'a', 'e', 'i'—from the word, (c) **keyboard typos**: randomly replacing letters of a word with characters that are nearby the original characters on an English keyboard, (d) **phonetic**: changing a word's spelling in such a way that its pronunciation stays the same, and (e) **visual**: replacing characters with a symbol that is its visually nearest neighbor (Eger et al., 2019). We can control

[4] We need human annotations for evaluating the correlation of evaluation metrics with human judgments, and such annotations are available for standard domains like WMT datasets. As a result, we introduce unknown tokens by using character-level attacks to artificially introduce more unknown tokens.

| Setting | Sentence |
|---|---|
| no-attack | Now they have come to an agreement. |
| intrude | Now they have c/o/me t+o a>n agreement. |
| disemvowel | Nw thy have come to an grmnt. |
| keyboard-typo | No3 they have come to xn agrrement. |
| phonetic | Nau they have cohm to an agrimand. |
| visual | Now thẸŸ hẪve come to ȃ aȥʁèȨmȆnţ. |

Table 3: Examples for the character-level attacks (Eger and Benz, 2020; Keller et al., 2021) at perturbation level $p = 0.3$, i.e., the probability that each letter in a sentence is attacked is 0.3.

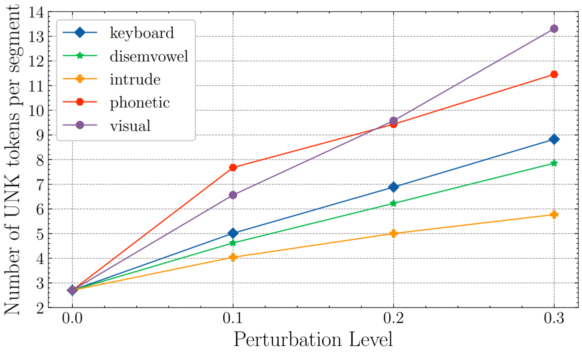


Figure 1: The number of average unknown tokens per segment across seven to-English language pairs in WMT19 given different attacks and perturbation levels.

the ratio of tokens that are affected by the adversarial attack by the *perturbation level* ($p$), e.g., $p = 0$ denotes no attack and $p = 0.3$ indicates that each letter in the sentence is attacked by the probability of 0.3. Table 3 shows an example of each of these attacks at $p = 0.3$.

Figure 1 shows the average number of unknown tokens, as determined based on BERT's tokenizer, per segment across seven to-English language pairs given different attacks and perturbation levels. We count a token as an unknown token if (1) BERT represents it as [UNK], or (2) BERT splits it into subwords, e.g., *'pre-trained'* to *'pre', '##train', '##ed'*.[5] As we see from the figure, the number of unknown tokens increases as we apply these character-level attacks with higher perturbation levels. In our experiments in Section 4, we report the results using visual attacks. The results using other attacks are also reported in Appendix B, and they follow the same patterns as those using the visual attack.

**Evaluation on low-resource language pairs.** Apart from the experiments on WMT19, we also perform the evaluations on the (Xhosa, Zulu) and (Bengali, Hindi) language pairs from WMT21 (Freitag et al., 2021). BERTScore uses multilingual BERT for evaluating non-English languages. Multilingual models contain a higher ratio of unknown tokens for low-resource languages, and therefore, evaluating the correlation of embedding-based metrics with human judgments on low-resource languages is also an indicator of their robustness. Table 4 shows the number of unknowns tokens per segment to multilingual BERT in four different low-resources language pairs in WMT21 dataset. We refer to the number of segments of low-resources dataset in Table 7 in Appendix C.

| Language pair | No. unknown tokens |
|---|---|
| bn-hi (Bengali-Hindi) | 19.235 |
| hi-bn (Hindi-Bengali) | 23.478 |
| xh-zu (Xhosa-Zulu) | 28.930 |
| zu-xh (Zulu-Xhosa) | 28.743 |

Table 4: The number of average unknown tokens per segment for each language pair in our low-resource datasets.

## 4 Experiments

### 4.1 Impact of Character-level Embeddings

Table 5 shows the results of BERTScore using different embeddings on WMT19's to-English language pairs (using $p = 0$). Figure 2 shows the average correlation score over all seven to-English language pairs given different perturbation level from $p = 0$ to $p = 0.3$ using the visual attack.

We observe that computing BERTScore using the ByT5-small models results in a slightly lower average correlation with human scores over the seven to-English pairs at $p = 0$ compared to BERTScore using BERT and larger ByT5 models.

However, the average correlation using ByT5-small remains around the same value given different ratio of unknown tokens, indicating higher *robustness* of the metrics using ByT5-small. On the other hand, while using BERT-large embeddings results in the highest average correlation with human scores in Table 5, its correlation drops considerably in the presence of more unknown tokens in Figure 2.

For Hindi-Bengali and Zulu-Xhosa, we compare the results against using the BERT-base-multilingual model in Table 6. We observe that the BERTScore metric that uses ByT5-small achieves

[5] Please refer to the detailed algorithm in Appendix A.

|  | de-en | fi-en | gu-en | kk-en | lt-en | ru-en | zh-en | Average |
|---|---|---|---|---|---|---|---|---|
| BERT-base | 0.180 | 0.339 | 0.288 | 0.438 | 0.364 | 0.209 | 0.410 | 0.318 |
| BERT-large | 0.194 | **0.346** | 0.292 | **0.442** | **0.375** | 0.208 | **0.418** | **0.325** |
| ByT5-small | 0.172 | 0.286 | 0.278 | 0.422 | 0.307 | 0.194 | 0.373 | 0.290 |
| ByT5-base | **0.197** | 0.326 | 0.297 | 0.419 | 0.358 | **0.215** | **0.418** | 0.319 |
| ByT5-large | 0.193 | 0.333 | **0.304** | 0.427 | 0.354 | 0.208 | 0.415 | 0.319 |

Table 5: Segment-level Kendall correlation results for to-English language pairs in WMT19 without any attack, i.e. $p = 0$. The correlation of BERTScore with human are reported using different embeddings including bert-base-uncased, bert-large-uncased, ByT5-small, ByT5-base, and ByT5-large.

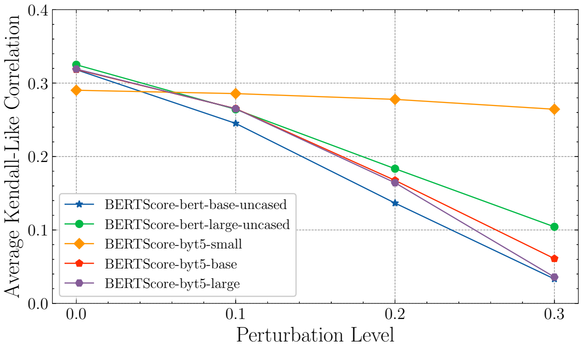


Figure 2: Average Kendall correlation of 7 to-English language pairs in WMT19 given different perturbation level from $p = 0.0$ to $p = 0.3$ using the visual attack.

| Model | bn-hi | hi-bn | xh-zu | zu-xh |
|---|---|---|---|---|
| BERT-multi | 0.073 | 0.364 | 0.266 | 0.488 |
| ByT5-small | **0.096** | **0.411** | **0.311** | **0.523** |

Table 6: Kendall correlation scores of BERTScore for WMT21 low-resource language pairs Hindi-Bengali and Zulu-Xhosa using BERT-base-multilingual and ByT5-small embeddings.

higher correlations with humans throughout. Given that low resources languages contain more out-of-vocabulary words for pretrained models, this observation confirms our previous results using character-level attacks on the WTM19 dataset.

### 4.2 Impact of the Selected Hidden Layer

Our results in Section 4.1 show the robustness of BERTScore when using the ByT5-small model for computing the embeddings. However, as Table 1 shows, the selected hidden layer for getting embeddings varies when using different pretrained models. For instance, when using ByT5-small embeddings, the model uses the embeddings of the first layer while it uses the embeddings of the 30th layer for ByT5-large. Zhang et al. (2020) show that BERTScore correlation scores with humans drop as they select the last few layers of BERT for getting the embeddings. Therefore, the robustness of examined metrics may also depend on their corresponding selected layers for computing embeddings.

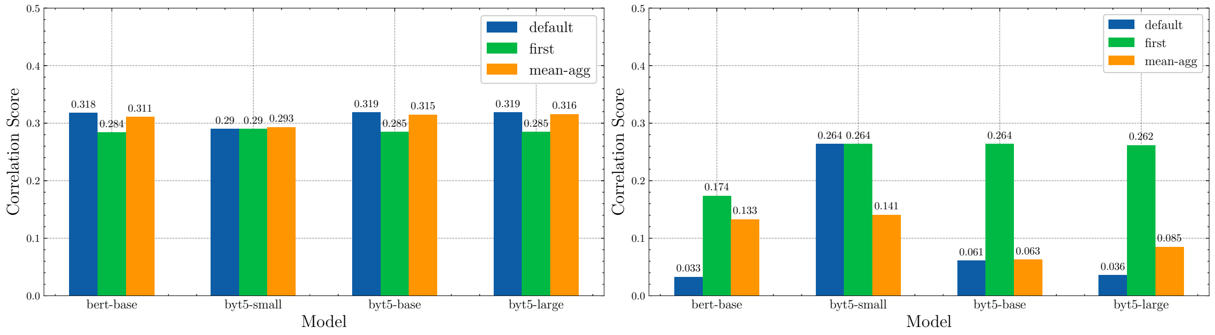
In this section, we evaluate the impact of the selected hidden layer on the robustness of the metric. We evaluate three settings where we use: (a) the embeddings of the first layer for all models, (b) the embeddings of the best layer for each model (cf. Table 3), and (c) the mean of aggregated embeddings over all layers. We perform the robustness evaluations using the visual attack at $p = 0.3$. Figure 3 shows the average results of this experiment[6]. We make the following observations.

First, using the embeddings of the first layer closes the gap between the correlations of different variations of the ByT5 model, i.e., small, base, and large, in the presence of more unknown tokens, i.e., $p = 0.3$.

Second, using the embeddings of the first layer improves the robustness of BERTScore using BERT embeddings, i.e., improving the correlation from 0.033 to 0.174 for BERT-base given $p = 0.3$. However, the correlation of the resulting BERTScore is still considerably lower than using ByT5 embeddings at the presence of more unknown tokens. This indicates that **both** the choices of the hidden layer as well as the pretrained model play an important role in the robustness of the resulting embedding-based metric. A reason why the first layer may be more effective in our setup is that, in the presence of input noise or unknown tokens, embeddings of higher layers may become less and less meaningful, as the noise may propagate and accumulate along layers. We provide an example from the similarity matrix of the resulting embeddings for different layers in Figure 5 in the Appendix E.

Overall, our results indicate that optimizing the layer on a standard data set such as WMT16 may

[6] In Table 8 and 9 in Appendix D, we report scores for each language pair.



(a) No attack

(b) Visual attack at $p = 0.3$

Figure 3: Average segment-level Kendall correlation results for seven to-non-English language pairs in WMT19 to fist layer, default layer, and mean of aggregated embeddings setting in BERTScore.

be suboptimal in terms of the generalization of the resulting metrics. Concerning efficiency of the resulting metrics (a core aspect of modern NLP (Moosavi et al., 2020)), BERT-base has 110 million parameters, while ByT5-small has 300 million parameters. With the default BERTScore setting, passing the input through 9 layers results in a longer inference time. However, using the embeddings of the first layer results in a very fast inference for both models.

## 5 Conclusion

Embedding-based evaluation metrics will be used across different tasks and datasets that may contain data from very different domains. However, such metrics are only evaluated on standard datasets that contain similar domains as those used for pretraining embeddings. As a result, it is not clear how reliable the results of such evaluation metrics will be on new domains. In this work, we investigate the robustness of embedding-based metrics in the presence of different ratios of unknown tokens. We show that (a) the results of the BERTScore using BERT-based embeddings is not robust, and its correlation with human evaluations drops significantly as the ratio of unknown tokens increases, and (b) using character-level embeddings from the first layer of ByT5 significantly improves the robustness of BERTScore and results in reliable results given different ratios of unknown tokens. We encourage the community to use this setting for their embedding-based evaluations, especially when applying the metrics to less standard domains.

In future work, we aim to address other aspects of robustness of evaluation metrics beyond an increased amount of unknown tokens as a result of spelling variation, such as how metrics cope with varying factuality (Chen and Eger, 2022) or with fluency and grammatical acceptability issues (Rony et al., 2022). We also plan to investigate the impact of pixel-based representations (Rust et al., 2022) (which are even more lower-level) for enhancing the robustness of evaluation metrics.

## Acknowledgments

We thank the Ubiquitous Knowledge Processing (UKP) Lab and the system administrators of the lab for providing the computing resources to perform the experiments. We thank the anonymous reviewers for their comments and suggestions that improved the final version of the paper.

## 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.

Ondřej Bojar, Yvette Graham, Amir Kamran, and Miloš Stanojević. 2016. Results of the WMT16 metrics shared task. In *Proceedings of the First Conference on Machine Translation: Volume 2, Shared Task Papers*, pages 199–231, Berlin, Germany. Association for Computational Linguistics.

Yanran Chen and Steffen Eger. 2022. Menli: Robust evaluation metrics from natural language inference. *ArXiv*, abs/2208.07316.

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

Steffen Eger and Yannik Benz. 2020. From hero to zéroe: A benchmark of low-level adversarial attacks. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing, AACL/IJCNLP 2020, Suzhou, China, December 4-7, 2020*, pages 786–803. Association for Computational Linguistics.

Steffen Eger, Gözde Gül Şahin, Andreas Rücklé, Ji-Ung Lee, Claudia Schulz, Mohsen Mesgar, Krishnkant Swarnkar, Edwin Simpson, and Iryna Gurevych. 2019. Text processing like humans do: Visually attacking and shielding NLP systems. 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 1634–1647, Minneapolis, Minnesota. Association for Computational Linguistics.

Marie Escribe. 2019. Human evaluation of neural machine translation: The case of deep learning. In *Proceedings of the Human-Informed Translation and Interpreting Technology Workshop (HiT-IT 2019)*, pages 36–46, Varna, Bulgaria. Incoma Ltd., Shoumen, Bulgaria.

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.

Marvin Kaster, Wei Zhao, and Steffen Eger. 2021. Global explainability of BERT-based evaluation metrics by disentangling along linguistic factors. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 8912–8925, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Yannik Keller, Jan Mackensen, and Steffen Eger. 2021. BERT-defense: A probabilistic model based on BERT to combat cognitively inspired orthographic adversarial attacks. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1616–1629, Online. Association for Computational Linguistics.

Chris van der Lee, Albert Gatt, Emiel van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368, Tokyo, Japan. Association for Computational Linguistics.

Christoph Leiter, Piyawat Lertvittayakumjorn, M. Fomicheva, Wei Zhao, Yang Gao, and Steffen Eger. 2022. Towards explainable evaluation metrics for natural language generation. *ArXiv*, abs/2203.11131.

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

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

Nitika Mathur, 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.

Nafise Sadat Moosavi, Angela Fan, Vered Shwartz, Goran Glavaš, Shafiq Joty, Alex Wang, and Thomas Wolf, editors. 2020. *Proceedings of SustaiNLP: Workshop on Simple and Efficient Natural Language Processing*. Association for Computational Linguistics, Online.

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.

Maja Popović. 2020. Informative manual evaluation of machine translation output. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 5059–5069, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Ricardo Rei, 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. Association for Computational Linguistics.

Md Rashad Al Hasan Rony, Liubov Kovriguina, Debanjan Chaudhuri, Ricardo Usbeck, and Jens Lehmann. 2022. RoMe: A robust metric for evaluating natural language generation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5645–5657, Dublin, Ireland. Association for Computational Linguistics.

Phillip Rust, J.F. Lotz, Emanuele Bugliarello, Elizabeth Salesky, Miryam de Lhoneux, and Desmond Elliott. 2022. Language modelling with pixels. *ArXiv*, abs/2207.06991.

Ananya B. Sai, Tanay Dixit, Dev Yashpal Sheth, Sreyas Mohan, and Mitesh M. Khapra. 2021. Perturbation CheckLists for evaluating NLG evaluation metrics. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7219–7234, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Linting Xue, Aditya Barua, Noah Constant, Rami Al-Rfou, Sharan Narang, Mihir Kale, Adam Roberts, and Colin Raffel. 2021. Byt5: Towards a token-free future with pre-trained byte-to-byte models.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34.

Laura Zeidler, Juri Opitz, and Anette Frank. 2022. A dynamic, interpreted CheckList for meaning-oriented NLG metric evaluation – through the lens of semantic similarity rating. In *Proceedings of the 11th Joint Conference on Lexical and Computational Semantics*, pages 157–172, Seattle, Washington. Association for Computational Linguistics.

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, 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.

**Algorithm 1:** Count UNK token in a BERT tokenized sentence

```
def count_UNK:
    Data: sentence: a tokenized sentence as a list of string
    output: count: number of UNK token of input tokenized sentence
    count $\longleftarrow$ 0
    buffer $\longleftarrow$ empty list
    for token in sentence do
        if [UNK] in token then
            count $\longleftarrow$ count $+1$
        else if ## in token then
            Add token to buffer
        else
            if len(buffer) != 0 then
                count $\longleftarrow$ count $+1$
                Empty buffer
            end
        end
    end
    if len(sentence) $\geq$ 2 then
        if ## in last token of sentence then
            count $\longleftarrow$ count $+1$
        end
    end
    return count
```

## A Counting UNK token

Algorithm 1 shows how we count UNK tokens that the BERT tokenizer creates from a sentence. In BERT, `[UNK]` represents the UNK tokens that are not in their given vocabulary. Besides `[UNK]`, BERT use WordPiece tokenizer concept, which breaks the unknown word into sub-words using a greedy longest-match-first algorithm, such as splits "*bassing*" into '*bass*' and '*##ing*' where '*##*' denotes the join of sub-words. Thus, the UNK word becomes two known words. '*##*' is the indication for the starting of a UNK word if the previous token does not contain '*##*'. In case the next token still contains '*##*', it indicates that the token still belongs to a word and does not count as a UNK token, e.g., "*verständlich*" to *'vers', '##tä', '##nd', '##lich'* and count it as one UNK token. It lasted until we finally found non contain '*##*' token. With a word-piece tokenizer, the beginning token of a tokenized sentence is either `[UNK]` or known word, and we also consider the case where the last token contains "*##*".

| Language Pair | No. Segment |
|---|---|
| bn-hi (Bengali → Hindi) | 4,461 |
| hi-bn (Hindi → Bengali) | 4,512 |
| xh-zu (Xhosa → Zulu) | 2,952 |
| zu-xh (Zulu → Xhosa) | 2,502 |

Table 7: Amount of segments in WMT21 for Hindi ⟷ Bengali and Zulu ⟷ Xhosa.

## B WMT19

The results of other attacks are illustrated in Figure 4.

## C FLORES

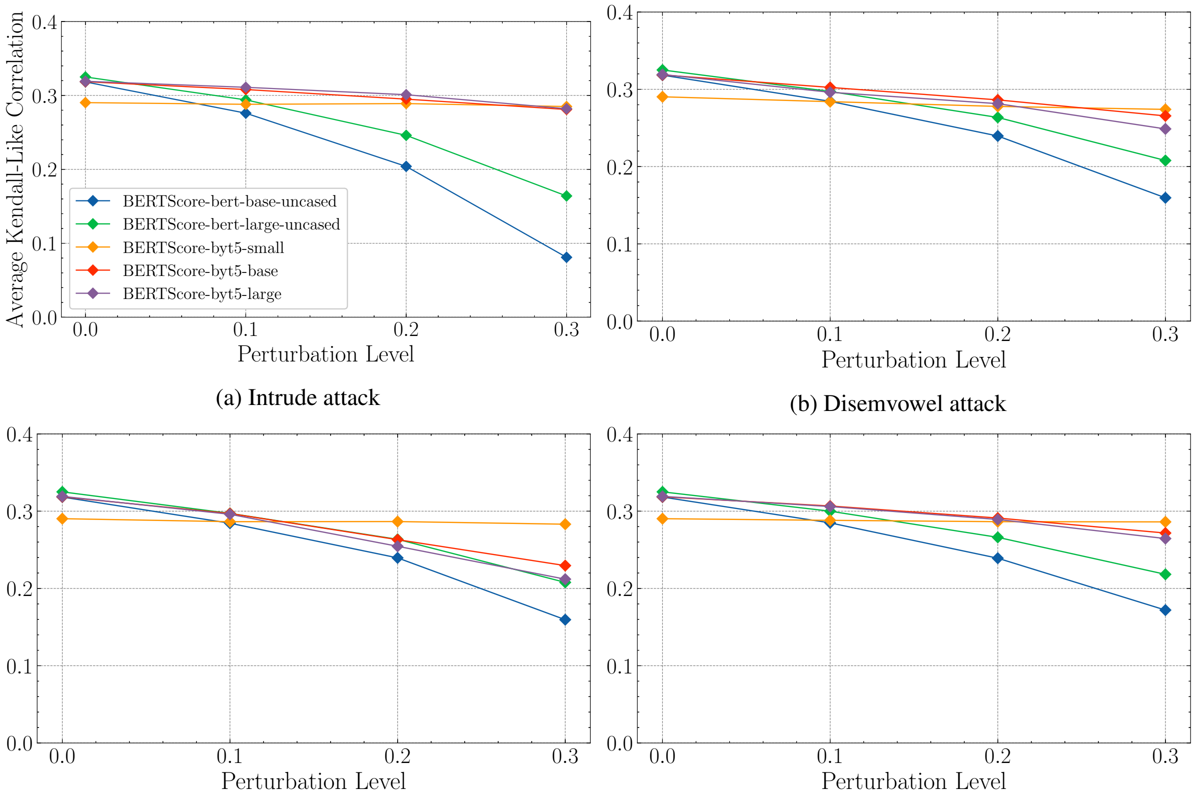
Table 7 shows the number of provided human annotations in FLORES.

## D Impact of layer choice in BERTScore

Table 8 and 9 show the particular results of each language pair with different settings in BERTScore without attack and with visual attack at $p = 0.3$ respectively.

## E Effectiveness of the first layer

In Figure 5, we show four different settings and their cosine similarity matrix computed by BERTScore using bert-base-uncased. In both *normal reference* with 1st or 9th setups, matched tokens get higher similarity score. 9th layer setting gathers information for relevant tokens, which results in higher similarity score across the matrix. As in the case with *attacked reference*, 1st layer setting penalizes the unmatched tokens and the magnitude of matched tokens are as high as using *normal reference* with 1st layer setup. However, by using 9th layer for *attacked reference*, we can observe the hue color of matched tokens with low score. Thus, we conclude the accumulated noise to higher layer cause the problem with effectiveness in our previous setup with WMT19 dataset.



(a) Intrude attack

(b) Disemvowel attack

(c) Phonetic attack

(d) Keyboard-typo attack

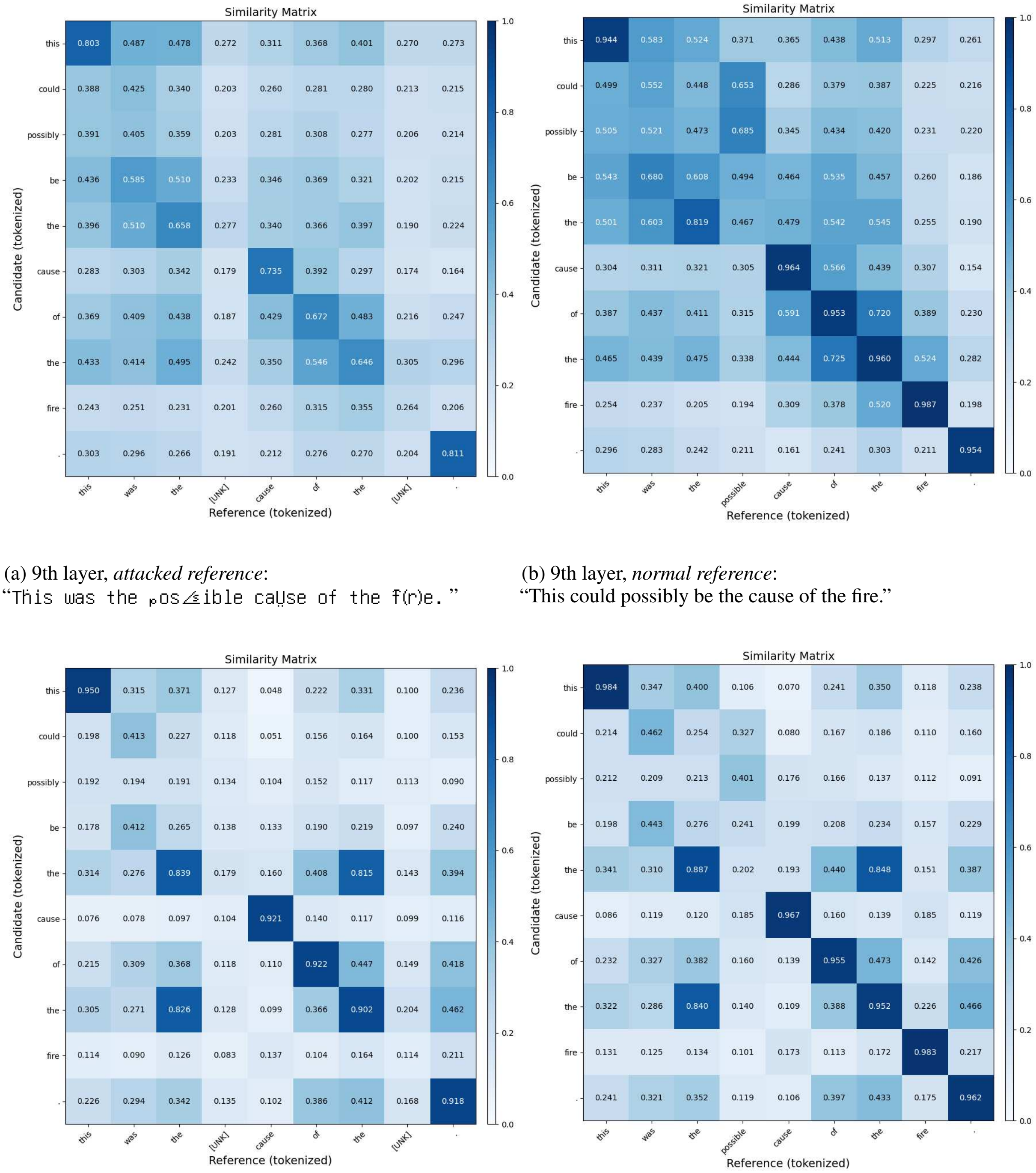
Figure 4: Average Kendall correlation of seven to-English language pairs in WMT19 under attack with perturbation level from $p = 0.0$ to $p = 0.3$

| Setting | Metric | de-en | fi-en | gu-en | kk-en | lt-en | ru-en | zh-en | Average |
|---|---|---|---|---|---|---|---|---|---|
| Default | BERTScore-bert-base-uncased | 0.18 | 0.339 | 0.288 | 0.438 | 0.364 | 0.209 | 0.41 | 0.318 |
| | BERTScore-byt5-small | 0.172 | 0.286 | 0.278 | 0.422 | 0.307 | 0.194 | 0.373 | 0.290 |
| | BERTScore-byt5-base | **0.197** | 0.326 | 0.297 | 0.419 | 0.358 | **0.215** | **0.418** | **0.319** |
| | BERTScore-byt5-large | 0.193 | 0.333 | 0.304 | 0.427 | 0.354 | 0.208 | 0.415 | **0.319** |
| First | BERTScore-bert-base-uncased | 0.147 | 0.295 | 0.263 | 0.421 | 0.318 | 0.183 | 0.361 | 0.284 |
| | BERTScore-byt5-small | 0.171 | 0.285 | 0.279 | 0.422 | 0.307 | 0.194 | 0.370 | 0.290 |
| | BERTScore-byt5-base | 0.164 | 0.276 | 0.280 | 0.414 | 0.307 | 0.191 | 0.362 | 0.285 |
| | BERTScore-byt5-large | 0.161 | 0.277 | 0.280 | 0.416 | 0.308 | 0.189 | 0.361 | 0.285 |
| Mean of aggregation | BERTScore-bert-base-uncased | 0.17 | **0.326** | 0.289 | **0.437** | **0.351** | 0.206 | 0.397 | 0.311 |
| | BERTScore-byt5-small | 0.170 | 0.292 | 0.284 | 0.420 | 0.313 | 0.202 | 0.372 | 0.293 |
| | BERTScore-byt5-base | 0.188 | 0.324 | **0.305** | 0.427 | 0.347 | 0.207 | 0.408 | 0.315 |
| | BERTScore-byt5-large | 0.185 | 0.322 | 0.311 | 0.431 | 0.343 | 0.208 | 0.411 | 0.316 |

Table 8: Segment-level correlation metric results Kendall for seven to-non-English language pairs in WMT19 with respect to fist layer, default layer and mean of aggregated embeddings setting without any attack i.e. $p = 0$.

| Setting | Metric | de-en | fi-en | gu-en | kk-en | lt-en | ru-en | zh-en | Average |
|---|---|---|---|---|---|---|---|---|---|
| Default | BERTScore-bert-base-uncased | -0.003 | -0.014 | -0.027 | 0.149 | -0.022 | 0.024 | 0.126 | 0.033 |
| | BERTScore-byt5-small | **0.155** | **0.266** | 0.239 | 0.392 | **0.264** | **0.175** | **0.360** | **0.264** |
| | BERTScore-byt5-base | 0.014 | -0.009 | 0.026 | 0.147 | 0.052 | 0.042 | 0.155 | 0.061 |
| | BERTScore-byt5-large | 0.011 | -0.055 | -0.018 | 0.141 | -0.015 | 0.032 | 0.155 | 0.036 |
| First | BERTScore-bert-base-uncased | 0.074 | 0.215 | 0.082 | 0.215 | 0.234 | 0.120 | 0.278 | 0.174 |
| | BERTScore-byt5-small | **0.155** | **0.266** | 0.239 | 0.392 | **0.264** | **0.175** | **0.360** | **0.264** |
| | BERTScore-byt5-base | 0.147 | 0.256 | **0.262** | **0.403** | 0.264 | 0.166 | 0.348 | **0.264** |
| | BERTScore-byt5-large | 0.138 | 0.258 | 0.259 | 0.394 | 0.262 | 0.170 | 0.352 | 0.262 |
| Mean of aggregation | BERTScore-bert-base-uncased | 0.053 | 0.144 | 0.052 | 0.214 | 0.149 | 0.082 | 0.240 | 0.133 |
| | BERTScore-byt5-small | 0.070 | 0.089 | 0.094 | 0.244 | 0.109 | 0.107 | 0.273 | 0.141 |
| | BERTScore-byt5-base | 0.025 | -0.029 | 0.022 | 0.263 | -0.019 | 0.056 | 0.123 | 0.063 |
| | BERTScore-byt5-large | 0.054 | 0.005 | 0.020 | 0.255 | 0.013 | 0.095 | 0.156 | 0.085 |

Table 9: Segment-level correlation metric results Kendall for seven to-non-English language pairs in WMT19 with respect to fist layer, default layer and mean of aggregated embeddings setting under visual attack at 0.3 perturbation level.



(a) 9th layer, *attacked reference*:
“Th̄is was the ₚos⦣ible caṳse of the f(r)e.”

(b) 9th layer, *normal reference*:
“This could possibly be the cause of the fire.”

(c) 1st layer, *attacked reference*:
“Th̄is was the ₚos⦣ible caṳse of the f(r)e.”

(d) 1th layer, *normal reference*:
“ This could possibly be the cause of the fire.”

Figure 5: Similarity Matrix using BERTScore with bert-base-uncased for *candidate*: “ This could possibly be the cause of the fire.” in different setups.