Synthetic Data Generation for Low-resource Grammatical Error Correction with Tagged Corruption Models

Felix Stahlberg and **Shankar Kumar**
Google Research
{fstahlberg,shankarkumar}@google.com

Abstract

Tagged corruption models provide precise control over the introduction of grammatical errors into clean text. This capability has made them a powerful tool for generating pre-training data for grammatical error correction (GEC) in English. In this work, we demonstrate their application to four languages with substantially fewer GEC resources than English: German, Romanian, Russian, and Spanish. We release a new tagged-corruption dataset consisting of 2.5M examples per language that was generated by a fine-tuned PaLM 2 foundation model. Pre-training on tagged corruptions yields consistent gains across all four languages, especially for small model sizes and languages with limited human-labelled data.

1 Introduction

Grammatical error correction (GEC) is the task of correcting writing errors in text (see Bryant et al. (2023) for an overview). Neural sequence-to-sequence models, commonly used for GEC, are hard to train due to limited human-labelled data. A common strategy to mitigate data sparsity is to generate synthetic training data, but most existing methods do not generate sufficiently diverse errors. Modern GEC systems are expected to handle a broad range of errors involving grammar, spelling, word choice, punctuation and orthography. However, many existing data generation methods that employ rules or character- or word- level noising strategies, cover only a small subset of error types (Grundkiewicz et al., 2019; Grundkiewicz and Junczys-Dowmunt, 2019; Náplava and Straka, 2019; Lichtarge et al., 2019; Flachs et al., 2021). Stahlberg and Kumar (2021) improved the diversity of *model-based* data generation (Xie et al., 2018; Kiyono et al., 2019) by introducing *tagged corruption* models. Tagged corruption models are trained to generate an ungrammatical version of a clean sentence given a specific error type tag. For example, the incorrect plural "sheeps" of "sheep" (i.e. a noun inflection error – `NOUN:INFL`) would be represented in a sentence as follows (Stahlberg and Kumar, 2021):

> "`NOUN:INFL` There were a lot of sheep."
> $\rightarrow$ "There were a lot of sheeps."

In this work, we adapt the tagged corruption approach of Stahlberg and Kumar (2021) to languages with fewer GEC resources than English such as German, Spanish, Romanian, and Russian. We faced two major challenges: First, training tagged corruption models is more challenging due to training data scarcity. We mitigated this issue by leveraging the large language model PaLM 2 (Anil et al., 2023). Second, automatic error type annotation tools such as ERRANT (Felice et al., 2016; Bryant et al., 2017) for English are not available for most other languages. Therefore, we developed a multilingual annotation tool based on classification rules that apply to multiple languages and writing systems. Using our framework, we generated a new synthetic pre-training dataset with 2.5M examples per language. We demonstrate consistent gains from pre-training mT5 (Xue et al., 2021) models on our new dataset and then fine-tuning them on gold data. We achieve the largest improvements (up to 30% relative) for smaller models and languages with limited gold data. We have released the dataset and the error annotation tool to the scientific community.

2 Multilingual rule-based error type annotation

ERRANT (Felice et al., 2016; Bryant et al., 2017) is a rule-based system for English that classifies writing errors into 25 different error categories. Some ERRANT rules are specific to English and do not apply to other languages. German (Boyd, 2018)

Proceedings of the 19th Workshop on Innovative Use of NLP for Building Educational Applications, pages 11–16

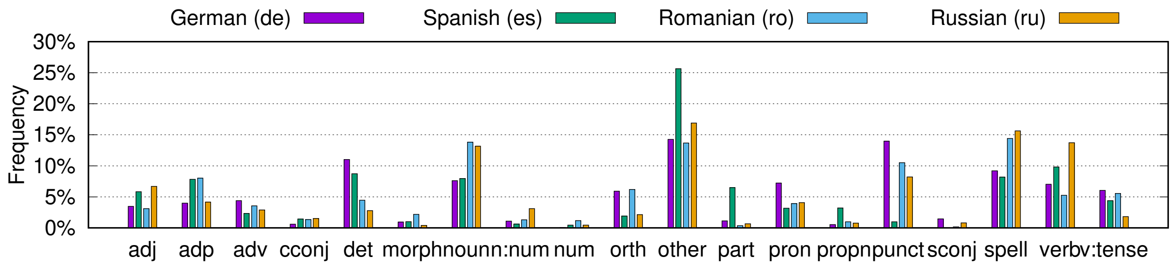

Figure 1: Development set tag distributions for German, Spanish, Romanian, and Russian.

Tag	Description
adj, adp, adv, cconj, det, part, pron, propn, sconj	Error classified by SpaCy part-of-speech (POS) tag.
morph	Morphology error.
noun	Noun or noun phrase error.
n:num	Noun number error.
num	Number error.
orth	Orthography error.
other	Unclassified error (no rule matched).
punct	Punctuation error.
spell	Spelling error according to GNU Aspell 0.60.
verb	Verb or verb phrase error.
v:tense	Verb tense error.
wo	Word order error.

Table 1: The error type tag set of our multilingual annotation tool. We use the same tag set for all languages. Rules are defined based on Aspell suggestions and SpaCy POS tags.

and Romanian (Cotet et al., 2020) versions of ERRANT have been developed, but they continue to be language-specific. Since our goal is to develop a recipe for low-resource GEC that is applicable to a large set of languages, we developed an annotation toolkit that implements a small set of general rules relying on multi-lingual NLP toolkits such as SpaCy's[1] part-of-speech (POS) tagger or GNU Aspell[2] for spelling correction. The error tag set of our tool is shown in Table 1.[3] We intentionally did not implement rules that rely on any language-specific knowledge beyond SpaCy's POS tags or Aspell suggestions. Therefore, compared to ERRANT, our tag set is more coarse-grained and less expressive. Despite the drawback, the tool's multilingual nature makes it useful for synthetic data generation across a range of languages.

[1] `https://spacy.io/`

[2] `http://aspell.net/`

[3] An open-source version of our tool is released on the dataset Github page. Please see the source code for more details about the implemented rules.

3 Synthetic data generation using a tagged corruption model

Tagged corruption models are neural models that corrupt a clean sentence according to an error type tag. We adapt Stahlberg and Kumar's (2021) recipe for English data generation as follows: for each language:

1. Annotate the gold development set with error type tags using our tool from Sec. 2.
2. Compute the unigram distribution of error tags on the gold development set.
3. Sample sentences from the large clean text corpus mC4[4] (Xue et al., 2021).
4. Randomly assign an error tag to each sentence according to the tag distribution.
5. Use the tagged corruption model with temperature sampling to generate corrupted versions of the sentences. Pair them with the original sentences to build a parallel GEC dataset.
6. Filter the dataset with language identification and simple heuristics based on length offsets and edit distances.

Fig. 1 shows the tag distributions on the development set for German, Spanish, Romanian, and Russian. Our corruption model is a PaLM 2 (Anil et al., 2023) model[5] that was jointly fine-tuned on the gold training sets of all four languages. The corruption model uses the following format:

"Corrupt $\langle$`lang`$\rangle$ $\langle$`tag`$\rangle$: $\langle$`clean_sentence`$\rangle$" $\rightarrow$ "$\langle$`corrupted_sentence`$\rangle$"

Fig. 2 illustrates how a training example for the corruption model is derived from the gold data. If a

[4] `https://www.tensorflow.org/datasets/catalog/c4#c4multilingual`

[5] "Bison" model size available via the Google Cloud API.

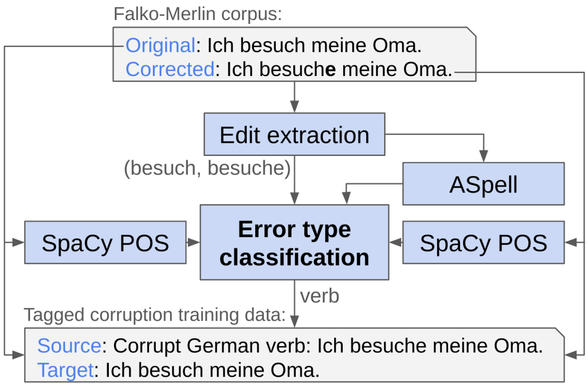

Figure 2: Example training instance for the tagged corruption model with a German verb error.

	de	es	ro	ru
Number of examples	2.5M			
Avg. sentence length (words)	18.9	22.0	20.8	19.1
Avg. edit distance (words)	2.8	1.9	2.3	1.5
Avg. sentence length (chars)	131.8	134.1	130.6	137.1
Avg. edit distance (chars)	5.6	5.2	3.6	4.1

Table 2: Average sentence lengths and source/target edit distances in the PRE corpus.

Language	Corpus	Train	Dev	Test
German (de)	Falko-Merlin	19.2K	2.5K	2.3K
Spanish (es)	COWS-L2H	10.1K	1.4K	1.1K
Romanian (ro)	RONACC	7.1K	1.5K	1.5K
Russian (ru)	RULEC	5.0K	2.5K	5.0K

Table 3: Number of training examples in the GOLD datasets.

sentence has multiple errors, the training example is repeated with each error tag.

Using the recipe (steps 1-6) we generated a large synthetic dataset[6] consisting of 2.5M examples per language. Table 2 lists some basic statistics of our new dataset. We will refer to this dataset as PRE.

4 Experimental setting

4.1 Gold datasets

We use the following GOLD GEC datasets for training the corruption model and for fine-tuning our GEC models: the Falko-Merlin corpus (Boyd, 2018) for German (de), the COWS-L2H corpus (Davidson et al., 2020) for Spanish (es), the RONACC corpus (Cotet et al., 2020) for Romanian (ro), and the RULEC-GEC corpus (Rozovskaya and Roth, 2019) for Russian (ru). Table 3 lists the dataset sizes.

[6]https://github.com/google-research-datasets/C4_200M-synthetic-dataset-for-grammatical-error-correction

4.2 Training setups

We train monolingual GEC models by fine-tuning the publicly available mT5 (Xue et al., 2021) checkpoints using the T5X (Roberts et al., 2023) framework on 4x4 TPUs (v3). We chose mT5 because it is available for a wide range of languages and model sizes. We use the default hyper-parameters,[7] but tune the learning rate (0.0001-0.001) and the number of training steps (1K-20K) on the respective development set. The model sizes range from *mT5-base* (580M parameters) to *mT5-xxl* (13B parameters). We compare four different training pipelines:

- GOLD: Fine-tune on the gold dataset (Sec. 4.1).
- PRE: Fine-tune on the synthetic tagged corruption dataset (Sec. 3).
- PRE→GOLD: Fine-tune first on the synthetic dataset, and then on the gold dataset.
- PRE+CLANG8→GOLD (only German and Russian): Fine-tune first on a 1:1 mix of the synthetic dataset and the CLANG8 corpus (Rothe et al., 2021), and then on the gold dataset. The CLANG8 corpus is a re-annotated version of the the language learner corpus Lang-8[8] (Mizumoto et al., 2011) available in German (114K examples) and Russian (45K examples).

5 Results

Like prior work we compute $F_{0.5}$-scores on the German, Russian, and Spanish test sets with the M2 scorer (Dahlmeier and Ng, 2012), and on the Romanian test set with Cotet et al.'s (2020) version of ERRANT.[9]

Table 4 contains the results for the three training setups for all four languages and model sizes. $F_{0.5}$-scores after training on PRE do not always surpass the GOLD baseline, which indicates that our synthetic dataset is not a replacement for human-labelled data. However, subsequent fine-tuning on GOLD after PRE consistently outperforms fine-tuning on GOLD alone, which shows the benefit of

[7]https://github.com/google-research/t5x/tree/main/t5x/examples/t5/mt5

[8]https://lang-8.com/

[9]https://github.com/teodor-cotet/errant/tree/0cb0f61af39ffb8c560ed6f92065f3b9e43e10dd

Setup	mT5-base				mT5-large				mT5-xl				mT5-xxl			
	de	es	ro	ru	de	es	ro	ru	de	es	ro	ru	de	es	ro	ru
GOLD	65.6	45.9	59.4	17.3	70.6	50.5	63.2	22.8	73.5	54.8	72.4	35.0	74.9	58.1	74.4	39.5
PRE	60.8	38.6	60.7	15.9	63.9	43.6	64.0	28.4	67.3	46.6	66.1	34.7	68.4	46.4	66.6	37.8
PRE→GOLD	**70.5**	**50.1**	**68.1**	**19.8**	**71.8**	**54.2**	**71.9**	**29.6**	**74.6**	**56.5**	**72.8**	**38.2**	**75.5**	**58.9**	**75.5**	**40.0**

Table 4: Test set $F_{0.5}$-scores for all four languages and model sizes. The systems highlighted in green outperform the GOLD baseline.

System	German (de)	Spanish (es)	Romanian (ro)	Russian (ru)
Grundkiewicz and Junczys-Dowmunt (2019)	70.24			34.46
Náplava and Straka (2019)	73.71			50.20
Katsumata and Komachi (2020)	68.86			44.36
Cotet et al. (2020)			53.80	
Niculescu et al. (2021)			69.01	
Flachs et al. (2021)	69.24	57.32		44.72
Rothe et al. (2021)	75.96			**51.62**
Náplava et al. (2022)	73.71			50.20
Kementchedjhieva and Søgaard (2023)	73.60	55.20	68.60	49.20
This work (mT5-xxl)				
PRE→GOLD	75.46	**58.89**	**75.47**	39.96
PRE+CLANG8→GOLD	**76.08**			44.31

Table 5: Comparison of the test set $F_{0.5}$-scores of our best systems to other results from the literature.

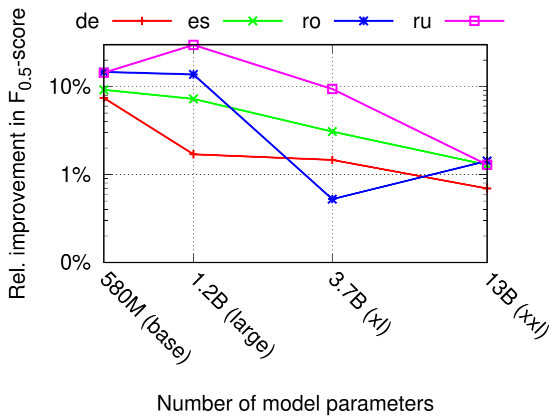

Figure 3: Relative improvements of the PRE→GOLD setup over GOLD-only.

adapting the model to the GEC domain before the final fine-tuning stage.

Fig. 3 shows a log-log plot of the relative improvements between the GOLD baseline and the PRE→GOLD setup across various model sizes. The improvements range between 0.5% and 30% depending on the language and model size. Our PRE dataset is particularly useful for small training sets (ru) and small models (left side of the plot). Grammatical error correction models deployed in practice are often small because a low latency is less disruptive for writers.

To investigate if pre-training can be further improved by adding external data, we performed experiments using the CLANG8 corpus (Rothe et al., 2021). Table 6 shows that pre-training on a 1:1 mix of PRE and CLANG8 outperforms pre-training on only one of them.

Setup	mT5-base		mT5-xxl	
	de	ru	de	ru
Rothe et al. (2021)	69.21	26.24	75.96	51.62
This work				
CLANG8	66.39	24.58	74.83	40.37
CLANG8→GOLD	70.59	26.24	75.65	43.62
PRE+CLANG8	69.87	25.74	74.47	**44.48**
PRE+CLANG8→GOLD	**72.02**	**26.39**	**76.08**	44.31

Table 6: Combining our PRE dataset with the CLANG8 corpus from Rothe et al. (2021). We report $F_{0.5}$-scores on the German and Russian test sets.

Table 5 lists our best setups in relation to prior work. We advance the state-of-the-art on Spanish and Romanian and match the best published results on German despite using a relatively simple training setup (standard 2-stage fine-tuning of off-the-shelf T5 models with normal cross-entropy loss).

6 Conclusion

We have introduced a new large synthetic dataset for GEC that was generated by an LLM-based tagged corruption model in German, Spanish, Romanian, and Russian. Our dataset consists of 2.5M examples per language. Pre-training GEC models on this dataset yields consistent gains on all four languages, especially for small gold training sets and small model sizes.

7 Limitations

Even though we took into account the distribution of the error tags on the development sets for synthetic data generation, it is possible that the synthetic dataset does not capture all its error characteristics. First, our tag set is not sufficient to represent more complex inter-dependencies between error types. Second, our automated annotation tool operates on the lexical level, so clausal, sentential, or discourse level errors are not represented in the error tag set. Third, the tagged corruption model is not guaranteed to always synthesize the correct error type. Fourth, error type tags are assigned to sentences randomly, but it is sometimes not even possible to enforce an error type in a particular sentence (e.g. corrupting a sentence without a conjunction with `cconj`). Despite these limitations, we confirm Stahlberg and Kumar's (2021) findings by demonstrating the effectiveness of tagged corruption models to generate diverse synthetic training data for GEC across a range of languages.

References

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Adriane Boyd. 2018. Using Wikipedia edits in low resource grammatical error correction. In *Proceedings of the 2018 EMNLP Workshop W-NUT: The 4th Workshop on Noisy User-generated Text*, pages 79–84, Brussels, Belgium. Association for Computational Linguistics.

Christopher Bryant, Mariano Felice, and Ted Briscoe. 2017. Automatic annotation and evaluation of error types for grammatical error correction. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 793–805, Vancouver, Canada. Association for Computational Linguistics.

Christopher Bryant, Zheng Yuan, Muhammad Reza Qorib, Hannan Cao, Hwee Tou Ng, and Ted Briscoe. 2023. Grammatical Error Correction: A Survey of the State of the Art. *Computational Linguistics*, 49(3):643–701.

Teodor-Mihai Cotet, Stefan Ruseti, and Mihai Dascalu. 2020. Neural grammatical error correction for romanian. In *2020 IEEE 32nd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 625–631.

Daniel Dahlmeier and Hwee Tou Ng. 2012. Better evaluation for grammatical error correction. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 568–572, Montréal, Canada. Association for Computational Linguistics.

Sam Davidson, Aaron Yamada, Paloma Fernandez Mira, Agustina Carando, Claudia H. Sanchez Gutierrez, and Kenji Sagae. 2020. Developing NLP tools with a new corpus of learner Spanish. In *Proceedings of the Twelfth Language Resources and Evaluation Conference*, pages 7238–7243, Marseille, France. European Language Resources Association.

Mariano Felice, Christopher Bryant, and Ted Briscoe. 2016. Automatic extraction of learner errors in ESL sentences using linguistically enhanced alignments. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 825–835, Osaka, Japan. The COLING 2016 Organizing Committee.

Simon Flachs, Felix Stahlberg, and Shankar Kumar. 2021. Data strategies for low-resource grammatical error correction. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 117–122, Online. Association for Computational Linguistics.

Roman Grundkiewicz and Marcin Junczys-Dowmunt. 2019. Minimally-augmented grammatical error correction. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 357–363, Hong Kong, China. Association for Computational Linguistics.

Roman Grundkiewicz, Marcin Junczys-Dowmunt, and Kenneth Heafield. 2019. Neural grammatical error correction systems with unsupervised pre-training on synthetic data. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 252–263, Florence, Italy. Association for Computational Linguistics.

Satoru Katsumata and Mamoru Komachi. 2020. Stronger baselines for grammatical error correction using a pretrained encoder-decoder model. 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*, pages 827–832, Suzhou, China. Association for Computational Linguistics.

Yova Kementchedjhieva and Anders Søgaard. 2023. Grammatical error correction through round-trip machine translation. In *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2208–2215, Dubrovnik, Croatia. Association for Computational Linguistics.

Shun Kiyono, Jun Suzuki, Masato Mita, Tomoya Mizumoto, and Kentaro Inui. 2019. An empirical study of incorporating pseudo data into grammatical error correction. 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 1236–1242, Hong Kong, China. Association for Computational Linguistics.

Jared Lichtarge, Chris Alberti, Shankar Kumar, Noam Shazeer, Niki Parmar, and Simon Tong. 2019. Corpora generation for grammatical error correction. 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 3291–3301, Minneapolis, Minnesota. Association for Computational Linguistics.

Tomoya Mizumoto, Mamoru Komachi, Masaaki Nagata, and Yuji Matsumoto. 2011. Mining revision log of language learning SNS for automated Japanese error correction of second language learners. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pages 147–155, Chiang Mai, Thailand. Asian Federation of Natural Language Processing.

Jakub Náplava and Milan Straka. 2019. Grammatical error correction in low-resource scenarios. In *Proceedings of the 5th Workshop on Noisy User-generated Text (W-NUT 2019)*, pages 346–356, Hong Kong, China. Association for Computational Linguistics.

Jakub Náplava, Milan Straka, Jana Straková, and Alexandr Rosen. 2022. Czech grammar error correction with a large and diverse corpus. *Transactions of the Association for Computational Linguistics*, 10:452–467.

Mihai Alexandru Niculescu, Stefan Ruseti, and Mihai Dascalu. 2021. Rogpt2: Romanian gpt2 for text generation. In *2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI)*, pages 1154–1161.

Adam Roberts, Hyung Won Chung, Gaurav Mishra, Anselm Levskaya, James Bradbury, Daniel Andor, Sharan Narang, Brian Lester, Colin Gaffney, Afroz Mohiuddin, Curtis Hawthorne, Aitor Lewkowycz, Alex Salcianu, Marc van Zee, Jacob Austin, Sebastian Goodman, Livio Baldini Soares, Haitang Hu, Sasha Tsvyashchenko, Aakanksha Chowdhery, Jasmijn Bastings, Jannis Bulian, Xavier Garcia, Jianmo Ni, Andrew Chen, Kathleen Kenealy, Kehang Han, Michelle Casbon, Jonathan H. Clark, Stephan Lee, Dan Garrette, James Lee-Thorp, Colin Raffel, Noam Shazeer, Marvin Ritter, Maarten Bosma, Alexandre Passos, Jeremy Maitin-Shepard, Noah Fiedel, Mark Omernick, Brennan Saeta, Ryan Sepassi, Alexander Spiridonov, Joshua Newlan, and Andrea Gesmundo. 2023. Scaling up models and data with t5x and seqio. *Journal of Machine Learning Research*, 24(377):1–8.

Sascha Rothe, Jonathan Mallinson, Eric Malmi, Sebastian Krause, and Aliaksei Severyn. 2021. A simple recipe for multilingual grammatical error correction. 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 702–707, Online. Association for Computational Linguistics.

Alla Rozovskaya and Dan Roth. 2019. Grammar error correction in morphologically rich languages: The case of Russian. *Transactions of the Association for Computational Linguistics*, 7:1–17.

Felix Stahlberg and Shankar Kumar. 2021. Synthetic data generation for grammatical error correction with tagged corruption models. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 37–47, Online. Association for Computational Linguistics.

Ziang Xie, Guillaume Genthial, Stanley Xie, Andrew Ng, and Dan Jurafsky. 2018. Noising and denoising natural language: Diverse backtranslation for grammar correction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 619–628, New Orleans, Louisiana. Association for Computational Linguistics.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2021. mT5: A massively multilingual pre-trained text-to-text transformer. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 483–498, Online. Association for Computational Linguistics.